

» بے نام خدا «

درس: مباحث و تشریح

استاد: محمد احمد زارہ

رشتہ: مهندس کامپیوٹر

دانشجو: زمر اکبریں امل - زمر امالی

دانشگاه: ملی مہارت اقدردہ مہتاب

«مقدمات»

«مفهوم»

«پنجشنبه»

1 A و چرا Python زبان برنامه نویسی محبوب علم داده است؟

2 B, NumPy و Pandas چه تفاوتی دارند؟

3 C, چرا Matplotlib برای تجسم داده ها استفاده می شود؟

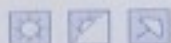
4 D, Seaborn چرا برای تجسم داده های پیچیده کاربرد دارد؟

5 E, چگونه می توانیم Function در Python استفاده می شود؟

6 F, چرا List Comprehension در Python استفاده می شود؟

7 G, چگونه می توانیم CSV file را در Python خواند؟

8 H, JSON و XML چه تفاوتی دارند؟



Subject: _____

Sa Su Mo Tu We Th Fr

Date _____

بخش دوم « Python Programming »

A چرا Python زبان برنامه نویسی محبوب علم داده است؟

- * **سادگی و خوانایی** ... Python با سینک ساده و قابل فهم طراحی شده است. این ویژگی ها باعث می شود که یادگیری و استفاده از آن برنامه نویسان مبتدی و حرفه ای امکان پذیر باشد.
- * **کتابخانه های قدرتمند** ... Python مجموعه از کتابخانه های قوی و متنوع بر علم داده ها دارد.
- * **قابلیت چند منظوره** ... Python نه تنها برای علم داده بلکه برای توسعه وب، اتوماسیون، تحلیل داده و بسیاری از کاربردهای دیگر نیز استفاده می شود.
- * **جامعه بزرگ و فعال** ... Python دارای یک جامعه بزرگ و فعال است که منابع آموزشی، مستندات و راهنمای های متعددی را ارائه می دهد.
- * **یستبان از ابزار های مختلف** ... Python به راحتی با ابزار های مختلف و قابلیت های توسعه پذیری برای پروژه های بزرگ و پیچیده مناسب است.
- * **مناسب برای پروژه های بزرگ** ... Python به راحتی با ابزار های مختلف و زبان های پلاتن یکپارچه می شود که این ویژگی ها به کاربر اجازه می دهد تا از ابزار و تکنولوژی های مختلف به راحتی استفاده کنند.

B. Numpy و Pandas چه تفاوتی دارند؟

Numpy و Pandas دو کتابخانه محبوب در میان برنامه‌نویس Python هستند که به طور عمده برای تجزیه و تحلیل داده‌ها و محاسبات علمی استفاده می‌شود.

تعریف ...

Numpy یک کتابخانه برای محاسبات عددی در Python است. این کتابخانه برای محاسبات عددی در Python است و این کتابخانه به خصوص برای کار با آرایه‌ها و ماتریس‌ها طراحی شده است.

Pandas ← یک کتابخانه برای تجزیه و تحلیل داده‌ها است که به خصوص به روی داده‌های جدولی (مثل داده‌های اکسل و دیتابیس‌ها) تمرکز دارد...

تفاوت‌ها...

Numpy ← آرایه چند بعدی ... این ابعاد آرایه و مدیریت آرایه‌های چند بعدی را فراهم می‌کنند که بسیار کارآمد از لیست‌های Python در انجام محاسبات عددی هستند.

Numpy ← بیشتر برای محاسبات عددی و کار با آرایه‌های تکرارپذیر در حالتی که

Pandas ← برای تجزیه و تحلیل داده‌های جدولی و کار با داده‌های علاقه‌مندی است.

C، چرا Matplotlib برای تجسم داده ها استفاده می شود؟

Matplotlib یک کتابخانه قدرتمند و محبوب در زبان برنامه نویسی پایتون است که برای تجسم داده ها (Data Visualization) استفاده می شود.

* **سادگی و قابلیت دسترسی:** Matplotlib دارای یک رابط کاربری ساده و واضح است که به کاربر اجازه می دهد به راحتی انواع مختلف نمودارها را ایجاد کند.

* **انعطاف پذیری:** این کتابخانه امکان می دهد که نمودارهای سفارشی و پیچیده ای را بسازد.

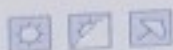
* **پشتیبانی از انواع مختلف نمودارها:** Matplotlib از انواع مختلف نمودارها، پشتیبانی می کند از جمله نمودارهای خطی، میله ای، پراکنش، و بیاری دیگر.

* **یکپارچگی با دیگر کتابخانه ها:** به راحتی با دیگر کتابخانه های پایتون مانند NumPy و Pandas یکپارچه می شود.

* **قابلیت ذخیره سازی نمودارها:** می توانیم نمودارهای ایجاد شده را در فرمت های مختلف ذخیره کنیم مانند PNG و SVG و PDF و ...

* **مستندات و منابع آموزشی:** Matplotlib دارای مستندات جامع و منابع آموزشی زیادی است که به کاربران کمک می کند تا به راحتی با این کتابخانه آشنا شوند.

* **پشتیبانی از محیط های مختلف:** می توان در محیط های مختلف مانند Jupyter Notebook و برنامه های وب استفاده می شوند.



Date _____

Subject: _____

Sa Su Mo Tu We Th Fr

Seaborn یک ابزار برای تجسم داده‌های پیچیده‌تر دارد؟

Seaborn یک کتابخانه تجسم داده در پایتون است که بر روی Matplotlib ساخته شده و به طور خاص برای تجزیه و تحلیل داده‌های آماری طراحی شده است.

* **تجسم آماری پیچیده**: ابزارهای متنوعی برای تجسم انواع مختلف داده‌های آماری ارائه می‌دهد.

* **استفاده آسان**: به کاربران این امکان می‌دهد به راحتی فلاتر نمودارها با استفاده از تنظیمات بین فرفر و سفارش‌های خود تعریف می‌دهد.

* **یکپارچگی با Seaborn**: به خوبی با DataFrame های کتابخانه‌های دسته‌ای دارد که این امکان را فراهم می‌کند تا داده‌ها به آسانی با یکدیگر ویرایش شوند.

* **استفاده از رنگ و طراحی مناسب**: مجموعه‌ای از رنگ‌ها و طرح‌های پیش‌فرض جذاب را فراهم می‌کند که به تجسم‌های شفاف و زیاده‌ای می‌بخشد.

* **پیشانی از تجزیه و تحلیل چند متغیره**: ابزارهایی برای تجزیه و تحلیل داده‌های چند متغیره ارائه می‌دهد مانند نمودارهای جفتی که می‌تواند به شناسایی روابط بین متغیرها کمک کند.

Seaborn به عنوان یک ابزار قدرتمند برای تجسم داده‌های پیچیده شناخته می‌شود و به

تحلیل‌گران داده و دانشمندان داده این امکان را می‌دهد به بیش‌ترین تری از داده‌ها خود به دست آورند.

E. چگونه می‌توانیم یک Python Function تعریف کنیم؟

مراحل ...

* استفاده از کلمه کلیدی `def` ← برای تعریف یک تابع ابتدا باید از کلمه کلیدی `def` استفاده کنیم.

* نام تابع ← نام تابع باید بایک حرف شروع شود و می‌توان شامل حروف، اعداد، خط باشد.

* پارامترها ← در حین فراخوانی می‌توانید پارامترهایی که تابع می‌پذیرد را مشخص کنید.

* دستور ← در ابتدای خط تعریف تابع، باید یک دو نقطه `:` قرار دهید.

* بدنه تابع ← بدنه تابع باید بایک فاصله مناسب مشخص شود و شامل دستورات است که می‌خواهید تابع اجرا کند.

* دستور اختیاری ← اگر می‌خواهید تابع مقداری را بازگرداند باید از دستور `return` استفاده کنید.

F. چرا List Comprehension در Python استفاده می شود؟

این روش مختصر و مفید برای ایجاد لیست ها است. این ویژگی به شما کمک می کند که با استفاده از یک بیان ساده لیست را از عناصر موجود در یک iterable (مقداریت مجموعه) یا هر شیء مشابه (بازیر) بسازید.

*** ساده تر و خواناتر ...**

به شما این امکان را می دهد که لیست های جدید را با یک خط کد ایجاد کنید این امر باعث می شود کد شما

*** عملکرد بهتری ...**

در اکثر موارد List Comprehension سریع تر از حلقه for می باشد. این به دلیل بهینه سازی های داخلی Python است که می تواند عملکرد بهتری را در هنگام ایجاد لیست ها ارائه دهد.

*** قابلیت فیلتر کردن ...**

می توانید به راحتی شرط ها را در List Comprehension قرار دهید تا فقط عناصر خاص را از لیست اصلی انتخاب کنید این کار باعث می شود کد شما تمیزتر و مختصرتر باشد.

*** سمبولیک در کار با داده های تودرتو ...**

به شما این اجازه را می دهد تا به راحتی با داده های تودرتو (مثل لیست های تودرتو) کار کنید و می توانید برای تبدیل یا فیلتر کردن آنها بسیار مفید باشد.

*** بهبود قابلیت نگهداری کد ...**

به معنای ارتقاء کارایی خوانایی و نگهداری پذیری کد است.

F. چرا List Comprehension در Python استفاده می شود؟

این روش مختصر و مفید برای ایجاد لیست ها است. این ویژگی به شما کمک می کند که با استفاده از یک بیان ساده لیست را از عناصر موجود در یک iterable (مقداریت مجموعه یا یک تپه مشابه) بسازید.

*** ساده تر و خواناتر ...**

به شما این امکان را می دهد که لیست های جدید را با یک خط کد ایجاد کنید این امر باعث می شود کد شما

*** عملکرد بهتری ...**

در اکثر موارد List Comprehension سریع تر از حلقه for می باشد. این به دلیل بهینه سازی های داخلی Python است که می تواند عملکرد بهتری را در هنگام ایجاد لیست ها ارائه دهد.

*** قابلیت فیلتر کردن ...**

می توانید به راحتی شرط ها را در List Comprehension قرار دهید تا فقط عناصر خاص را از لیست اصلی انتخاب کنید این کار باعث می شود کد شما تمیزتر و مختصرتر باشد.

*** سهولت در کار با داده های تودرتو ...**

به شما این اجازه را می دهد تا به راحتی با داده های تودرتو (مثل لیست های تودرتو) کار کنید و می توانید برای تبدیل یا فیلتر کردن آنها بسیار سریع باشید.

*** بهبود قابلیت نگهداری ...**

به معنای ارتقاء کد برای خوانایی و انعطاف پذیری کد است.

G. چگونه می‌تواند یک CSV file را در Python خواند؟

برای خواندن یک فایل CSV می‌تواند از کتابخانه‌های مختلفی استفاده کند. یکی از محبوب‌ترین و ساده‌ترین روش‌ها استفاده از کتابخانه Pandas است.

روش اول: استفاده از Pandas

* کتابخانه Pandas ابزاری قدرتمند برای تجزیه و تحلیل داده‌هاست و کار با فایل‌های CSV را بسیار آسان می‌کند.

روش دوم: استفاده از کتابخانه CSV

* این روش ممکن است کمی پیچیده‌تر باشد اما هنوز هم ساده است.

برها

```
Pip install Pandas
```

```
import Pandas as pd
```

```
data = pd.read_csv('Path/to/your/file.csv')
```

```
(Print (data.head()))
```

روش کتابخانه

```
import csv
```

```
with open('Path/to/your/file.csv') as file:
```

```
reader = csv.reader(file)
```

```
for row in reader:
```

```
print(row)
```


H. JSON و XML چه تفاوتی دارند؟

* تفاوت ها...

JSON: ساختار یکپارچه - مقادیر استفاده می کنند داده ها در قالب اشیاء و آرایه ها سازماندهی شوند.

XML: از تگ ها برای تعریف داده ها استفاده می کند و به صورت درختی سازماندهی می شوند.

* خوانایی...

JSON: به دلیل ساختار ساده تر و کمتر بودن تعداد کاما آکس ها و معمولاً خوانایی بیشتر دارد.

XML: به دلیل استفاده از تگ ها و ساختارهای درختی ممکن است خوانایی کمتری داشته باشد.

* اندازه و بارایی...

JSON: به طور کلی اندازه فایل ها کمتر از XML است زیرا از تگ های اضافی استفاده نمی کنند و ساختار آن ساده است.

XML: معمولاً حجم بیشتری دارد و می تواند پذیرد ارزش داده ها کند تر باشد.

* قابلیت گسترش...

XML: به دلیل قابلیت تعریف تگ های سفارشی و ساختارهای پیچیده در مواردی که نیاز به قابلیت های گسترش پذیر وجود دارد مناسب است.

JSON: در مقایسه با XML قابلیت گسترش کمتری دارد اما برای بسیاری از برنامه های کاربردی وب و API کافی است.