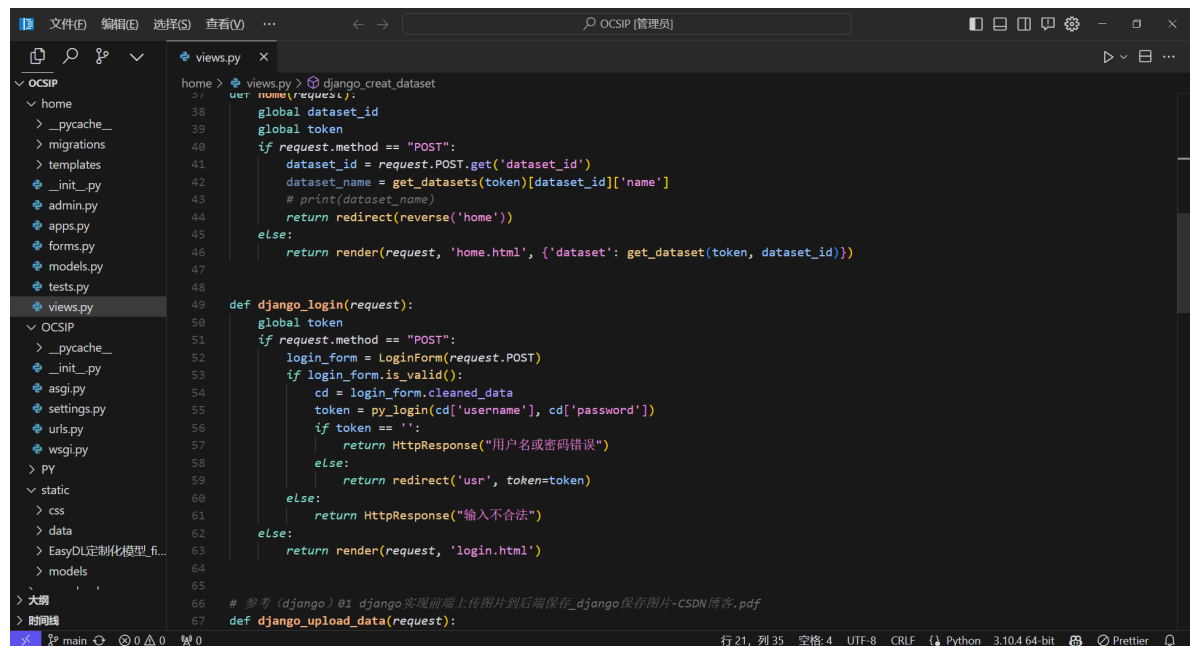


王贤义

## 使用工具

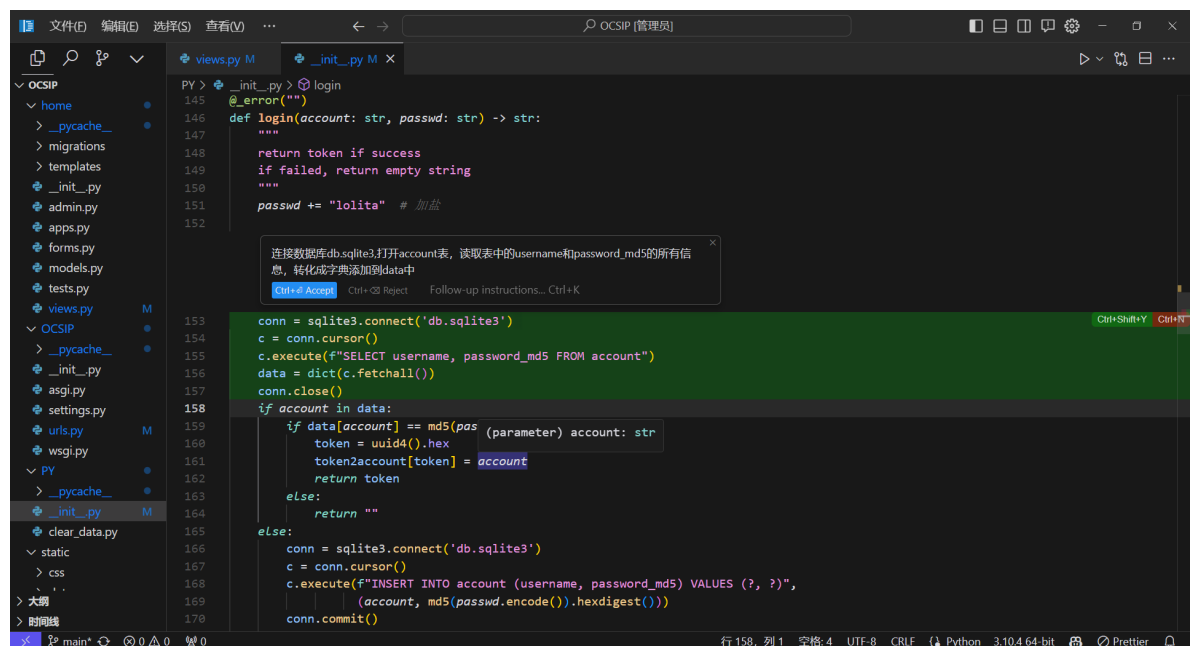
使用cursor进行AI编程助力，软件主界面如下



## 使用场景

在使用python操作sqlite时，由于并未学习任何数据库的相关知识，所以本次项目中关于数据库的增删查改操作全部由cursor生成。

### 子场景1：使用数据库进行account表的查找



### 子场景2：使用数据库进行datasets表的增加

```
def creat_dataset(token: str, dataset_name: str) -> str:
    """
    return id
    if failed, return empty string
    """
    id = uuid4().hex
    name = dataset_name
    created_time = updated_time = time.strftime(
        "%Y-%m-%d %H:%M:%S", time.localtime())
    # freshtoken2account()

    conn = sqlite3.connect('db.sqlite3')
    c = conn.cursor()
    c.execute("INSERT INTO datasets (dataset_id, dataset_name, dataset_created_time, dataset_updated_time, account_id) VALUES (?, ?, ?, ?, ?)",
              (id, name, created_time, updated_time, account_id))
    conn.commit()
    conn.close()

    # _dump_json(
    #     f"static/data/datasets/{account}/{id}.json",
    #     {},
    # )
    return id
```

### 子场景3|4：使用数据库进行删和改，效果如下所示

```
def delete_dataset(token: str, dataset_id: str) -> bool:
    """
    return success or not
    """
    # freshtoken2account()
    account = token2account[token]
    conn = sqlite3.connect('db.sqlite3')
    c = conn.cursor()
    c.execute("SELECT * FROM datasets WHERE dataset_id = ? AND account_id = (SELECT id FROM account WHERE username = ?)", (dataset_id, account_id))
    rows = c.fetchall()
    if len(rows) == 0:
        return False
    c.execute("DELETE FROM datasets WHERE dataset_id = ? AND account_id = (SELECT id FROM account WHERE username = ?)", (dataset_id, account_id))
    conn.commit()
    conn.close()
    return True

def rename_dataset(token: str, dataset_id: str, new_name: str) -> bool:
    """
    return success or not
    """
    # freshtoken2account()
    account = token2account[token]
    conn = sqlite3.connect('db.sqlite3')
    c = conn.cursor()
    c.execute("SELECT * FROM datasets WHERE dataset_id = ? AND account_id = (SELECT id FROM account WHERE username = ?)", (dataset_id, account_id))
    rows = c.fetchall()
    if len(rows) == 0:
        return False
    updated_time = time.strftime("%Y-%m-%d %H:%M:%S", time.localtime())
    c.execute("UPDATE datasets SET dataset_name = ?, dataset_updated_time = ? WHERE dataset_id = ? AND account_id = (SELECT id FROM account WHERE username = ?)",
              (new_name, updated_time, dataset_id, account_id))
    conn.commit()
    conn.close()
    return True
```

```
def upload_data(
    token: str, dataset_id: str, imgs: List[Tuple[str, ByteString]]
) -> Dict[str, Dict[str, int | str]]:
```

## 使用心得

这次使用场景中使用AI助力编程能够在陌生的场景中，比如本次的数据库操作，快速的进行实现，只需要把需求，条件提供清楚，那么AI还是可以完成一份基本正确的答卷的，不然从头开始学习成本太高，也不是本次项目的中心。