علـــوی

مجموعه مدارس متوسطه اول علوی

سه‌گانه فناوری

(سخت‌افزار، نرم‌افزار و ذهن‌افزار)

پایه نهم



فهرست

Contents

بخش اول: الکترونیک ۵

درس ۱: آشنایی با ال‌ای‌دی، مقاومت و قانون اهم ۶

درس۲: محافظت از ال‌ای‌دی با استفاده از مقاومت ۱۱

درس۳: ساخت مدارها روی بردبورد ۱۴

درس۴: آشنایی با خازن ۱۸

درس۵: توصیف مدارها با استفاده از نمادها ۲۳

بخش دوم: نرم افزار ۲۸

درس ۶ : مرور مبانی بینایی ماشین ۲۹

درس۷: هنر تمرکز ۳۵

درس۸: آشنایی با تشخیص دست و ورود به دنیای MediaPipe ۳۹

درس۹: اتصال دوربین به مدل MediaPipe و تشخیص دست ۴۵

درس۱۰: استخراج مختصات و منطق ریاضی تشخیص وضعیت انگشتان ۴۹

درس۱۱: توسعه منطق تشخیص به تمام انگشتان و ساخت شمارشگر ۵۴

درس۱۲: الگوریتم شمارش انگشتان ۶۰

درس۱۳: طراحی رابط کاربری گرافیکی (GUI) و نمایش اطلاعات روی تصویر ۶۴

درس۱۴: آشنایی با HTML و ساخت اولین صفحه وب ۶۹

درس ۱۵: تگ‌ها و عناصر پرکاربرد HTML ۷۵

درس ۱۶: ویژگی‌ها، قالب‌بندی و طراحی صفحه وب با استایل ۸۱

درس۱۷: آشنایی با چاپ سه‌بعدی ۸۸

درس۱۸: آشنایی با اجزای پرینتر سه‌بعدی و فرآیند چاپ ۹۳

درس۱۹: طراحی سه‌بعدی با نرم‌افزار Tinkercad ۹۹

بخش سوم: ذهن افزار ۱۰۶

درس۲۰: آشنایی با هوش مصنوعی و کاربردهای آن ۱۰۷

درس۲۱: آینده هوش مصنوعی و چالش‌های آن ۱۱۲

درس۲۲: آشنایی با متاورس، واقعیت مجازی و واقعیت افزوده ۱۱۷

درس۲۳: فناوری‌های پشتیبان متاورس و کاربردهای آن ۱۲۲

سخنی با دبیر ارجمند

جزوه که پیش رو دارید، تلاشی است برای ترسیم نقشه راهی نوین در آموزش علوم رایانه برای دانش‌آموزان پایه نهم. ما در این مسیر تلاش کرده‌ایم از قالب‌های سنتی فاصله بگیریم و به جای صرفِ آموزش تئوری، بر «تفکر فناورانه» و «خلق کردن» تمرکز کنیم.

این جزوه در سه محور سخت‌افزار، نرم‌افزار و ذهن‌افزار طراحی شده است. هدف ما در این مجموعه تنها آموزش کدنویسی یا بستن یک مدار ساده نیست؛ بلکه می‌خواهیم دانش‌آموز بیاموزد که چگونه تکنولوژی را در خدمت حل مسائل واقعی به کار گیرد.

بخش اول: سخت‌افزار

در این بخش، دانش‌آموزان با مفاهیم الکترونیک و فناوری سخت‌افزاری آشنا می‌شوند و اجزای پایه مدارهای الکتریکی و الکترونیکی، نقش آن‌ها و برخی کاربردهای عملی این مفاهیم را بررسی می‌کنند. هدف این بخش، ایجاد درکی ابتدایی اما کاربردی از ساختار و عملکرد اجزای فیزیکی فناوری‌های دیجیتال است.

بخش دوم: نرم‌افزار

در این بخش، دانش‌آموزان با برخی ابزارها و محیط‌های نرم‌افزاری کاربردی آشنا می‌شوند و مهارت‌هایی در زمینه طراحی و ویرایش برنامه‌نویسی با پایتون، طراحی سایت و آشنایی با فرآیند طراحی و تولید به کمک چاپگر سه‌بعدی کسب می‌کنند. تأکید اصلی این بخش بر یادگیری از طریق فعالیت و پروژه و تبدیل دانش نظری به مهارت عملی است.

بخش سوم: ذهن‌افزار

در این بخش، تمرکز بر پرورش توانایی‌های ذهنی و شناختی موردنیاز در عصر فناوری است.

از شما همکار گرامی انتظار می‌رود :

تسهیل‌گر باشید: در بخش‌هایی مانند برنامه‌نویسی و بینایی ماشین، بیش از آنکه متصدی انتقال دانش باشید، راهنمای مسیر برای خطایابی و خلاقیت دانش‌آموزان باشید.

به تفاوت‌های فردی توجه کنید: سطح مهارت‌ها در این جزوه متنوع است. به دانش‌آموزانی که زودتر به نتیجه می‌رسند، فرصت چالش‌های تکمیلی بدهید و از آنانی که درگیرِ درکِ مفاهیم انتزاعی هستند، با مثال‌های ملموس‌تر حمایت کنید.

تفکر انتقادی را پرورش دهید: در بخش‌های هوش مصنوعی و اخلاق فناوری، کلاس درس باید فضایی برای پرسشگری و بحث‌های اخلاقی باشد، نه صرفاً حفظ کردن تعاریف.

ارزشیابی درس می‌تواند شامل آزمون‌های کتبی و عملی، فعالیت‌های کلاسی و پروژه‌های دانش‌آموزی باشد تا علاوه بر سنجش دانش و مفاهیم نظری، توانایی دانش‌آموزان در انجام فعالیت‌های واقعی و به‌کارگیری مهارت‌های آموخته‌شده نیز ارزیابی شود.

امید است این جزوه بتواند زمینه‌ای مناسب برای رشد مهارت‌های علمی و عملی، تقویت خلاقیت و حل مسئله و شکل‌گیری نگرشی آگاهانه و مسئولانه نسبت به رایانه، فناوری‌های نوین، هوش مصنوعی و دنیای دیجیتال فراهم آورد و مسیر یادگیری و استفاده درست از فناوری را برای دانش‌آموزان عزیز هموار سازد.

سخنی با دانش‌آموزان گرامی

فرزند خوبم، سلام!

تا امروز شاید رایانه را بیشتر به‌عنوان وسیله‌ای برای انجام کارهای روزمره، بازی، جست‌وجو یا ارتباط با دیگران می‌شناختیم؛ اما در این جزوه قرار است نگاه متفاوتی به فناوری داشته باشیم. می‌خواهیم یاد بگیریم رایانه و ابزارهای دیجیتال چگونه ساخته و کار می‌کنند، چگونه می‌توان مشکلات آن‌ها را شناسایی و برطرف کرد، چگونه از نرم‌افزارهای مختلف استفاده کنیم و چگونه با کمک فناوری، ایده‌های خود را به طرح‌ها و پروژه‌های واقعی تبدیل کنیم.

در این جزوه، قرار نیست فقط مطالبی را بخوانیم و به خاطر بسپاریم؛ قرار است تجربه کنیم، بسازیم، آزمایش کنیم، خطا کنیم، دوباره تلاش کنیم و از تجربه‌های خود یاد بگیریم. فعالیت‌ها و پروژه‌های هر درس فرصتی هستند تا آموخته‌های خود را در عمل به کار بگیریم و مهارت‌هایمان را گام‌به‌گام توسعه دهیم.

هدف ما این است که در پایان این جزوه، تنها یک «کاربر فناوری» نباشیم؛ بلکه به دانش‌آموزانی آگاه، خلاق، توانمند و مسئول تبدیل شویم که می‌توانند فناوری را درست بشناسند، هوشمندانه به کار بگیرند، مسئله‌ها را حل کنند و برای ساختن ایده‌های تازه از آن بهره ببرند.

پس آماده باش؛ در «فناوری‌های سه‌گانه» قرار است هم ابزارها را بشناسیم، هم نرم‌افزارها را به کار بگیریم و هم ذهن خود را برای ساختن آینده‌ای بهتر آماده کنیم.

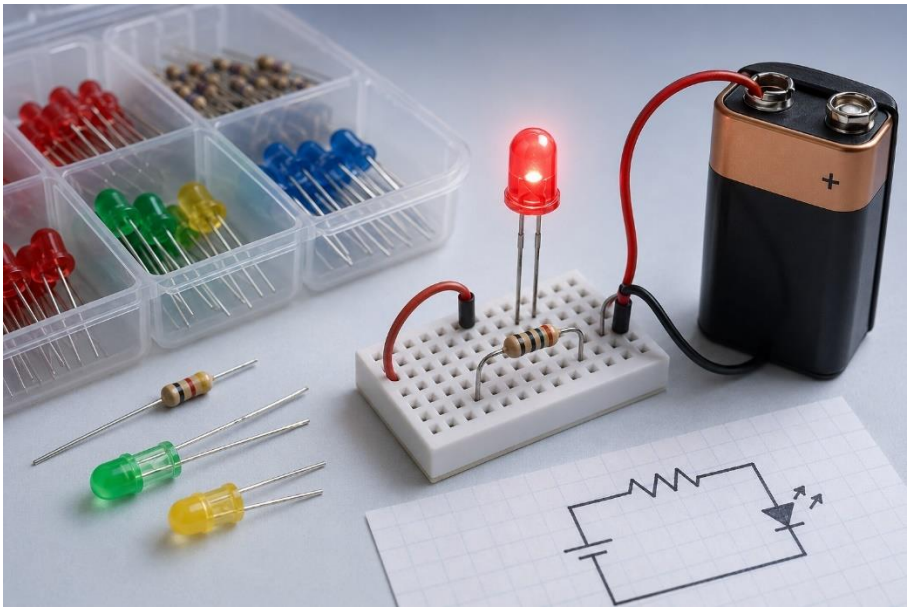
در این جزوه برای ساده‌تر شدن مطالب، از برخی علامت‌های اختصاری استفاده شده است:

	مقدمه و شروع هر درس		خلاصه درس
	ورود به مطالب اصلی و توضیح مفاهیم		تمرین در منزل
	دستیار هوشمند و هوش مصنوعی		پروژه نهایی درس
	آیا می‌دانید و اطلاعات بیشتر		چالش برای دانش‌آموزان قوی
	فعالیت کلاسی		

بخش اول: الکترونیک

Electronics

درس ۱: آشنایی با ال‌ای‌دی، مقاومت و قانون اهم



در پایان این درس خواهید توانست:

- کاربردهای ال‌ای‌دی را بیان کنید.
- وظیفه مقاومت را در مدار توضیح دهید.
- قانون اهم را در محاسبه‌های ساده به کار ببرید.

آیا تاکنون به چراغ کوچک تلویزیون، مودم یا رایانه دقت کرده‌اید؟
چرا این چراغ‌ها ساعت‌ها روشن می‌مانند اما نمی‌سوزند؟ اگر یک ال‌ای‌دی را مستقیم به باتری وصل کنیم، چه اتفاقی می‌افتد؟

در این درس با یکی از مهم‌ترین قطعات الکترونیکی یعنی ال‌ای‌دی و قطعه‌ای که از آن محافظت می‌کند، یعنی مقاومت آشنا می‌شویم. یک ال‌ای‌دی، یک مقاومت و یک باتری را مشاهده کنید. به نظر شما:

۱. ال‌ای‌دی چه کاری انجام می‌دهد؟
۲. آیا می‌توان ال‌ای‌دی را مستقیماً به باتری وصل کرد؟

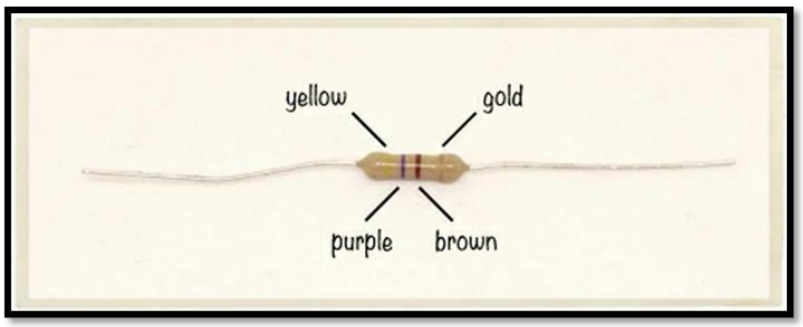
نظر خود را با اعضای گروه در میان بگذارید.

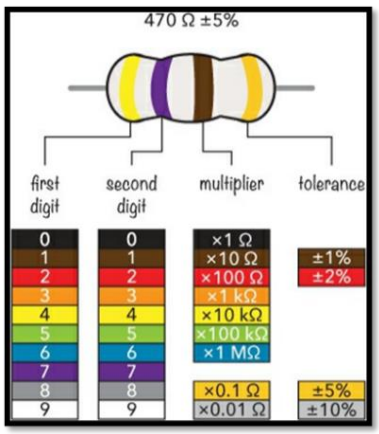
ایجاد نور با ال‌ای‌دی‌ها

ال‌ای‌دی‌ها (دیودهای نورافشان) قطعات الکترونیکی پرکاربردی هستند که نور تولید می‌کنند. آن‌ها هم به عنوان چراغ نشانگر ساده (مثل چراغ روشن/خاموش دستگاه‌ها) و هم در دستگاه‌های پیچیده‌تر مانند نمایشگرها (که از هزاران ال‌ای‌دی ریز تشکیل شده‌اند) استفاده می‌شوند.

مقاومت چیست و چه کاربردی دارد؟

مقاومت قطعه‌ای است که در برابر عبور جریان الکتریکی مقاومت می‌کند. در یک مدار با ولتاژ ثابت، هرچه مقدار مقاومت بیشتر باشد، جریان کمتری از مدار عبور می‌کند. واحد اندازه‌گیری مقاومت، اُهم (Ω) است.





چگونه مقدار مقاومت را بخوانیم؟

مقاومت‌ها معمولاً با نوارهای رنگی نشانه‌گذاری می‌شوند:

دو نوار اول: ارقام عدد مقاومت.

نوار سوم: ضریب (تعداد صفرها).

نوار چهارم: تلرانس (درصد خطای مجاز).

مثال:

نوارهای زرد (۴)، بنفش (۷)، قهوه‌ای (۱۰) و طلایی (تلرانس ۵٪) به معنای مقاومت با تلرانس ±۵٪ است.

$$47 \times 10 = 470\Omega$$

توجه: اگر یک مقاومت به جای چهار نوار، پنج نوار داشته باشد، در آن صورت سه نوار اول ارقام هستند و نوار چهارم ضریب است.

چرا مقاومت مهم است؟

مقاومت‌ها برای محدود کردن جریان و ایجاد افت ولتاژ در مدار به کار می‌روند. اگر از آن‌ها به درستی استفاده نشود، ممکن است باعث آسیب دیدن قطعات (مثل ال‌ای‌دی) یا تخلیه سریع باتری شوند.

قانون اهم: رابطه کلیدی در الکترونیک

قانون اهم رابطه بین ولتاژ(V) ، جریان (I) و مقاومت (R) را بیان می‌کند:

$$V = I \times R$$

در اینجا معنی این حروف آمده است:

- V ولتاژ، که با واحد ولت (V) اندازه‌گیری می‌شود.
- I جریان، که با واحد آمپر (A) اندازه‌گیری می‌شود.
- R مقاومت، که با واحد اهم (Ω) اندازه‌گیری می‌شود.

با توجه به این تعاریف، قانون اُهم به زبان ساده می‌گوید: «ولتاژ برابر است با جریان ضربدر مقاومت.» شما همچنین می‌توانید فرمول قانون اُهم را به دو صورت زیر بنویسید:

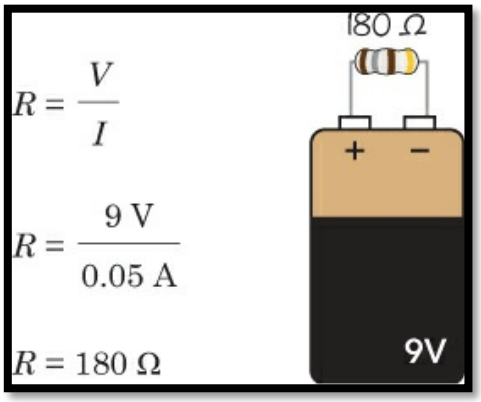
$$I = \frac{V}{R}$$

$$R = \frac{V}{I}$$

بیایید قانون اُهم را به کار بگیریم. تصور کنید یک مقاومت و یک باتری ۹ ولتی دارید و می‌خواهید جریان ۰.۰۵ آمپر در مقاومت جاری شود. برای به دست آوردن مقدار جریان مورد نظر، به چه مقدار مقاومت نیاز دارید؟ از قانون اُهم برای یافتن پاسخ استفاده کنید:

$$R = \frac{V}{I}$$

با جایگذاری مقادیر داده شده:



بنابراین، برای اینکه جریان ۰.۰۵ آمپر از یک باتری ۹ ولتی عبور کند، به یک مقاومت ۱۸۰ اُهمی نیاز دارید.

فعالیت کلاسی

در بسیاری از وسایل الکترونیکی مانند تلویزیون، رایانه، ماشین لباسشویی و روترهای وای‌فای، از الای‌دی برای نشان دادن وضعیت دستگاه یا عملکرد بخش‌های مختلف آن استفاده می‌شود. برای اینکه جریان عبوری از الای‌دی از مقدار مجاز بیشتر نشود، در بسیاری از مدارها از مقاومت یا مدارهای محدودکننده جریان استفاده می‌شود.

در این فعالیت می‌خواهیم با ساخت یک مدار ساده، نقش مقاومت در مدار الای‌دی را بررسی کنیم و ببینیم چرا استفاده از مقاومت مناسب اهمیت دارد.

وسایل مورد نیاز

- یک الای‌دی
- یک باتری کتابی
- یک مقاومت ۳۳۰ اهم
- سیم رابط، در صورت نیاز

شرح فعالیت

۱. الای‌دی، مقاومت و باتری را با راهنمایی دبیر به‌صورت صحیح به یکدیگر متصل کنید.
۲. مدار را کامل کنید و روشن شدن الای‌دی را مشاهده کنید.
۳. توجه کنید که مقاومت در مسیر الای‌دی قرار گرفته است و جریان عبوری از آن را محدود می‌کند.

خلاصه درس

- الای‌دی یک قطعه الکترونیکی است که با عبور جریان مناسب، نور تولید می‌کند. جریان عبوری از الای‌دی باید در محدوده مجاز آن باشد.
- مقاومت وظیفه محدود کردن جریان و ایجاد افت ولتاژ در مدار را بر عهده دارد و استفاده از مقاومت مناسب می‌تواند از آسیب دیدن قطعاتی مانند الای‌دی جلوگیری کند.
- قانون اهم رابطه بین ولتاژ، جریان و مقاومت را بیان می‌کند و برای محاسبه مقادیر این سه کمیت در مدارهای الکترونیکی به کار می‌رود.

تمرین

۱- جاهای خالی را کامل کنید.

الف) واحد اندازه‌گیری مقاومت ..... است.

ب) الای‌دی هنگام عبور جریان ..... تولید می‌کند.

ج) هرچه مقدار مقاومت بیشتر باشد، ..... کمتری از مدار عبور می‌کند.

۲- وظیفه مقاومت در مدار چیست؟

.....

۳-چرا بیشتر مدارهای دارای ال‌ای‌دی از مقاومت استفاده می‌کنند؟

.....

۴-قانون اهم چه کمیت‌هایی را به هم مرتبط می‌کند؟

.....

۵-اگر ولتاژ یک باتری ۶ ولت و جریان مدار ۰٫۰۳ آمپر باشد، مقدار مقاومت را با استفاده از قانون اهم محاسبه کنید.

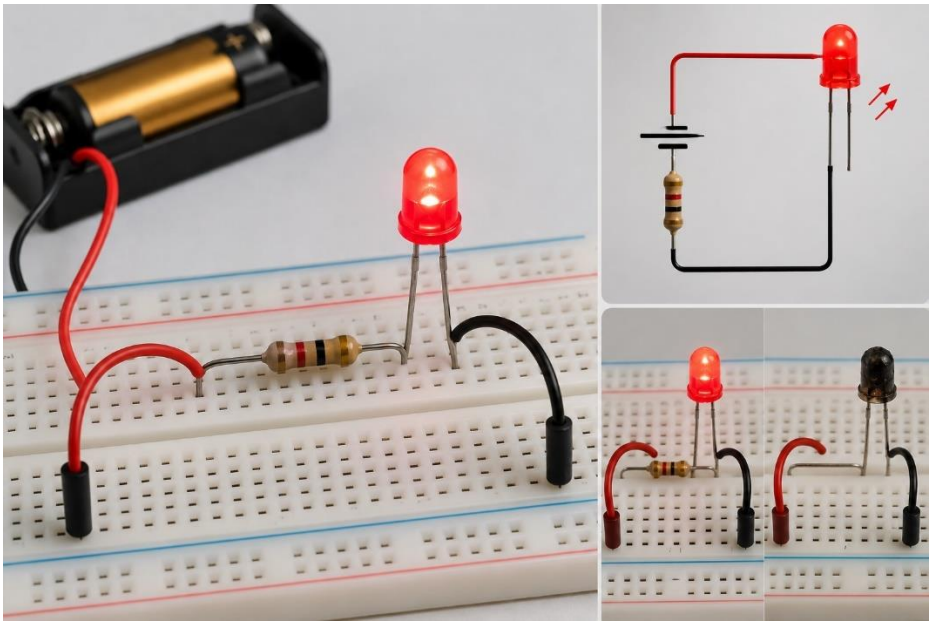
.....

پروژه

با استفاده از یک ال‌ای‌دی، یک باتری و یک مقاومت ۳۳۰ اهمی، یک مدار ساده بسازید و ال‌ای‌دی را روشن کنید. پس از ساخت مدار، به پرسش‌های زیر پاسخ دهید:

- مقاومت چه نقشی در مدار داشت؟
- اگر مقاومت را حذف کنید چه اتفاقی ممکن است رخ دهد؟

درس۲: محافظت از ال‌ای‌دی با استفاده از مقاومت



در پایان این درس خواهید توانست:

- دلیل استفاده از مقاومت در مدار ال‌ای‌دی را توضیح دهید.
- مقدار مقاومت مناسب برای یک ال‌ای‌دی را محاسبه کنید.
- مدار ساده روشن کردن ال‌ای‌دی را ببندید.
- از مقاومت مناسب برای جلوگیری از آسیب دیدن ال‌ای‌دی استفاده کنید.


الای‌دی یکی از پرکاربردترین قطعات الکترونیکی است که در وسایل مختلف مانند چراغ‌ها، نمایشگرها و بردهای الکترونیکی استفاده می‌شود. اگر الای‌دی بدون یک مدار مناسب برای محدود کردن جریان به منبع تغذیه متصل شود، ممکن است جریان بیش از حد مجاز از آن عبور کند و به الای‌دی آسیب برساند. بنابراین، در بسیاری از مدارها برای محدود کردن جریان الای‌دی از یک مقاومت مناسب به‌صورت سری استفاده می‌شود.

به پرسش‌های زیر پاسخ دهید.

۱. آیا تاکنون سوختن یک الای‌دی را مشاهده کرده‌اید؟
۲. اگر جریان زیادی از یک وسیله الکترونیکی عبور کند، چه اتفاقی می‌افتد؟
۳. به نظر شما چگونه می‌توان از الای‌دی محافظت کرد؟

چرا از مقاومت استفاده می‌کنیم؟

در بسیاری از مدارهای ساده، برای محدود کردن جریان الای‌دی از یک مقاومت مناسب به‌صورت سری با آن استفاده می‌شود. البته، مقاومت‌ها با مقادیر مختلفی در دسترس هستند و برای یافتن مقدار مناسب برای مدار خود، باید کمی محاسبات انجام دهید. بسیاری از الای‌دی‌های معمولی، در جریان‌های حدود چند میلی‌آمپر تا چند ده میلی‌آمپر کار می‌کنند و افت ولتاژ مستقیم آن‌ها نیز به نوع و رنگ الای‌دی بستگی دارد. برای یک مثال ساده آموزشی می‌توان افت ولتاژ حدود ۲ ولت و جریان ۲۰ میلی‌آمپر را در نظر گرفت.این دو مقدار، به همراه ولتاژ باتری شما، تمام آن چیزی است که برای محاسبه مقاومت مناسب نیاز دارید. فقط این دو مقدار را در فرمول زیر قرار دهید:

$$R = \frac{V_{battery} - V_{LED}}{I_{LED}}$$

در این فرمول:

- R مقدار مقاومت مورد نیاز بر حسب اهم (Ω) است.
- $V_{battery}$ ولتاژ باتری شما بر حسب ولت (V) است.
- V_{LED} ولتاژ مورد نیاز الای‌دی برای روشن شدن بر حسب ولت (V) است (معمولاً حدود ۲ ولت).
- I_{LED} جریان مورد نیاز الای‌دی بر حسب آمپر (A) است (معمولاً حدود ۰.۰۲ آمپر).

فعالیت کلاسی

وسایل مورد نیاز

- یک عدد الای‌دی
- یک عدد مقاومت ۳۳۰ اهم
- یک عدد باتری کتابی ۹ ولت
- یک سر باتری کتابی
- سیم رابط

مراحل اجرا

۱. پایه‌های الای‌دی را شناسایی کنید و جهت صحیح اتصال آن را مشخص کنید.
۲. مقاومت ۳۳۰ اهم را به‌صورت سری با الای‌دی متصل کنید.
۳. دو سر مدار را با استفاده از سر باتری به باتری ۹ ولت متصل کنید.

خلاصه درس

- الای‌دی برای کارکرد صحیح به جریان مشخصی نیاز دارد.
- مقاومت از عبور جریان زیاد جلوگیری می‌کند.
- مقاومت همیشه به صورت سری با الای‌دی قرار می‌گیرد.
- مقدار مقاومت با استفاده از ولتاژ باتری، ولتاژ الای‌دی و جریان الای‌دی محاسبه می‌شود.
- در عمل معمولاً از نزدیک‌ترین مقدار استاندارد مقاومت استفاده می‌شود.

تمرین

۱. چرا باید از مقاومت در مدار الای‌دی استفاده کنیم؟

.....

۲. اگر ولتاژ باتری ۹ ولت باشد، چرا نمی‌توان الای‌دی را مستقیماً به آن متصل کرد؟

.....

۳. اجزای فرمول محاسبه مقاومت را نام ببرید.

.....

٤. اگر مقدار مقاومت محاسبه شده ۳۵۰ اهم باشد، از چه مقاومت استانداردی می‌توان استفاده کرد؟

.....

پروژه

با استفاده از یک باتری، یک الای‌دی و یک مقاومت، مدار روشن کردن الای‌دی را روی بردبورد یا تخته آزمایش ببندید.

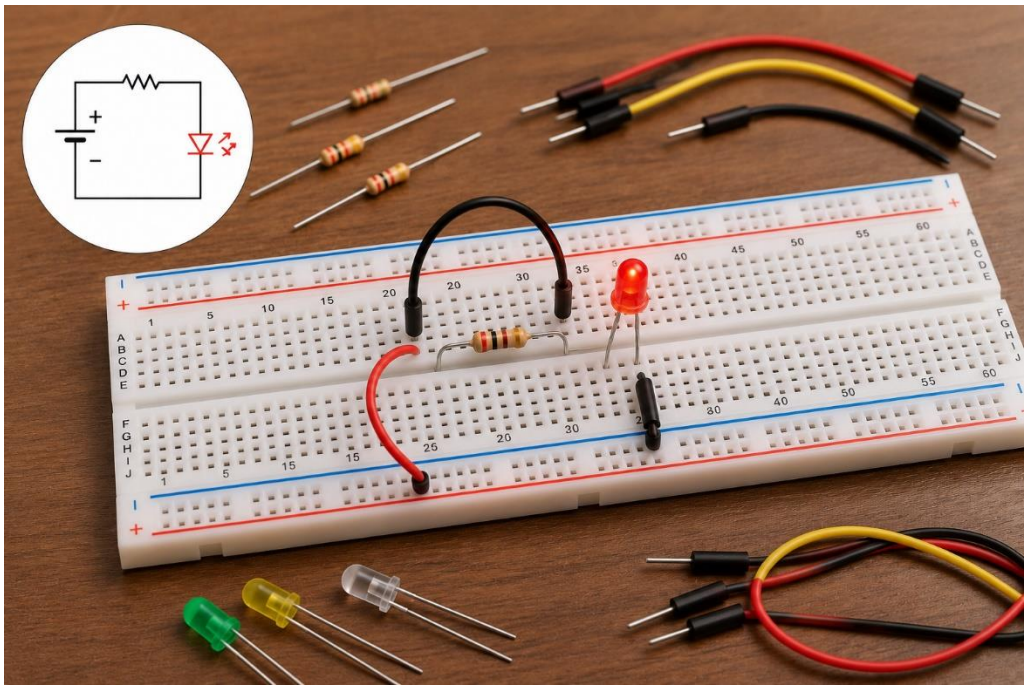
سپس:

- از مدار خود عکس بگیرید.
- مقدار مقاومت استفاده شده را یادداشت کنید.
- توضیح دهید چرا این مقاومت را انتخاب کرده‌اید.

چالش

اگر بخواهید دو الای‌دی را با یک باتری ۹ ولت روشن کنید، آیا می‌توان از همان مقاومت ۳۳۰ اهم استفاده کرد؟ دلیل خود را توضیح دهید.

درس۳: ساخت مدارها روی بردبورد



در پایان این درس خواهید توانست:

- بردبورد و کاربرد آن را توضیح دهید.
- نحوه اتصال قطعات روی بردبورد را بیان کنید.
- ریل‌های تغذیه و ناحیه قطعات را تشخیص دهید.
- از سیم مناسب برای بردبورد استفاده کنید.
- یک مدار ساده ال‌ای‌دی را روی بردبورد مونتاژ کنید.

با افزایش تعداد قطعات در یک مدار، اتصال آن‌ها با سیم یا پیچاندن پایه‌ها دشوار و نامنظم می‌شود. **بردبورد** وسیله‌ای است که امکان ساخت و آزمایش مدارهای الکترونیکی را بدون لحیم‌کاری فراهم می‌کند. همچنین می‌توان پس از پایان آزمایش، قطعات را به‌راحتی جدا کرده و دوباره در پروژه‌های دیگر استفاده کرد.

به پرسش‌های زیر پاسخ دهید.

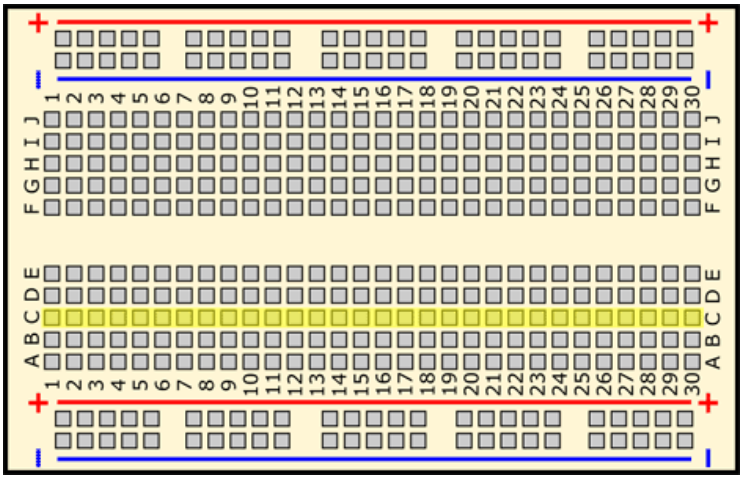
۱. اگر بخواهید مداری با چند قطعه الکترونیکی بسازید، چه روشی را انتخاب می‌کنید؟
۲. آیا تاکنون از بردبورد استفاده کرده‌اید؟
۳. به نظر شما چرا در آموزش الکترونیک از بردبورد استفاده می‌شود؟

ساخت مدارها روی بردبورد

تا به حال، شما مدارها را با نوار یا پیچاندن پایه‌های قطعات به یکدیگر متصل کرده‌اید، اما این روش زمانی که یک مدار بیش از چند قطعه داشته باشد، چندان عملی نیست. خوشبختانه، یک بردبورد می‌تواند اتصال قطعات را آسان‌تر کند. بردبوردها دارای سوراخ‌هایی هستند که می‌توانید پایه‌های قطعات را در آن‌ها فرو کنید تا مدارها را ایجاد کنید. وقتی کارتان تمام شد، می‌توانید به راحتی تمام قطعات را جدا کرده و از آن‌ها در پروژه‌های مختلف استفاده مجدد کنید!

نحوه اتصال قطعات و سیم‌ها

در داخل یک بردبورد، صفحات فلزی سوراخ‌هایی را که در بیرون می‌بینید، با یک الگوی خاص به هم متصل می‌کنند. بیایید به یک بردبورد با چهار ناحیه اتصال نگاه کنیم، دو ناحیه تغذیه و دو ناحیه قطعه. در نواحی تغذیه در هر دو طرف، تمام سوراخ‌های موجود در هر ستون به هم متصل هستند. شما معمولاً طرف مثبت منبع تغذیه مدار خود (مانند باتری‌هایی که تاکنون استفاده کرده‌اید) را به ستون‌های قرمز و طرف منفی منبع تغذیه را به ستون‌های آبی وصل خواهید کرد. در سراسر این مباحث، من به ستون تغذیه‌ای که با یک خط قرمز مشخص شده است، به عنوان ستون تغذیه مثبت و به ستون تغذیه‌ای که با یک خط آبی مشخص شده است، به عنوان ستون تغذیه منفی اشاره خواهم کرد.



در نواحی قطعات، تمام سوراخ‌های موجود در هر ردیف به هم متصل هستند و ستون‌ها به هم متصل نیستند. نواحی قطعات چپ و راست از هم جدا شده‌اند به طوری که هیچ اتصالی بین آن‌ها وجود ندارد. برای مثال، سوراخ‌های A، B، C، D

و E در ردیف ۱ به هم متصل هستند و سوراخ‌های F ، G، H، I و J در ردیف ۱ نیز به هم متصل هستند، اما سوراخ‌های E و F در ردیف ۱ به هم متصل نیستند.

برای اتصال یک قطعه به بردبورد، کافی است آن را در سوراخی که می‌خواهید اتصال برقرار شود فشار دهید. برای مثال، اگر می‌خواهید یک طرف یک مقاومت را به طرف مثبت یک الای‌دی وصل کنید، فقط کافی است پایه مقاومت و پایه الای‌دی را در دو سوراخ در یک ردیف در ناحیه قطعات چپ یا راست فرو کنید. اگر دو پایه قطعه یا سیم دارید که نباید به هم متصل شوند، فقط مطمئن شوید که آن‌ها یا در ردیف‌های مختلف در ناحیه قطعات قرار دارند یا در طرف‌های مقابل ناحیه قطعات هستند.

سیم‌های مناسب برای استفاده در بردبورد

در نهایت، شما می‌خواهید یک ردیف روی بردبورد خود را به یک ردیف دیگر متصل کنید. می‌توانید از یک سیم برای ایجاد این اتصال استفاده کنید، اما همه سیم‌ها برای بردبورد مناسب نیستند. سیم باید به اندازه‌ای محکم باشد که بتوانید بدون خم شدن آن را در سوراخ فرو کنید و باید به اندازه‌ای ضخیم باشد که کاملاً داخل سوراخ بردبورد قرار بگیرد و بیرون نیفتد. سیم‌های تک‌رشته‌ای بهترین سیم‌ها برای ساخت مدارها روی بردبورد هستند زیرا به جای داشتن چندین سیم نازک که به هم پیچیده شده‌اند، یک هسته جامد در داخل دارند. ضخامت سیم مورد نیاز شما به بردبورد شما بستگی دارد، اما سیم‌هایی با قطر ۰.۰۱۶ تا ۰.۰۲۸ اینچ باید کار کنند.

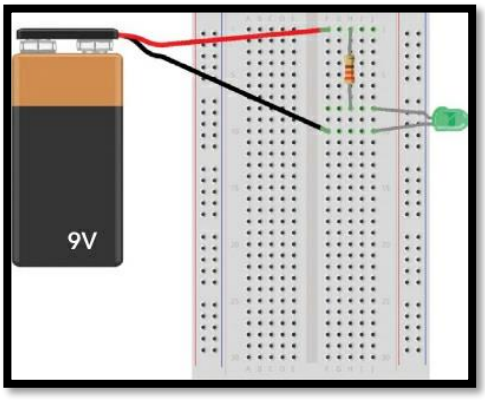
ضخامت سیم اغلب با واحد AWG (American Wire Gauge) اندازه‌گیری می‌شود و توصیه می‌کنیم از سیم‌هایی با اندازه ۲۱ تا ۲۶ AWG استفاده کنید. می‌توانید سیم‌هایی را بخرید که برای استفاده آسان با بردبورد بریده و لخت شده‌اند، یا می‌توانید با استفاده از سیم‌چین، سیم‌های خود را ببرید و لخت کنید. گزینه دیگر استفاده از سیم‌های جامپر بردبورد است. این سیم‌ها دارای سرهای محکمی هستند که اتصال آن‌ها به بردبورد بسیار آسان است. اگر قصد دارید مدارهای زیادی را روی بردبورد متصل کنید (که باید این کار را انجام دهید!)، تعدادی سیم جامپر بردبورد در دسترس داشته باشید تا کارتان آسان‌تر شود.



فعالیت کلاسی: اولین مدار روی بردبورد

وسایل مورد نیاز

- یک بردبورد
- یک باتری ۹ ولتی
- سر باتری ۹ ولتی



- یک ال‌ای‌دی
- یک مقاومت ۳۳۰ اهم
- سیم جامپر

مراحل اجرا

۱. ال‌ای‌دی را روی بردبورد قرار دهید.
۲. مقاومت ۳۳۰ اهم را به صورت سری با ال‌ای‌دی متصل کنید.
۳. باتری را با استفاده از سر باتری به مدار وصل کنید.
٤. از سیم‌های جامپر برای برقراری اتصال بین قطعات استفاده کنید.
٥. روشن شدن ال‌ای‌دی را بررسی کنید.

خلاصه درس

- بردبورد وسیله‌ای برای ساخت مدار بدون لحیم‌کاری است.
- بردبورد دارای ریل‌های تغذیه و ناحیه نصب قطعات است.
- در هر ردیف از ناحیه قطعات، پنج سوراخ به یکدیگر متصل هستند.
- برای اتصال بخش‌های مختلف مدار از سیم تک‌رشته یا سیم جامپر استفاده می‌شود.
- با استفاده از بردبورد می‌توان مدارهای مختلف را به‌راحتی ساخت، آزمایش و دوباره استفاده کرد.

تمرین

۱. بردبورد چه مزیتی نسبت به اتصال مستقیم قطعات با سیم دارد؟

.....

۲. ریل‌های قرمز و آبی در بردبورد چه کاربردی دارند؟

.....

۳. اگر دو قطعه را در یک ردیف از بردبورد قرار دهید، چه اتفاقی می‌افتد؟

.....

٤. چرا استفاده از سیم جامپر در بردبورد توصیه می‌شود؟

.....

درس۴: آشنایی با خازن



در پایان این درس خواهید توانست:

- خازن و کاربرد آن را توضیح دهید.
- نحوه ذخیره انرژی در خازن را بیان کنید.
- تفاوت خازن‌های قطبی و غیرقطبی را تشخیص دهید.
- واحد اندازه‌گیری ظرفیت خازن را بشناسید.
- یک مدار ساده با استفاده از خازن را مونتاژ و آزمایش کنید.

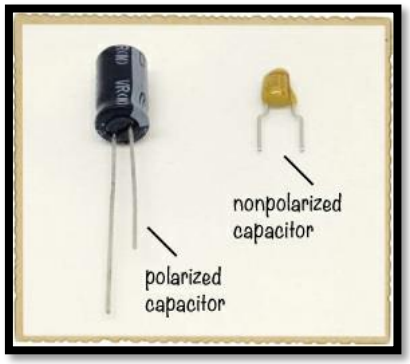
در بسیاری از مدارهای الکترونیکی، علاوه بر مقاومت و ال‌ای‌دی از قطعه‌ای به نام **خازن** استفاده می‌شود. خازن می‌تواند مقدار کمی انرژی الکتریکی را در خود ذخیره کرده و در زمان مورد نیاز آن را آزاد کند. به همین دلیل از خازن در مدارهای الکترونیکی برای ایجاد تأخیر زمانی، حذف نویز و ذخیره موقت انرژی استفاده می‌شود.

به پرسش‌های زیر پاسخ دهید.

۱. اگر برق یک مدار برای چند لحظه قطع شود، آیا می‌توان ال‌ای‌دی را برای مدت کوتاهی روشن نگه داشت؟
۲. به نظر شما چه وسیله‌ای می‌تواند انرژی الکتریکی را برای مدت کوتاه ذخیره کند؟
۳. تفاوت یک باتری با یک خازن چیست؟

آشنایی با خازن

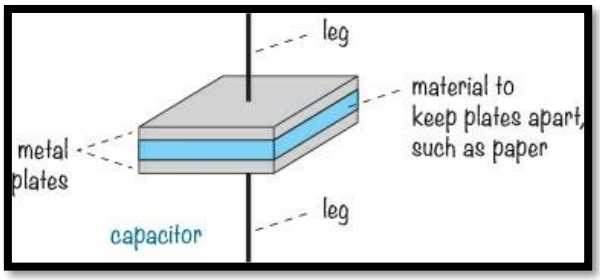
خازن مانند یک باتری قابل شارژ است؛ شما می‌توانید یک خازن را شارژ کنید و از انرژی آن برای روشن کردن چیزی استفاده کنید. اما یک باتری می‌تواند انرژی بسیار بیشتری نسبت به یک خازن ذخیره کند. یک باتری می‌تواند یک دیود نوری (ال‌ای‌دی) را به مدت چندین روز بدون نیاز به شارژ مجدد روشن نگه دارد، در حالی که بیشتر خازن‌ها حداکثر می‌توانند یک ال‌ای‌دی را برای چند ثانیه روشن نگه دارند.



خازن‌ها اغلب برای ایجاد تاخیر زمانی در یک مدار استفاده می‌شوند. به عنوان مثال، از آنجا که یک خازن انرژی را ذخیره می‌کند، می‌توان از آن برای نگه داشتن یک ال‌ای‌دی روشن برای مدت زمان کمی بیشتر استفاده کرد، حتی پس از قطع شدن برق. این ترفند ساده می‌تواند با سایر قطعات الکترونیکی نیز به کار رود تا نتایج جالبی حاصل شود.

روش کار خازن‌ها:

از داخل، خازن‌ها دستگاه‌هایی بسیار ساده هستند. آن‌ها از دو صفحه فلزی تشکیل شده‌اند که بسیار نزدیک به هم قرار دارند و معمولاً ماده‌ای مثل کاغذ در میان آن‌ها قرار دارد. برای صرفه‌جویی در فضا، این صفحات فلزی و ماده بین آن‌ها را تا می‌زنند یا می‌پیچند تا یک بسته فشرده ایجاد شود.



وقتی باتری را به دو سر یک خازن متصل می‌کنید، جریانی برقرار می‌شود که باتری سعی می‌کند الکترون‌ها را از میان خازن عبور دهد. الکترون‌ها از عایق بین دو صفحه خازن به‌صورت عادی عبور نمی‌کنند؛ اما هنگام شارژ و تخلیه، جریان در مدار برقرار است و خازن می‌تواند انرژی الکتریکی ذخیره و آزاد کند. در نهایت، یکی از صفحات دیگر نمی‌تواند الکترون بیشتری در خود نگه دارد و جریان متوقف می‌شود؛ در این لحظه می‌گوییم خازن کاملاً شارژ شده است.

اما همانطور که الکترون‌ها در یک باتری دوست ندارند با هم فشرده شوند، الکترون‌های خازن هم دوست ندارند که با هم روی یک صفحه جمع شوند. آن‌ها می‌خواهند به طرف صفحه‌ای که الکترون کمتری دارد، بروند. این موضوع یعنی شما انرژی پتانسیل الکتریکی را در خازن ذخیره کرده‌اید. اگر باتری را قطع کنید و به عنوان مثال، یک مقاومت (رزیستور) را بین دو سر خازن وصل کنید، الکترون‌های ذخیره شده روی یک صفحه شروع به حرکت می‌کنند و از طریق مقاومت به سمت صفحه‌ای که الکترون کمتری دارد جریان پیدا می‌کنند.

خازن‌های قطبی و غیرقطبی

خازن‌ها می‌توانند قطبی یا غیرقطبی باشند.مانند یک ال‌ای‌دی، یک خازن قطبی دارای یک پایه مثبت و یک پایه منفی است و باید طوری در مدار قرار گیرد که پایه مثبت آن همیشه به سمت ترمینال مثبت باتری باشد. در عکس بالا، خازن سیاه یک خازن قطبی است و پایه منفی آن با یک نوار و علامت‌های "منفی" در روی بدنه مشخص شده است.اما خازن زرد رنگ غیرقطبی است، بنابراین فرقی نمی‌کند که کدام پایه از آن به کدام طرف مدار متصل شود.

در تمام مدارهایی که از خازن استفاده می‌شود، می‌توانید از یک خازن غیرقطبی استفاده کنید، به شرطی که بتوانید خازنی با ظرفیت (خازن) مناسب پیدا کنید. ظرفیت خازنی را بر حسب فاراد (F) اندازه‌گیری می‌کنند و هرچه ظرفیت یک خازن بیشتر باشد، می‌تواند انرژی بیشتری را ذخیره کند. خازن‌های غیرقطبی با ظرفیت بالا عموماً ساخته نمی‌شوند، چون باید از نظر فیزیکی بسیار بزرگ باشند. خازن‌های قطبی این توانایی را دارند که در فضای کوچک‌تری، انرژی بیشتری ذخیره کنند، اما معایبی هم دارند؛ مهم‌ترین آن‌ها این است که باید حتماً به درستی در مدار متصل شوند.وقتی مدارهایی با خازن‌های با ظرفیت بالا می‌سازید، احتمالاً از خازن‌های قطبی استفاده خواهید کرد.همیشه مراقب باشید که پایه مثبت هر خازن قطبی به سمت نزدیک‌ترین نقطه از مدار که با قطب مثبت باتری متصل است، وصل شود.

مقادیر خازن‌ها

خازن‌هایی که در این کتاب از آن‌ها استفاده می‌شود، دارای ظرفیتی در محدوده μF (میکروفاراد) ، nF (نانوفاراد) یا pF(پیکوفاراد) هستند. ظرفیت خازن‌ها معمولاً بسیار کوچک است و اغلب با پیشوندهای "میکرو"، "نانو" و "پیکو" نوشته می‌شوند. تعریف این پیشوندها به شرح زیر است:

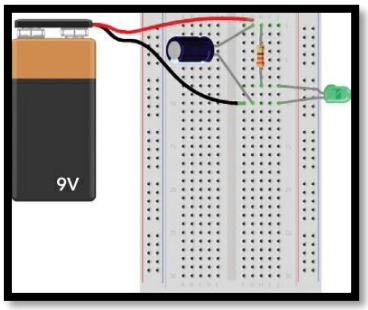
- μ(میکرو) به معنای یک میلیونیوم (10^{-6}) است، بنابراین: $۱f = ۱۰۰۰۰۰۰uf$
- n (نانو) به معنای یک میلیاردیوم (10^{-9}) است، بنابراین: $۱f = ۱۰۰۰۰۰۰۰۰۰uf$
- p (پیکو) به معنای یک تریلیونیوم (10^{-12}) است، بنابراین: $۱f = ۱۰۰۰۰۰۰۰۰۰۰۰۰uf$

خازن‌های قطبی به اندازه‌ای بزرگ هستند که مقدار ظرفیتشان را مستقیماً روی بدنه‌شان چاپ می‌کنند. اما خازن‌های غیرقطبی معمولاً کوچک‌ترند و مقادیر آن‌ها با کدهایی مانند ۱۰۴ یا ۲۰۲ مشخص می‌شود که ممکن است کمی گیج‌کننده باشند.

من خودم همیشه فراموش می‌کنم این کدها چه معنایی دارند، بنابراین وقتی لازم است معنای یک کد خاص را پیدا کنم، ساده‌ترین کار این است که در جدولی معادل آن را پیدا کنم. اما مثل بیشتر چیزهای الکترونیکی، خازن‌ها وقتی که با آن‌ها کار کنید خیلی جالب‌ترند! همین حالا پروژه بعدی را بسازید و خودتان نحوه کار یک خازن را امتحان کنید.

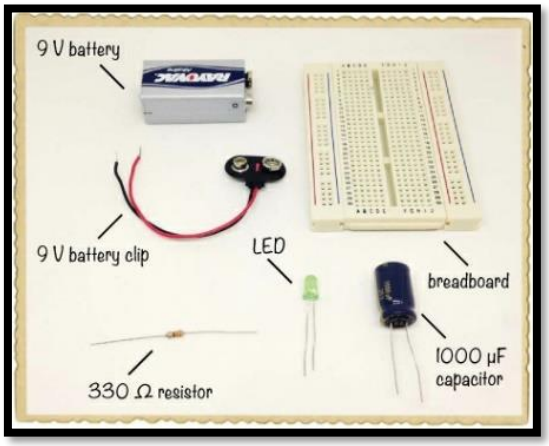
فعالیت کلاسی

تست یک خازن



این پروژه نشان می‌دهد که یک خازن قادر است انرژی را ذخیره کند. این مدار تقریباً همان مداری است که درپروژه شماره ۳: اولین مدار بردبُردی شما، با این تفاوت که این بار قرار است یک خازن به مدار اضافه کنید.همان‌طور که در تصویر بالا مشخص است، وقتی باتری را از مدار جدا کنید، متوجه خواهید شد که الای‌دی هنوز برای یک یا دو ثانیه روشن باقی می‌ماند. دلیل این موضوع این است که خازن انرژی خود را در هنگامی که باتری متصل بود ذخیره کرده بود و حالا این انرژی را برای روشن نگه داشتن الای‌دی استفاده می‌کند.

وسایل مورد نیاز برد بوردی یک باتری استاندارد ۹ ولتی برای تغذیه مدار، یک گیره باتری ۹ ولتی برای اتصال باتری به مدار، یک الای‌دی، یک مقاومت 330Ω ، خازن با ظرفیت ۱۰۰۰ میکروفاراد .



خلاصه درس

- خازن انرژی الکتریکی را به‌طور موقت ذخیره می‌کند.
- خازن نسبت به باتری انرژی بسیار کمتری ذخیره می‌کند.
- خازن‌ها در ایجاد تأخیر زمانی و ذخیره موقت انرژی کاربرد دارند.
- خازن‌ها به دو نوع قطبی و غیرقطبی تقسیم می‌شوند.
- ظرفیت خازن با واحد فاراد و واحدهای کوچک‌تر مانند میکروفاراد، نانوفاراد و پیکوفاراد اندازه‌گیری می‌شود.



تمرین

۱. خازن چه تفاوتی با باتری دارد؟

.....

۲. دو کاربرد مهم خازن را نام ببرید.

.....

۳. تفاوت خازن قطبی و غیرقطبی چیست؟

.....

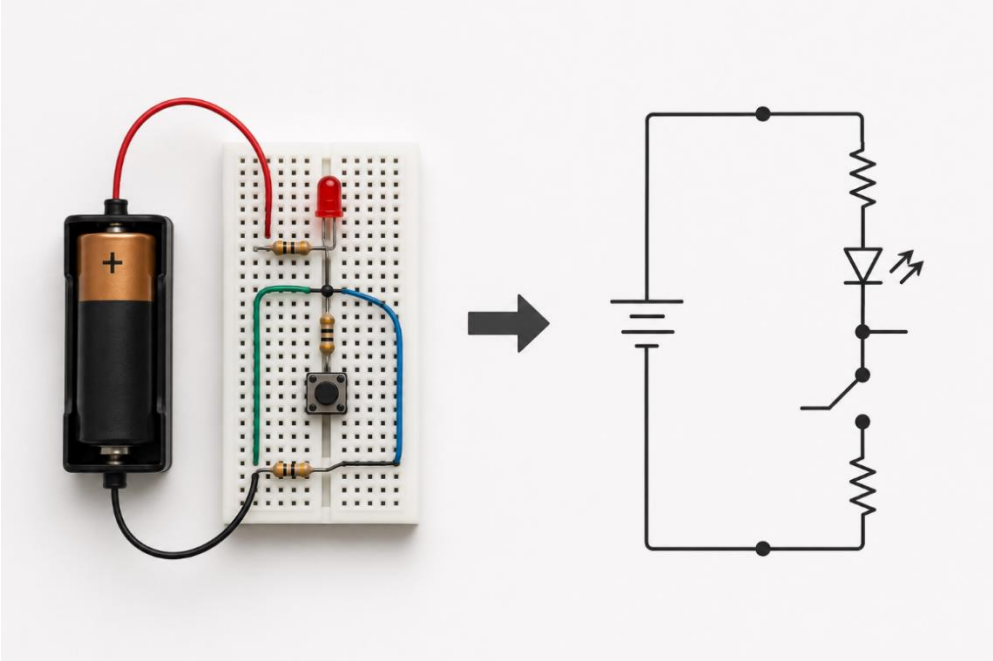
٤. ظرفیت خازن با چه واحدی اندازه‌گیری می‌شود؟

.....

٥. اگر خازن قطبی را برعکس در مدار قرار دهیم، چه مشکلی ممکن است ایجاد شود؟

.....

درس۵: توصیف مدارها با استفاده از نمادها



در پایان این درس خواهید توانست:

- مفهوم نقشه یا شماتیک مدار را توضیح دهید.
- نمادهای رایج قطعات الکترونیکی را شناسایی کنید.
- مدارهای ساده را از روی نقشه مداری مونتاژ کنید.
- اهمیت استفاده از شماتیک در طراحی و عیب‌یابی مدارها را بیان کنید.
- یک مدار چشمک‌زن ساده را روی بردبورد اجرا کنید.

با پیچیده‌تر شدن مدارهای الکترونیکی، رسم شکل واقعی قطعات دشوار و زمان‌بر می‌شود. به همین دلیل، در الکترونیک از **نقشه مداری (شماتیک)** استفاده می‌شود. در شماتیک، هر قطعه با یک نماد استاندارد نمایش داده می‌شود تا مدارها به‌صورت ساده، خوانا و قابل فهم رسم شوند.

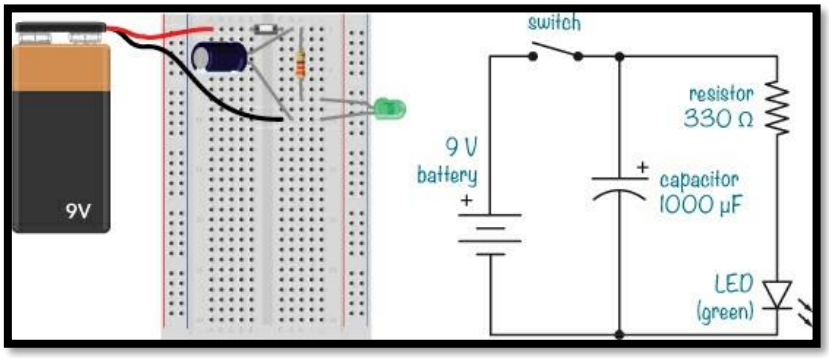
یادگیری نمادهای الکترونیکی به شما کمک می‌کند مدارهای مختلف را بخوانید، طراحی کنید و بسازید.

به پرسش‌های زیر پاسخ دهید.

۱. اگر بخواهید یک مدار بزرگ را روی کاغذ رسم کنید، آیا رسم شکل واقعی همه قطعات مناسب است؟
۲. چرا مهندسان از نمادهای استاندارد به جای شکل واقعی قطعات استفاده می‌کنند؟
۳. آیا تاکنون نقشه یک مدار الکترونیکی را دیده‌اید؟

توصیف مدارها با استفاده از نمادها

تاکنون، شما مدارهایی را تنها با تعداد کمی قطعه ساخته‌اید. برای ساخت پروژه‌های الکترونیکی جالب‌تر، معمولاً به قطعات بیشتری نیاز دارید. اما رسم دقیق هر یک از قطعات در یک مدار بزرگ به همان شکل واقعی‌شان می‌تواند پیچیده و زمان‌بر باشد. در اینجا نقشه‌های مداری (معروف به شِماتیک)، کمک ما خواهند کرد. در یک نقشه مداری، هر قطعه نماد ساده‌ی خاص خودش را دارد؛ نمادی که به شما اجازه می‌دهد تمام مدار را به سرعت رسم کنید. شبیه به کلمات و کتاب‌ها، تا زمانی که معنای نمادهای مختلف را ندانید، یک نقشه مداری ممکن است کمی پیچیده به نظر برسد. بیایید مستقیم وارد شویم و چند نماد را یاد بگیریم! در زیر یک مدار شامل یک ال‌ای‌دی، یک مقاومت و یک باتری را مشاهده می‌کنید، و در کنار آن نقشه مداری مربوطه:

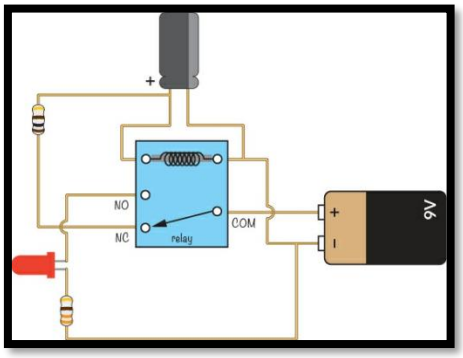


نماد باتری دارای یک علامت «+» است که محل قرارگیری پایه مثبت را نشان می‌دهد. گاهی اوقات نماد باتری بدون علامت مثبت دیده می‌شود؛ در این حالت، سمت مثبت آن، طرفی با خط بلندتر است. نماد کلید بسیار ساده است و اهمیتی ندارد که آن را در چه جهتی رسم کنید. با مقایسه این نماد با تصویر "واقعی" مدار، می‌بینید که پایه مثبت ال‌ای‌دی به مقاومت متصل شده است. مقاومت از نوع قطعات قطبی نیست؛ بنابراین نماد آن علامت جهت‌داری ندارد. همچنین انواع خازن‌ها شامل خازن‌های قطبی و غیرقطبی هستند، اما در مثال ما خازن قطبی نشان داده شده است که سمت مثبت آن با علامت «+» مشخص شده است. دوباره به مدار و نقشه مداری آن نگاه کنید – آیا فکر می‌کنید آن‌ها یکسان هستند؟ وقتی یاد بگیرید چگونه فقط با دیدن یک نقشه مداری، یک مدار را بسازید، دنیایی جدید برایتان گشوده خواهد شد. امروزه

می‌توانید تقریباً نقشه مداری هر چیزی را در اینترنت پیدا کنید، مثلاً رادیوها، پخش‌کننده‌های MP۳، رادیوهای دوطرفه (واکی‌تاکی) یا هر چیز دیگری که بخواهید بسازید!در طول این مباحث، با هر بخشی که با قطعات جدیدتری آشنا شویم، نمادهای بیشتری از مدارها را نیز به شما خواهم آموخت.

کند کردن چشمک زدن

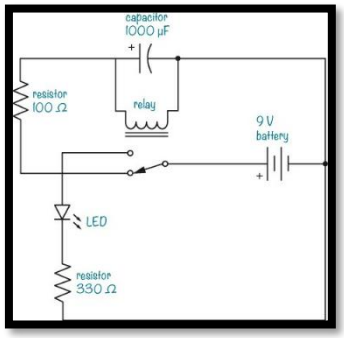
خازن در مدار ال‌ای‌دی باعث شد برای مدت کوتاهی پس از جدا شدن باتری روشن بماند. اگر یک خازن را به موازات آهنربای الکتریکی در مدار چراغ چشمک زن خیلی سریع خود وصل کنید، آهنربای الکتریکی نیز برای مدتی روشن می‌ماند. اما آهنربای الکتریکی برای مدت طولانی خاموش نمی‌ماند، بنابراین در این حالت، به نظر می‌رسد که چراغ همیشه روشن است. برای اینکه مدت بیشتری خاموش بماند، باید شارژ خازن را کند کنید تا به محض تخلیه دوباره به طور کامل شارژ نشود. برای انجام این کار، می‌توانید مقدار جریان ورودی به خازن را کاهش دهید و چگونه مقدار جریان را کاهش می‌دهید؟ با یک مقاومت! برای چشمک زدن یک چراغ ال‌ای‌دی با رله، از مداری مانند این استفاده می‌کنید، که قرار است بعداً آن را بسازیم:



فعالیت کلاسی

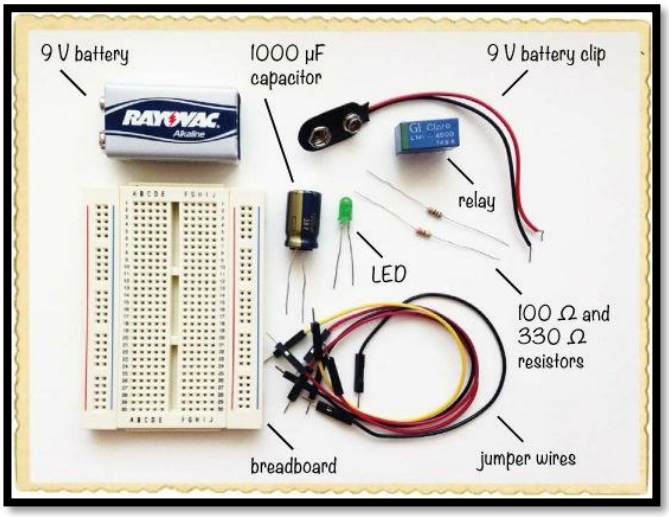
لامپی که چشمک می‌زند!

حالا وقت آن است که اولین لامپ چشمک‌زن خود را بسازید و یک ال ای دی را به صورت پالسی روشن و خاموش کنید. در ادامه، دیاگرام کامل مدار را مشاهده می‌کنید – آیا می‌توانید قطعات مختلف را تشخیص دهید؟



هنگام ساخت یک مدار الکتریکی، بهتر است آن را به شکلی ببندید که شبیه به دیاگرام مدار باشد. این کار به شما کمک می‌کند تا در صورت بروز مشکل و کار نکردن مدار، راحت‌تر بتوانید عیب یابی کنید و بفهمید چه چیزی اشتباه است. اغلب اوقات مدار اولین بار که آن را می‌بندید کار نمی‌کند و پیدا کردن اشتباهات و فهمیدن اینکه چگونه آن‌ها را رفع کنید، جزئی از فرآیند یادگیری است!

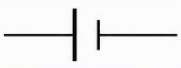
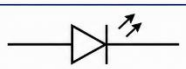
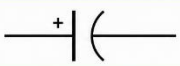
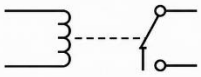
وسایل مورد نیاز



- برد برد(Breadboard)
- سیم‌های جامپر برای برد برد
- باتری ۹ ولتی استاندارد
- کلیپ باتری ۹ ولتی
- رله DPDT یاSPDT
- یک ال ای دی) الای‌دی (استاندارد
- خازن قطبی ۱۰۰۰ میکروفارادی (µF)
- مقاومت ۱۰۰ اهمی(Ω)
- مقاومت ۳۳۰ اهمی(Ω)

نمادهای پرکاربرد قطعات الکترونیکی

در نقشه‌های شماتیک، برای نمایش قطعات الکترونیکی از نمادهای استاندارد استفاده می‌شود. آشنایی با این نمادها به شما کمک می‌کند مدارها را به‌درستی بخوانید، تحلیل کنید و آن‌ها را روی بردبورد یا برد مدار چاپی (PCB) پیاده‌سازی کنید. در جدول زیر، تصویر واقعی هر قطعه در کنار نماد شماتیک و نام آن آورده شده است تا بتوانید ارتباط بین قطعه واقعی و نماد آن را به‌خوبی درک کنید.

نام قطعه	نماد شماتیک	تصویر قطعه
باتری		
مقاومت		
دیود نورانی		
کلید		
خازن		
رله		

خلاصه درس

- مدارهای الکترونیکی با استفاده از شماتیک نمایش داده می‌شوند.
- هر قطعه الکترونیکی دارای نماد استاندارد مخصوص خود است.
- شماتیک، ساخت و عیب‌یابی مدار را آسان‌تر می‌کند.
- مقاومت و خازن در مدارهای چشمک‌زن نقش مهمی در کنترل زمان روشن و خاموش شدن ال‌ای‌دی دارند.
- هنگام مونتاژ مدار بهتر است چیدمان قطعات مشابه نقشه مداری باشد.

تمرین

۱. شماتیک مدار چیست و چه کاربردی دارد؟

.....

۲. چرا در نقشه‌های مداری از نمادها به جای شکل واقعی قطعات استفاده می‌شود؟

.....

۳. نام چهار قطعه الکترونیکی و کاربرد هر یک را بنویسید.

.....

٤. چرا هنگام ساخت مدار باید به شماتیک توجه کنیم؟

.....

٥. مقاومت در مدار چشمک‌زن چه نقشی دارد؟

.....

بخش دوم: نرم افزار

SoftWare

درس ۶: مرور مبانی بینایی ماشین



در پایان این درس خواهید توانست:

- ساختار تصویر و فریم را در بینایی ماشین توضیح دهید.
- مختصات تصویر را به‌درستی به کار ببرید.
- تصاویر رنگی و خاکستری را از یکدیگر تشخیص دهید.



وقتی انسان به دست خود نگاه می‌کند، فوراً می‌تواند آن را تشخیص دهد؛ اما رایانه این توانایی را به‌صورت طبیعی ندارد. برای رایانه، تصویر فقط مجموعه‌ای از عددهاست. هر پیکسل یک مقدار دارد و کنار هم قرار گرفتن این پیکسل‌ها، تصویر را می‌سازد. برای اینکه رایانه بتواند در آینده دست را تشخیص دهد، ابتدا باید چند مفهوم مهم را یاد بگیرد:

۱. تصویر چگونه به برنامه می‌رسد؟
۲. هر تصویر چگونه به بخش‌های کوچک‌تر تقسیم می‌شود؟
۳. مختصات یک نقطه در تصویر چگونه مشخص می‌شود؟
٤. چرا گاهی از تصویر خاکستری استفاده می‌کنیم؟
٥. فریم چیست و چه ارتباطی با ویدئو دارد؟

این درس دقیقاً برای پاسخ دادن به همین پرسش‌ها طراحی شده است. در واقع، این درس مانند ساختن پیِ یک ساختمان است. اگر این بخش درست و محکم باشد، در جلسات بعدی می‌توانیم با اطمینان به سمت تشخیص دست، شمارش انگشتان و کنترل برنامه با حرکت دست برویم.



تصویر چیست؟

در بینایی ماشین، تصویر فقط یک عکس ساده نیست؛ بلکه یک ماتریس از پیکسل‌ها است. هر پیکسل کوچک‌ترین واحد تشکیل‌دهنده تصویر است. اگر تصویر را خیلی نزدیک ببینیم، متوجه می‌شویم که از تعداد زیادی نقطه‌ی ریز ساخته شده است. در تصویر رنگی، هر پیکسل معمولاً از سه مقدار تشکیل می‌شود:

- آبی
- سبز
- قرمز

در OpenCV این ترتیب به صورت BGR استفاده می‌شود.

فریم چیست؟

وقتی دوربین روشن است و تصویر زنده را نشان می‌دهد، در واقع در هر لحظه یک تصویر جداگانه ثبت می‌شود. به هر یک از این تصویرهای جداگانه، فریم گفته می‌شود. پس:

- تصویر = یک عکس ثابت
- فریم = یک تصویر از ویدئو یا تصویر زنده

اگر دوربین را به‌سرعت برای ما چندین تصویر پشت سر هم بفرستد، ذهن ما آن‌ها را به‌صورت حرکت پیوسته می‌بیند. به همین دلیل ویدئو مانند حرکت واقعی به نظر می‌رسد.

مختصات تصویر چگونه کار می‌کند؟

در ریاضی، برای مشخص کردن یک نقطه از مختصات استفاده می‌کنیم. در تصویر هم همین‌طور است. اما یک تفاوت مهم وجود دارد:

- مبدأ مختصات در تصویر، گوشه‌ی بالا-چپ است.

- محور x از چپ به راست زیاد می‌شود.
- محور y از بالا به پایین زیاد می‌شود.

یعنی نقطه‌ی (۰, ۰) در بالا-چپ تصویر قرار دارد.

اگر بخواهیم نقطه‌ای را در مختصات زیر مشخص کنیم:

$$(x, y) = (۲۰۰, ۱۵۰)$$

یعنی:

- ۲۰۰ پیکسل از سمت چپ جلو رفته‌ایم.
- ۱۵۰ پیکسل از بالا پایین آمده‌ایم.

این موضوع در تشخیص دست بسیار مهم است، چون بعداً باید بدانیم انگشت یا کف دست دقیقاً در کجای تصویر قرار دارد.

در پروژه تشخیص دست، برنامه باید بتواند:

- محل دست را پیدا کند،
- روی دست کادر یا نقاط مشخص کند،
- فاصله‌ی بین انگشتان را بسنجد،
- و گاهی بر اساس موقعیت دست تصمیم بگیرد.
- همه‌ی این کارها با مختصات انجام می‌شوند.

اگر برنامه نداند یک نقطه در کجای تصویر قرار دارد، نمی‌تواند دست را تحلیل کند.

تصویر رنگی و خاکستری

تصویر رنگی: تصویر رنگی همان تصویری است که در دوربین می‌بینیم. این تصویر دارای سه کانال رنگی است:

- آبی
- سبز
- قرمز

تصویر خاکستری: در تصویر خاکستری، رنگ‌ها حذف می‌شوند و فقط شدت روشنایی باقی می‌ماند. مزایای تصویر خاکستری:

- حجم اطلاعات کمتر می‌شود.
- پردازش سریع‌تر انجام می‌شود.
- در بعضی الگوریتم‌های تشخیص، کارکرد بهتری دارد.

در بسیاری از پروژه‌های بینایی ماشین، ابتدا تصویر به خاکستری تبدیل می‌شود تا الگوریتم راحت‌تر بخش‌های مهم را پیدا کند.

برنامه دریافت تصویر از دوربین

```
import cv2

cap = cv2.VideoCapture(0)

while True:
    ret, frame = cap.read()

    if not ret:
        print("دوربین در دسترس نیست")
        break

    cv2.imshow("Camera", frame)

    if cv2.waitKey(1) & 0xFF == ord('q'):
        break

cap.release()
cv2.destroyAllWindows()
```



فعالیت کلاسی

برنامه را اجرا کنید و ببینید تصویر دوربین چگونه نمایش داده می‌شود.

پرسش‌ها:

- آیا تصویر روان حرکت می‌کند؟
- اگر دست خود را جلوی دوربین ببرید چه اتفاقی می‌افتد؟
- چند فریم در هر ثانیه دیده می‌شود؟
- تغییر عنوان پنجره: در کد زیر:

```
import cv2

# تصال به دوربین۱
cap = cv2.VideoCapture(0)

# حلقه دریافت و نمایش فریم‌ها
while True:
    # خواندن یک فریم از دوربین
    ret, frame = cap.read()

    # بررسی موفق بودن دریافت فریم
    if not ret:
        print("خطا در دریافت تصویر از دوربین")
        break

    # نمایش فریم دریافتی
    cv2.imshow("Live Camera", frame)

    # خروج از برنامه با فشردن کلید q
    if cv2.waitKey(1) & 0xFF == ord('q'):
        break

# آزادسازی منابع
cap.release()
cv2.destroyAllWindows()
```

$$cv۲.imshow("Live\ Camera", frame)$$

عبارت "Live Camera" را به نام دلخواه خود تغییر دهید.

- آزمایش شماره دوربین: اگر دوربین باز نشد، عدد ۰ را به ۱ یا ۲ تغییر دهید و نتیجه را بررسی کنید.

تمرین

۱. تصویر چیست؟

.....

۲. فریم چیست؟

.....

۳. تفاوت آن‌ها در چیست؟

.....

٤. مختصات نقطه‌ای که در تصویر ۲۰۰ پیکسل از چپ و ۱۲۰ پیکسل از بالا قرار دارد را بنویسید.

.....

٥. کدی بنویسید که تصویر دوربین را نمایش دهد و با زدن کلید q بسته شود.

.....

٦. در یک پاراگراف توضیح دهید چرا پردازش تصویر خاکستری در بعضی پروژه‌ها سریع‌تر است.

.....

۷. توضیح دهید چرا cap.release() و cv۲.destroyAllWindows() مهم هستند.

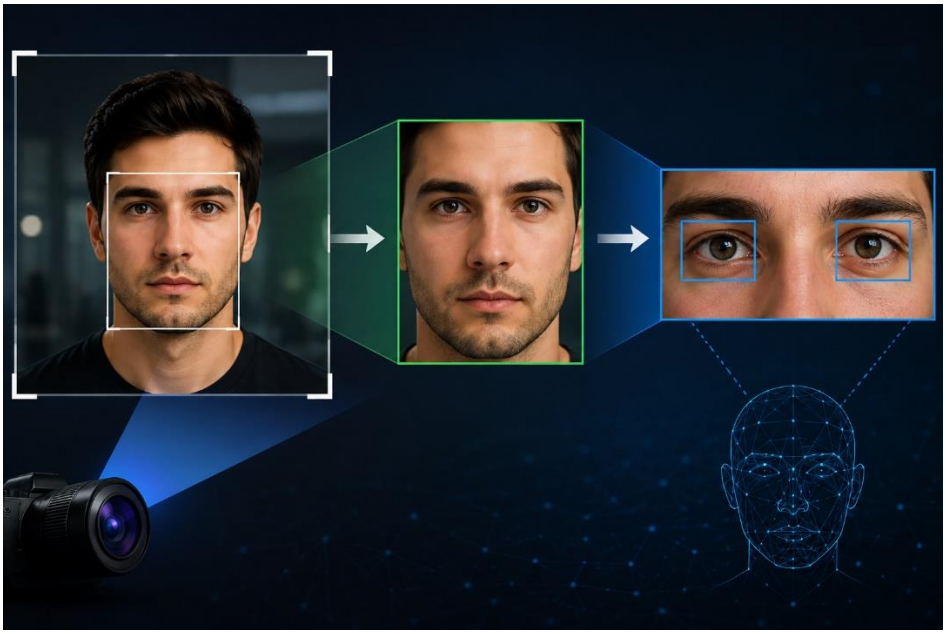
.....

خلاصه درس

- تصویر در بینایی ماشین از تعداد زیادی پیکسل تشکیل شده است و هر پیکسل بخشی از اطلاعات تصویر را نگه می‌دارد.
- در تصاویر رنگی، هر پیکسل معمولاً دارای سه کانال آبی، سبز و قرمز است و OpenCV از ترتیب BGR استفاده می‌کند.
- فریم یک تصویر جداگانه از ویدئو یا دوربین زنده است و نمایش سریع فریم‌ها باعث ایجاد حس حرکت می‌شود.
- تصویر رنگی اطلاعات سه کانال رنگی دارد، اما تصویر خاکستری فقط شدت روشنایی را نشان می‌دهد.
- تبدیل تصویر به خاکستری باعث کاهش اطلاعات، افزایش سرعت پردازش و ساده‌تر شدن بعضی الگوریتم‌های تشخیص می‌شود.

درس۷: هنر تمرکز

مدیریت ناحیه مورد علاقه (ROI) و منطق تشخیص چشم



در پایان این درس خواهید توانست:

- مفهوم ROI یا «ناحیه مورد علاقه» را در پردازش تصویر توضیح و کاربرد آن را بیان کنید.
- یک ROI ایجاد کرده و تشخیص چشم را فقط در محدوده چهره انجام دهید.


در درس گذشته یاد گرفتیم که چگونه دوربین را روشن کنیم و فریم‌ها را یکی پس از دیگری نمایش دهیم. اما یک چالش بزرگ وجود دارد: وقتی رایانه به یک تصویر نگاه می‌کند، تمام پیکسل‌ها را با یک ارزش می‌بیند. برای او، پیکسل‌های یک دیوار سفید با پیکسل‌های چشم شما تفاوتی ندارد. اگر بخواهیم در کلِ اتاق به دنبال یک شیء کوچک مثل «مردمک چشم» بگردیم، رایانه باید میلیون‌ها پیکسل را بررسی کند. این کار نه تنها سرعت برنامه را کند می‌کند، بلکه احتمال خطا را بالا می‌برد؛ چون ممکن است طرح روی پرده یا دکمه لباس شما را به اشتباه «چشم» تشخیص دهد. راه حل حرفه‌ای در بینایی ماشین، استفاده از ROI یا همان «ناحیه مورد علاقه» است.



مفهوم ROI: از کل به جزء

تصور کنید می‌خواهید در یک کتابِ بزرگ، کلمه خاصی را پیدا کنید. آیا تمام صفحات را از ابتدا تا انتها می‌گردید؟ خیر؛ ابتدا به سراغ «فصل» مربوطه می‌روید و فقط همان چند صفحه را جست‌وجو می‌کنید. در پردازش تصویر، ROI دقیقاً همین کار را انجام می‌دهد. ما به جای اینکه کل فریم دوربین را برای یافتن چشم‌ها جست‌وجو کنیم، ابتدا یک محدوده بزرگتر که حاوی چشم است (یعنی چهره) را پیدا می‌کنیم. سپس، یک «تصویرِ کوچکِ جدید» فقط از چهره می‌سازیم و جست‌وجوی چشم را به آن تصویر کوچک محدود می‌کنیم. این یعنی:

۱. دقت بیشتر: احتمال تشخیص اشتباه در محیط اطراف حذف می‌شود.
۲. سرعت بالاتر: پیکسل‌های کمتری پردازش می‌شوند.

چالش مختصات: گم شدن در نقشه اصلی

وقتی ما یک ناحیه (مثل چهره) را از تصویر اصلی جدا می‌کنیم، یک سیستم مختصات جدید متولد می‌شود. در تصویر اصلی، نقطه (۰,۰) گوشه بالا سمت چپ کل فریم است. در تصویر ROI (چهره)، نقطه (۰,۰) گوشه بالا سمت چپِ همان کادرِ چهره است. اینجاست که دانش‌آموزان معمولاً دچار خطا می‌شوند. اگر برنامه بگوید چشم در نقطه (۲۰, ۳۰) از چهره قرار دارد، و ما بخواهیم کادر چشم را روی تصویر اصلی (رنگی) رسم کنیم، نباید از عدد ۲۰ و ۳۰ استفاده کنیم؛ چون این مختصات نسبت به شروعِ چهره است. ما باید این اعداد را با مختصاتِ شروعِ خودِ چهره جمع کنیم تا به «مختصات جهانی» یا اصلی برسیم.

کد نویسی قدم به قدم

برای اینکه این مفاهیم را در کد پایتون پیاده کنیم، برنامه ما باید یک مسیر منطقی را طی کند. ما ابتدا مدل‌های آماده را بارگذاری می‌کنیم. این مدل‌ها مانند یک «کارآگاه آموزش‌دیده» هستند که یکی تخصصش پیدا کردن چهره و دیگری تخصصش پیدا کردن چشم است. در داخل حلقه اصلی برنامه، مراحل زیر به صورت زنجیروار اتفاق می‌افتد:

الف) آماده‌سازی و تشخیص چهره

ابتدا تصویر را به خاکستری تبدیل می‌کنیم (چون کارآگاه ما برای تشخیص الگوها نیازی به رنگ ندارد). سپس با تابع detectMultiScale چهره را پیدا می‌کنیم. این تابع به ما چهار عدد می‌دهد: x و y (محل شروع چهره) و w و h (عرض و ارتفاع چهره).

ب) ساخت ROI:

با استفاده از مختصاتی که از مرحله قبل گرفتیم، تصویر چهره را از فریم اصلی برش می‌زنیم. در پایتون این کار با دستور $frame[y:y+h, x:x+w]$ انجام می‌شود. حالا ما یک تصویر کوچک داریم که فقط شامل چهره است.

ج) تشخیص چشم و بازگشت به مختصات اصلی

حالا کارآگاهِ چشم را فقط روی این تصویر کوچک می‌فرستیم. وقتی او چشم را پیدا کرد، مختصات آن را نسبت به ابتدای چهره به ما می‌دهد. در نهایت، با یک جمع ساده، کادر چشم را روی همان تصویر اصلی و رنگی رسم می‌کنیم تا کاربر نتیجه نهایی را ببیند.

```
import cv2
# ۱. بارگذاری کارآگاه‌ها (مدل‌های تشخیص چهره و چشم)
face_detector = cv2.CascadeClassifier(cv2.data.haarcascades + "haarcascade_frontalface_default.xml")
eye_detector = cv2.CascadeClassifier(cv2.data.haarcascades + "haarcascade_eye.xml")

# ۲. اتصال به دوربین
cap = cv2.VideoCapture(0)

while True:
    ret, frame = cap.read()
    if not ret:
        break
    # ۳. تبدیل به خاکستری برای پردازش سریع‌تر و دقیق‌تر
    gray_frame = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)

    # ۴. مرحله اول: پیدا کردن چهره‌ها در کل تصویر
    faces = face_detector.detectMultiScale(gray_frame, 1.3, 5)

    for (x, y, w, h) in faces:
        # رسم کادر سبز دور چهره در تصویر اصلی
        cv2.rectangle(frame, (x, y), (x + w, y + h), (0, 255, 0), 2)

        # ۵. ایجاد ROI (جدا کردن ناحیه چهره)
        # ما به نسخه خاکستری برای تشخیص و نسخه رنگی برای رسم نیاز داریم
        roi_gray = gray_frame[y:y + h, x:x + w]
        roi_color = frame[y:y + h, x:x + w]

        # ۶. مرحله دوم: پیدا کردن چشم‌ها فقط در محدوده چهره (ROI)
        eyes = eye_detector.detectMultiScale(roi_gray)

        for (ex, ey, ew, eh) in eyes:
            cv2.rectangle(roi_color, (ex, ey), (ex + ew, ey + eh), (255, 0, 0), 2)

    # ۷. نمایش نتیجه نهایی
    cv2.imshow("Smart ROI Detection", frame)

    # خروج با کلید q
    if cv2.waitKey(1) & 0xFF == ord('q'):
        break
cap.release()
cv2.destroyAllWindows()
```



فعالیت کلاسی

سعی کنید کد را طوری تغییر دهید که کادر دور چشم‌ها قرمز شود و ضخامت خط کادر چهره بیشتر شود. سپس بررسی کنید اگر دستتان را روی چشمانتان بگیرید، کادرهای آبی چه تغییری می‌کنند؟

تمرین

تحقیق کنید که اگر بخواهیم به جای چشم، «لبخند» را تشخیص دهیم، باید از کدام فایل .xml استفاده کنیم؟ سعی کنید کد را برای تشخیص لبخند در ناحیه پایین چهره (یک ROI جدید در دل ROI چهره) بازنویسی کنید.

خلاصه درس

- در ROI فقط ناحیه مهم تصویر بررسی می‌شود تا سرعت و دقت برنامه افزایش یابد.
- چشم در محدوده چهره تشخیص داده می‌شود تا خطاهای احتمالی کاهش یابد.
- برای نمایش چشم روی تصویر اصلی، مختصات ROI به مختصات چشم اضافه می‌شود.

درس۸: آشنایی با تشخیص دست و ورود به دنیای MediaPipe



در پایان این درس خواهید توانست:

- مفهوم MediaPipe و کاربرد آن در تشخیص دست را توضیح دهید.
- مفهوم Landmark و ۲۱ نقطه کلیدی دست را شناسایی کنید.
- کتابخانه‌های موردنیاز برای ساخت برنامه تشخیص دست را نصب و آماده کنید.

در جلسات گذشته یاد گرفتیم که رایانه چگونه تصویر را از دوربین دریافت می‌کند، چگونه هر تصویر از میلیون‌ها پیکسل ساخته شده است و چگونه می‌توان با استفاده از مفهوم ROI فقط بخشی از تصویر را بررسی کرد. تا اینجا، تمرکز ما بیشتر روی مفاهیم پایه پردازش تصویر بود؛ یعنی اینکه رایانه «تصویر» را ببیند و بخش‌هایی از آن را تحلیل کند.

اما اکنون وارد مرحله‌ای می‌شویم که در آن رایانه فقط تصویر را نمی‌بیند، بلکه سعی می‌کند «اعضای بدن انسان» را نیز تشخیص دهد. این دقیقاً همان جایی است که هوش مصنوعی وارد ماجرا می‌شود.

پروژه اصلی ما در این دوره، ساخت یک سیستم هوشمند برای تشخیص دست و شمارش انگشتان است. بنابراین از این درس به بعد، تمرکز اصلی ما روی دست انسان خواهد بود:

۱. دست در تصویر چگونه پیدا می‌شود؟
۲. رایانه چگونه متوجه می‌شود انگشتان کجا قرار دارند؟
۳. چگونه می‌توان حرکات دست را به فرمان تبدیل کرد؟

برای رسیدن به این هدف، با یکی از مهم‌ترین ابزارهای مدرن بینایی ماشین آشنا می‌شویم: کتابخانه MediaPipe.

چرا تشخیص دست کار سختی است؟

در نگاه اول شاید تصور کنیم تشخیص دست کار ساده‌ای است؛ چون انسان‌ها خیلی راحت دست دیگران را تشخیص می‌دهند. اما برای رایانه این مسئله بسیار پیچیده است.

دست انسان:

- شکل ثابتی ندارد
- دائماً حرکت می‌کند
- می‌چرخد
- خم می‌شود
- اندازه‌اش تغییر می‌کند
- و ممکن است بخشی از آن پشت جسم دیگری قرار بگیرد.

علاوه بر این، نور محیط نیز مدام تغییر می‌کند. گاهی دست زیر نور شدید قرار دارد و گاهی در سایه دیده می‌شود. حتی رنگ پوست افراد نیز با یکدیگر متفاوت است.

اگر بخواهیم فقط با فرمول‌های ساده ریاضی یا رنگ پوست دست را پیدا کنیم، برنامه دائماً اشتباه خواهد کرد. به همین دلیل در پروژه‌های واقعی، از مدل‌های هوش مصنوعی استفاده می‌شود؛ مدل‌هایی که قبلاً با میلیون‌ها تصویر آموزش دیده‌اند.

دستیار هوشمند: ورود به دنیای هوش مصنوعی در بینایی ماشین

در سال‌های گذشته، برنامه‌نویسان مجبور بودند بسیاری از ویژگی‌های تصویر را به صورت دستی تعریف کنند. اما امروزه، هوش مصنوعی می‌تواند خودش الگوها را یاد بگیرد.

شرکت گوگل مجموعه‌ای بسیار بزرگ از تصاویر دست انسان را به یک مدل هوشمند نشان داده است:

- دست در زاویه‌های مختلف
- دست کودکان و بزرگسالان
- دست در نور کم و زیاد
- دست باز و بسته
- و حتی دست‌هایی که بخشی از آن‌ها مخفی است.

مدل هوش مصنوعی پس از مشاهده این تصاویر، یاد گرفته است که ساختار دست انسان را تشخیص دهد. نتیجه این آموزش‌ها اکنون در قالب کتابخانه‌ای به نام MediaPipe در اختیار برنامه‌نویسان قرار گرفته است.

MediaPipe چیست؟

MediaPipe یک چارچوب قدرتمند برای پردازش تصویر و هوش مصنوعی است که توسط گوگل توسعه داده شده است. این کتابخانه به برنامه‌نویسان اجازه می‌دهد تا بدون نوشتن الگوریتم‌های پیچیده یادگیری ماشین، از مدل‌های آماده هوش مصنوعی استفاده کنند.



MediaPipe می‌تواند:

- چهره را تشخیص دهد،
- بدن انسان را ردیابی کند،
- حالات صورت را تحلیل کند،
- و مهم‌تر از همه، دست و انگشتان را شناسایی کند.

مزیت اصلی MediaPipe سرعت بالای آن است. این کتابخانه به‌گونه‌ای طراحی شده که حتی روی لپ‌تاپ‌های معمولی نیز بتواند پردازش تصویر زنده را به‌صورت روان اجرا کند.

مفهوم نقطه کلیدی

در بینایی ماشین، به نقاط مهم و استاندارد روی بدن انسان یا اشیا، «Landmark» گفته می‌شود. مدل Hands برای هر دست، ۲۱ نقطه کلیدی تولید می‌کند.هر نقطه:

- شماره مخصوص خودش را دارد،
- و مختصات دقیق آن در تصویر مشخص می‌شود.

این نقاط مانند اسکلت دیجیتال دست عمل می‌کنند.



آماده‌سازی محیط برنامه‌نویسی

قبل از شروع کار، باید کتابخانه‌های موردنیاز را نصب کنیم.

$$pip\ install\ opencv - python$$
$$pip\ install\ mediapipe$$

اکنون اولین ساختار واقعی برنامه تشخیص دست را می‌بینیم. در این درس هدف ما فقط آشنایی با ساختار کلی برنامه است؛ در درس بعد تصویر زنده دوربین را به آن متصل خواهیم کرد.

```
import cv2
import mediapipe as mp

# ابزار رسم نقاط و خطوط
mp_draw = mp.solutions.drawing_utils

# مدل تشخیص دست
mp_hands = mp.solutions.hands

# راه اندازی مدل Hands
hands = mp_hands.Hands(
    static_image_mode=False,
    max_num_hands=2,
    min_detection_confidence=0.5,
    min_tracking_confidence=0.5
)

print("مدل تشخیص دست آماده است.")
```


تحلیل کـــد
درک منطق برنامه
در ابتدای برنامه، دو کتابخانه وارد می‌شوند:
import cv2
import mediapipe as mp
MediaPipe
وظیفهٔ تحلیل هوشمند تصویر را بر عهده دارد
OpenCV
مسئول دریافت تصویر از دوربین و نمایش آن
1
سپس وارد بخش مهم برنامه می‌شویم:
mp_hands = mp.solutions.hands
در اینجا به مدل مخصوص تشخیص دست دسترسی پیدا می‌کنیم.
MediaPipe
2
بعد از آن، شیء اصلی مدل ساخته می‌شود:
hands = mp_hands.Hands(...)
این شیء مغز مُتفکر برنامهٔ ماست.
شیء اصلی مدل
مغز متفکر برنامه
3


بررسی تنظیمات مدل
static_image_mode
static_image_mode=False
این گزینه مشخص می‌کند که ورودی برنامه تصویر ثابت است یا ویدئوی زنده.
وقتی مقدار آن False باشد، مدل فرض می‌کند که فریم‌ها پشت سر هم هستند و تلاش می‌کند دست را بین فریم‌ها دنبال کند.
این کار سرعت پردازش را بسیار افزایش می‌دهد.
تصویر ثابت
ویدئوی زنده
1
max_num_hands
max_num_hands =2
تعداد دست‌هایی که مدل همزمان تشخیص می‌دهد.
2
min_detection_confidence
min_detection_confidence=0.5
حداقل میزان اطمینان لازم برای تشخیص اولیهٔ دست.
3
min_tracking_confidence
min_tracking_confidence=0.5
حداقل میزان اطمینان برای ادامهٔ ردیابی دست در فریم‌های بعدی.
4
ترکیب این تنظیمات باعث می‌شود مدل بتواند دست‌ها را در ویدئوی زنده
به‌صورت سریع، دقیق و پایدار تشخیص داده و دنبال کند.

فعالیت کلاسی

روی یک برگه، شکل ساده‌ای از دست بکشید و شماره نقاط زیر را مشخص کنید:

- مچ دست
- نوک انگشت شست
- نوک انگشت اشاره
- نوک انگشت وسط
- نوک انگشت حلقه
- نوک انگشت کوچک

این فعالیت کمک می‌کند ساختار ۲۱ نقطه MediaPipe بهتر در ذهن شما شکل بگیرد.

خلاصه درس

- MediaPipe کتابخانه‌ای قدرتمند است که برای تشخیص و ردیابی دست و انگشتان استفاده می‌شود.
- مدل MediaPipe Landmark برای هر دست ۲۱ نقطه کلیدی با مختصات مشخص ایجاد می‌کند.
- نصب کتابخانه برای ساخت برنامه تشخیص دست از کتابخانه‌های OpenCV و MediaPipe استفاده می‌شود.

تمرین

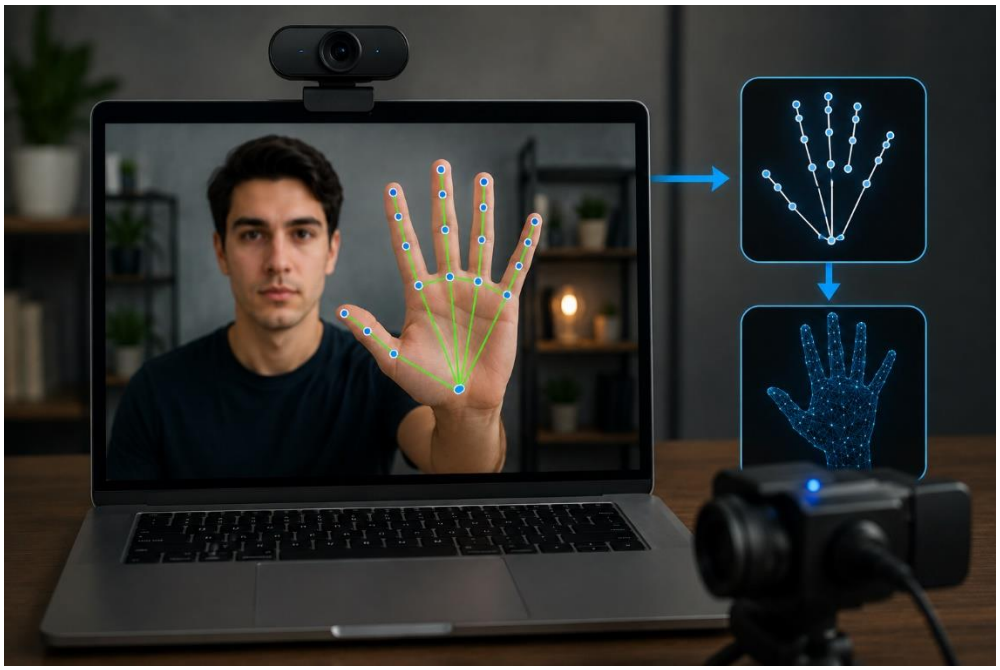
۱. کتابخانه MediaPipe را نصب کنید.

۲. کد این درس را اجرا کنید و مطمئن شوید خطایی نمایش داده نمی‌شود.

۳. تحقیق کنید:

- چرا مدل‌های هوش مصنوعی نسبت به روش‌های سنتی دقت بیشتری دارند؟
- تفاوت «تشخیص» و «ردیابی» چیست؟

درس۹: اتصال دوربین به مدل MediaPipe و تشخیص دست



در پایان این درس خواهید توانست:

- دوربین را به MediaPipe متصل کرده و تصویر زنده دریافت کنید.
- دست و نقاط کلیدی آن را در تصویر زنده تشخیص داده و رسم کنید.
- تأثیر تعداد دست‌ها و شرایط نوری را بر عملکرد تشخیص دست بررسی کنید.


در درس قبل یاد گرفتیم که MediaPipe Hands چگونه می‌تواند دست را تشخیص دهد و برای آن ۲۱ نقطه کلیدی بسازد.

اما این شناخت هنوز فقط روی کاغذ است؛ یعنی ما مدل را می‌شناسیم، ولی هنوز آن را به تصویر دوربین وصل نکرده‌ایم.

در این درس، هدف ما این است که برنامه را از حالت «آموزشی» به حالت «عملی» برسانیم.

یعنی رایانه باید بتواند:

۱. دوربین را باز کند،
۲. هر فریم را دریافت کند،
۳. آن را به MediaPipe بدهد،
٤. دست را پیدا کند،
٥. نقاط کلیدی را رسم کند،
٦. و نتیجه را همان لحظه نمایش دهد.

این همان لحظه‌ای است که دانش‌آموز حس می‌کند بینایی ماشین فقط یک مفهوم تئوری نیست، بلکه واقعاً می‌تواند با جهان واقعی ارتباط برقرار کند.



چرخه پردازش تصویر و تشخیص زنده دست باMediaPipe

۱. دوربین یک تصویر جدید می‌فرستد.
۲. برنامه آن تصویر را می‌گیرد.
۳. تصویر برای مدل هوش مصنوعی آماده می‌شود.
٤. مدل بررسی می‌کند آیا دستی وجود دارد یا نه.
٥. اگر دست پیدا شود، نقاط آن مشخص می‌شوند.
٦. این نقاط روی تصویر رسم می‌شوند.
۷. تصویر نهایی نمایش داده می‌شود.
۸. این روند برای فریم بعدی تکرار می‌شود.

به این ترتیب، برنامه مثل یک چشم زنده عمل می‌کند که مدام محیط را می‌بیند و تحلیل می‌کند.


تحلیل و توضیح کد تشخیص دست
OpenCV و MediaPipe
OpenCV
MediaPipe
ایده کلی برنامه
OpenCV تصویر را از دوربین می‌گیرد و نمایش می‌دهد.
MediaPipe: تصویر را تحلیل می‌کند و نقاط دست را پیدا می‌کند.
دوربین (OpenCV)
تحلیل هوشمند (MediaPipe)
بسرعت
نقاط و خطوط دست روی تصویر
جریان کلی برنامه
شروع
وارد کردن کتابخانه‌ها
آماده‌سازی ابزارها
باز کردن دوربین
4 ساخت مدل تشخیص دست
5 حلقهٔ اصلی برنامه
1 دریافت فریم
2 بررسی موفقیت دریافت
3 تبدیل رنگ BGR به RGB
4 ارسال فریم به مدل
5 بررسی وجود دست
6 رسم نقاط و اتصالات
7 نمایش تصویر
6 خروج از برنامه (کلید q)
7 آزادسازی منابع
پایان
1 وارد کردن کتابخانه‌ها
import cv2
import mediapipe as mp
cv2: برای کار با دوربین، تصویر و نمایش.
mediapipe: برای پردازش هوشمند تصویر.
2 آمادهٔ ابزارها
mp_draw = mp.solutions.drawing_utils
mp_hands = mp.solutions.hands
mp_draw: ابزار رسم نقطه‌ها و خطوط.
mp_hands: مدل آماده تشخیص دست.
3 باز کردن دوربین
cap = cv2.VideoCapture(0)
از دوربین شماره 0 (اولین دوربین) تصویر می‌گیرد.
4 ساخت مدل تشخیص دست
hands = mp_hands.Hands(
static_image_mode=False,
max_num_hands=2,
min_detection_confidence=0.5,
min_tracking_confidence=0.5
)
static_image_mode=False: پردازش ویدئوی زنده.
max_num_hands=2: حداکثر ۲ دست.
min_detection_confidence=0.5: حداقل اطمینان برای تشخیص اولیه (۵۰٪).
min_tracking_confidence=0.5: حداقل اطمینان برای ادامه ردیابی در فریم‌های بعدی (۵۰٪).
حلقهٔ اصلی برنامه (قلب برنامه)
1 دریافت فریم از دوربین
success, frame = cap.read()
success: آیا فریم با موفقیت گرفته شده؟
frame: تصویر دریافتی از دوربین.
2 بررسی موفقیت دریافت فریم
if not success:
print("خطا در دریافت تصویر از دوربین")
break
اگر فریم دریافت نشد، پیام خطا چاپ می‌شود و برنامه متوقف می‌شود.
3 تبدیل رنگ BGR به RGB
frame_rgb = cv2.cvtColor(frame, cv2.COLOR_BGR2RGB)
Opencv از BGR استفاده می‌کند، اما MediaPipe به RGB نیاز دارد.
4 ارسال فریم به مدل
results = hands.process(frame_rgb)
فریم به مدل داده می‌شود و مدل آن را بررسی و نقاط دست را برمی‌گرداند.
5 بررسی وجود دست
if results.multi_hand_landmarks:
اگر مدل دست را پیدا کند، وارد مرحله رسم نقاط می‌شویم.
6 رسم نقاط و اتصالات دست
for hand_landmarks in results.multi_hand_landmarks:
mp_draw.draw_landmarks(
frame,
hand_landmarks,
mp_hands.HAND_CONNECTIONS
)
نقاط کلیدی دست و خطوط اتصال بین آن‌ها روی تصویر رسم می‌شود.
7 نمایش تصویر نهایی
cv2.imshow("Hand Detection", frame)
تصویر نهایی (شامل نقاط و خطوط دست) در یک پنجره نمایش داده می‌شود.
6 خروج
if cv2.waitKey(1) & 0xFF == ord('q'):
break
اگر کاربر کلید q را فشار دهد، حلقه متوقف می‌شود و برنامه خاتمه می‌یابد.
خروج با فشردن کلید q
7 آزادسازی
آزاد کردن دوربین
cap.release()
دوربین آزاد می‌شود تا در برنامه‌های دیگر قابل استفاده باشد.
بستن تمام پنجره‌ها
cv2.destroyAllWindows()
تمام پنجره‌های باز شده بسته می‌شوند.
نکات مهم
آزادسازی منابع در پایان برنامه، باعث جلوگیری از خطا در اجراهای بعدی می‌شود.
حلقهٔ اصلی باید سریع اجرا شود تا تصویر روان و بدون لگ نمایش داده شود.
تنظیم مقادیر اطمینان بالاتر، دقت را بیشتر اما سرعت را کمتر می‌کند.
بدون تبدیل رنگ BGR به RGB، دقت مدل کاهش پیدا می‌کند.

فعالیت کلاسی

۱- برنامه را اجرا کنید و دست خود را جلوی دوربین بگیرید.

سعی کنید تشخیص دهید:

- مچ دست کدام نقطه است؟
- نوک انگشت‌ها کجا هستند؟
- کدام خطوط بین مفصل‌ها رسم شده‌اند؟

۲-مقدار max_num_hands را از ۲ به ۱ تغییر دهید و دوباره اجرا کنید.

مشاهده کنید چه تغییری در خروجی رخ می‌دهد.

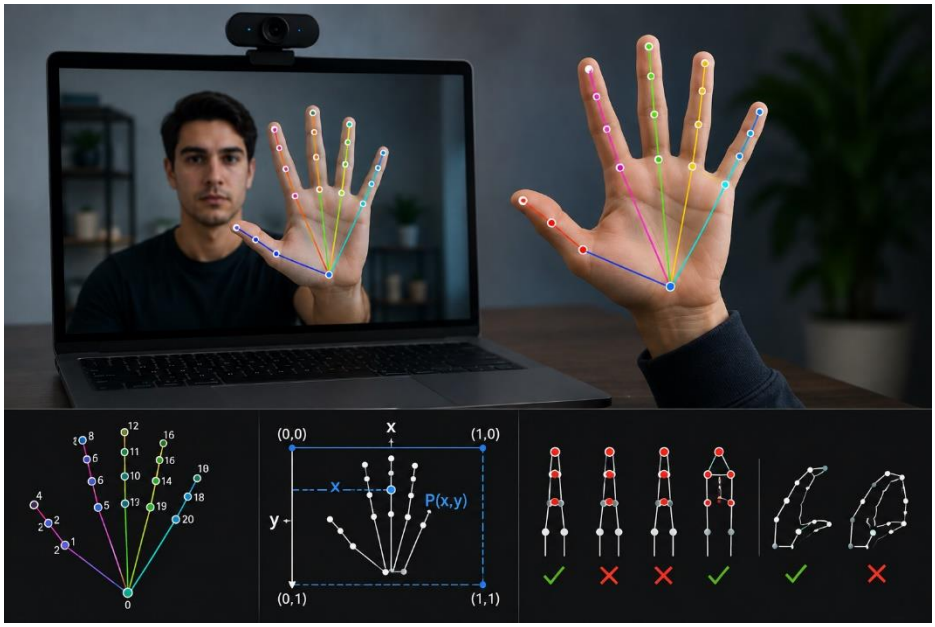
خلاصه درس

- دوربین تصویر زنده را دریافت کرده و هر فریم را به برنامه ارسال می‌کند.
- MediaPipe تصویر را بررسی کرده و دست و نقاط کلیدی آن را تشخیص می‌دهد.
- رسم نقاط :نقاط و خطوط دست روی تصویر نمایش داده می‌شوند.
- حلقه تکرار این فرایند برای هر فریم تکرار می‌شود تا تشخیص دست به‌صورت زنده انجام شود.

تمرین

۱. کد درس را اجرا کنید.
۲. دست خود را در فاصله‌های مختلف از دوربین قرار دهید.
۳. در نورهای مختلف برنامه را امتحان کنید.
٤. بررسی کنید آیا مدل در نور کم یا زیاد همچنان درست کار می‌کند یا نه.
٥. پاسخ دهید:
   - چرا تبدیل BGR به RGB لازم است؟
   - چرا حلقه while True در این برنامه ضروری است؟

درس۱۰: استخراج مختصات و منطق ریاضی تشخیص وضعیت انگشتان



در پایان این درس خواهید توانست:

- مختصات ۲۱ نقطه دست را از خروجی MediaPipe استخراج کنید.
- مختصات نسبی MediaPipe را به مختصات واقعی پیکسل تبدیل کنید.
- با استفاده از مقایسه مختصات، باز یا بسته بودن انگشتان را تشخیص دهید.



موفق شدیم دوربین را به مدل هوشمند MediaPipe Hands متصل کنیم و اسکلت دیجیتالی دست (۲۱ نقطه کلیدی) را روی تصویر زنده نشان دهیم. اما برای اینکه بتوانیم یک «دستیار هوشمند حرکتی» بسازیم، رایانه باید فراتر از رسم خطوط عمل کند؛ رایانه باید معنای موقعیت این نقاط را بفهمد. رایانه به خودی خود متوجه نمی‌شود که آیا انگشت اشاره ما باز است یا خم شده؛ او فقط تعدادی عدد به عنوان مختصات (x, y)در اختیار دارد. در این درس یاد می‌گیریم که:

۱. چگونه مختصات ریاضی این ۲۱ نقطه را از خروجی MediaPipe استخراج کنیم.
۲. چگونه تفاوت میان مختصات نسبی (اعشاری) و مختصات واقعی پیکسل‌ها را حل کنیم.
۳. با چه منطق و فرمول ریاضی ساده‌ای، باز یا بسته بودن تک‌تک انگشتان را تشخیص دهیم.

این درس، پل اصلی میان «بینایی ماشین» و «برنامه‌نویسی منطقی» برای کنترل برنامه‌ها با دست است.



راز مختصات اعشاری در MediaPipe چیست؟

وقتی MediaPipe مختصات ۲۱ نقطه دست را به ما می‌دهد، این اعداد بین ۰ تا ۱ هستند (مثلاً $x = ۰.۵۲$ و $y = ۰.۳۴$).
به این نوع نمایش، مختصات نرمال‌شده (نسبی) می‌گویند.

چرا MediaPipe مختصات را مستقیم به پیکسل نمی‌دهد؟

اندازه پنجره دوربین در سیستم‌های مختلف متفاوت است (مثلاً ۴۸۰ × ۶۴۰ یا ۷۲۰ × ۱۲۸۰). اگر MediaPipe مختصات را بر اساس پیکسل ثابت می‌داد، با تغییر اندازه پنجره، تمام محاسبات برنامه به هم می‌خورد.

وقتی اعداد بین ۰ و ۱ باشند، مستقل از اندازه تصویر هستند:

- $x = ۰$ یعنی منتهی‌الیه سمت چپ تصویر.
- $x = ۱$ یعنی منتهی‌الیه سمت راست تصویر.
- $y = ۰$ یعنی بالاترین نقطه تصویر.
- $y = ۱$ یعنی پایین‌ترین نقطه تصویر.

فرمول تبدیل به مختصات واقعی پیکسل:

برای رسم دایره یا نوشتن متن روی یک نقطه خاص، باید این نسبت اعشاری را در عرض (Width) و ارتفاع (Height) واقعی فریم ضرب کنیم:

$$\text{Pixel } X = x \times \text{Image Height}$$
$$\text{Pixel } Y = y \times \text{Image Height}$$

تحلیل اسکلت دست: انگشت باز است یا بسته؟

برای درک وضعیت انگشتان، باید به رفتار طبیعی آن‌ها هنگام باز و بسته شدن دقت کنیم.

اگر دست خود را به صورت عمودی (رو به بالا) جلوی صورت بگیرید و انگشت اشاره خود را خم کنید، چه اتفاقی می‌افتد؟

- وقتی انگشت باز است: نوک انگشت (نقطه ۸) در بالاترین وضعیت قرار دارد. یعنی روی صفحه تصویر، مقدار yنقطه ۸ کمتر از مقدار yبند دوم (نقطه ۶) است (به یاد داشته باشید که در تصویر دیجیتال، جهت محور yرو به پایین افزایش می‌یابد؛ پس نقاط بالاتر، yکمتری دارند).
- وقتی انگشت بسته (خم) می‌شود: نوک انگشت (نقطه ۸) پایین‌تر از بند دوم (نقطه ۶) قرار می‌گیرد. یعنی مقدار yنقطه ۸ بزرگتر از مقدار yنقطه ۶ می‌شود.

فرمول منطقی انگشت اشاره

$$\text{If } Y_۸ < Y_۶ \longrightarrow \text{انگشت باز است}$$

$$\text{If } Y_۸ > Y_۶ \longrightarrow \text{انگشت بسته است}$$

- این منطق برای انگشت‌های وسط (نقاط ۱۲ و ۱۰)، حلقه (نقاط ۱۶ و ۱۴) و کوچک (نقاط ۲۰ و ۱۸) نیز دقیقاً به همین ترتیب برقرار است.

انگشت شست انسان به صورت عمودی خم نمی‌شود، بلکه حرکت آن به صورت افقی (چپ و راست) است. بنابراین مقایسه مقدار yبرای شست کارایی ندارد.

- برای شست، مختصات افقی یعنی xنوک شست (نقطه ۴) را با xبند دوم شست (نقطه ۲) مقایسه می‌کنیم.
- بسته به اینکه دست چپ باشد یا دست راست، جهت باز شدن شست فرق می‌کند که در جلسات آینده آن را دقیق‌تر بررسی می‌کنیم. اما اساس کار، تمرکز روی محور xبه جای محور yاست.

کد نویسی: استخراج مختصات و بررسی وضعیت انگشت اشاره

در این کد، ما فریم دوربین را پردازش کرده، مختصات واقعی پیکسلی نقاط ۵، ۶ و ۸ را استخراج می‌کنیم و با یک دستور شرطی ساده (if)، وضعیت انگشت اشاره را روی تصویر می‌نویسیم.

```python
import cv2
import mediapipe as mp

# MediaPipe آماده‌سازی ابزارهای .۱
mp_draw = mp.solutions.drawing_utils
mp_hands = mp.solutions.hands

cap = cv2.VideoCapture(0)

# ۲: راه‌اندازی مدل
hands = mp_hands.Hands(
    static_image_mode=False,
    max_num_hands=1, # برای سادگی پردازش ، فقط یک دست
    min_detection_confidence=0.7,
    min_tracking_confidence=0.7
)

while True:
    success, frame = cap.read()
    if not success:
        break

    # دریافت واقعی وتصویر (عرض و ارتفاع)
    h, w, c = frame.shape

    # MediaPipe تبدیل برای #
    frame_rgb = cv2.cvtColor(frame, cv2.COLOR_BGR2RGB)
    results = hands.process(frame_rgb)

    if results.multi_hand_landmarks:
        for hand_landmarks in results.multi_hand_landmarks:
            # رسم کل اسکلت دست برای راهنمایی
            mp_draw.draw_landmarks(frame, hand_landmarks, mp_hands.HAND_CONNECTIONS)

            # استخراج لیست  نقاط کلیدی دست
            lm_list = hand_landmarks.landmark

            # استخراج مختصات پیکسلی نوک انگشت اشاره (نقطه ۸) و بند دوم (نقطه ۶)
            ضرب در عرض (w) و ارتفاع (h) برای تبدیل اعشار به پیکسل واقعی
            x8, y8 = int(lm_list[8].x * w), int(lm_list[8].y * h)
            x6, y6 = int(lm_list[6].x * w), int(lm_list[6].y * h)

            # رسم دایره‌های رنگی روی نقاط هدف برای وضوح بیشتر
            cv2.circle(frame, (x8, y8), 10, (0, 0, 255), cv2.FILLED) # دایره قرمز روی نوک انگشت
            cv2.circle(frame, (x6, y6), 10, (0, 255, 0), cv2.FILLED) # دایره سبز روی بند دوم

            # اعمال منطق ریاضی : مقایسه محور Y
            # اگر Y نوک انگشت کمتر ازY بند دوم باشد (یعنی بالاتر باشد)
            if y8 < y6:
                status = "Open"
                color = (0, 255, 0) # سبز
            else:
                status = "Closed"
                color = (0, 0, 255) # قرمز

            # نمایش وضعیت روی صفحه
            cv2.putText(frame, f"Index Finger: {status}", (10, 70),
                        cv2.FONT_HERSHEY_SIMPLEX, 1, color, 3)

    cv2.imshow("Finger Detection Logic", frame)

    if cv2.waitKey(1) & 0xFF == ord('q'):
        break
cap.release()
cv2.destroyAllWindows()
```

فعالیت کلاسی

۱. برنامه را اجرا کنید و دست خود را مقابل دوربین قرار دهید. انگشت اشاره را خم و راست کنید و مطمئن شوید که متن روی صفحه در لحظه بین "Open" و "Closed" تغییر می‌کند.

۲. با تغییر موقعیت دست خود به حالت افقی (خوابیده)، بررسی کنید که آیا منطق مقایسه yهمچنان درست کار می‌کند یا خیر؟ در مورد چرخش دست چه چالشی وجود دارد؟ بحث کنید.

خلاصه درس

- مختصات نرمال‌شده MediaPipe موقعیت نقاط دست را به‌صورت اعداد بین ۰ و ۱ نمایش می‌دهد.
- مختصات پیکسلی با تبدیل مختصات نسبی به اندازه واقعی تصویر برای نمایش نقاط روی تصویر به دست می‌آیند.
- تشخیص انگشت با مقایسه مختصات نقاط کلیدی می‌تواند باز یا بسته بودن انگشتان را مشخص کند.
- تشخیص شست با بررسی موقعیت افقی آن و مقایسه مختصات محور X انجام می‌شود.

تمرین

کد این درس را روی رایانه خود پیاده‌سازی و تست کنید.

بر اساس منطق ارائه شده برای انگشت اشاره، کد را به گونه‌ای توسعه دهید که علاوه بر انگشت اشاره، وضعیت انگشت وسط (نقاط ۱۲ و ۱۰) را نیز بررسی کرده و وضعیت آن را هم روی صفحه چاپ کند.

بررسی کنید و در یک پاراگراف بنویسید که چگونه می‌توانیم از باز بودن همزمان انگشت اشاره و وسط (حالت علامت V یا پیروزی) برای انجام یک کار خاص در رایانه استفاده کنیم؟

درس۱۱: توسعه منطق تشخیص به تمام انگشتان و ساخت شمارشگر



در پایان این درس خواهید توانست:

- وضعیت باز یا بسته بودن هر پنج انگشت را با استفاده از نقاط کلیدی دست تشخیص دهید.
- وضعیت انگشتان را در یک لیست ذخیره کرده و تعداد انگشتان باز را شمارش کنید.
- محدودیت روش شمارش انگشتان در هنگام چرخش دست را توضیح دهید.


ما فهمیدیم چگونه می‌توان از میان ۲۱ نقطه کلیدی دست، دو نقطه خاص را انتخاب کرد و با مقایسه ساده ارتفاع آن‌ها روی محور y، وضعیت باز یا بسته بودن یک انگشت را تشخیص داد. ما این کار را فقط برای انگشت اشاره انجام دادیم و نتیجه را روی صفحه به صورت متن «Open» یا «Closed» نمایش دادیم. اما در دنیای واقعی، وقتی دست را جلوی دوربین می‌گیریم، معمولاً فقط یک انگشت نیست که حرکت می‌کند. ما برای نشان دادن اعداد، برای دادن فرمان و برای ارتباط با رایانه، از ترکیب چند انگشت استفاده می‌کنیم. مثلاً نشان دادن عدد ۲ یعنی انگشت اشاره و وسط هم‌زمان باز باشند، یا نشان دادن عدد ۵ یعنی تمام انگشتان باز باشند. پس برنامه ما باید از حالت «بررسی یک انگشت» خارج شود و بتواند تمام پنج انگشت را در همان لحظه بررسی کند و در نهایت یک عدد کلی به ما بدهد: «چند انگشت باز است؟» این یعنی ما در این درس، اولین پروژه نیمه‌کامل بینایی ماشین خود را می‌سازیم: شمارشگر انگشتان.

اگر ارتفاع نوک انگشت (نقطه ۸) کمتر از ارتفاع بند دوم (نقطه ۶) باشد، یعنی نوک انگشت بالاتر قرار دارد و انگشت باز است. این منطق عیناً برای سه انگشت دیگر هم صادق است:

$$y_{tip} < y_{pip} \Rightarrow \text{انگشت باز است}$$

$$y_{tip} > y_{pip} \Rightarrow \text{انگشت بسته است}$$


مشکل انگشت شست: چرا منطق متفاوت دارد؟

اگر به دست خود نگاه کنید، متوجه می‌شوید که انگشت شست رفتار متفاوتی دارد. وقتی شست را باز و بسته می‌کنید، حرکت آن بیشتر در جهت افقی است، نه عمودی. یعنی نوک شست به سمت چپ یا راست حرکت می‌کند، نه به سمت بالا یا پایین. پس مقایسه محور yبرای شست کار نمی‌کند. در عوض باید محور xرا مقایسه کنیم. منطق شست برای دست راست:

وقتی دست راست را به صورت کفِ دست به سمت دوربین نگه می‌دارید:

- اگر شست باز باشد، نوک شست (نقطه ۴) در سمت چپ‌تر از بند شست (نقطه ۳) قرار می‌گیرد.یعنی مقدار x_4بزرگ‌تر از x_3می‌شود.
- اگر شست بسته باشد، نوک شست به سمت کف دست نزدیک می‌شود و x_4کوچک‌تر از x_3می‌شود.

توجه: این منطق فقط برای دست راست به این شکل است. اگر دست چپ جلوی دوربین باشد، جهت باز شدن شست برعکس می‌شود. در این مرحله از دوره، ما فرض می‌کنیم که کاربر دست راست را به دوربین نشان می‌دهد. در جلسات آینده، وقتی که سیستم پیچیده‌تر شود، تشخیص دست چپ و راست را هم اضافه خواهیم کرد.

استفاده از لیست برای ذخیره وضعیت انگشتان

اگر بخواهیم وضعیت ۵ انگشت را ذخیره کنیم، می‌توانیم ۵ متغیر جداگانه بسازیم:

```
thumb = 0
index = 0
middle = 0
ring = 0
pinky = 0
```

اما این روش زمانی که تعداد انگشتان زیاد شود، گیج‌کننده است. راه بهتر این است که از لیست استفاده کنیم:

```
fingers_status = []
```

در این لیست، ما برای هر انگشت یک عدد می‌گذاریم:

- عدد ۱ یعنی انگشت باز است.
- عدد ۰ یعنی انگشت بسته است.

مثلاً اگر فقط انگشت اشاره و وسط باز باشند، لیست ما این‌طور می‌شود:

```
# شست=۰ , اشاره=۱ , وسط=۱ , حلقه=۰ , کوچک=۰
fingers_status = [0, 1, 1, 0, 0]
```

حالا برای اینکه بفهمیم چند انگشت باز است، کافی است تعداد عددهای ۱ در این لیست را بشماریم. پایتون برای این کار یک تابع خیلی راحت دارد:

```
total = fingers_status.count(1)
# نتیجه: ۲
```

یعنی ۲ انگشت باز است. همین!

کد شمارشگر انگشتان :

بخش ۱: آماده‌سازی ابزارها

```
import cv2
import mediapipe as mp

mp_draw = mp.solutions.drawing_utils
mp_hands = mp.solutions.hands
```

در این بخش ابزارهای مورد نیاز از کتابخانه‌های MediaPipe و OpenCV بارگذاری می‌شوند.

بخش ۲: باز کردن دوربین

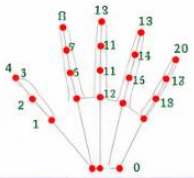
```
cap = cv2.VideoCapture(0)
```

دوربین پیش‌فرض سیستم (0) باز می‌شود.



بخش ۳: ساخت مدل تشخیص دست

```
hands = mp_hands.Hands(
  static_image_mode=False,
  max_num_hands=1,
  min_detection_confidence=0.7,
  min_tracking_confidence=0.7
)
```

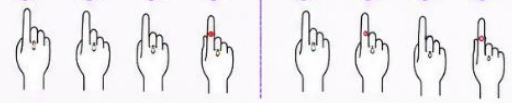


شماره نقاط بند دوم (PIP) برای همان انگشتان

اشاره	وسط	حلقه	کوچک
6	10	14	18

شماره نقاط نوک انگشتان (برای ۴ انگشت معمولی)

اشاره	وسط	حلقه	کوچک
8	12	16	20



بخش ۴: حلقهٔ اصلی برنامه

اگر دستی شناسایی شد

۱. دریافت فریم از دوربین

```
success, frame = cap.read()
if not success:
  print("خطا در دریافت تصویر")
  break
```

۲. دریافت ابعاد تصویر

```
h, w, c = frame.shape
```

۳. تبدیل رنگ و پردازش توسط MediaPipe

```
frame_rgb = cv2.cvtColor(frame,
            cv2.COLOR_BGR2RGB)
results = hands.process(frame_rgb)
```

۱. رسم اسکلت دست

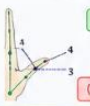


```
mp_draw.draw_landmarks(
  frame,
  hand_landmarks,
  mp_hands.HAND_CONNECTIONS
)
```

۲. بررسی وضعیت انگشتان

الف) انگشت شست (منطق افقی)

اگر نقطه 2 x 3 نقطه 3



شست باز (1)

در غیر این صورت

شست بسته (0)

ب) چهار انگشت دیگر (منطق عمودی)

برای هر انگشت:

اگر y نوک انگشت < y بند دوم

انگشت باز (1)

در غیر این صورت

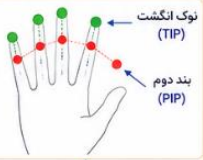
انگشت بسته (0)

۳. شمارش انگشتان باز

```
total_fingers = fingers_status.count(1)
```

انگشت باز

انگشت بسته

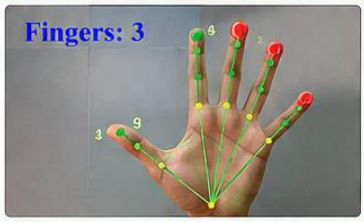


رسم دایره روی نوک انگشت

سبز (باز)

قرمز (بسته)

بخش ۵: نمایش نتیجه روی تصویر




```
cv2.putText(
  frame,
  f"Fingers: {total_fingers}",
  (30, 80),
  cv2.FONT_HERSHEY_SIMPLEX,
  2, (255, 0, 0), 4
)
```

تعداد انگشتان باز در گوشه بالا چپ تصویر نمایش داده می‌شود.

تصویر نهایی نمایش داده می‌شود.

خروج از برنامه

با فشردن کلید q برنامه متوقف می‌شود.

بخش ۶: خروجی اطلاعات

```
cap.release()
cv2.destroyAllWindows()
```

آزاد کردن دوربین

بستن تمام پنجره‌ها و پایان برنامه

احتمالاً وقتی برنامه را اجرا می‌کنید، متوجه می‌شوید که گاهی عدد روی صفحه لرزشی است. مثلاً وقتی ۳ انگشت باز دارید، گاهی عدد برای چند فریم به ۲ یا ۴ می‌پرد و دوباره برمی‌گردد. این به دلیل نویز تصویر و نوسان نور در دوربین است. مدل در هر فریم تصویر را از صفر تحلیل می‌کند و گاهی در تشخیص دقیق محل یک نقطه دچار خطای جزئی می‌شود.

راه‌حل ساده در این مرحله:
فعلاً نیازی نیست این مشکل را حل کنیم. این رفتار طبیعی سیستم‌های بینایی ماشین در سطح مقدماتی است. در جلسات آینده، وقتی که پروژه‌های پیچیده‌تری بسازیم، تکنیک‌هایی برای «هموارسازی» (Smoothing) خروجی خواهیم آموخت.

- ❖ دقت کنید که در این کد، ما فرض کردیم دست رو به بالا نگه داشته می‌شود. یعنی نوک انگشتان به سمت بالای تصویر اشاره می‌کند.

سؤال: اگر کاربر دست خود را ۹۰ درجه بچرخاند و انگشتان به سمت راست یا چپ اشاره کنند، آیا منطق مقایسه محور yهنوز کار می‌کند؟

پاسخ: خیر. وقتی دست افقی می‌شود، باز و بسته شدن انگشتان دیگر روی محور yاتفاق نمی‌افتد، بلکه روی محور xرخ می‌دهد. این یکی از محدودیت‌های مهم این روش است. در جلسات آینده خواهیم آموخت که چگونه با محاسبه «زاویه مچ دست» می‌توانیم این مشکل را حل کنیم و برنامه را مقاوم در برابر چرخش دست کنیم.



فعالیت کلاسی

۱- تست اعداد

برنامه را اجرا کنید و اعداد ۰ تا ۵ را به ترتیب به دوربین نشان دهید (از مشت بسته تا دست کاملاً باز). ثبت کنید که برنامه در چه اعدادی دچار خطا می‌شود و در چه اعدادی کاملاً دقیق است.

۲- مشاهده دایره‌های رنگی

دقت کنید که دایره‌های سبز و قرمز روی نوک انگشتان به‌درستی رسم می‌شوند یا خیر. آیا رنگ هر انگشت با وضعیت واقعی آن هماهنگ است؟

۳- تست دو دست

مقدار max_num_hands را از ۱ به ۲ تغییر دهید و دو دست را جلوی دوربین بگیرید. چه اتفاقی می‌افتد؟ آیا برنامه هر دو دست را به‌درستی می‌شمارد؟ (توجه: کد فعلی فقط یک دست را می‌شمارد. فکر کنید چرا؟)

خلاصه درس

- تشخیص انگشتان با مقایسه نقاط کلیدی می‌توان وضعیت باز یا بسته بودن پنج انگشت را مشخص کرد.
- انگشت شست به دلیل حرکت افقی با مقایسه مختصات محور x تشخیص داده می‌شود.
- ذخیره وضعیت انگشتان با استفاده از عدد ۱ برای انگشت باز و عدد ۰ برای انگشت بسته انجام می‌شود.
- شمارش انگشتان با شمارش تعداد عددهای ۱ در لیست انجام می‌شود.
- محدودیت برنامه زمانی ایجاد می‌شود که دست بچرخد و انگشتان دیگر در جهت عمودی قرار نداشته باشند.

تمرین

۱. کد این درس را روی رایانه خود پیاده‌سازی کنید و نتیجه را تست کنید.

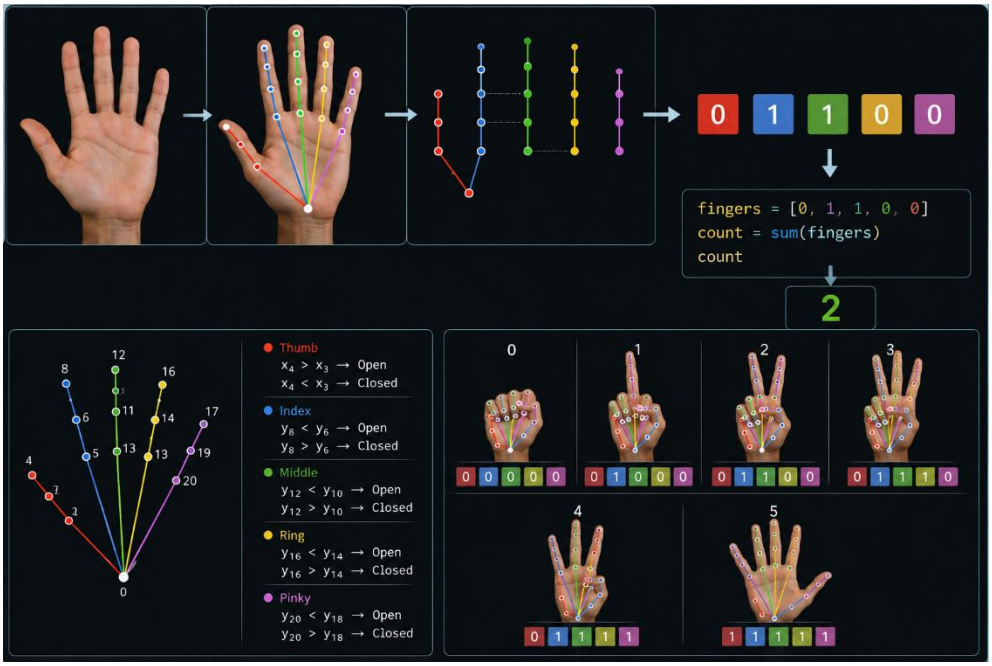
۲. یک عکس از خروجی برنامه بگیرید وقتی که ۳ انگشت باز دارید و دایره‌های سبز و قرمز روی انگشتان دیده می‌شوند.

۳. متن نمایش‌داده‌شده روی تصویر را تغییر دهید تا به‌جای Fingers: 3، بنویسد تعداد انگشتان باز: ۳. (نکته: برای متن فارسی به زودی راه‌حل ارائه خواهیم داد؛ فعلاً با همین متن انگلیسی تمرین کنید.)

چالش

اگر بخواهیم برنامه به جای شمارش انگشتان، فقط تشخیص دهد که کاربر «مشت بسته» (۰ انگشت باز) یا «دست باز» (۵ انگشت باز) نشان می‌دهد، چگونه می‌توانیم کد را ساده‌تر کنیم؟

درس۱۲: الگوریتم شمارش انگشتان



در پایان این درس خواهید توانست:

- وضعیت انگشتان را به‌صورت صفر و یک در یک لیست ذخیره کنید.
- با استفاده از تابع sum و متد count تعداد انگشتان باز را محاسبه کنید.
- تعداد انگشتان باز را به یک خروجی عددی و قابل استفاده در برنامه تبدیل کنید.

در درس قبل، وضعیت هر انگشت به‌صورت جداگانه بررسی شد و نتیجه آن در یک لیست ذخیره گردید. به این ترتیب، برنامه می‌توانست مشخص کند که هر انگشت باز است یا بسته. خروجی این مرحله معمولاً به صورت مجموعه‌ای از صفر و یک نمایش داده می‌شود. در این نمایش، هر عدد نشان‌دهنده وضعیت یکی از انگشتان دست است. عدد ۱ به معنای باز بودن انگشت و عدد ۰ به معنای بسته بودن آن است. بنابراین در مثال بالا، دو انگشت باز هستند.

در این درس، هدف آن است که این داده خام به یک خروجی معنادارتر تبدیل شود؛ یعنی برنامه بتواند از روی وضعیت انگشتان، تعداد انگشتان باز را محاسبه کند. این مرحله از نظر آموزشی اهمیت زیادی دارد، زیرا دانش‌آموز در اینجا یاد می‌گیرد که چگونه داده‌های استخراج‌شده از تصویر را به یک نتیجه عددی و قابل استفاده تبدیل کند.

مفهوم شمارش انگشتان

فرض کنید برنامه پس از بررسی دست، لیست زیر را ساخته است:

$$fingers = [۰, ۱, ۱, ۰, ۰]$$

این لیست یعنی:

- شست بسته است،
- انگشت اشاره باز است،
- انگشت وسط باز است،
- انگشت حلقه بسته است،
- انگشت کوچک بسته است.

پس اگر بخواهیم بدانیم چند انگشت باز است، کافی است تعداد ۱ها را بشماریم. از آنجا که هر انگشت باز با ۱ نمایش داده شده، جمع کردن اعضای لیست نیز همین نتیجه را می‌دهد:

$$۰ + ۱ + ۱ + ۰ + ۰ = ۲$$

بنابراین تعداد انگشتان باز برابر با ۲ خواهد بود. این ایده بسیار مهم است، زیرا نشان می‌دهد که چگونه می‌توان اطلاعات بینایی ماشین را به داده‌ای ساده و قابل پردازش تبدیل کرد.

یکی از ساده‌ترین راه‌ها برای شمارش انگشتان باز، استفاده از تابع sum() است. این تابع مجموع اعضای یک لیست عددی را محاسبه می‌کند.

```
fingers = [0, 1, 1, 0, 0]
count = sum(fingers)

print(count)
```

علت این نتیجه روشن است، زیرا sum()همه اعضای لیست را با هم جمع می‌کند و در لیستی که فقط از صفر و یک تشکیل شده باشد، حاصل جمع دقیقاً برابر با تعداد انگشتان باز(۲) خواهد بود.

روش دیگری که می‌تواند همین کار را انجام دهد، استفاده از متد count() است. این متد تعداد تکرار یک مقدار مشخص را در لیست می‌شمارد. اگر بخواهیم تعداد ۱ها را بشماریم، می‌نویسیم:

```
fingers = [0, 1, 1, 0, 0]
count = fingers.count(1)

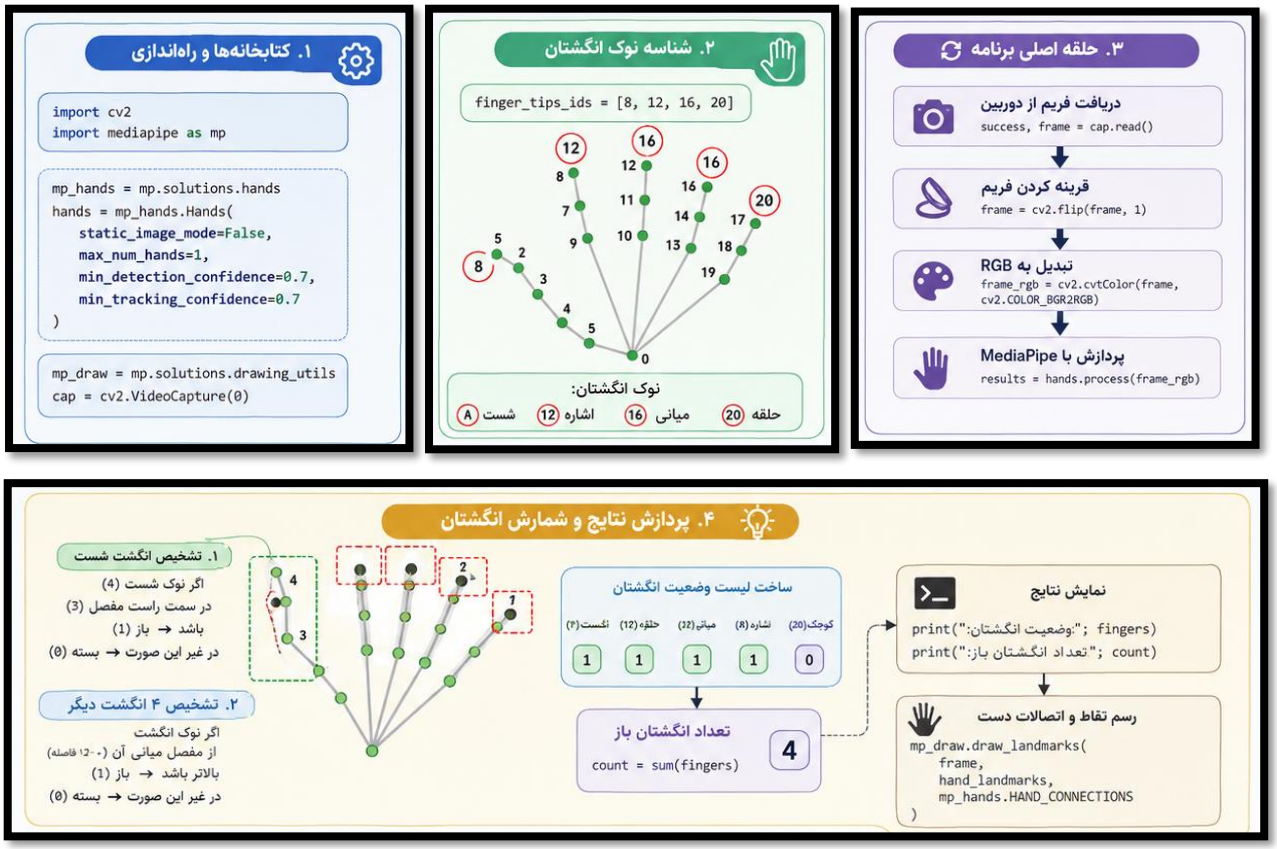
print(count)
```

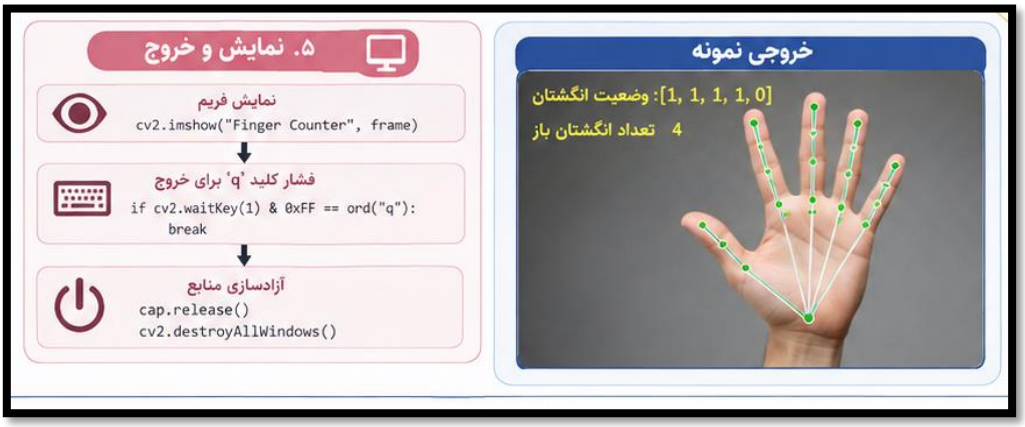
در اینجا، برنامه مستقیماً تعداد عضوهایی را می‌شمارد که مقدارشان ۱ است.

مقایسه دو روش

هر دو روش نتیجه یکسانی تولید می‌کنند ، با این حال، در این پروژه معمولاً sum() انتخاب مناسب‌تری است، زیرا دانش‌آموز به‌خوبی می‌بیند که «جمع صفر و یک‌ها» چگونه به تعداد انگشتان باز تبدیل می‌شود.

اکنون باید این ایده را به برنامه تشخیص دست متصل کنیم. برنامه ابتدا وضعیت هر انگشت را در لیست fingers ذخیره می‌کند و سپس تعداد انگشتان باز را محاسبه می‌نماید.





فعالیت کلاسی

۱. برنامه را اجرا کنید و دست خود را به صورت مشت کامل در برابر دوربین قرار دهید. بررسی کنید که آیا خروجی برابر با ۰ می‌شود یا نه.

۲. فقط یک انگشت خود را باز نگه دارید. ببینید آیا برنامه عدد ۱ را نمایش می‌دهد.

۳. دو، سه، چهار و پنج انگشت را به‌ترتیب باز کنید و خروجی برنامه را یادداشت نمایید.

٤. خط زیر را:

$$count = sum(fingers)$$

با این خط جایگزین کنید:

$$count = fingers.count(۱)$$

سپس برنامه را دوباره اجرا کنید و نتیجه‌ها را مقایسه نمایید.

خلاصه درس

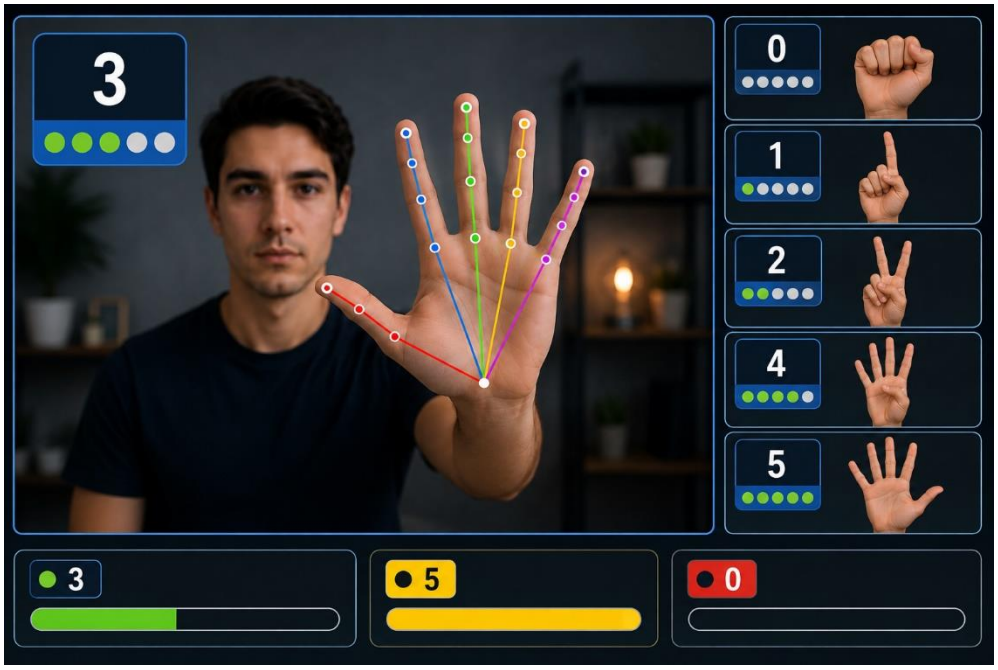
- وضعیت انگشتان هر انگشت با عدد ۱ برای باز و عدد ۰ برای بسته در لیست ذخیره می‌شود.
- شمارش انگشتان با جمع کردن عددهای لیست می‌تواند تعداد انگشتان باز را مشخص کند.
- تابع sum مجموع صفر و یک‌های لیست را محاسبه می‌کند و تعداد انگشتان باز را به دست می‌آورد.
- متد count تعداد عددهای ۱ را در لیست می‌شمارد و تعداد انگشتان باز را مشخص می‌کند.
- حلقه for می‌تواند بدون استفاده از sum تعداد انگشتان باز را محاسبه کند.

پروژه

برنامه‌ای بنویسید که اگر تعداد انگشتان باز برابر با ۰ بود، پیام «دست بسته است» را چاپ کند و اگر برابر با ۵ بود، پیام «همه انگشتان باز هستند» را نشان دهد.

بدون استفاده از ()sum، تعداد انگشتان باز را با استفاده از یک حلقه for محاسبه کنید.

درس۱۳: طراحی رابط کاربری گرافیکی (GUI) و نمایش اطلاعات روی تصویر



در پایان این درس خواهید توانست:

- سیستم مختصات تصویر را در OpenCV برای رسم عناصر گرافیکی استفاده کنید.
- با استفاده از توابع cv.rectangle و cv.putText روی تصویر مستطیل و متن رسم کنید.
- تعداد انگشتان باز را به‌صورت زنده روی تصویر دوربین نمایش دهید.

در درس قبل، ما موفق شدیم تعداد انگشتان باز را در قالب یک متغیر عددی به نام count در محیط خروجی (Terminal/Console) چاپ کنیم. اگرچه این یک موفقیت برنامه‌نویسی است، اما برای کاربر نهایی که به تصویر دوربین نگاه می‌کند، خواندن اعداد در محیط متنی ترمینال جذاب و کاربردی نیست.

یک سیستم بینایی ماشین استاندارد باید اطلاعات پردازش‌شده را به صورت زنده و مستقیم روی خود فریم‌های ویدئو نمایش دهد. در این درس، یاد می‌گیریم که چگونه با استفاده از توابع ترسیم گرافیکی در کتابخانه OpenCV، یک رابط کاربری گرافیکی ساده اما حرفه‌ای بسازیم تا تعداد انگشتان باز و وضعیت پردازش را به‌صورت بصری روی تصویر دوربین نمایش دهد.

مفهوم سیستم مختصات تصویر در OpenCV

پیش از نوشتن متن یا ترسیم شکل روی تصویر، باید درک کنیم که OpenCV چگونه نقاط روی صفحه را آدرس‌دهی می‌کند. تصویر از پیکسل‌ها تشکیل شده است. مختصات هر پیکسل با دو مقدار (x,y)مشخص می‌شود:

- نقطه (۰،۰): گوشه سمت چپ و بالای تصویر است.
- محور x: با حرکت به سمت راست، مقدار آن افزایش می‌یابد.
- محور y: با حرکت به سمت پایین، مقدار آن افزایش می‌یابد.

برای ترسیم هر المان گرافیکی، باید مختصات پیکسلی دقیق محل قرارگیری آن را به توابع OpenCV پاس بدهیم.

ترسیم مستطیل با cv2.rectangle()

برای اینکه متن‌های خروجی ما خوانا باشند، بهتر است آن‌ها را روی یک مستطیل پس‌زمینه (مثلاً یک جعبه اطلاعاتی تیره) بنویسیم. تابع ترسیم مستطیل در OpenCV به شکل زیر استفاده می‌شود:

```
cv2.rectangle(image, pt1, pt2, color, thickness)
```

- image: تصویری که می‌خواهیم شکل روی آن رسم شود.
- pt۱: مختصات گوشه سمت چپ و بالای مستطیل به صورت (x۱, y۱).
- pt۲: مختصات گوشه سمت راست و پایین مستطیل به صورت (x۲, y۲).
- color: رنگ مستطیل به صورت باینری در قالب BGR مثلاً (۰, ۲۵۵, ۰) برای سبز
- Thickness: ضخامت خطوط. اگر این مقدار را برابر با ۱- یا cv۲.FILLED قرار دهیم، داخل مستطیل کاملاً رنگ می‌شود.

مثال ترسیم یک جعبه توپر رنگی:

```
# رسم یک مستطیل توپر بنفش در بالای صفحه
cv2.rectangle(frame, (10, 10), (200, 100), (128, 0, 128), -1)
```

نوشتن متن با cv2.putText()

برای نمایش متنی عدد شمارش‌شده روی فریم دوربین، از تابع **cv2.putText()** استفاده می‌کنیم:

```
cv2.putText(image, text, org, fontFace, fontScale, color, thickness)
```

- image: فریم تصویر.
- text: رشته متنی که می‌خواهیم چاپ شود باید به صورت String باشد.
- org: مختصات گوشه پایین و سمت چپ شروع متن به صورت . (x, y)
- fontFace: نوع فونت مثلاً .cv۲.FONT_HERSHEY_SIMPLEX.
- fontScale: اندازه فونت (ضریب بزرگ‌نمایی).
- color: رنگ متن در قالبBGR .
- thickness: ضخامت خطوط تشکیل‌دهنده حروف متن.

ادغام بخش گرافیکی با برنامه اصلی شمارش

اکنون کدهای رسم مستطیل پس‌زمینه و چاپ عدد شمارش‌شده را به برنامه اصلی اضافه می‌کنیم تا در هر فریم، تغییرات به صورت زنده روی تصویر اعمال شوند.

```python
import cv2
import mediapipe as mp

# MediaPipe راه‌اندازی ماژول‌های تشخیص دست
mp_hands = mp.solutions.hands
hands = mp_hands.Hands(
    static_image_mode=False,
    max_num_hands=1,
    mix_detection_confidence=0.7,
    min_tracking_confidence=0.7
)
mp_draw = mp.solutions.drawing_utils

اتصال به دوربین
cap = cv2.VideoCapture(0)

# اندیس نوک انگشتان (به جز شست)
finger_tips_ids = [8, 12, 16, 20]
while True:
    success, frame = cap.read()
    if not success:
        break
    # برگرداندن آینه‌ای تصویر برای طبیعی‌تر شدن حرکت دست
    frame = cv2.flip(frame, 1)
    frame_rgb = cv2.cvtColor(frame, cv2.COLOR_BGR2RGB)
    results = hands.process(frame_rgb)
    # مقدار اولیه شمارش در صورتی که دستی در تصویر نباشد
    count = 0
    if results.multi_hand_landmarks:
        for hand_landmarks in results.multi_hand_landmarks:
            lm_list = hand_landmarks.landmark
            fingers = []
            # منطق بررسی شست (محور X)
            if lm_list[4].x > lm_list[3].x:
                frngers.append(1)
            else:
                fingers.append(0)
            # منطق بررسی سایر انگشتان (محور Y)
            for tip_id in finger_tips_ids:
                if lm_list[tip_id].y < lm_list[tip_id - 2].y:
                    fingers.append(1)
                else:
                    fingers.append(0)
            # محاسبه تعداد انگشتان باز
            count = sum(fingers)
            رسم اسکلت و خطوط دست روی تصویر
            mp_draw.draw_landmarks(
                frame,
                hand_landmarks,
                mp_hands.HAND_CONNECTIONS
            )
# --- طراحی رابط کاربری گرافیکی ---
1 رسم جعبه پس‌زمینه برای نمایش عدد (مستطیل توپر آبی رنگ)
cv2.rectangle(frame, (20, 20), (120, 120), (255, 0, 0) ,
# 2 قرار دادن عدد شمارش‌شده در وسط جعبه (تبدیل عدد count به رشته)
cv2.putText(
    frame,
    str(count),
    (45, 95),
    cv2.FONT_HERSHEY_SIMPLEX,
    2.5,
    (255, 255, 255),
    5
)
# 3 نمایش وضعیت پردازش سیستم به صورت متن کوچک‌تر بالای صفحه
cv2.putText(
    frame,
    "FINGERS ACTIVE",
    (150, 50),
    cv2.FONT_HERSHEY_PLAIN,
    1.5,
    (0, 255, 0),
    2
)
نمایش پنجره خروجی گرافیکی
cv2.imsbow("Smart Finger Counter GUI", frame)
شرط خروج با کلید q
if cv2.waitKey(1) & 0xFF == ord("q"):
    break
cap.release()
cv2.destroyAllWindows()
```

فعالیت کلاسی

۱. برنامه بالا را اجرا کرده و هماهنگی تغییر اعداد داخل جعبه آبی‌رنگ را با حرکت انگشتان دست خود بررسی کنید.

۲. با تغییر مقادیر پیکسلی در دستور cv۲.rectangle موقعیت جعبه نمایش را از سمت چپ و بالای تصویر، به گوشه سمت راست و پایین منتقل کنید. (راهنمایی: اندازه فریم ورودی معمولاً ۴۸۰ × ۶۴۰پیکسل است).

۳. فونت‌های مختلف موجود در OpenCV مانند FONT_HERSHEY_DOUBLE_TRIPLEXcv۲. را امتحان کنید و تفاوت ظاهر نوشته‌ها را ببینید.

٤. رنگ پس‌زمینه جعبه عدد را به زرد و رنگ خود نوشته عدد را به مشکی تغییر دهید.

خلاصه درس

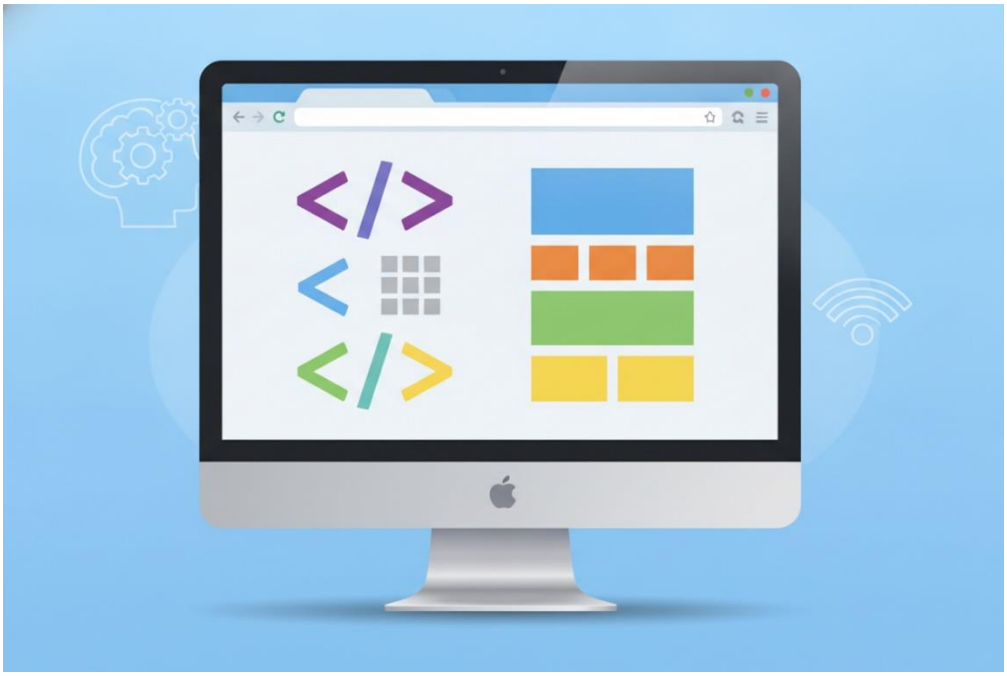
- سیستم مختصات تصویر در OpenCV از گوشه بالا و سمت چپ تصویر با مختصات صفر شروع می‌شود.
- ترسیم مستطیل با تابع cv.rectangle برای ساخت جعبه نمایش اطلاعات روی تصویر انجام می‌شود.
- نمایش متن با تابع cv.putText برای نمایش تعداد انگشتان و وضعیت برنامه روی فریم استفاده می‌شود.
- رابط کاربری با ترکیب متن و شکل‌های گرافیکی، اطلاعات پردازش‌شده را به‌صورت زنده روی تصویر نمایش می‌دهد.

پروژه

۱. با استفاده از آموخته‌های خود در شرط‌های پایتون، برنامه‌ای بنویسید که اگر تعداد انگشتان باز ۵ بود، متنی با عنوان "ALL OPEN" در وسط صفحه نمایش داده شود و اگر ۰ بود، متن "CLOSED" نمایش یابد.

۲. طرحی ایجاد کنید که علاوه بر نمایش تعداد کل انگشتان، یک نوار باریک (مانند نوار بار پیشرفت یا Progress Bar) در پایین صفحه رسم کند که طول آن با توجه به تعداد انگشتان باز تغییر کند (مثلاً با هر انگشت باز، ۱۰۰ پیکسل به طول نوار اضافه شود).

درس۱۴: آشنایی با HTML و ساخت اولین صفحه وب



در پایان این درس خواهید توانست:

- HTML را تعریف کنید.
- کاربرد HTML در طراحی صفحات وب را توضیح دهید.
- ساختار اصلی یک سند HTML را بشناسید.
- با تگ‌ها و عناصر اصلی HTML آشنا شوید.
- اولین صفحه وب خود را ایجاد و در مرورگر مشاهده کنید.



امروزه بیشتر فعالیت‌های آموزشی، خرید، بانکداری، ارتباطات و سرگرمی از طریق وب‌سایت‌ها انجام می‌شود. هر صفحه وب که در اینترنت مشاهده می‌کنیم، با استفاده از زبان‌های مختلف برنامه‌نویسی و نشانه‌گذاری ساخته شده است. مهم‌ترین زبان برای ایجاد ساختار صفحات وب، HTML است.

HTML مشخص می‌کند هر بخش از یک صفحه، مانند عنوان، متن، تصویر یا پیوند، چگونه در مرورگر نمایش داده شود. به عبارت دیگر، HTML اسکلت و ساختار اصلی یک صفحه وب را تشکیل می‌دهد و مرورگر بر اساس آن، محتوای صفحه را نمایش می‌دهد.

در این درس باHTML ، ساختار یک صفحه وب و روش ایجاد اولین صفحه HTML آشنا می‌شوید.

به پرسش‌های زیر پاسخ دهید.

۱. روزانه از چه وب‌سایت‌هایی استفاده می‌کنید؟
۲. برای مشاهده صفحات وب از چه نرم‌افزاری استفاده می‌کنید؟
۳. به نظر شما مرورگر چگونه تشخیص می‌دهد کدام متن عنوان و کدام متن پاراگراف است؟



HTML چیست؟

HTML مخفف عبارت HyperText Markup Language به معنای زبان نشانه‌گذاری ابرمتن است. این زبان برای ایجاد ساختار صفحات وب استفاده می‌شود و مشخص می‌کند هر بخش از صفحه چه نقشی دارد.

برای مثال، با HTML می‌توان تعیین کرد که:

یک متن عنوان اصلی صفحه باشد.

بخشی از متن به صورت پاراگراف نمایش داده شود.

تصویری در صفحه قرار گیرد.

کاربر با کلیک روی یک متن به صفحه دیگری منتقل شود.

نکته مهم این است که HTML ظاهر صفحات را طراحی نمی‌کند؛ بلکه ساختار و محتوای آن‌ها را مشخص می‌کند. برای زیباسازی صفحات وب از زبان CSS استفاده می‌شود که در درس‌های آینده با آن آشنا خواهید شد.

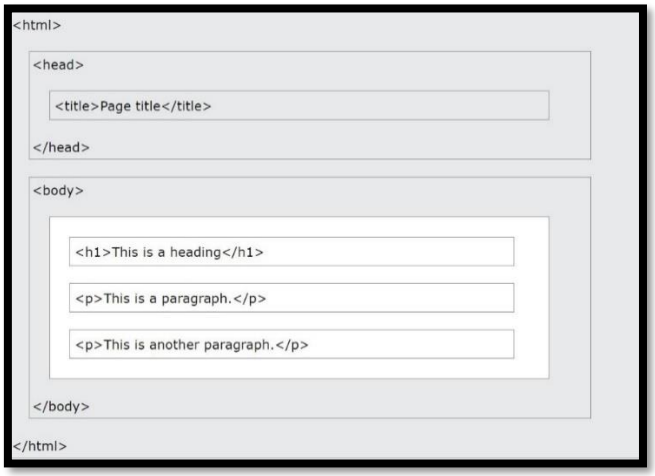
مرورگر چیست؟

مرورگر (Browser) نرم‌افزاری است که فایل‌های HTML را خوانده و آن‌ها را به صورت یک صفحه وب نمایش می‌دهد.

برخی از مرورگرهای پرکاربرد عبارت‌اند از:

Google Chrome
Microsoft Edge
Mozilla Firefox
Safari

مرورگر، کدهای HTML را تفسیر می‌کند و نتیجه را به صورت متن، تصویر، جدول، پیوند و سایر عناصر به کاربر نمایش می‌دهد.



ساختار یک سند HTML

تمام صفحات وب دارای ساختار مشخصی هستند. این ساختار از چند بخش اصلی تشکیل شده است.

```
<!DOCTYPE html>
<html>
<head>
<title>Page Title</title>
</head>
<body>

<h1>My First Heading</h1>
<p>My first paragraph.</p>

</body>
</html>
```

My First Heading

My first paragraph.

۱- اعلام نوع سند

<!DOCTYPE html>

این دستور به مرورگر اعلام می‌کند که فایل، یک سند **HTML۵** است.

۲- عنصر اصلی صفحه

<html>

تمام محتوای صفحه وب داخل این تگ قرار می‌گیرد.

۳- بخش Head

<head>

این بخش شامل اطلاعات مربوط به صفحه است؛ مانند عنوان صفحه و اطلاعات موردنیاز مرورگر.

محتویات این بخش معمولاً در صفحه وب نمایش داده نمی‌شوند.

٤- عنوان صفحه

<title>

عنوان صفحه را مشخص می‌کند.

این عنوان در **تب مرورگر** یا نوار عنوان پنجره نمایش داده می‌شود.

۵-بخش Body

<body>

تمام محتوایی که کاربر در صفحه مشاهده می‌کند، داخل این بخش قرار می‌گیرد؛ مانند:

- عنوان‌ها
- متن‌ها
- تصاویر
- پیوندها
- جدول‌ها
- فهرست‌ها

عنصر (Element) چیست؟

هر عنصر HTML معمولاً از سه بخش تشکیل شده است:

- تگ آغازین (Start Tag)
- محتوا (Content)
- تگ پایانی (End Tag)

به عنوان مثال:

<h۱>به درس HTML خوش آمدید</h۱>

در این مثال:

<h۱>تگ آغازین است.

«به درس HTML خوش آمدید» محتوا است.

</h۱> تگ پایانی است.

تگ آغازین	عملکرد	تگ پایانی
< h۱>	سربرگ اول (عنوان بزرگ)	</ h۱>
<P>	پاراگراف معمولی	</P>
< br >	شکستن خط (خط جدید)	

برخی عناصر پرکاربرد

توجه: برخی از تگ‌ها مانند
 تگ پایانی ندارند و به آن‌ها **عناصر خالی** گفته می‌شود.

تاریخچه‌ای کوتاه از HTML

HTMLنخستین بار در سال **۱۹۹۳** معرفی شد و تاکنون نسخه‌های مختلفی از آن منتشر شده است.

امروزه **HTML۵** جدیدترین و پرکاربردترین نسخه این زبان است که امکانات فراوانی برای طراحی صفحات وب در اختیار برنامه‌نویسان قرار می‌دهد.

نسخه HTML	سال انتشار	ویژگی‌های اصلی
HTML ۱,۰	۱۹۹۳	اولین نسخه پایه‌ای، بسیار محدود، بدون استاندارد رسمی
HTML ۲,۰	۱۹۹۵	اولین نسخه استاندارد شده، شامل فرم‌ها و تصاویر
HTML ۳,۲	۱۹۹۷	پشتیبانی از جداول، ابرپیوندها، و فونت‌ها
HTML ٤,۰۱	۱۹۹۹	پشتیبانی از CSS، اسکریپت‌نویسی، بهبود قابلیت‌های تعاملی
XHTML ۱,۰	۲۰۰۰	نسخه‌ای با ساختار XML، منظم‌تر و قابلیت خواندن بهتر برای ماشین
HTML۵	۲۰۱۴	آخرین استاندارد، شامل تگ‌های جدید

آیا می‌دانید؟

اگر فایل HTML را با یک ویرایشگر متن ساده مانند Notepad بنویسید و با پسوند html. ذخیره کنید، می‌توانید آن را در هر مرورگری مشاهده کنید.

فعالیت کلاسی: ساخت اولین صفحه وب

وسایل مورد نیاز:

نرم‌افزار Notepad یا هر ویرایشگر متن ساده

یک مرورگر وب

مراحل اجرا:

مرحله ۱: برنامه Notepad را اجرا کنید.

مرحله ۲: کد زیر را تایپ کنید.

```
<!DOCTYPE html>
<html>
<head>
</title>اولین صفحه وب<title>
</head>
<body>
</h۱>سلام دنیا<h۱>
.</p>این اولین صفحه HTML من است<p>
</body>
</html>
```

مرحله ۳: فایل را با نام index.html ذخیره کنید.

مرحله ۴: فایل ذخیره‌شده را در مرورگر باز کنید و نتیجه را مشاهده نمایید.

خلاصه درس

- HTML زبان استاندارد ایجاد صفحات وب است.
- HTML ساختار صفحه را مشخص می‌کند.
- مرورگر کدهای HTML را خوانده و نمایش می‌دهد.
- هر صفحه HTML از بخش‌های اصلی مانند html، head و body تشکیل شده است.
- هر عنصر HTML معمولاً شامل تگ آغازین، محتوا و تگ پایانی است.
- با استفاده از یک ویرایشگر متن ساده می‌توان اولین صفحه وب را ایجاد کرد.

تمرین

۱. HTMLمخفف چه عبارتی است؟

.....

۲. یک فایل HTML ایجاد کنید که شامل:

عنوان صفحه

یک عنوان اصلی

دو پاراگراف

باشد.

پروژه

یک صفحه وب با موضوع «**معرفی مدرسه من**» طراحی کنید که شامل موارد زیر باشد:

عنوان صفحه

نام مدرسه

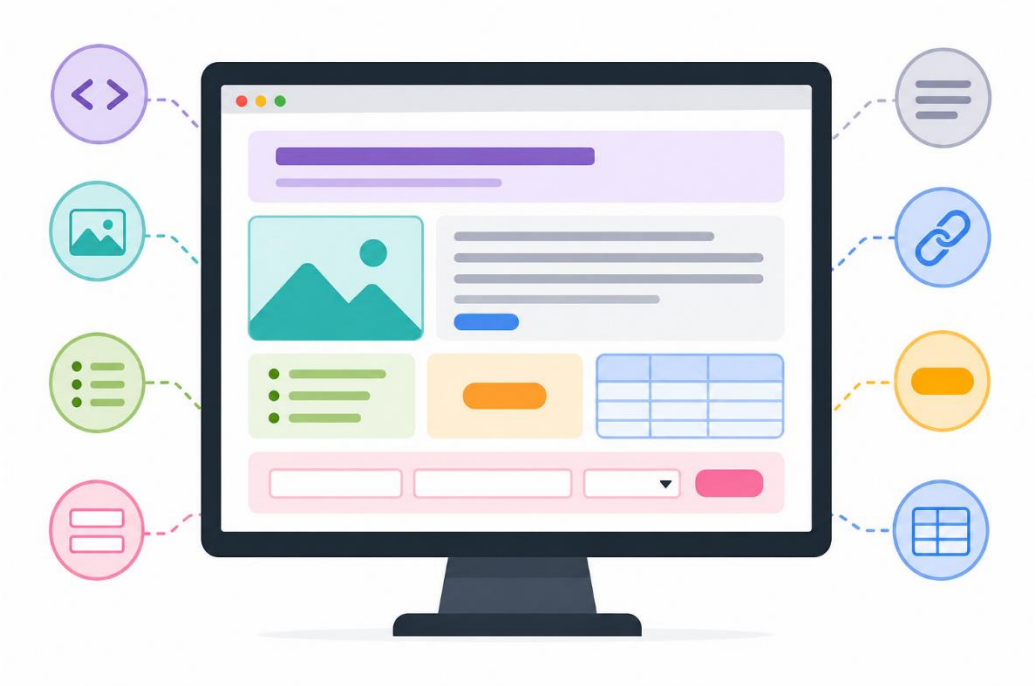
نام دانش‌آموز

یک پاراگراف درباره مدرسه

یک پیام خوش‌آمدگویی

فایل را با نام **school.html** ذخیره کرده و در مرورگر اجرا کنید.

درس ۱۵: تگ‌ها و عناصر پرکاربردHTML



در پایان این درس خواهید توانست:

- تگ‌های پرکاربرد HTML را شناسایی کنید.
- از سرخط‌ها و پاراگراف‌ها برای سازمان‌دهی متن استفاده کنید.
- پیوند (لینک) و تصویر را در صفحات وب قرار دهید.
- با ویژگی‌های مهم عناصر HTML آشنا شوید.
- یک صفحه وب ساده شامل متن، تصویر و لینک طراحی کنید.

اکنون که با ساختار کلی یک صفحه HTML آشنا شده‌اید، زمان آن رسیده است که عناصر مختلف یک صفحه وب را بشناسید. صفحات وب تنها از متن تشکیل نشده‌اند؛ بلکه شامل عنوان‌ها، پاراگراف‌ها، تصاویر، پیوندها، خطوط جداکننده و عناصر دیگری هستند که هر کدام با تگ‌های HTML ایجاد می‌شوند.

به پرسش‌های زیر پاسخ دهید.

۱. در یک صفحه وب، عنوان اصلی چه تفاوتی با متن عادی دارد؟
۲. چگونه می‌توان کاربر را از یک صفحه به صفحه دیگر هدایت کرد؟
۳. اگر بخواهید تصویری را در صفحه وب نمایش دهید، از چه روشی استفاده می‌کنید؟

سرخط‌ها (Headings)

عنوان‌ها و زیرعنوان‌های صفحات وب با تگ‌های <h۱>تا <h٦>ایجاد می‌شوند.

<h۱>بزرگ‌ترین و مهم‌ترین عنوان صفحه است.

<h٦>کوچک‌ترین عنوان را نمایش می‌دهد.

```
<h1>عنوان اصلی صفحه</h1>
<h2>زیرعنوان اول</h2>
<h3>زیرعنوان فرعی</h3>
<h4>بخش چهارم</h4>
<h5>بخش پنجم</h5>
<h6>بخش آخر و کوچکترین سرخط</h6>
```

هرچه شماره تگ بزرگ‌تر باشد، اندازه و اهمیت عنوان کمتر خواهد بود.

نمونه:

<h۱> مدرسه علوی h۱>

<h۲>درباره مدرسه</h۲>

<h۳>فعالیت‌های آموزشی </
</h۳>

اهمیت سرخط‌ها

استفاده صحیح از سرخط‌ها باعث می‌شود:

- مطالب صفحه منظم‌تر شوند.
- مطالعه مطالب آسان‌تر شود.
- موتورهای جستجو موضوع صفحه را بهتر تشخیص دهند.

نکته: بهتر است در هر صفحه فقط یک عنوان اصلی (<h۱>) وجود داشته باشد.

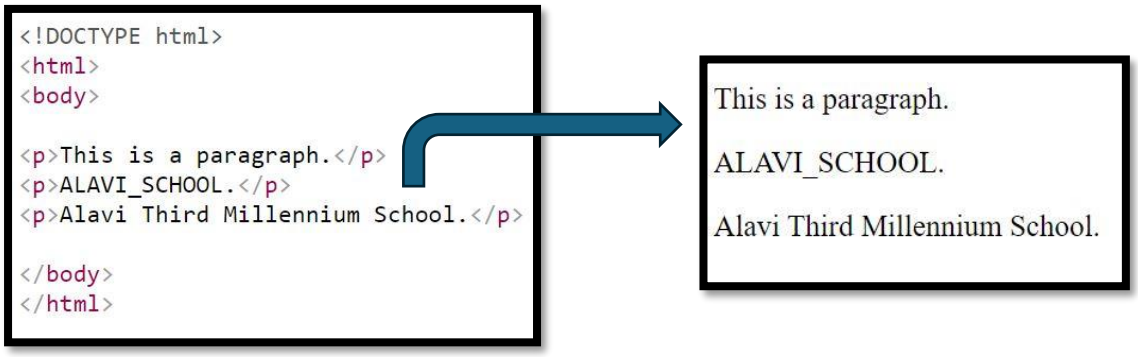
پاراگراف‌ها (Paragraph)

برای نوشتن متن‌های معمولی از تگ <**p**> استفاده می‌شود.

هر پاراگراف از خط جدید آغاز شده و مرورگر به‌طور خودکار فاصله‌ای قبل و بعد از آن ایجاد می‌کند.

نمونه:

مدرسه علوی محیطی مناسب برای یادگیری، پژوهش و رشد دانش‌آموزان فراهم کرده است.



ایجاد پیوند(Hyperlink)

برای انتقال کاربر به صفحه‌ای دیگر از تگ <**a**> استفاده می‌شود.

آدرس مقصد در ویژگی **href** نوشته می‌شود.

نمونه:

ورود به سایت

کاربرد لینک‌ها

با استفاده از لینک می‌توان:

به صفحه‌ای دیگر در همان وب‌سایت رفت.

وارد یک وب‌سایت دیگر شد.

فایل یا سندی را برای دانلود در اختیار کاربران قرار داد.

نمایش تصویر

برای قرار دادن تصویر در صفحه وب از تگ <**img**>استفاده می‌شود.

برخلاف بسیاری از تگ‌ها، این تگ **تگ پایانی ندارد.**

مهم‌ترین ویژگی‌های آن عبارت‌اند از:

ویژگی	کاربرد
src	آدرس تصویر
alt	متن جایگزین تصویر
width	عرض تصویر
height	ارتفاع تصویر

نمونه:

```
<img src="school.jpg" alt="مدرسه علوی" width="۳۰۰">
```

ویژگی alt

اگر تصویر به هر دلیلی نمایش داده نشود، متن موجود در ویژگی **alt** به جای آن نمایش داده می‌شود.

همچنین این ویژگی برای نرم‌افزارهای ویژه افراد کم‌بینا و نابینا اهمیت زیادی دارد.

آدرس‌دهی تصاویر

برای معرفی محل تصویر دو روش وجود دارد.

آدرس نسبی (Relative URL)

در این روش تصویر داخل پوشه پروژه قرار دارد.

```
src="images/logo.png"
```

این روش برای بیشتر پروژه‌ها مناسب‌تر است.

آدرس مطلق (Absolute URL)

در این روش تصویر از یک وب‌سایت دیگر بارگذاری می‌شود.

```
src="https://example.com/logo.png"
```

ویژگی‌های عناصر HTML

هر عنصر HTML می‌تواند ویژگی‌هایی داشته باشد که اطلاعات بیشتری درباره آن عنصر در اختیار مرورگر قرار می‌دهند.

ویژگی‌ها همیشه داخل **تگ آغازین** نوشته می‌شوند.

برای مثال:

```
<img src="logo.png" width="۲۰۰">
```

در این مثال:

src محل تصویر را مشخص می‌کند.

width عرض تصویر را تعیین می‌کند.

ویژگی‌های پرکاربرد

title

این ویژگی توضیح کوتاهی درباره عنصر ارائه می‌دهد.

اگر نشانگر ماوس چند لحظه روی عنصر قرار گیرد، متن آن نمایش داده می‌شود.

lang

زبان صفحه وب را مشخص می‌کند.

نمونه:

```
<html lang="fa">
```

مشاهده کد صفحات وب

تمام صفحات اینترنت از کدهای HTML تشکیل شده‌اند.

برای مشاهده کد یک صفحه می‌توانید:

روی صفحه کلیک راست کنید.

گزینه **View Page Source** را انتخاب کنید.

همچنین با انتخاب گزینه **Inspect** می‌توانید ساختار عناصر صفحه را مشاهده و بررسی کنید.

آیا می‌دانید؟

- بیشتر وب‌سایت‌های دنیا از همین تگ‌های ساده HTML برای ایجاد ساختار صفحات خود استفاده می‌کنند.
- موتورهای جستجو مانند گوگل، ابتدا عنوان‌ها و سرخط‌های صفحات را بررسی می‌کنند تا موضوع صفحه را تشخیص دهند.
- استفاده از ویژگی **alt** باعث می‌شود صفحات وب برای همه کاربران، از جمله افراد دارای معلولیت بینایی، قابل استفاده باشند.

فعالیت کلاسی: طراحی یک صفحه وب ساده برای "مدرسه علوی"

شما باید یک صفحه HTML بنویسید که اطلاعات زیر را نمایش دهد:

۱. عنوان اصلی صفحه "Alavi School" :

۲. یک سرخط با متن "About Our School" :

۳. یک پاراگراف درباره مدرسه

٤. یک خط افقی (**<hr>**) برای جدا کردن بخش‌ها

خلاصه درس

- عنوان‌های صفحات وب با تگ‌های <h۱> تا <h٦> ایجاد می‌شوند.
- متن‌های معمولی با تگ <p> نوشته می‌شوند.
- برای ایجاد پیوند از تگ <a> استفاده می‌شود.
- تصاویر با تگ <img> در صفحه قرار می‌گیرند.
- ویژگی‌هایی مانند **src**، **alt**، **width** و **height** اطلاعات بیشتری درباره عناصر HTML ارائه می‌کنند.
- می‌توان کد صفحات وب را در مرورگر مشاهده و بررسی کرد.

تمرین

۱. تفاوت تگ <h۱> و <h٦> چیست؟

.....

۲. کاربرد تگ <p> را توضیح دهید.

.....

۳. ویژگی href در کدام تگ استفاده می‌شود و چه وظیفه‌ای دارد؟

.....

٤. ویژگی alt چه کاربردی دارد؟

.....

٥. دو تفاوت بین آدرس نسبی و آدرس مطلق را بیان کنید.

.....

پروژه

یک صفحه وب با موضوع «**شهر من**» طراحی کنید که شامل موارد زیر باشد:

- عنوان اصلی
- دو زیرعنوان
- دو پاراگراف
- یک تصویر
- یک لینک به وب‌سایت شهرداری یا یکی از جاذبه‌های گردشگری شهر

پس از اجرای صفحه، از عملکرد صحیح لینک و نمایش تصویر اطمینان حاصل کنید.

درس ۱۶: ویژگی‌ها، قالب‌بندی و طراحی صفحه وب با استایل



در پایان این درس خواهید توانست:

- مفهوم ویژگی (Attribute) در HTML را توضیح دهید.
- کاربرد ویژگی styleرا در طراحی صفحات وب بیان کنید.
- رنگ متن و رنگ پس‌زمینه عناصر را تغییر دهید.
- نوع، اندازه و محل قرارگیری متن را تنظیم کنید.
- از تگ‌های
، <hr>و <pre>به‌درستی استفاده کنید.
- یک صفحه وب ساده را با استفاده از Style طراحی و قالب‌بندی کنید.

در درس‌های قبل یاد گرفتید چگونه ساختار یک صفحه وب را ایجاد کرده و از تگ‌هایی مانند عنوان، پاراگراف، تصویر و پیوند استفاده کنید. اما بیشتر صفحات وب تنها از متن ساده تشکیل نشده‌اند؛ بلکه دارای رنگ، اندازه‌های مختلف، فونت‌های متنوع و چیدمان مناسب هستند.

برای تغییر ظاهر عناصر HTML از CSS استفاده می‌شود. یکی از ساده‌ترین روش‌های استفاده از CSS ، به‌کارگیری ویژگی style در داخل تگ‌های HTML است. در این درس با ویژگی‌های پرکاربرد Style آشنا می‌شوید و اولین صفحه وب خود را قالب‌بندی خواهید کرد.

به پرسش‌های زیر پاسخ دهید.

۱. چرا صفحات وب مختلف ظاهر یکسانی ندارند؟
۲. چگونه می‌توان رنگ نوشته‌های یک صفحه را تغییر داد؟
۳. اگر بخواهید عنوان صفحه بزرگ‌تر و در وسط صفحه نمایش داده شود، چه باید کرد؟

ویژگی (Attribute) چیست؟

ویژگی‌ها اطلاعات بیشتری درباره عناصر HTML در اختیار مرورگر قرار می‌دهند و در تگ آغازین نوشته می‌شوند. برای مثال، در تگ تصویر، ویژگی src محل قرارگیری تصویر را مشخص می‌کند و ویژگی width عرض آن را تعیین می‌کند.

نمونه:

```
<img src="school.jpg" width="۳۰۰">
```

ویژگی Style چیست؟

ویژگی استایل برای تعیین ظاهر عناصر HTML به کار می‌رود. با استفاده از آن می‌توان رنگ، اندازه، نوع قلم و محل قرارگیری متن را تغییر داد.

ساختار کلی آن به صورت زیر است:

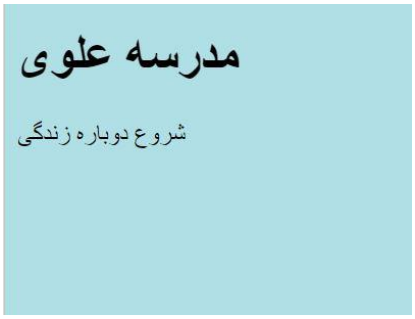
```
<tag style="ویژگی: مقدار;">
```

```
<!DOCTYPE html>
<html>
<body style="background-color:powderblue;">

<h1>مدرسه علوی </h1>
<p>شروع دوباره زندگی</p>

</body>
</html>
```





تغییر رنگ متن

برای تعیین رنگ نوشته‌ها از ویژگی **color** استفاده می‌شود.

نمونه:

```
<h۱ style="color:blue;">
مدرسه علوی
</h۱>
```

```
<!DOCTYPE html>
<html>
<body>

<h1 style="color:blue;">مدرسه علوی </h1>
<p style="color:red;"> مدرسه هزاره سوم </p>

</body>
</html>
```



مدرسه علوی

مدرسه هزاره سوم

تغییر رنگ پس‌زمینه

برای تغییر رنگ زمینه یک عنصر از ویژگی **background-color** استفاده می‌شود.

نمونه:

```
<p style="background-color:yellow;">
به مدرسه ما خوش آمدید.
</p>
```

تعیین نوع قلم

برای انتخاب نوع فونت از ویژگی **font-family** استفاده می‌شود.

نمونه:

```
<p style="font-family:Tahoma;">
متن نمونه
</p>
```

استفاده از فونت مناسب باعث افزایش خوانایی صفحه می‌شود.

تعیین اندازه قلم

برای تغییر اندازه نوشته‌ها از ویژگی font-size استفاده می‌شود.

نمونه:

```
<p style="font-size:۲۰px;">
متن نمونه
</p>
```

واحد px (پیکسل) یکی از رایج‌ترین واحدهای اندازه‌گیری در صفحات وب است.

```
<h1 style="font-size:300%;">ALAVI</h1>
<p style="font-size:160%;">MATH AND ART .</p>
```



ALAVI

MATH AND ART .

هم‌تراز کردن متن

برای تعیین محل قرارگیری متن از ویژگی text-align استفاده می‌شود.

مقادیر متداول آن عبارت‌اند از:

- left
- center
- right

نمونه:

<h۱ style="text-align:center;">

طراحی صفحات وب

</h۱>

```
<h1 style="text-align:center;">Centered Heading</h1>
<p style="text-align:center;">Centered paragraph.</p>
```

ایجاد خط افقی

گاهی لازم است بخش‌های مختلف صفحه از یکدیگر جدا شوند. برای این کار از تگ **<hr>** استفاده می‌شود.

نمونه:

<hr>

این تگ یک خط افقی در صفحه ایجاد می‌کند.

شکستن خط

مرورگر فاصله‌ها و خطوط خالی اضافی موجود در کد HTML را نادیده می‌گیرد. برای رفتن به خط بعد بدون ایجاد پاراگراف جدید از تگ **
** استفاده می‌شود.

نمونه:

نام مدرسه

شهر تهران

نمایش متن با قالب اصلی

در حالت عادی مرورگر فاصله‌های اضافی را حذف می‌کند. اگر بخواهیم متن دقیقاً با همان فاصله‌ها و خطوط نوشته‌شده نمایش داده شود، از تگ **<pre>** استفاده می‌کنیم.

این تگ برای نمایش موارد زیر مناسب است:

- شعر
- کدهای برنامه‌نویسی
- متن‌هایی که فاصله‌ها اهمیت دارند.

نمونه:

<pre>
- دانایی توانایی است.
- تلاش رمز موفقیت است.
</pre>

طراحی یک صفحه وب ساده

با ترکیب تگ‌های HTML و ویژگی **style** می‌توان صفحات زیباتری ایجاد کرد.

برای مثال می‌توان:

- عنوان را در مرکز صفحه قرار داد.
- رنگ متن را تغییر داد.
- اندازه نوشته را بزرگ‌تر کرد.
- برای پاراگراف‌ها رنگ پس‌زمینه تعیین نمود.
- این تغییرات باعث افزایش خوانایی و جذابیت صفحه می‌شوند.



فعالیت کلاسی: طراحی اولین صفحه وب رنگی

وسایل مورد نیاز

نرم‌افزار Notepad یا Visual Studio Code

مرورگر وب

مراحل اجرا

۱. یک فایل جدید با نام **style.html** ایجاد کنید.
۲. عنوان صفحه را در مرکز قرار دهید.
۳. برای عنوان رنگ متن و رنگ پس‌زمینه تعیین کنید.
٤. یک پاراگراف با رنگ و اندازه دلخواه بنویسید.
٥. یک خط افقی بین دو بخش صفحه قرار دهید.
٦. یک شعر کوتاه را با استفاده از تگ **<pre>** نمایش دهید.
۷. فایل را ذخیره کرده و نتیجه را در مرورگر مشاهده کنید.

خلاصه درس

- ویژگی‌ها اطلاعات بیشتری درباره عناصر HTML ارائه می‌کنند.
- با استفاده از ویژگی style می‌توان ظاهر عناصر را تغییر داد.
- ویژگی‌های color، background-color، font-family، font-size و text-align برای قالب‌بندی صفحات به کار می‌روند.
- تگ <hr> برای ایجاد خط افقی استفاده می‌شود.
- تگ
 متن را به خط بعد منتقل می‌کند.
- تگ <pre> فاصله‌ها و خطوط متن را همان‌گونه که نوشته شده‌اند نمایش می‌دهد.

تمرین

۱. ویژگی style چه کاربردی دارد؟

.....

۲. تفاوت color و background-color را توضیح دهید.

.....

۳. از کدام ویژگی برای تغییر اندازه نوشته استفاده می‌شود؟

.....

٤. کاربرد تگ‌های زیر را بنویسید.

<hr>.....

.....

<pre>.....

پروژه

یک صفحه وب با عنوان «مدرسه من» طراحی کنید که شامل موارد زیر باشد:

۱. عنوان اصلی در مرکز صفحه
۲. یک تصویر از مدرسه
۳. یک پاراگراف معرفی
٤. یک فهرست از امکانات مدرسه
٥. یک خط افقی بین بخش‌ها
٦. یک شعار مدرسه با استفاده از تگ <pre>
۷. رنگ‌بندی مناسب برای متن و پس‌زمینه
۸. استفاده از حداقل پنج ویژگی style

پس از اجرا، صفحه را بررسی کرده و در صورت نیاز ظاهر آن را بهبود دهید.

چالش

صفحه معرفی یک مدرسه یا کتابخانه طراحی کنید که در آن از تمام مطالب سه درس این فصل استفاده شده باشد؛ از جمله:

۱. ساختار کامل HTML
۲. عنوان‌ها و پاراگراف‌ها
۳. تصویر و پیوند
٤. ویژگی‌های style
٥. تگ‌های <hr>،
و <pre>

درس۱۷: آشنایی با چاپ سه‌بعدی



در پایان این درس خواهید توانست:

- چاپ سه‌بعدی را به زبان ساده توضیح دهید.
- تفاوت پرینتر معمولی و پرینتر سه‌بعدی را بیان کنید.
- چند کاربرد چاپ سه‌بعدی را در زندگی روزمره نام ببرید.
- مراحل کلی ساخت یک قطعه با پرینتر سه‌بعدی را توضیح دهید.
- اهمیت طراحی سه‌بعدی در ساخت قطعات را درک کنید.


امروزه فناوری‌های نوین، روش ساخت بسیاری از وسایل را تغییر داده‌اند. یکی از این فناوری‌ها چاپ سه‌بعدی است. با استفاده از پرینتر سه‌بعدی می‌توان بسیاری از وسایل مورد نیاز را تنها با طراحی یک مدل در رایانه تولید کرد.
چاپ سه‌بعدی در صنایع مختلف مانند پزشکی، معماری، خودروسازی، آموزش و حتی ساخت اسباب‌بازی کاربرد فراوانی دارد. آشنایی با این فناوری، نخستین گام برای ورود به دنیای طراحی و ساخت دیجیتال است.

۱. آیا تاکنون وسیله‌ای را دیده‌اید که با پرینتر سه‌بعدی ساخته شده باشد؟
۲. به نظر شما پرینتر سه‌بعدی چگونه یک جسم واقعی را تولید می‌کند؟
۳. اگر بتوانید هر وسیله‌ای را طراحی و چاپ کنید، دوست دارید چه چیزی بسازید؟
٤. پرینتر معمولی چه تفاوتی با پرینتر سه‌بعدی دارد؟



چاپ سه‌بعدی چیست؟

چاپ سه‌بعدی روشی برای ساخت اجسام واقعی از روی یک مدل دیجیتالی است. در این روش، پرینتر جسم را لایه به لایه می‌سازد تا در پایان، قطعه کامل ایجاد شود.
در پرینترهای معمولی، تنها جوهر روی کاغذ قرار می‌گیرد؛ اما پرینتر سه‌بعدی از مواد پلاستیکی مخصوص استفاده می‌کند و یک جسم واقعی تولید می‌کند.
به همین دلیل به این فناوری ساخت افزایشی (Additive Manufacturing) نیز گفته می‌شود؛ زیرا قطعه با اضافه شدن لایه‌های متوالی ساخته می‌شود.

تاریخچه چاپ سه‌بعدی

در ابتدا، چاپ سه‌بعدی با عنوان نمونه‌سازی سریع [۱] شناخته می‌شد اصطلاحی که امروزه هنوز گاهی استفاده می‌شود، البته به ندرت. پیش از اینکه پرینترهای سه‌بعدی مقرون‌به‌صرفه در دسترس عموم قرار گیرند، این فناوری تنها برای ساخت نمونه‌های اولیه به‌کار می‌رفت.
یکی از کاربردهای معمول نمونه‌سازی سریع، طراحی و تولید یک دورکنترل تلویزیون بود.
هزینه‌های مقدماتی برای تولید انبوه مانند ساخت قالب‌ها، فرآیند تولید خود، آزمایش‌ها و غیره می‌تواند به ده‌ها هزار دلار برسد. بنابراین، تولیدکننده باید کاملاً مطمئن باشد که دورکنترل به‌خوبی در دست کاربر قرار می‌گیرد و تمام دکمه‌ها به‌راحتی در دسترس هستند. در اینجا نمونه‌سازی وارد عمل می‌شود.
اگرچه قیمت پرینترهای سه‌بعدی در آن زمان بسیار بالا بود، اما هزینه تولید یک نمونه واحد با روش‌های قدیمی حدود هزار دلار بود که باز هم صرفه‌جویی قابل‌توجهی محسوب می‌شد. با این حال، به‌دلیل هزینه بالای دستگاه‌ها، هیچ امیدی برای ورود آن‌ها به دست کاربران عادی وجود نداشت خوشبختانه این وضعیت تغییر کرد.

[۱] Rapid Prototyping

آیا می‌دانید؟

چاپ سه‌بعدی به شکلی که امروزه آن را می‌شناسیم، در سال ۱۹۸۴ کشف شد، زمانی که چارلز دبلیو. هال بنیان‌گذار شرکت 3 D Systems، برای اختراع خود **استریولیتوگرافی** درخواست ثبت اختراع داد. هال اولین کسی بود که داده‌های دیجیتال سه‌بعدی را به‌صورت فیزیکی چاپ کرد. این فناوری که معمولاً به اختصار SLA نامیده می‌شود، تا به امروز نیز مورد استفاده قرار می‌گیرد. در بخش مربوط به روش‌های مختلف چاپ سه‌بعدی می‌توانید اطلاعات بیشتری درباره آن کسب کنید.

پرینتر سه‌بعدی چگونه کار می‌کند؟

برای ساخت یک قطعه با پرینتر سه‌بعدی، معمولاً مراحل زیر انجام می‌شود:

۱. ایده‌پردازی

ابتدا مشخص می‌کنیم چه وسیله‌ای قرار است ساخته شود.

۲. طراحی سه‌بعدی

مدل وسیله در نرم‌افزارهای طراحی مانند تینکرکد (Tinkercad) طراحی می‌شود.

۳. آماده‌سازی فایل

مدل طراحی شده به نرم‌افزار مخصوص چاپ منتقل می‌شود تا برای چاپ آماده شود.

۴. چاپ قطعه

پرینتر با ذوب کردن فیلامنت و قرار دادن آن به صورت لایه‌لایه، قطعه را می‌سازد.



کاربردهای چاپ سه‌بعدی

چاپ سه‌بعدی در بسیاری از زمینه‌ها استفاده می‌شود، از جمله:

- ساخت نمونه اولیه محصولات
- تولید قطعات صنعتی
- ساخت وسایل پزشکی
- طراحی ماکت‌های معماری
- تولید قطعات آموزشی
- ساخت اسباب‌بازی و وسایل تزئینی
- ساخت قطعات یدکی

مزایای چاپ سه‌بعدی

چاپ سه‌بعدی مزایای فراوانی دارد:

- ساخت سریع نمونه‌ها
- کاهش هزینه تولید نمونه اولیه
- امکان ساخت قطعات سفارشی
- کاهش دورریز مواد
- تبدیل ایده به محصول واقعی در مدت کوتاه



فعالیت کلاسی

به تصویر یک پرینتر سه‌بعدی نگاه کنید و حدس بزنید هر قسمت چه وظیفه‌ای دارد. (در درس بعد با اجزای آن آشنا خواهید شد.)

آیا می‌دانید؟

- پرینترهای سه‌بعدی می‌توانند هزاران قطعه مختلف را تنها با تغییر فایل طراحی تولید کنند.
- بسیاری از مدارس، دانشگاه‌ها و کارخانه‌ها از پرینتر سه‌بعدی برای آموزش و ساخت نمونه‌های اولیه استفاده می‌کنند.
- امروزه حتی برخی قطعات پزشکی و دندان‌پزشکی نیز با فناوری چاپ سه‌بعدی ساخته می‌شوند.

خلاصه درس

- چاپ سه‌بعدی روشی برای ساخت اجسام واقعی از روی یک مدل دیجیتالی است.
- پرینتر سه‌بعدی قطعه را به صورت لایه‌لایه تولید می‌کند.
- طراحی سه‌بعدی نخستین مرحله ساخت یک قطعه است.
- چاپ سه‌بعدی در صنایع، پزشکی، آموزش و بسیاری از مشاغل کاربرد دارد.
- این فناوری امکان ساخت سریع و دقیق بسیاری از قطعات را فراهم می‌کند.

تمرین

۱. چاپ سه‌بعدی را به زبان خود تعریف کنید.

.....

۲. دو تفاوت بین پرینتر معمولی و پرینتر سه‌بعدی را بنویسید.

.....

۳. سه کاربرد چاپ سه‌بعدی را نام ببرید.

.....

٤. مراحل کلی ساخت یک قطعه با پرینتر سه‌بعدی را به ترتیب بنویسید.

.....

۵. به نظر شما مهم‌ترین مزیت چاپ سه‌بعدی چیست؟ دلیل خود را بنویسید.

.....

درس۱۸: آشنایی با اجزای پرینتر سه‌بعدی و فرآیند چاپ



در پایان این درس خواهید توانست:

- با اجزای اصلی پرینتر سه‌بعدی آشنا شوید.
- با مراحل آماده‌سازی و فرایند چاپ سه‌بعدی آشنا شوید.

در درس قبل با مفهوم چاپ سه‌بعدی و کاربردهای آن آشنا شدید. اما پرینتر سه‌بعدی چگونه یک طرح دیجیتالی را به یک جسم واقعی تبدیل می‌کند؟

برای پاسخ به این پرسش، ابتدا باید با اجزای اصلی پرینتر و وظیفه هر بخش آشنا شویم. شناخت این قسمت‌ها به ما کمک می‌کند هنگام طراحی و چاپ، عملکرد دستگاه را بهتر درک کنیم و از آن به‌درستی استفاده کنیم.

۱. به نظر شما کدام قسمت پرینتر، قطعه را می‌سازد؟
۲. ماده اولیه پرینتر سه‌بعدی چیست؟
۳. اگر هنگام چاپ، مواد اولیه تمام شود، چه اتفاقی می‌افتد؟
٤. چرا پرینتر باید قطعه را لایه‌به‌لایه بسازد؟

تکنولوژی پرینتر سه بعدی

تمامی روش‌های پرینت سه‌بعدی بر پایه یک اصل واحد استوارند: ایجاد اشیاء از طریق افزودن لایه‌ها بر روی لایه‌های قبلی. در حال حاضر، هیچ فناوری پرینت سه‌بعدی وجود ندارد که کاملاً جامع و برای هر هدفی مناسب باشد. به همین دلیل، بسیار مهم است که تصمیم بگیرید چگونه و برای چه هدفی قصد استفاده از پرینتر را دارید. برای ساده‌تر شدن موضوع، بیایید انواع پرینترهای سه‌بعدی را به سه دسته اصلی تقسیم کنیم:

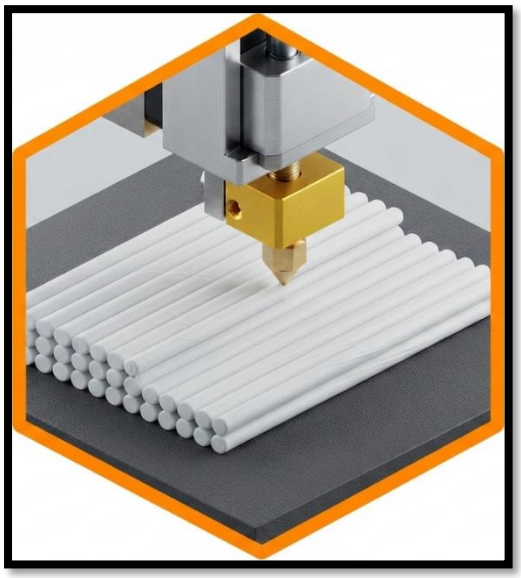
۱. رشته‌ای از پلاستیک که توسط یک المنت حرارتی ذوب شده و به‌وسیله هد چاپ (اکسترودر) از میان یک نازل خارج می‌شود. این توصیف استانداردی از فناوری‌های FFF[۲] (تولید با فیلامنت ذوبی) یا FDM[۳](مدل‌سازی لایه‌های ذوبی) است. این دو اصطلاح را می‌توان مترادف یکدیگر دانست (لازم به ذکر است که FDM یک علامت تجاری ثبت‌شده متعلق به شرکت Stratasys است).

۲. مواد مایعی که در نواحی از پیش تعیین‌شده به لایه‌های جامد تبدیل می‌شوند. این فناوری معمولاً با نام SLA یا استریولیتوگرافی شناخته می‌شود. در این روش، ماده مایع (رزین) توسط تابش نور (لیزر فرابنفش، پنل LED یا پروژکتور DLP[۴]) سخت یا پخت می‌شود.

۳. پودر نرمی که توسط لیزر زینتر (فشرده و شکل‌دهی می‌شود، نه ذوب کامل) می‌گردد. این فناوری SLS[۵](زینتر لیزری انتخابی) نام دارد و در مقایسه با دو روش قبلی، بسیار گران‌تر است.

[۲] Fused Filament Fabrication
[۳] Fused Deposition Modelling
[٤] Digital Light Processing
[٥] Selective Laser Sintering

پرینترهای FDM/FFF



این مدل‌ها رایج‌ترین و مقرون‌به‌صرفه‌ترین فناوری پرینت سه‌بعدی می‌باشد، که برای چاپ قطعات عملکردی/مکانیکی و نمونه‌های اولیه مناسب است. پرینتر از رشته‌های پلاستیکی به عنوان منبع اصلی استفاده می‌کند. به این قرقره‌های پلاستیکی «فیلامنت» گفته می‌شود که معمولاً با قطر ۱.۷۵ میلی‌متر در دسترس هستند. هنوز برخی فیلامنت‌های ۳ میلی‌متری در بازار وجود دارند، اما دقت پرینت آن‌ها بسیار پایین است و استفاده از آن‌ها توصیه نمی‌شود. در مقایسه با رزین‌های مایع یا مواد پودری، کار با فیلامنت‌ها ایمن و آسان است. نکته منفی این است که لایه‌ها روی اشیاء پرینت‌شده با چشم غیرمسلح قابل مشاهده هستند. ارتفاع معمول لایه (هنگام استفاده از نازل ۰.۴ میلی‌متری) بین ۰.۰۵ تا ۰.۳ میلی‌متر است.

یکی از فاکتورهای کلیدی در طبقه‌بندی چاپگرهای سه‌بعدی، «هندسه حرکتی» یا مکانیزم جابه‌جایی هد چاپ (اکسترودر) نسبت به صفحه ساخت است. در این بخش، سه نوع اصلی این چاپگرها را از منظر سیستم مختصات بررسی می‌کنیم:

۱. چاپگرهای کارتزین

این چاپگرها بر پایه سیستم مختصات دکارتی (XYZ) طراحی شده‌اند که رایج‌ترین نوع در صنعت چاپ سه‌بعدی رومیزی محسوب می‌شوند. در این ساختار، کنترل حرکت به تفکیک محورها انجام می‌گیرد: هد چاپ یا اکسترودر در امتداد محورهای X و Z حرکت می‌کند و صفحه چاپ به صورت رفت و برگشتی در امتداد محور Y جابه‌جا می‌شود. این ویژگی مکانیکی باعث شده است که اکثر این دستگاه‌ها از صفحات چاپ مربعی یا مستطیلی بهره ببرند. سادگی در طراحی و کالیبراسیون، ویژگی اصلی این چاپگرهاست.

۲. چاپگرهای دلتا

چاپگرهای دلتا از ساختاری متفاوت و الهام‌گرفته از بازوهای رباتیک بهره می‌برند. در این مدل، اکسترودر توسط سه بازوی متحرک که در یک نقطه مشترک به هم متصل می‌شوند، هدایت می‌گردد.

- مزایا: ویژگی بارز این مدل‌ها، سرعت بسیار بالا در چاپ و امکان دستیابی به حجم چاپ عمودی بیشتر است.

- چالش‌ها: به دلیل ماهیت حرکتی آن، کالیبراسیون اولیه و مونتاژ این دستگاه‌ها نیازمند دقت بسیار بالایی است. همچنین، کنترل نرم‌افزاری این دستگاه‌ها مستلزم انجام محاسبات ریاضی پیچیده‌ای برای هماهنگ‌سازی حرکت موتورهای پله‌ای در هر بازو است.

۳. چاپگرهای قطبی

چاپگرهای قطبی بر اساس سیستم مختصات قطبی عمل می‌کنند و در مقایسه با دو مدل قبلی، کمتر در بازار رایج هستند. مکانیزم حرکتی در این دستگاه‌ها به گونه‌ای است که اکسترودر در دو محور محدود حرکت کرده و صفحه چاپ به صورت چرخشی (دوار) عمل می‌کند.

- مزایا: از نظر ساختار مکانیکی، این چاپگرها به دلیل استفاده از قطعات کمتر، نسبتاً ساده هستند.
- چالش‌ها: اگرچه ساختار فیزیکی دستگاه ساده است، اما آماده‌سازی فایل‌های مدل (Slicing) برای این چاپگرها فرآیندی پیچیده دارد و به نرم‌افزارهای خاصی نیاز است که بتوانند مختصات سه‌بعدی را به حرکات چرخشی و شعاعی تبدیل کنند.

اجزای اصلی پرینتر سه‌بعدی

همه پرینترهای سه‌بعدی از نظر ساختار بسیار شبیه به هم هستند. آن‌ها معمولاً از بخش‌های زیر تشکیل می‌شوند:

اکسترودر: اکسترودر یا هد چاپ برای ایجاد لایه‌های چاپی از طریق بیرون راندن پلاستیک ذوب‌شده طراحی شده است. ابتدا رشته فیلامنت از طریق یک لوله وارد اکسترودر می‌شود. در این مرحله، فیلامنت به صورت یک رشته پلاستیکی جامد است و در دمای اتاق قرار دارد. سپس از هیت‌سینک عبور می‌کند؛ هیت‌سینک قطعه‌ای است که برای دفع گرمای منتقل‌شده از هیت‌بریک طراحی شده و ناحیه گذار بین فیلامنت جامد و ذوب‌شده را به حداقل می‌رساند. معمولاً یک فن در کنار هیت‌سینک نصب می‌شود تا کارایی خنک‌سازی را افزایش دهد. هیت‌بریک در اصل نوعی لوله رزوه‌دار است که در یک سر باریک‌تر می‌شود تا قطر آن تا حد ممکن کاهش یابد؛ به این ترتیب گرمای کمتری به سمت ناحیه‌ای بالا می‌رود که فیلامنت باید در آنجا جامد باقی بماند.



بلوک گرم‌کن از مواد رسانای گرما، معمولاً آلومینیوم، ساخته می‌شود و شامل یک المنت حرارتی الکتریکی کوچک به همراه یک ترمیستور برای اندازه‌گیری دما است. ماده در داخل بلوک گرم‌کن ذوب می‌شود و سپس به جلو رانده شده و از طریق نازل خارج می‌شود. نازل می‌تواند قطرهای متفاوتی داشته باشد و بسیاری از پرینترها به کاربران اجازه می‌دهند نازل را با نازلی جدید و با قطر متفاوت تعویض کنند.

مراحل آماده‌سازی چاپ

قبل از شروع چاپ، باید مدل سه‌بعدی آماده شود.

مراحل کلی عبارت‌اند از:

۱. طراحی مدل در نرم‌افزار طراحی سه‌بعدی
۲. ذخیره فایل با فرمت **STL**
۳. وارد کردن فایل به نرم‌افزار آماده‌سازی چاپ مانند تینکرکد
٤. تنظیم اندازه و محل قرارگیری قطعه
٥. تبدیل فایل به دستورهای قابل فهم برای پرینتر
٦. آغاز چاپ

نکات ایمنی

هنگام کار با پرینتر سه‌بعدی باید نکات زیر رعایت شود:

- هنگام چاپ به نازل دست نزنید؛ زیرا بسیار داغ است.
- هنگام حرکت دستگاه، دست خود را وارد قسمت‌های متحرک نکنید.
- از کشیدن یا جدا کردن قطعه قبل از پایان چاپ خودداری کنید.
- دستگاه باید روی سطح صاف و محکم قرار گیرد.
- پس از پایان کار، محیط اطراف پرینتر را مرتب نگه دارید.

فعالیت کلاسی

در گروه‌های دو نفره درباره وظیفه هر یک از اجزای پرینتر گفت‌وگو کنید و نتیجه را برای کلاس توضیح دهید.

خلاصه درس

- پرینتر سه‌بعدی از چند بخش اصلی تشکیل شده است.
- نازل، فیلامنت را ذوب کرده و به صورت لایه‌لایه روی صفحه چاپ قرار می‌دهد.

- فیلامنت ماده اولیه چاپ سه‌بعدی است.
- قبل از چاپ، مدل سه‌بعدی باید برای پرینتر آماده‌سازی شود.
- رعایت نکات ایمنی هنگام کار با پرینتر ضروری است.



تمرین

۱. انواع پرینتر سه‌بعدی؟

.....

۲. فیلامنت چیست و چه کاربردی دارد؟

.....

۳. مراحل آماده‌سازی فایل برای چاپ را نام ببرید.

.....



پروژه

یک پرینتر سه‌بعدی را از طریق اینترنت یا آزمایشگاه مدرسه مشاهده کنید و گزارشی کوتاه تهیه کنید.

در گزارش خود به پرسش‌های زیر پاسخ دهید:

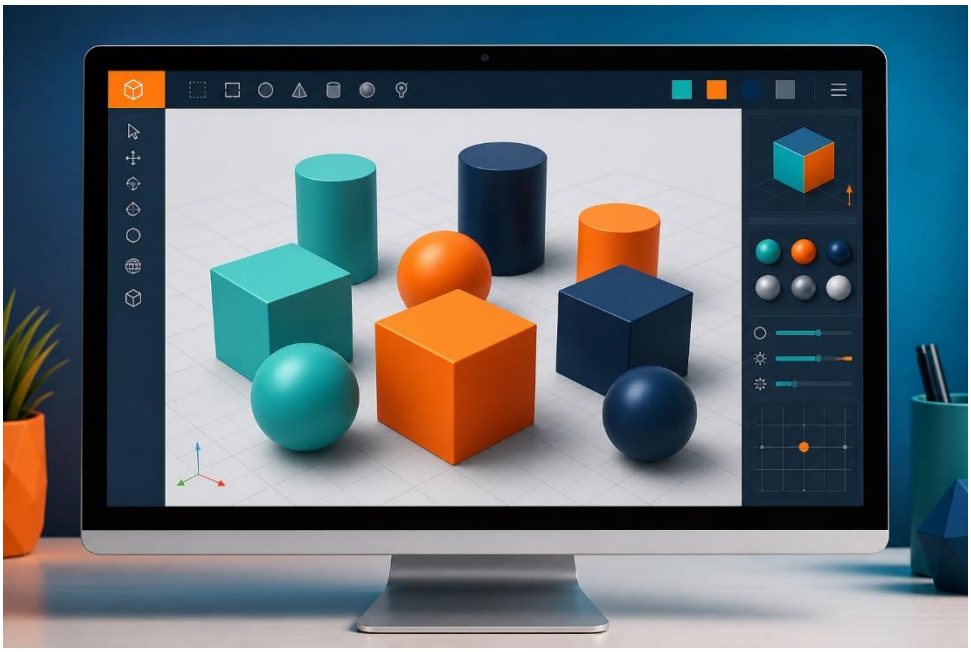
۱. نام پرینتر چیست؟

۲. از چه نوع فیلامنتی استفاده می‌کند؟

۳. مهم‌ترین اجزای آن کدام‌اند؟

٤. این پرینتر بیشتر برای چه کاربردی مناسب است؟

درس۱۹: طراحی سه‌بعدی با نرم‌افزار Tinkercad



در پایان این درس خواهید توانست:

- محیط نرم‌افزار Tinkercad را بشناسید.
- اشکال سه‌بعدی ساده را در صفحه طراحی قرار دهید.
- اندازه، محل و جهت اشکال را تغییر دهید.
- با ابزارهای Group و Hole مدل‌های جدید ایجاد کنید.
- یک مدل ساده را طراحی کرده و با فرمت STL ذخیره کنید.



پس از آشنایی با پرینتر سه‌بعدی، اکنون زمان آن رسیده است که خودمان یک مدل سه‌بعدی طراحی کنیم.
برای این کار از نرم‌افزار Tinkercad استفاده می‌کنیم. این نرم‌افزار محیطی ساده و کاربردی دارد و به کاربران کمک می‌کند بدون نیاز به برنامه‌نویسی، مدل‌های سه‌بعدی مختلف را طراحی کنند.
بسیاری از دانش‌آموزان، طراحان و علاقه‌مندان به چاپ سه‌بعدی، نخستین تجربه طراحی خود را با این نرم‌افزار آغاز می‌کنند.

- اگر بخواهید یک جاکلیدی برای خود طراحی کنید، از چه شکل‌هایی استفاده می‌کنید؟
- آیا می‌توان با کنار هم قرار دادن چند شکل ساده، یک وسیله جدید ساخت؟
- برای اینکه پرینتر بتواند یک مدل را چاپ کند، ابتدا چه کاری باید انجام دهیم؟
- اگر بخواهید روی یک قطعه نام خود را بنویسید، چگونه این کار را انجام می‌دهید؟



نرم‌افزارهای طراحی سه‌بعدی

امروزه طیف گسترده‌ای از برنامه‌های کاربردی برای طراحی و مدل‌سازی سه‌بعدی در دسترس کاربران قرار دارد که هریک با توجه به سطح نیاز و مهارت، قابلیت‌های متفاوتی را ارائه می‌دهند:

- ابزارهای ساده و مبتدی: برخی برنامه‌ها مانند «تینک‌کد» (TinkerCad)، بسیار ساده و زودآموز هستند و اغلب به‌صورت تحت وب اجرا می‌شوند؛ این ابزارها برای کسانی که تازه قصد ورود به دنیای چاپ سه‌بعدی را دارند، گزینه‌ای بسیار مناسب محسوب می‌شوند.
- مدل‌سازی پارامتریک: برای کاربرانی که به طراحی بر مبنای محاسبات و پارامترهای هندسی علاقه‌مند هستند، نرم‌افزاری نظیر «اوپن‌اسکد» (OpenSCAD) امکانات ویژه‌ای فراهم می‌کند.
- ابزارهای حرفه‌ای: برای پروژه‌های پیچیده مهندسی و صنعتی، استفاده از برنامه‌های قدرتمندی مانند «اتودسک فیوژن ۳۶۰» (Autodesk Fusion ۳۶۰) پیشنهاد می‌شود که ابزارهای کامل و دقیقی را در اختیار طراحان قرار می‌دهد.

نرم‌افزار تینکرکد چیست؟

Tinkercadیک نرم‌افزار طراحی سه‌بعدی آنلاین است که توسط شرکت Autodesk توسعه یافته است.
در این نرم‌افزار، مدل‌ها با استفاده از اشکال هندسی مانند مکعب، استوانه، کره و منشور ساخته می‌شوند. با تغییر اندازه، جابه‌جایی و ترکیب این اشکال می‌توان مدل‌های گوناگونی طراحی کرد.

آشنایی با محیط نرم‌افزار

پس از ورود به نرم‌افزار، بخش‌های زیر را مشاهده می‌کنید.

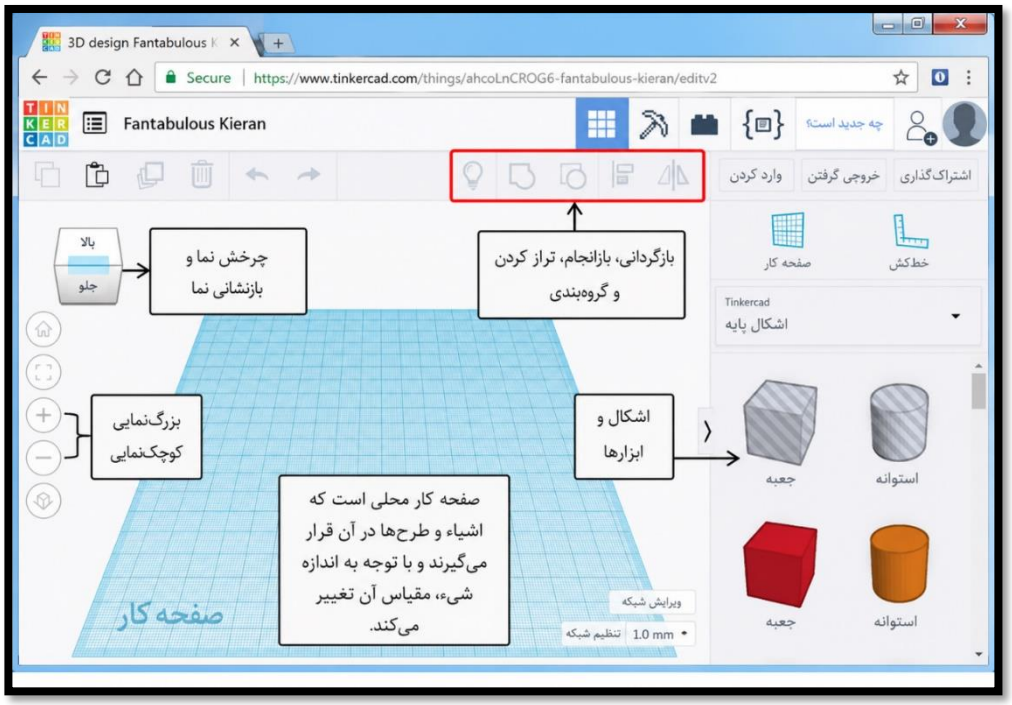
صفحه طراحی(Workplane) : محلی است که مدل روی آن ساخته می‌شود.

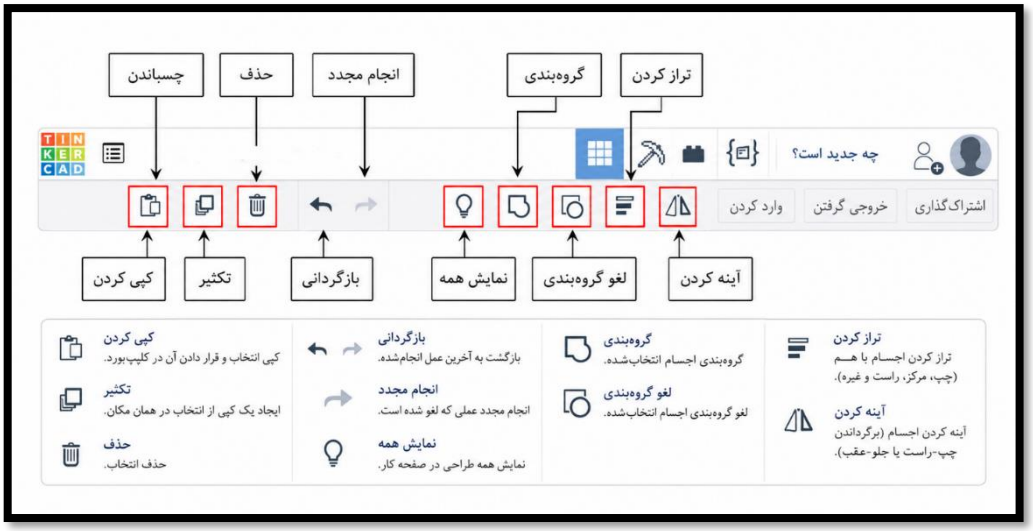
اشکال سه‌بعدی(Basic Shapes) : در این قسمت اشکال مختلف مانند مکعب، استوانه، کره و سایر حجم‌ها قرار دارند و می‌توان آن‌ها را وارد صفحه طراحی کرد.

ابزار جابه‌جایی: برای تغییر محل قرار گرفتن مدل روی صفحه استفاده می‌شود.

ابزار تغییر اندازه: با کشیدن دسته‌های اطراف شکل، طول، عرض و ارتفاع آن تغییر می‌کند.

ابزار چرخش: این ابزار برای تغییر جهت قرار گرفتن مدل استفاده می‌شود.





ساخت مدل با اشکال ساده

در Tinkercad تقریباً همه مدل‌ها از ترکیب چند شکل ساده ساخته می‌شوند.

برای مثال یک جاکلیدی می‌تواند از:

- یک مکعب
- یک استوانه
- یک سوراخ دایره‌ای

تشکیل شود. ابتدا اشکال را روی صفحه قرار می‌دهیم، سپس اندازه آن‌ها را تغییر داده و در محل مناسب قرار می‌دهیم.

ابزار Hole

گاهی لازم است بخشی از مدل حذف شود.

در Tinkercad این کار با ابزار Hole انجام می‌شود.

ابتدا شکل موردنظر را به حالت Hole تبدیل می‌کنیم، سپس آن را روی مدل قرار داده و هر دو شکل را انتخاب می‌کنیم.

پس از استفاده از دستور **Group**، قسمت موردنظر از مدل حذف می‌شود.

ابزار Goup

ابزار Group چند شکل را به یک مدل واحد تبدیل می‌کند.

این ابزار باعث می‌شود تمام اجزای مدل به صورت یک قطعه واحد جابه‌جا یا ویرایش شوند.

ذخیره فایل برای چاپ

پس از پایان طراحی، باید مدل ذخیره شود.

برای چاپ سه‌بعدی معمولاً از فرمت STL استفاده می‌شود.

فعالیت کلاسی

یک مکعب و استوانه را وارد صفحه طراحی کنید و اندازه آن را به دلخواه تغییر دهید.

همه اشکال را انتخاب کرده و با استفاده از دستور Group آن‌ها را به یک مدل تبدیل کنید.

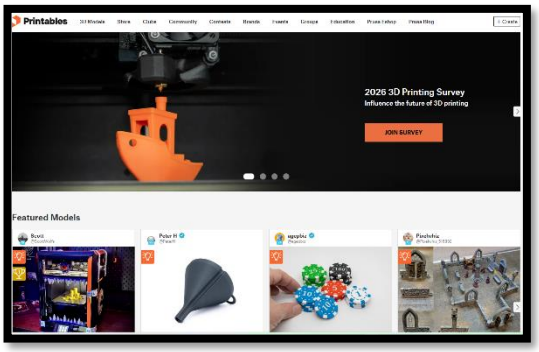
مدل نهایی را با فرمت STL ذخیره کنید.

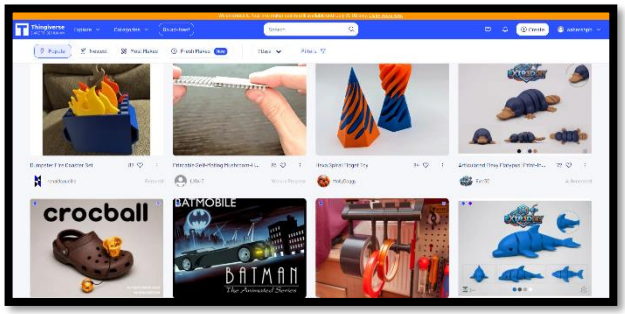
کتابخانه‌ها و پایگاه‌های آنلاین مدل‌های سه‌بعدی

سریع‌ترین و ساده‌ترین روش برای آغاز کار با پرینترهای سه‌بعدی، استفاده از مدل‌های آماده و رایگانی است که در شبکه جهانی اینترنت در دسترس قرار دارند. این مدل‌ها معمولاً با پسوندهای رایجِ (STL) یا (OBJ) ارائه می‌شوند که استانداردهای جهانی برای تبادل داده‌های هندسی سه‌بعدی محسوب می‌شوند.وب‌سایت‌های متعددی وجود دارند که مجموعه‌ای گسترده از این مدل‌ها را برای بارگیری فراهم کرده‌اند. برترین و کاربردی‌ترینِ این پایگاه‌ها در فهرست زیر معرفی شده‌اند:

- پروساپرینترز
- آدرس وب‌سایت
  www.prusaprinters.org
- نوع مدل‌ها: رایگان

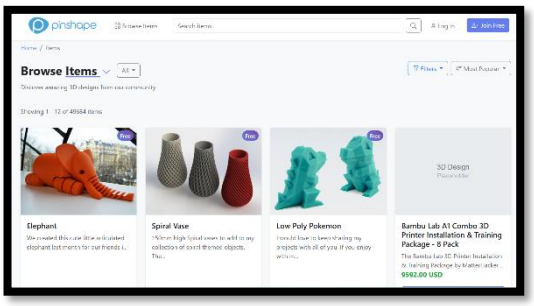
- تینگی‌ورس
- آدرس وب‌سایت www.thingiverse.com
- نوع مدل‌ها: رایگان

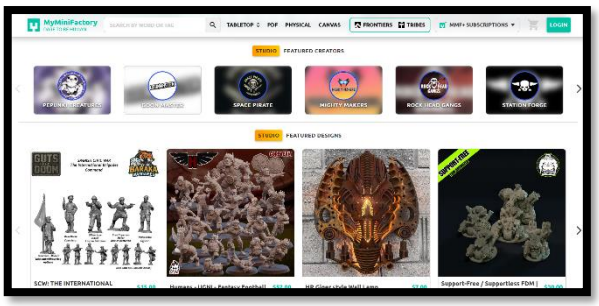




- پروساپرینترز
- آدرس وب‌سایت
  www.pinshape.com
- نوع مدل‌ها: رایگان و غیر رایگان

- پروساپرینترز
- آدرس وب سایت
  www.myminifactory.com
- نوع مدل‌ها: رایگان و غیر رایگان





دستیار هوشمند

استفاده از اینترنت و هوش مصنوعی برای طراحی بهتر

گاهی اوقات ایده‌ای برای طراحی داریم، اما نمی‌دانیم شکل آن چگونه است یا از چه قسمت‌هایی تشکیل شده است. در این شرایط می‌توانیم از موتورهای جست‌وجو و ابزارهای هوش مصنوعی کمک بگیریم.

جست‌وجوی اینترنتی

پیش از شروع طراحی، می‌توانید تصاویر وسیله مورد نظر را در اینترنت جست‌وجو کنید. مشاهده تصاویر از زوایای مختلف به شما کمک می‌کند شکل و ابعاد قطعه را بهتر بشناسید.

برای مثال اگر قصد طراحی یک **جاکلیدی، گلدان یا پایه تلفن همراه** را دارید، ابتدا نمونه‌های مختلف آن را مشاهده کنید و سپس طراحی خود را آغاز کنید.

کمک گرفتن از هوش مصنوعی

ابزارهای هوش مصنوعی می‌توانند در مراحل مختلف طراحی به شما کمک کنند. برای مثال می‌توانید از آن‌ها بخواهید:

- ایده‌هایی برای طراحی یک وسیله ارائه دهند.
- مراحل طراحی یک مدل در Tinkercad را توضیح دهند.
- اندازه تقریبی یک قطعه را پیشنهاد کنند.
- روش ساخت یک مدل را مرحله‌به‌مرحله بیان کنند.

نمونه پرامپت

می‌خواهم در نرم‌افزار Tinkercad یک جامدادی ساده طراحی کنم. مراحل طراحی را به زبان ساده و مناسب دانش‌آموز پایه هشتم توضیح بده.

یا

برای طراحی یک پایه تلفن همراه در Tinkercad از چه اشکال هندسی استفاده کنم؟

یا

یک جاکلیدی با طرح ستاره پیشنهاد بده و مراحل ساخت آن را توضیح بده.

از تصویر تا مدل سه‌بعدی

بعضی از ابزارهای هوش مصنوعی می‌توانند از روی یک تصویر، مدل سه‌بعدی تولید کنند یا فایل‌هایی برای ویرایش در نرم‌افزارهای طراحی ایجاد نمایند. البته در بسیاری از موارد لازم است مدل تولیدشده را بررسی و اصلاح کنید تا برای چاپ مناسب باشد.

بنابراین بهتر است همیشه پیش از چاپ:

- اندازه مدل را بررسی کنید.
- ضخامت قطعات را کنترل کنید.
- از قابل چاپ بودن مدل مطمئن شوید.

نکته: هوش مصنوعی یک **دستیار طراحی** است، نه جایگزین خلاقیت شما. بهترین نتیجه زمانی به دست می‌آید که ابتدا خودتان ایده‌پردازی کنید و سپس از هوش مصنوعی برای کامل‌تر کردن طرح یا رفع اشکال‌های آن کمک بگیرید.

خلاصه درس

- Tinkercadیک نرم‌افزار ساده برای طراحی سه‌بعدی است.
- مدل‌های سه‌بعدی از ترکیب اشکال هندسی ساخته می‌شوند.
- با ابزارهایMove ، Rotateو Scale می‌توان مدل را ویرایش کرد.
- ابزار Hole برای حذف بخشی از مدل و ابزار Group برای ترکیب اشکال به کار می‌رود.
- پس از پایان طراحی، مدل با فرمت STL ذخیره می‌شود.

پروژه

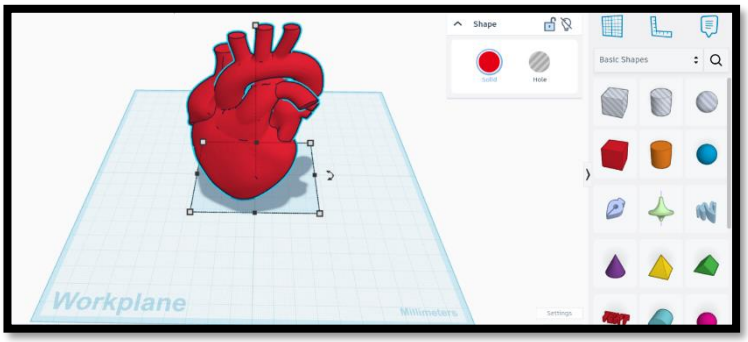
یکی از مدل‌های زیر را در نرم‌افزار تینکرکد طراحی کنید:

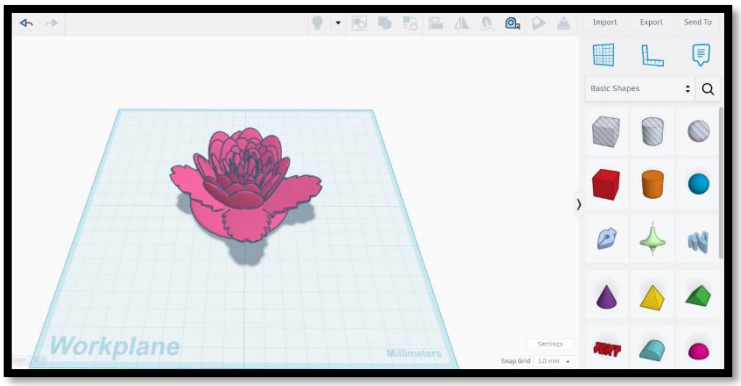
- پایه تلفن همراه

پس از پایان طراحی:

۱. فایل را با فرمت STLذخیره کنید.
۲. از محیط طراحی یک تصویر تهیه کنید.
۳. توضیح دهید در طراحی خود از چه اشکالی استفاده کرده‌اید.

نمون فایل:





بخش سوم: ذهن افزار

Mindware

درس۲۰: آشنایی با هوش مصنوعی و کاربردهای آن



در پایان این درس خواهید توانست:

- مفهوم هوش مصنوعی را توضیح دهید.
- تفاوت هوش مصنوعی با برنامه‌های معمولی را بیان کنید.
- چند نمونه از کاربردهای هوش مصنوعی در زندگی روزمره را معرفی کنید.
- نقش هوش مصنوعی را در آینده فناوری درک کنید.
- نمونه‌هایی از ابزارهای مبتنی بر هوش مصنوعی را شناسایی کنید.

در سال‌های اخیر، هوش مصنوعی به یکی از مهم‌ترین فناوری‌های جهان تبدیل شده است. بسیاری از ابزارها و برنامه‌هایی که هر روز از آن‌ها استفاده می‌کنیم، مانند موتورهای جست‌وجو، مترجم‌های آنلاین، دستیارهای صوتی و سامانه‌های پیشنهاددهنده فیلم یا موسیقی، از هوش مصنوعی بهره می‌برند.

هوش مصنوعی تلاش می‌کند رایانه‌ها و ماشین‌ها را به گونه‌ای طراحی کند که بتوانند برخی از کارهایی را انجام دهند که معمولاً به هوش انسان نیاز دارد؛ مانند یادگیری، تصمیم‌گیری، تشخیص تصویر و درک گفتار.

در این درس با مفهوم هوش مصنوعی، ویژگی‌های آن و کاربردهای مهم آن آشنا می‌شوید.

آیا تاکنون از برنامه‌ای استفاده کرده‌اید که بتواند به پرسش‌های شما پاسخ دهد؟

به نظر شما تلفن همراه چگونه چهره افراد را تشخیص می‌دهد؟

فکر می‌کنید هوش مصنوعی در چه کارهای دیگری می‌تواند به انسان کمک کند؟

هوش مصنوعی چیست؟

هوش مصنوعی Artificial Intelligence یا AI شاخه‌ای از علوم رایانه است که به طراحی سیستم‌هایی می‌پردازد که بتوانند وظایفی را انجام دهند که معمولاً به هوش انسان نیاز دارند.

برخی از این توانایی‌ها عبارت‌اند از:

- یادگیری از اطلاعات
- حل مسئله
- تشخیص تصویر و صدا
- درک زبان انسان
- تصمیم‌گیری در شرایط مختلف

هوش مصنوعی مانند انسان فکر یا احساس نمی‌کند، بلکه با استفاده از داده‌ها و الگوریتم‌ها، الگوها را شناسایی کرده و بر اساس آن‌ها تصمیم می‌گیرد.

تفاوت هوش مصنوعی با برنامه‌های معمولی

برنامه‌های رایانه‌ای معمولی فقط دستورهایی را اجرا می‌کنند که برنامه‌نویس از قبل برای آن‌ها نوشته است.

اما سامانه‌های هوش مصنوعی می‌توانند با بررسی اطلاعات، الگوها را یاد بگیرند و عملکرد خود را بهبود دهند.

برای مثال:

یک ماشین‌حساب همیشه طبق فرمول‌های مشخص محاسبه می‌کند .

اما یک برنامه تشخیص چهره با مشاهده تصاویر فراوان، چهره افراد را یاد می‌گیرد و آن‌ها را شناسایی می‌کند .

کاربردهای هوش مصنوعی

امروزه هوش مصنوعی در بسیاری از زمینه‌ها به کار می‌رود، از جمله:

آموزش

- آموزش هوشمند
- تصحیح خودکار آزمون‌ها
- پاسخ‌گویی به پرسش‌های دانش‌آموزان

پزشکی

- کمک به تشخیص بیماری‌ها
- بررسی تصاویر پزشکی
- پیشنهاد روش‌های درمان

حمل‌ونقل

- خودروهای هوشمند
- مسیریابی دقیق
- مدیریت ترافیک

ارتباطات

- ترجمه خودکار زبان‌ها
- دستیارهای صوتی
- پیشنهاد محتوا در شبکه‌های اجتماعی

سرگرمی

- پیشنهاد فیلم و موسیقی
- طراحی بازی‌های هوشمند
- تولید تصویر و متن

مزایای هوش مصنوعی

هوش مصنوعی می‌تواند:

- سرعت انجام کارها را افزایش دهد.
- دقت برخی فعالیت‌ها را بیشتر کند.
- در انجام کارهای تکراری به انسان کمک کند.
- حجم زیادی از اطلاعات را در زمان کوتاه بررسی کند.

محدودیت‌های هوش مصنوعی

با وجود توانایی‌های فراوان، هوش مصنوعی محدودیت‌هایی نیز دارد.

برای مثال:

احساسات واقعی ندارد.

ممکن است بر اساس اطلاعات نادرست، نتیجه اشتباه ارائه دهد.

برای عملکرد مناسب به داده‌های کافی و صحیح نیاز دارد.



آیا می‌دانید؟

بسیاری از موتورهای جست‌وجو، مترجم‌های آنلاین و دستیارهای صوتی از هوش مصنوعی استفاده می‌کنند.



فعالیت کلاسی

شناسایی کاربردهای هوش مصنوعی

- به‌صورت گروهی سه نرم‌افزار یا وب‌سایت مبتنی بر هوش مصنوعی را انتخاب کنید.
- کاربرد هر یک را بررسی کنید.
- مشخص کنید این ابزار چگونه به کاربران کمک می‌کند.
- نتایج را در کلاس ارائه دهید.

➢ هوش مصنوعی **جایگزین انسان نیست**، بلکه ابزاری برای کمک به انجام بهتر و سریع‌تر بسیاری از فعالیت‌هاست. انسان همچنان نقش اصلی را در تصمیم‌گیری، خلاقیت و نظارت بر عملکرد این فناوری بر عهده دارد.



خلاصه درس

- هوش مصنوعی شاخه‌ای از علوم رایانه است.
- سامانه‌های هوش مصنوعی می‌توانند از داده‌ها یاد بگیرند و تصمیم‌گیری کنند.
- هوش مصنوعی در آموزش، پزشکی، حمل‌ونقل، ارتباطات و سرگرمی کاربرد دارد.
- این فناوری مزایای فراوانی دارد، اما محدودیت‌هایی نیز دارد و باید به‌درستی از آن استفاده شود.



تمرین

۱. هوش مصنوعی را به زبان خود تعریف کنید.

.....

۲. دو تفاوت بین برنامه‌های معمولی و سامانه‌های هوش مصنوعی را بنویسید.

.....

۳. چهار کاربرد هوش مصنوعی را نام ببرید.

.....

٤. به نظر شما هوش مصنوعی در آینده چه تأثیری بر زندگی انسان خواهد داشت؟

.....

۵. یکی از ابزارهای هوش مصنوعی که تاکنون از آن استفاده کرده‌اید معرفی کرده و کاربرد آن را توضیح دهید.

.....



پروژه

یک پوستر یا ارائه کوتاه با موضوع **هوش مصنوعی در زندگی من** تهیه کنید.

در این پروژه:

- سه کاربرد هوش مصنوعی را معرفی کنید.
- مزایا و محدودیت‌های آن را بیان کنید.
- نظر خود را درباره استفاده از هوش مصنوعی در آینده بنویسید.



چالش

تحقیق کنید «یادگیری ماشین (Machine Learning) و «هوش مصنوعی» چه تفاوتی با یکدیگر دارند.

درس۲۱: آینده هوش مصنوعی و چالش‌های آن



در پایان این درس خواهید توانست:

- درباره آینده هوش مصنوعی و تأثیر آن بر زندگی انسان گفت‌وگو کنید.
- برخی فرصت‌ها و چالش‌های توسعه هوش مصنوعی را توضیح دهید.
- اهمیت تفکر انتقادی در استفاده از فناوری‌های نوین را درک کنید.
- دیدگاه خود را درباره آینده هوش مصنوعی بیان کنید.


هوش مصنوعی هر روز پیشرفته‌تر می‌شود و توانایی انجام کارهای بیشتری را پیدا می‌کند. امروزه این فناوری در تشخیص تصویر، ترجمه زبان، رانندگی خودروهای هوشمند و حتی تولید متن و تصویر به کار می‌رود. بسیاری از پژوهشگران معتقدند که در آینده، هوش مصنوعی نقش مهم‌تری در زندگی انسان خواهد داشت.

با این حال، پیشرفت این فناوری پرسش‌ها و چالش‌های جدیدی نیز به همراه دارد. آیا هوش مصنوعی می‌تواند مانند انسان فکر کند؟ آیا روزی ماشین‌ها همه کارهای انسان را انجام خواهند داد؟ چگونه می‌توان از این فناوری به شیوه‌ای ایمن و مسئولانه استفاده کرد؟

در این درس با برخی از فرصت‌ها و چالش‌های آینده هوش مصنوعی آشنا می‌شوید.

به پرسش‌های زیر پاسخ دهید.

به نظر شما هوش مصنوعی در آینده چه کارهای جدیدی خواهد توانست انجام دهد؟

آیا ماشین‌ها می‌توانند جای همه انسان‌ها را بگیرند؟ چرا؟

اگر یک ربات بتواند مانند انسان صحبت کند، آیا او واقعاً مانند انسان فکر می‌کند؟



آینده هوش مصنوعی

پیشرفت هوش مصنوعی با سرعت زیادی ادامه دارد و هر سال ابزارها و برنامه‌های جدیدی معرفی می‌شوند. کارشناسان پیش‌بینی می‌کنند که در آینده، هوش مصنوعی بتواند در زمینه‌هایی مانند:

- آموزش
- پزشکی
- حمل‌ونقل
- کشاورزی
- صنعت
- مدیریت شهرهای هوشمند

نقش گسترده‌تری داشته باشد. هدف اصلی این فناوری، کمک به انسان برای انجام بهتر، سریع‌تر و دقیق‌تر کارهاست.

هوش مصنوعی عمومی

امروزه بیشتر سامانه‌های هوش مصنوعی تنها برای انجام یک وظیفه مشخص طراحی شده‌اند؛ برای مثال، برخی فقط تصویر را تشخیص می‌دهند و برخی فقط متن تولید می‌کنند.

پژوهشگران در حال بررسی امکان ساخت هوش مصنوعی عمومی (AGI) هستند؛ سامانه‌ای که بتواند مانند انسان در زمینه‌های مختلف یاد بگیرد، استدلال کند و مسائل گوناگون را حل کند.

در حال حاضر چنین سامانه‌ای ساخته نشده است و هنوز در حد پژوهش و بررسی قرار دارد.

هوش مصنوعی فراتر از انسان

برخی دانشمندان درباره آینده‌ای صحبت می‌کنند که در آن، هوش مصنوعی بتواند در برخی زمینه‌ها از توانایی‌های انسان فراتر رود. به این مفهوم گاهی **ابرهوش مصنوعی (ASI)** گفته می‌شود.

این موضوع هنوز تحقق نیافته و درباره امکان یا زمان دستیابی به آن، دیدگاه‌های متفاوتی وجود دارد. به همین دلیل، باید آن را یک موضوع پژوهشی و آینده‌نگر دانست، نه یک واقعیت امروزی.

ربات‌های هوشمند

پیشرفت هوش مصنوعی باعث شده است ربات‌ها توانایی‌های بیشتری پیدا کنند.

برخی ربات‌ها می‌توانند:

- اشیا را جابه‌جا کنند.
- با انسان گفت‌وگو کنند.
- چهره افراد را تشخیص دهند.
- در کارخانه‌ها یا مراکز درمانی به انسان کمک کنند.

با وجود این، ربات‌ها همچنان برای انجام بسیاری از کارها به برنامه‌ریزی و نظارت انسان نیاز دارند.

فرصت‌های آینده

هوش مصنوعی می‌تواند فرصت‌های فراوانی ایجاد کند، از جمله:

- کمک به تشخیص سریع‌تر بیماری‌ها
- کاهش خطا در برخی فعالیت‌ها
- انجام کارهای خطرناک به جای انسان
- کمک به آموزش شخصی‌سازی‌شده
- مدیریت بهتر منابع و انرژی

چالش‌های آینده

هم‌زمان با پیشرفت هوش مصنوعی، پرسش‌ها و چالش‌های مهمی نیز مطرح می‌شود.

برای مثال:

- چگونه از اطلاعات شخصی کاربران محافظت کنیم؟
- اگر هوش مصنوعی تصمیم اشتباهی بگیرد، چه کسی مسئول است؟
- آیا ممکن است برخی شغل‌ها در آینده تغییر کنند؟
- چگونه اطمینان پیدا کنیم که هوش مصنوعی به نفع انسان استفاده می‌شود؟

پاسخ به این پرسش‌ها نیازمند همکاری دانشمندان، قانون‌گذاران و کاربران است.

آیا هوش مصنوعی احساس دارد؟

برخی سامانه‌های هوش مصنوعی می‌توانند گفت‌وگو کنند یا احساسات را شبیه‌سازی کنند، اما این به معنای داشتن احساس واقعی نیست.

هوش مصنوعی بر اساس داده‌ها و الگوریتم‌ها عمل می‌کند و نمی‌توان گفت مانند انسان دارای آگاهی یا احساس است.

این موضوع همچنان یکی از پرسش‌های علمی و فلسفی مهم به شمار می‌رود.



آیا می‌دانید؟

بسیاری از پژوهشگران معتقدند که هوش مصنوعی در آینده می‌تواند به حل مسائل مهمی مانند تغییرات اقلیمی، مدیریت منابع و درمان برخی بیماری‌ها کمک کند.

فعالیت کلاسی : گفت‌وگو درباره آینده هوش مصنوعی

کلاس را به گروه‌های کوچک تقسیم کنید.

هر گروه یکی از موضوع‌های زیر را انتخاب کند:

- هوش مصنوعی در آموزش
- هوش مصنوعی در پزشکی
- هوش مصنوعی در حمل‌ونقل
- ربات‌های هوشمند

فرصت‌ها و چالش‌های موضوع انتخاب‌شده را فهرست کنید. نتایج را در کلاس ارائه کرده و درباره آن گفت‌وگو کنید.



خلاصه درس

- هوش مصنوعی در آینده نقش گسترده‌تری در زندگی انسان خواهد داشت.
- هوش مصنوعی عمومی و ابرهوش مصنوعی، موضوعاتی هستند که هنوز در مرحله پژوهش قرار دارند.
- ربات‌های هوشمند می‌توانند در انجام بسیاری از کارها به انسان کمک کنند.
- توسعه هوش مصنوعی علاوه بر فرصت‌ها، چالش‌هایی نیز به همراه دارد.
- استفاده مسئولانه از این فناوری برای آینده جامعه اهمیت زیادی دارد.

تمرین

۱. دو فرصت مهم هوش مصنوعی در آینده را نام ببرید.

.....

۲. منظور از هوش مصنوعی عمومی چیست؟

.....

۳. چرا گفته می‌شود هوش مصنوعی هنوز مانند انسان احساس واقعی ندارد؟

.....

٤ .دو چالش مهم توسعه هوش مصنوعی را بیان کنید.

.....

٥ .به نظر شما مهم‌ترین کاربرد هوش مصنوعی در آینده چه خواهد بود؟ دلیل خود را بنویسید.

.....

پروژه

موضوع: آینده هوش مصنوعی

یک گزارش کوتاه یا ارائه کلاسی تهیه کنید که در آن:

- یکی از کاربردهای آینده هوش مصنوعی را معرفی کنید.
- مزایا و چالش‌های آن را بررسی کنید.

درس۲۲: آشنایی با متاورس، واقعیت مجازی و واقعیت افزوده



در پایان این درس خواهید توانست:

- مفهوم متاورس را توضیح دهید.
- تفاوت متاورس با اینترنت امروزی را بیان کنید.
- با فناوری‌های واقعیت مجازی (VR) و واقعیت افزوده (AR) آشنا شوید.
- تجهیزات موردنیاز برای ورود به متاورس را معرفی کنید.
- کاربردهای اولیه متاورس را در زندگی روزمره بیان کنید.


فناوری هر روز روش زندگی، یادگیری و ارتباط ما را تغییر می‌دهد. یکی از فناوری‌های نوظهور که در سال‌های اخیر توجه بسیاری را به خود جلب کرده است، متاورس (Metaverse) است.

متاورس فضایی مجازی و سه‌بعدی است که افراد می‌توانند با استفاده از اینترنت و ابزارهای دیجیتال در آن حضور پیدا کنند، با دیگران ارتباط برقرار کنند، آموزش ببینند، بازی کنند، خرید انجام دهند یا حتی در جلسات کاری شرکت کنند.

در این درس با مفهوم متاورس، فناوری‌های مرتبط با آن و تفاوت آن با اینترنت امروزی آشنا می‌شوید.

به پرسش‌های زیر پاسخ دهید.

اگر بتوانید وارد یک دنیای مجازی شوید، دوست دارید چه فعالیتی انجام دهید؟

آیا تاکنون از عینک‌های واقعیت مجازی استفاده کرده‌اید یا فیلمی درباره آن دیده‌اید؟

به نظر شما آینده اینترنت چگونه خواهد بود؟



متاورس چیست؟

واژه متاورس (Metaverse) از دو بخش تشکیل شده است:

Meta به معنای «فراتر»

Verseبه معنای «جهان»

بنابراین متاورس را می‌توان جهانی فراتر از فضای اینترنت امروزی دانست.

در متاورس، کاربران تنها صفحات وب را مشاهده نمی‌کنند، بلکه با استفاده از شخصیت مجازی (Avatar) وارد یک محیط سه‌بعدی می‌شوند و می‌توانند با افراد دیگر و اشیای مجازی تعامل داشته باشند.

متاورس چگونه کار می‌کند؟

در متاورس، هر کاربر یک آواتار (Avatar) یا شخصیت دیجیتالی برای خود انتخاب می‌کند.

این آواتار می‌تواند:

- حرکت کند.
- با دیگران گفت‌وگو کند.
- در کلاس‌های آموزشی شرکت کند.
- بازی انجام دهد.
- از فروشگاه‌های مجازی خرید کند.
- همه این فعالیت‌ها در محیطی سه‌بعدی و از طریق اینترنت انجام می‌شود.

تفاوت اینترنت امروزی و متاورس

امروزه بیشتر فعالیت‌های اینترنتی از طریق مشاهده صفحات وب یا استفاده از برنامه‌های مختلف انجام می‌شود.

اما در متاورس، کاربران احساس می‌کنند **در داخل محیط دیجیتال حضور دارند** و می‌توانند به‌صورت تعاملی با محیط ارتباط برقرار کنند.

به همین دلیل، متاورس را نسل جدیدی از اینترنت می‌دانند.

واقعیت مجازی(Virtual Reality)

واقعیت مجازی (VR) فناوری‌ای است که کاربر را وارد یک محیط کاملاً شبیه‌سازی‌شده می‌کند.

برای استفاده از این فناوری معمولاً از:

- هدست واقعیت مجازی
- کنترل‌کننده‌های حرکتی
- هدفون

استفاده می‌شود.

کاربر هنگام استفاده از هدست، محیط واقعی اطراف خود را کمتر می‌بیند و فضای مجازی را تجربه می‌کند.

نمونه کاربردها

- بازی‌های رایانه‌ای
- آموزش پزشکی
- شبیه‌ساز رانندگی و پرواز
- بازدید مجازی از مکان‌های تاریخی

واقعیت افزوده(Augmented Reality)

واقعیت افزوده (AR) فناوری‌ای است که اطلاعات یا تصاویر دیجیتال را روی دنیای واقعی نمایش می‌دهد.

در این فناوری، کاربر همچنان محیط واقعی را مشاهده می‌کند، اما اطلاعات مجازی نیز به آن افزوده می‌شود.

برای استفاده از واقعیت افزوده معمولاً به تلفن همراه یا تبلت نیاز است.

نمونه کاربردها

- نمایش اطلاعات محصولات در فروشگاه‌ها
- آموزش علوم
- مسیریابی
- بازی‌های موبایلی

تفاوت واقعیت مجازی و واقعیت افزوده

واقعیت افزوده(AR)	واقعیت مجازی(VR)
اطلاعات مجازی روی محیط واقعی نمایش داده می‌شود.	کاربر وارد محیط کاملاً مجازی می‌شود.
اغلب با تلفن همراه یا تبلت قابل استفاده است.	معمولاً به هدست نیاز دارد.
محیط واقعی همچنان قابل مشاهده است.	محیط واقعی دیده نمی‌شود.

تجهیزات موردنیاز برای ورود به متاورس

برای استفاده از متاورس، بسته به نوع فعالیت، ممکن است از تجهیزات زیر استفاده شود:

- رایانه یا تلفن همراه
- اینترنت پرسرعت
- هدست واقعیت مجازی (VR)
- عینک واقعیت افزوده (AR)
- هدفون
- کنترل‌کننده‌های حرکتی

این تجهیزات باعث می‌شوند تعامل کاربر با محیط مجازی طبیعی‌تر و جذاب‌تر باشد.

کاربردهای اولیه متاورس

متاورس می‌تواند در زمینه‌های مختلف مورد استفاده قرار گیرد، از جمله:

- آموزش مجازی
- بازی‌های سه‌بعدی
- برگزاری جلسات آنلاین
- خرید از فروشگاه‌های مجازی
- بازدید از موزه‌ها و مکان‌های تاریخی

با پیشرفت فناوری، کاربردهای متاورس نیز گسترده‌تر خواهد شد.

خلاصه درس

- متاورس محیطی سه‌بعدی و تعاملی در فضای دیجیتال است.
- کاربران با استفاده از آواتار در متاورس حضور پیدا می‌کنند.
- واقعیت مجازی (VR) کاربر را وارد محیطی کاملاً مجازی می‌کند.
- واقعیت افزوده (AR) اطلاعات دیجیتال را به محیط واقعی اضافه می‌کند.
- برای ورود به متاورس از تجهیزاتی مانند هدست، اینترنت و رایانه استفاده می‌شود.
- متاورس در آموزش، بازی، ارتباطات و بسیاری از فعالیت‌های دیگر کاربرد دارد.

تمرین

۱. متاورس را به زبان خود تعریف کنید.

.....

۲. تفاوت واقعیت مجازی و واقعیت افزوده را توضیح دهید.

.....

۳. سه وسیله موردنیاز برای استفاده از متاورس را نام ببرید.

.....

٤. دو کاربرد متاورس را در آموزش یا زندگی روزمره بیان کنید.

.....

پروژه

طراحی یک کلاس درس در متاورس

به‌صورت گروهی، یک کلاس درس مجازی طراحی کنید و موارد زیر را مشخص نمایید:

۱. محیط کلاس چگونه است؟

۲. دانش‌آموزان چگونه وارد کلاس می‌شوند؟

۳. معلم چگونه آموزش می‌دهد؟

٤. چه ابزارهایی برای یادگیری استفاده می‌شود؟

٥. چه تفاوتی با کلاس واقعی دارد؟

٦. نتیجه کار را در قالب نقاشی، پوستر یا اسلاید به کلاس ارائه دهید.

چالش

تحقیق کنید در حال حاضر کدام کشورها یا شرکت‌ها بیشترین سرمایه‌گذاری را در توسعه متاورس انجام داده‌اند و نمونه‌ای از فعالیت آن‌ها را معرفی کنید.

درس۲۳: فناوری‌های پشتیبان متاورس و کاربردهای آن



در پایان این درس خواهید توانست:

- نقش هوش مصنوعی در متاورس را توضیح دهید.
- مفهوم اینترنت اشیا (IoT) را بیان کنید.
- کاربردهای متاورس را در زمینه‌های مختلف معرفی کنید.
- مزایا و چالش‌های استفاده از متاورس را بررسی کنید.
- درباره تأثیر متاورس بر سبک زندگی آینده اظهار نظر کنید.

متاورس تنها یک محیط سه‌بعدی نیست، بلکه مجموعه‌ای از فناوری‌های مختلف است که در کنار یکدیگر تجربه‌ای جدید از دنیای دیجیتال را ایجاد می‌کنند. فناوری‌هایی مانند هوش مصنوعی، اینترنت اشیا، اینترنت پرسرعت و رایانش ابری باعث می‌شوند کاربران بتوانند در محیطی هوشمند و تعاملی فعالیت کنند.

در این درس با مهم‌ترین فناوری‌های پشتیبان متاورس، کاربردهای آن و همچنین مزایا و چالش‌های این فناوری آشنا می‌شوید.

به پرسش‌های زیر پاسخ دهید.

آیا وسایلی در خانه شما به اینترنت متصل هستند؟

به نظر شما هوش مصنوعی چگونه می‌تواند دنیای متاورس را هوشمندتر کند؟

اگر بتوانید از طریق متاورس از یک موزه یا شهر تاریخی بازدید کنید، چه مزایا و چه محدودیت‌هایی خواهد داشت؟

نقش هوش مصنوعی در متاورس

یکی از مهم‌ترین فناوری‌های مورد استفاده در متاورس، هوش مصنوعی (Artificial Intelligence) است.

هوش مصنوعی به متاورس کمک می‌کند تا:

- رفتار کاربران را تحلیل کند.
- شخصیت‌های مجازی (آواتارها) را هوشمندتر کند.
- محیط‌های سه‌بعدی را طبیعی‌تر نمایش دهد.
- ارتباط میان کاربران را آسان‌تر کند.

بدون استفاده از هوش مصنوعی، ایجاد محیطی پویا و تعاملی در متاورس بسیار دشوار خواهد بود.

اینترنت اشیا (Internet of Things)

اینترنت اشیا (IoT) به شبکه‌ای از وسایل و تجهیزات گفته می‌شود که از طریق اینترنت به یکدیگر متصل هستند و می‌توانند اطلاعات را ارسال، دریافت و پردازش کنند.



نمونه‌هایی از این وسایل عبارت‌اند از:

- ساعت هوشمند
- تلویزیون هوشمند
- دوربین‌های امنیتی
- سیستم روشنایی هوشمند
- یخچال و ماشین لباس‌شویی هوشمند

این وسایل می‌توانند بدون دخالت مستقیم انسان، برخی وظایف را انجام دهند یا از راه دور کنترل شوند.

ارتباط اینترنت اشیا با متاورس

در آینده، بسیاری از وسایل هوشمند می‌توانند با محیط متاورس ارتباط برقرار کنند.

برای مثال:

- کنترل وسایل خانه از طریق یک محیط سه‌بعدی
- مشاهده وضعیت خانه هنگام سفر
- مدیریت سیستم روشنایی یا گرمایش از راه دور
- دریافت اطلاعات لحظه‌ای از تجهیزات هوشمند

این ارتباط می‌تواند تجربه‌ای یکپارچه بین دنیای واقعی و دنیای مجازی ایجاد کند.

کاربردهای متاورس

متاورس تنها برای بازی و سرگرمی نیست و می‌تواند در زمینه‌های مختلف مورد استفاده قرار گیرد.

پزشکی

پزشکان و دانشجویان پزشکی می‌توانند از مدل‌های سه‌بعدی بدن انسان برای آموزش، تمرین جراحی و بررسی وضعیت بیماران استفاده کنند.

بازی و سرگرمی

یکی از شناخته‌شده‌ترین کاربردهای متاورس، بازی‌های سه‌بعدی و تعاملی است. در این محیط، کاربران احساس می‌کنند در فضای بازی حضور دارند و با دیگران تعامل می‌کنند.

گردشگری

متاورس می‌تواند امکان بازدید مجازی از شهرها، موزه‌ها، آثار تاریخی یا مناطق طبیعی را برای افرادی که امکان سفر ندارند، فراهم کند.

تجارت و خرید

در آینده ممکن است افراد بتوانند فروشگاه‌ها، خانه‌ها یا محصولات را به‌صورت سه‌بعدی مشاهده و بررسی کنند و سپس برای خرید یا انجام معاملات تصمیم بگیرند.

مزایای متاورس

استفاده از متاورس می‌تواند مزایای بسیاری داشته باشد، از جمله:

- یادگیری تعاملی و جذاب‌تر
- دسترسی آسان‌تر به خدمات آموزشی و فرهنگی
- برگزاری جلسات و همکاری از راه دور
- کاهش برخی هزینه‌های سفر و آموزش
- تجربه محیط‌های شبیه‌سازی‌شده بدون خطر

چالش‌ها و محدودیت‌های متاورس

در کنار مزایا، استفاده از متاورس با چالش‌هایی نیز همراه است، مانند:

- نیاز به تجهیزات پیشرفته
- هزینه نسبتاً زیاد برخی ابزارها
- نگرانی درباره امنیت و حریم خصوصی کاربران
- احتمال کاهش تعاملات حضوری
- وابستگی بیش از حد به فضای مجازی

برای بهره‌گیری مناسب از متاورس، باید میان استفاده از فناوری و زندگی واقعی تعادل برقرار شود.

متاورس و سبک زندگی

متاورس می‌تواند روش یادگیری، خرید، کار و سرگرمی را تغییر دهد، اما نباید جایگزین کامل ارتباطات و تجربه‌های واقعی انسان شود.

ارتباط با خانواده، دوستی، ورزش، طبیعت‌گردی و فعالیت‌های اجتماعی همچنان بخش مهمی از زندگی انسان هستند و فناوری باید در خدمت ارتقای این تجربه‌ها باشد، نه جایگزین آن‌ها.

فعالیت کلاسی: طراحی یک کاربرد برای متاورس

وسایل موردنیاز

کاغذ و قلم

رایانه (در صورت امکان)

مراحل اجرا

گروه‌های دو یا سه نفره تشکیل دهید.

یکی از موضوع‌های زیر را انتخاب کنید:

- مدرسه آینده
- بیمارستان هوشمند
- موزه مجازی
- فروشگاه سه‌بعدی

توضیح دهید متاورس چگونه می‌تواند به بهبود این محیط کمک کند.

ایده خود را در قالب یک نقاشی، پوستر یا ارائه کوتاه به کلاس معرفی کنید.

خلاصه درس

- هوش مصنوعی یکی از فناوری‌های اصلی متاورس است.
- اینترنت اشیا امکان ارتباط و کنترل وسایل هوشمند را فراهم می‌کند.
- متاورس در آموزش، پزشکی، بازی، گردشگری و تجارت کاربردهای فراوانی دارد.
- این فناوری می‌تواند فرصت‌های ارزشمندی ایجاد کند، اما چالش‌هایی مانند امنیت اطلاعات و کاهش تعاملات حضوری نیز به همراه دارد.
- استفاده آگاهانه و مسئولانه از متاورس اهمیت زیادی دارد.

تمرین

۱. نقش هوش مصنوعی در متاورس را توضیح دهید.

.....

۲. اینترنت اشیا چیست؟ دو نمونه از وسایل مبتنی بر اینترنت اشیا را نام ببرید.

.....

۳. سه کاربرد متاورس را در زندگی روزمره بیان کنید.

.....

٤. دو مزیت و دو چالش متاورس را بنویسید.

.....

٥. به نظر شما متاورس در آینده چه تأثیری بر آموزش خواهد داشت؟

.....

پروژه

طراحی «شهر هوشمند آینده»

به‌صورت گروهی، شهری را طراحی کنید که در آن از فناوری‌های متاورس، هوش مصنوعی و اینترنت اشیا استفاده شده باشد.

در طرح خود موارد زیر را مشخص کنید:

- مدرسه هوشمند
- بیمارستان هوشمند
- سیستم حمل‌ونقل
- مراکز خرید
- خدمات شهری
- نقش هوش مصنوعی و اینترنت اشیا در هر بخش

نتیجه کار را در قالب پوستر، ماکت، اسلاید یا نقشه مفهومی به کلاس ارائه دهید.