项目：交通记录应用

介绍

在本课程中，您将实现一个简单的流量捕获应用程序的实现。 该应用程序是一个基于Web的系统，由两部分组成。 用python编写的Web服务器和提供界面的基于浏览器的客户端。

您将获得：

1.该应用程序的完整的基于浏览器的前端，它使用HTML，CSS和Javascript。 您无需修改此代码。 或者了解它如何运行，而不是如何与服务器交互。

2.用Python编写的应用程序的框架后端框架，提供了核心的Web服务器功能。

您需要：

1.创建一个合适的sqlite3数据库来保存所有必需的数据。

2.通过根据需要添加代码来扩展框架代码以完成应用程序功能。 可以在预期的地方扩展/修改代码，在注释中使用“ ##”表示。 您可以根据需要在其他地方更改代码。

3.创建两个其他程序，这些程序可以从数据库中提取汇总数据作为CSV文件。 以及两个支持程序，可以对它们进行测试。

您将使用python进行程序开发，并使用SQL与sqlite数据库进行交互。 您应该自己编写SQL查询，而不要依赖诸如pandas之类的库来为您抽象它。

Web应用程序结构

该Web应用程序需要运行基于python的Web服务器，并且浏览器才能访问此服务器。 您的任务是将框架服务器开发为功能捕获应用程序。 您应该通过在包含支持文件的目录中从命令提示符处执行python server.py来运行服务器。

当浏览器请求html，css或javascript文件时，这些文件将由现有代码返回。

当请求的文件为“ / action”时，将检查参数“ command”，如果该参数有效，则调用适当的处理函数。 您应该在其中修改带有注释的代码（以“ ##”开头）。

支持以下命令：

1.login

Expects username and password, validates these to generate a session token (magic).

需要用户名和密码，验证它们以生成会话令牌（magic）。

2.add

Expects location, type and occupancy and records this in the traffic database.

需要位置、类型和占用率，并将其记录在流量数据库中。

3.undo

Expects the same as add, but is used to correct mistakes.

期望与add相同，但用于纠正错误。

4. back

Decides if the back button should go to the login page or summary page. No parameters.

决定“上一步”按钮是转到登录页还是摘要页。没有参数。

5. summary

Provides the summary statistics. No parameters.

提供摘要统计信息。没有参数。

6. logout

Logs the user out of the current session, ending it. No parameters.

将用户从当前会话中注销，并结束该会话。没有参数。

请注意，除了列出的命令所需的参数外，所有带有randn参数的“ / action”请求均具有randn参数。 这用于防止不必要的浏览器缓存行为，因此无需进一步考虑。

完成操作请求后，服务器必须提供XML格式的响应，其中包含外部标签<response> ... </ response>，其中外部标签可能包含以下一个或多个以下<action> ... </ action> 条目。

1. refill

<action>

<type>refill</type>

<where>...</where>

<what>...</what>

</action>

重新填充操作响应（refill action response）允许服务器提供应放置在当前页面内各个位置的文本。该位置由<where> target </ where>标识。使用占位符更新可用字段。

Location message is the gray box shown in the pages of Figure 1. total is the number shown at the base of each page. And the entries sum\_\* are the various count values shown on the summary page.

内容应采用base64编码，以避免出现字符转义问题，并占用<what> ... </ what>字段。

2. *redirect*

<action>

<type>redirect</type>

<where>...</where>

</action>

重定向操作响应指示客户端应加载<where> ... </ where>条目指定的页面。 它用于back命令以及会话结束或无效的位置。

提供了辅助函数来生成这些动作，并在框架代码中演示了其用法，包括使用base64编码。

Tasks （需要完成的任务）

Task 1 Setup （根据后续代码所需要的数据建立）

使用sqlite创建一个SQL数据库，该数据库将用于保存应用程序所需的所有数据。

No data should be hardcoded into the application or held between server requests within the application. The sole exceptions to this are the vehicle type and occupancy level definitions. You are advised to consider all the tasks in designing the database as later tasks may require columns in tables initially created for the earlier tasks and for these to be filled in as part of the behaviour of the earlier task.

不应将任何数据硬编码到应用程序中，也不应在应用程序中的服务器请求之间保留这些数据。唯一的例外是车辆类型和占用水平的定义。建议您在设计数据库时考虑所有任务，因为以后的任务可能需要在最初为以前的任务创建的表中填写列，并将这些列作为以前任务行为的一部分填写。

Task 2 Parameter Handling

验证传入请求的所有部分都存在是Web应用程序后端的重要第一步。 当前，如果用户将用户名或密码字段留为空白，则在Python程序内将发生错误，因为当输入为空时，将不填充用于存储输入值的参数变量。 您将需要为这种情况添加适当的处理。 作为要求的示例，参数['command']处理已经处理了相同的问题。

Task 3 User Login

在框架代码中，仅支持一个用户“ test”，并且他们可以使用任何非空密码登录。 这在handle\_login\_request（）中处理。 在数据库中添加一个适当的表，其中包括以下各项：用户名和密码。 两者都应该是字符串。 为了安全起见，应该对密码进行哈希处理，而不是以明文形式存储。 您可能会发现hashlib是完成此任务的有用的python模块。 用10个用户（test1，test2，...）和相应的密码（password1，password2，...）预填充数据库。 在任何给定时间只允许用户登录一次。 应该生成一个魔术会话令牌，该令牌将通过Cookie传递给客户端，并随每个/ action请求一起返回。 这将用于验证对其他操作的访问。 在此任务和其他任务中，您应该有意防范恶意输入。 多个用户可以同时登录。

Task 4 User Logout

当用户选择注销时，会话应结束。 每个用户会话的开始和结束时间都应保存在数据库中。 扩展handle\_logout\_request（）以支持此操作。

Task 5 Traffic Adding

扩展handle\_add\_request（）函数，使其将车辆记录在数据库中。 所使用的表格应至少包括记录的location, type, occupancy and time。

It should also be possible to identify which user created the record and which other records they created during the same login session.还应该可以识别在同一登录会话期间user创建的记录和它们创建的其他记录。响应中应包含‘message’ 和 ‘total’的更新。

Task 6 Traffic Correction

扩展handle\_undo\_request（）函数，使其在数据库中记录撤消车辆的需求。 使用的表应包括记录的location, type, occupancy and time。It should also be possible to identify which user created the record and which other records they created during the same login session. 您可以选择在相同或不同的表中添加和撤消记录。 撤消条目不应导致删除添加条目，仅防止它们在生成的统计信息中计数。 车辆必须存在才能撤消。 响应中应包含‘message’ 和 ‘total’的更新。

Task 7 Online Summary

扩展handle\_summary\_request（）函数，使其返回当前会话的正确流量统计信息，而不是代码中的当前sum\_ \*占位符的值。

（原文：Extend the handle\_summary\_request() function so that it returns the correct traffic statistics for the current session rather than the current sum\_\* placeholder value in the code.）

Task 8 Offline Summary

Create a program (separate from the web app) called task8\_out.py that is able to query the database and indicated the number of each type of vehicle and occupancy at each location during a period specified by a start date and time and an end date and time. Invalid dates and times should be intentionally rejected. The output should be delivered as a csv file with a row per location with the format as described in Figure 1.

创建一个名为task8\_out.py的程序（与web应用程序分开），该程序能够查询数据库，并指示在由开始日期和时间以及结束日期和时间指定的期间内，每种类型的车辆的数量和每个位置的占用率。无效的日期和时间应被故意拒绝。输出应该以csv文件的形式交付，每个位置有一行，格式如图1所示。

Create a second program (separate from the web app) called task8\_in.py that consumes a csv file containing entries of the form described in Figure 2 and updates your database to include these traffic observation records. The updated records should be correctly reflected when you run the first program of this task. You do not need to maintain user/session details beyond what is required to make this task possible for your given database schema. They will not be extracted.

The structure of the database should be the same as that used in the web app.

创建第二个名为task8\_in.py的程序（与web应用程序分开），该程序使用包含图2所示表单条目的csv文件，并更新数据库以包含这些流量观测记录。运行此任务的第一个程序时，应正确反映更新的记录。您不需要维护用户/会话的详细信息，这些信息超出了使给定数据库架构可以执行此任务所需的范围。它们不会被提取。

数据库的结构应与web应用中使用的结构相同。

Location,Type,Occ1,Occ2,Occ3,Occ4

Location is a string containing one or more lowercase letters, the digits 0 thru 9 and spaces. [0-9a-z ]+

Type is one of car,bus,taxi,bicycle,motorbike,van,truck,other.

Occ1 is the number of that type that was recorded with occupancy 1/25%

Occ2 is the number of that type that was recorded with occupancy 2/50%

Occ3 is the number of that type that was recorded with occupancy 3/75%

Occ4 is the number of that type that was recorded with occupancy 4+/100%

For Example:

main road,car,1,0,0,0

ring road,bus,0,0,0,1

位置，类型，Occ1，Occ2，Occ3，Occ4

Location是一个包含一个或多个小写字母、数字0到9和空格的字符串。[0-9a-z]+

车型有汽车、公共汽车、出租车、自行车、摩托车、厢式货车、卡车等。

Occ1是占用率为1/25%时记录的类型的数量

Occ2是在占用率为2/50%的情况下记录的该类型的数量

Occ3是入住率为3/75%时记录的类型数

Occ4是占用率为4+/100%时记录的类型数

例如：

主要道路，汽车，1,0,0,0

环路，公共汽车，0,0,0,1

Figure 1. Task 8 Output CSV Format.

Date, Time,Session,Mode,Location,Type,Occupancy

Date is in the form day/month/year.

Time is in the form hhmm

Mode is either add or undo

Location is a string containing one or more lowercase letters, the digits 0 thru 9 and spaces. [0-9a-z ]+

Type is one of car,bus,taxi,bicycle,motorbike,van,truck,other.

For Example

1/6/2019,1243,1,add,main road,car,1

1/6/2019,1243,1,add,main road,car,2

1/6/2019,1244,1,undo,main road,car,1

2/6/2019,1255,2, add,ring road,bus,4

日期，时间，会话方式，位置，类型，职业

日期在形式日/月/年。

时间在HHMM的形式

模式是第二个加载还是第二个加载

位置是一条带有一条或更多下载信件的弦乐，数字0 thru 9和空间。[（0-9A-Z+）

类型是一辆汽车，总线，出租车，自行车，摩托车，货车等。

举例来说

页：1

页：1

1/6/20191244，1，undo，main road，car，1

2/6/20191255.2，Add.ring road.bus，4

Figure 2: Task 8 Input CSV format.

Task 9 User Hours

Create a third python program (separate from the web app) called task9\_out.py that is able to query the database and indicate the total hours each user was involved in counting in order to allow pay to be determined. It should allow a date to be provided and provide the totals for that date, the week ending on that date and the preceding month ending on that date. The output should be delivered as a csv file with a row per user of the format described in Figure 3. Invalid dates should be intentionally rejected.

Create a fourth python program (separate from the web app) called task9\_in.py that is able to read a CSV file of the form described in Figure 4 and updates your database to reflect the user counting sessions specified.

The structure of the database should be the same as that used in the web app.

创建第三个名为task9\_out.py的python程序（独立于web应用程序），该程序能够查询数据库并指示每个用户参与计数的总小时数，以便确定工资。它应允许提供一个日期，并提供该日期、该日期结束的一周和该日期结束的上一个月的总数。输出应该以csv文件的形式传递，每个用户一行的格式如图3所示。无效日期应被故意拒绝。

创建第四个名为task9\_in.py的python程序（独立于web应用程序），该程序能够读取图4所示表单的CSV文件，并更新数据库以反映指定的用户计数会话。

数据库的结构应与web应用中使用的结构相同。

User,Day,Week,Month

User is one of the predefined usernames.

Day is the number of hours on the given data, with one decimal place rounded up.

Week is the number of hours in the preceding week including the date, with one decimal place rounded up.

Month is the number of hours in the preceding month including the date with one decimal place rounded up.

(A month means up to but not including the same day of the month from the preceding month.)

For Example:

test1,4.3,20.4,81.7

test1,6.5,32.4,100.7

用户，日，周，月

用户是预定义的用户名之一。

天是给定数据上的小时数，小数点后一位四舍五入。

Week是包括日期在内的前一周的小时数，小数点后一位四舍五入。

Month是上个月的小时数，包括小数点后一位四舍五入的日期。

（一个月指上一个月的同一天，但不包括同一天。）

例如：

试验1,4.3,20.4,81.7

试验1,6.5,32.4100.7

Figure 3. Task 9 Output CSV Format.

User,Date,Time,Mode

User is a one of the 10 predefined usernames

Date is in the form day/month/year.

Time is in the form hhmm

Mode is either login or logout

For Example

test1,1/6/2019,1243,login

test2,1/6/2019,1543,login

test1,1/6/2019,1455,logout

test2,1/6/2019,1800,logout

用户、日期、时间、模式

用户是10个预先定义的用户名之一

日期的格式为天/月/年。

时间是hhmm的形式

模式为登录或注销

例如

测试1，1/6/20191243，登录

测试2，1/6/20191543，登录

测试1，1/6/20191455，注销

测试2，1/6/20191800，注销

Figure 4: Task 9 Input CSV format.