

بخش اول: مقدمه و تنظیمات اولیه

1. شروع کار با لینوکس:

- از ابتدا از سیستم عامل لینوکس، به خصوص **اوبونتو**، استفاده کنید به دلیل سادگی و کاربرپسندی آن.
- می‌توانید اوبونتو را به صورت **دو سیستم عامله (Dual Boot)** یا در یک ماشین مجازی نصب کنید.

2. دوری از استفاده از IDE:

- در مراحل ابتدایی یادگیری از IDE‌هایی که تکمیل خودکار دارند استفاده نکنید.
- نوشتن کد با دست و حتی روی کاغذ تمرین شود تا مهارت بیشتری کسب کنید.

3. توسعه مهارت‌های جستجو:

- مهارت جستجو برای حل مشکلات برنامه‌نویسی حیاتی است.
- ابزارهایی مانند ChatGPT می‌توانند برای جستجو مفید باشند.

4. نسخه پایتون:

- کلاس از نسخه **Python 3.12** استفاده می‌کند، اما نسخه‌های قدیمی‌تر نیز قابل قبول هستند.

5. معرفی iPython:

- iPython به عنوان یک شل پیشرفته برای پایتون معرفی شد که تجربه کاربری بهتری ارائه می‌دهد.
- نصب با دستور `pip3 install ipython` انجام می‌شود.

6. استفاده از خط فرمان در ویندوز:

- می‌توانید از Command Prompt برای نصب بسته‌ها استفاده کنید.

بخش دوم: اصول اولیه پایتون

1. شل تعاملی پایتون:

- امکان اجرای کد خط به خط و دریافت بازخورد فوری را فراهم می‌کند.

2. زبان‌های کامپایل شده و تفسیری:

- **زبان‌های کامپایل شده:** قبل از اجرا به کد ماشین تبدیل می‌شوند و خطاها در مرحله کامپایل شناسایی می‌شوند.

- **زبان‌های تفسیری:** خط به خط اجرا می‌شوند و خطاها در زمان اجرا شناسایی می‌شوند.
- پایتون یک زبان تفسیری است.

3. سینتکس (Syntax):

- قوانین نوشتاری کد در یک زبان برنامه‌نویسی.

4. متغیرها:

- متغیرها برای ذخیره داده‌ها استفاده می‌شوند.

- نام متغیر نمی‌تواند شامل فاصله یا با عدد شروع شود.
- پایتون **دینامیک تایپ** است؛ نوع متغیر در زمان اجرا تعیین می‌شود.

5. تابع `print()`:

- برای نمایش خروجی استفاده می‌شود.

6. انواع داده:

- **str**: رشته‌ها.
- **int**: اعداد صحیح.
- **float**: اعداد اعشاری.
- **bool**: مقادیر درست یا غلط.
- **complex**: اعداد مختلط.

7. تابع `type()`:

- نوع داده‌ی متغیر یا مقدار را بازمی‌گرداند.

8. تبدیل نوع (Type Casting):

- تبدیل مقدار از یک نوع داده به نوع دیگر.
- `int()`, `float()`, `str()`, `bool` برای تبدیل نوع استفاده می‌شوند.

9. تابع `input()`:

- برای دریافت ورودی از کاربر استفاده می‌شود.
- مقدار ورودی همیشه به صورت رشته بازگردانده می‌شود.

بخش سوم: عملگرها در پایتون

1. عملگر انتساب:

- `=` برای تخصیص مقدار به متغیر استفاده می‌شود.

2. عملگرهای ریاضی:

- `+`, `-`, `*`, `/`, `//`, `%`, `**`.

3. عملگرهای مقایسه:

- `==`, `!=`, `<`, `>`, `<=`, `>=`.

4. عملگرهای منطقی:

- `and`, `or`, `not`.

5. عملگر هویت:

- `is` برای بررسی اینکه دو متغیر به یک شیء در حافظه اشاره دارند یا خیر.

6. عملگر عضویت:

- `in`, `not in` برای بررسی وجود یا عدم وجود یک مقدار در یک توالی.

7. عملگرهای انتساب افزوده:

○ +=, -=, *=, /=, //=, %, **.

8. رشته‌ها:

- اتصال با + و تکرار با *.
- کاراکترهای خاص مانند n\ (خط جدید) و t\ (تب).
- رشته‌های چندخطی با ''' یا "" ایجاد می‌شوند.

9. f-strings:

- فرمت‌دهی رشته‌ها با قرار دادن متغیرها داخل {} در رشته‌ای که با f شروع می‌شود.

بخش چهارم: نکات اضافی

1. ساختارهای کنترلی:

- شرط‌ها، حلقه‌ها، و توابع در کلاس‌های بعدی مورد بحث قرار خواهند گرفت.

2. اهمیت تمرین:

- تمرین و جستجوی مستقل کلید یادگیری مؤثر است.

3. استفاده از iPython:

- کدهای ارائه شده در کلاس با استفاده از iPython نمایش داده می‌شوند.