محمد مهدی جاوید

از پایگاه داده

پروژه نهایی

\*\*گزارش کار\*\*

### مقدمه:

در این گزارش، جزئیاتی از پروژه‌ای را که یک برنامه وب Flask برای نمایش داده‌های مختلف از منابع متفاوت اطلاعات از جمله MongoDB، MySQL و یک فایل CSV را پیاده‌سازی می‌کند، بررسی می‌شود.

### نصب و وابستگی‌ها:

- استفاده از کتابخانه‌های `pandas`، `Flask`، `mysql.connector`، `pymongo`، `BeautifulSoup`، و `csv` برای پردازش داده و توسعه برنامه وب.

- نصب پکیج‌های مورد نیاز با استفاده از مدیریت بسته `pip`.

### ماژول‌ها و توابع اصلی:

1. \*\*`write\_to\_html`:\*\* تابعی برای نوشتن داده‌ها به فایل HTML از طریق فراخوانی این تابع و پاس دادن لیست‌های داده‌ها.

2. \*\*`isorepublics`:\*\* تابعی برای اتصال به پایگاه داده MongoDB و بررسی وجود یا ساختن دیتابیس و کالکشن مورد نیاز.

3. \*\*`add`:\*\* تابعی برای افزودن داده‌ها به کالکشن MongoDB.

4. \*\*`shows`:\*\* تابعی برای نمایش داده‌های موجود در کالکشن MongoDB.

5. \*\*اتصال به پایگاه داده MySQL:\*\* برقراری اتصال و مدیریت ارتباط با پایگاه داده MySQL.

### جریان کار:

1. \*\*خواندن داده‌ها از فایل CSV:\*\*

- خواندن داده‌ها از فایل CSV و قراردادن آن‌ها در لیست‌های جداگانه.

2. \*\*نوشتن داده‌ها به فایل HTML:\*\*

- فراخوانی تابع `write\_to\_html` برای نوشتن داده‌ها به فایل HTML.

3. \*\*تبدیل داده‌ها به HTML و ذخیره:\*\*

- تبدیل داده‌ها از فایل CSV به فرمت HTML و ذخیره آن‌ها.

4. \*\*پیکربندی وب‌سایت:\*\*

- تعریف یک برنامه Flask و تعیین مسیرهای مربوطه.

5. \*\*اجرای برنامه:\*\*

- اجرای برنامه و امکانات اشکال‌زدایی.

##خلاصه:

به طور خلاصه، این کد داده‌ها را از یک فایل CSV می‌خواند، آن را در پایگاه داده MongoDB وارد می‌کند و سپس یک جدول HTML از داده‌ها را در یک برنامه وب Flask ارائه می‌کند.

### نتیجه‌گیری:

این گزارش جزئیات پیاده‌سازی یک برنامه وب Flask را برای نمایش داده‌ها از منابع مختلف اطلاعات ارائه داده است. این پروژه نشان می‌دهد که چگونه می‌توان از ابزارها و کتابخانه‌های مختلف برای ادغام داده‌ها و نمایش آن‌ها در یک وب‌سایت استفاده کرد.