



UTRIP!

مقدمه

این پروژه به جمع بندی آموخته های شما در درس می پردازد. انتظار می رود مهارت هایی را که در تمرین های پیشین و سایر بخش های درس آموخته اید، در پیاده سازی این پروژه به کار گیرید. در نهایت شما باید سایت رزرو هتلی را طراحی کنید، که دارای امکاناتی مانند رزرو هتل و گرفتن نظرات و امتیازات کاربران خواهد بود.

ابتدا اطلاعات هتل ها از فایل خوانده خواهند شد و سپس کاربران مختلف می توانند اتاق های این هتل ها را رزرو کنند، به هتل ها امتیاز دهند، و یا لیست هتل ها را طبق خواسته ی خودشان فیلتر کرده و مشاهده کنند.

در فاز اول این پروژه شما منطق برنامه ی این سایت را پیاده سازی می کنید که با استفاده از رابط کاربری خط فرمان¹ می توان از آن استفاده کرد.

شرح تمرین

در این فاز، منطق برنامه در قالب تعدادی دستور در ادامه توضیح داده شده است. روند استفاده از برنامه به این شکل است که کاربر در برنامه شما با استفاده از رابط خط فرمان، دستوری همراه با آرگومان های لازم برای اجرای آن در ورودی استاندارد وارد می کند. به عنوان مثال برای نمایش اطلاعات یک هتل خاص دستور GET hotels و بعد از دستور، id هتلی که مایل به گرفتن اطلاعات آن است را نیز وارد می کند تا دستور مورد نظرش اجرا شود که در اینجا دستور، گرفتن اطلاعات هتل مشخصی بود.

همینطور در نظر داشته باشید که این اطلاعات یعنی آرگومان های هر دستور پس از علامت ؟ در دستور می آیند و همینطور ترتیب خاصی برای آن ها وجود ندارد به این معنا که لزومی ندارد آرگومان ها به همان ترتیبی که در توضیحات هر دستور گفته شده، وارد شوند. برای مثال می توانید در نظر بگیرید که رمز عبور پس از ایمیل کاربر برای ورود، وارد شود. بعضی از دستورات نیز آرگومانی برای دادن به دستور نیاز ندارند که در آن صورت تنها دستور مربوط به آن ها وارد می شود.

¹ CLI(Command Line Interface)

نکته دیگری که دستورات وجود دارد این است که قبل از وارد شدن دستور، عبارت GET و POST وارد می‌شود که به این شکل دستورات به نحوی از هم جدا می‌شوند به این شکل که دستوراتی که بیشتر برای گرفتن اطلاعات از سیستم استفاده می‌شوند، از GET و دستوراتی که بیشتر برای فرستادن اطلاعات به سیستم استفاده می‌شوند، از POST استفاده می‌کنند.

همینطور دقت داشته باشید که تمامی دستورها پس از اجرا شدن دارای خروجی مشخص هستند که منحصر به ذکر می‌شود.

دقت داشته باشید با توجه به تمامی نکات ذکر شده اعم از عبارت قبل از دستورات و یا آرگومان‌های داده شده به دستورات، اگر در دستورات وارد شده کاربر، خطایی وجود داشته باشد باید با توجه به توضیحاتی که همراه با هر دستور آمده است، خطای آن را خروجی دهید. به عنوان مثالی از خطاهایی که ممکن است رخ دهد، می‌توانید وارد کردن دستور خاصی از طرف کاربری که وارد سیستم نشده را نیز در نظر بگیرید.

خروجی پروژه شما به صورت خودکار تست می‌شود، لذا خروجی شما باید دقیقاً همانند خروجی خواسته شده باشد. در غیر این صورت نمره‌ی بخش آزمون را از دست خواهید داد.

● خطاها

○ خطای عدم وجود

اگر دستوری که کاربر وارد می‌کند (قسمت پس از POST یا GET)، در لیست دستورهای تعریف شده نباشد یا کاربر با وارد کردن شناسه‌ی ناموجود در سیستم قصد دسترسی به موجودیتی را داشته باشد، این خطا در خروجی چاپ می‌شود.

خروجی

Not Found

○ خطای درخواست اشتباه

اگر اولین قسمت خط ورودی کاربر یکی از دو مقدار POST و GET نباشد یا این دستور اطلاعات کافی برای اجرا را در خود نداشته باشد، سیستم خروجی زیر را چاپ می‌کند.

خروجی

Bad Request

○ خطای عدم دسترسی

اگر کاربری به سامانه وارد نشده باشد و از دستورات غیر از ثبت نام یا ورود استفاده کند، سیستم باید خروجی زیر را به کاربر نشان دهد.



● اطلاعات هتل ها

فایل اطلاعات هتل ها در اختیارتان قرار گرفته است که توضیح اطلاعات هر هتل در ادامه آمده است.

هر هتل شامل اطلاعات زیر است:

1. unique_id شناسه یکتای هتل است که برای دسترسی به هتل دارنده آن شناسه از آن استفاده می شود.
2. property_name نام هتل است.
3. hotel_star_rating یک عدد طبیعی است و تعداد ستاره های هتل را مشخص می کند. (تعداد ستاره های یک هتل بین ۱ تا ۵ است)
4. hotel_overview توضیح کوتاهی درباره هتل است.
5. property_amenities امکانات هتل است که هر کدام از آن ها با کاراکتر | جدا شده اند. مانند {Telephone|24-hr Room Service|WiFi Internet}
6. city شهری است که هتل در آن قرار دارد.
7. longitude و latitude مختصات جغرافیایی هتل هستند. (این اعداد طبیعی نیستند)
8. image url آدرس اینترنتی از عکس هتل است که در قالب یک رشته ذخیره می شود.

هتل ها شامل 4 نوع اتاق مختلف هستند که تعداد اتاق های هر نوع و قیمتشان در ستون های زیر می آیند (قیمت ها به ازای هر اتاق و هر شب هستند):

9. num of standard rooms
10. num of deluxe rooms
11. num of luxury rooms
12. num of premium rooms
13. standard room price
14. deluxe room price
15. luxury room price
16. premium room price

● تاریخ

برای رزرو هتل کاربر باید زمان اقامت خود در هتل را مشخص کند. برای راحتی کار، فرض کنید تاریخ اقامت، در یک بازه‌ی 30 روزه تعریف می‌شود و تاریخ شروع و پایان اقامت همواره یک عدد بین 1 تا 30 خواهد بود.

● دستورها

● وارد کردن اطلاعات هتل ها

در ابتدای شروع برنامه، باید اطلاعات هتل‌ها از یک فایل csv که نام آن به عنوان آرگومان به برنامه‌ی شما داده می‌شود، خوانده شده و در حافظه‌ی برنامه ذخیره شود. دستور آغاز برنامه به شکل زیر خواهد بود.

توجه کنید نام خروجی برنامه شما باید **utrip.out** باشد.

ورودی

```
./utrip.out <csvFile_relative_address>
```

نمونه

```
./utrip.out folder1/folder2/Hotels.csv
```

در هنگام خواندن اطلاعات از فایل، باید به هر اتاق از هتل‌ها، یک شناسه‌ی یکتا اختصاص دهید تا قابل تمایز باشند. توجه کنید ساختار این شناسه باید به شکل زیر باشد وگرنه در تست‌های اتوماتیک دچار مشکل خواهید شد.

برای هر نوع اتاق حرف اول آن نوع اتاق در ابتدا آمده و پس از آن شماره اتاق که از ۱ شروع خواهد شد و تا شماره آخرین اتاق از آن نوع جلو خواهد رفت.

به عنوان مثال اگر یک هتل 3 اتاق standard و 1 اتاق deluxe و 2 اتاق luxury و 3 اتاق premium دارد، شناسه اتاق‌ها به شکل زیر خواهد بود.

s1, s2, s3 برای اتاق‌های استاندارد.

d1 برای اتاق deluxe.

l1, l2 برای اتاق‌های luxury.

p1, p2, p3 برای اتاق‌های premium.

در ادامه برای رزرو اتاق‌ها از این شناسه استفاده خواهد شد.

● احراز هویت

○ ثبت نام

با این دستور افراد می توانند در سامانه ثبت نام کنند که در این صورت پس از وارد کردن این دستور وارد سیستم نیز خواهند شد. ایمیل و نام کاربری هر کاربر یک رشته یکتا در سیستم است. اگر ایمیل یا نام کاربری در سیستم موجود باشد درخواست معتبر نیست و خروجی Bad Request چاپ می شود.

ورودی
POST signup ? email <email> username <username> password <password>
خروجی
OK Bad Request

○ ورود

اگر فردی قبلاً در سامانه ثبت نام کرده باشد، پیش از استفاده از امکانات آن باید وارد شود. توجه کنید که کاربران بلافاصله بعد از ثبت نام نیازی به دستور ورود ندارند و پس از ثبت نام به صورت ضمنی با همان دستور ثبت نام وارد سیستم نیز می شوند. در صورت درست نبودن اطلاعاتی که کاربر وارد کرده است پیام Bad Request در خروجی چاپ شود.

ورودی	خروجی
POST login ? email <email> password <password>	OK Bad Request

○ خروج

اگر شخصی قبلاً در سیستم وارد شده بود با وارد کردن این دستور از سیستم خارج می شود و می تواند با یک حساب کاربری دیگر وارد سامانه شود. اگر کاربری قبل از ورود به سیستم این دستور را وارد کرد، پیام Permission Denied باید خروجی داده شود.

ورودی	خروجی
POST logout	OK Permission Denied

● افزایش اعتبار کیف پول

کاربر می تواند به مقداری که وارد می کند (عدد اعشاری بزرگتر از 0)، حساب مالی خود را شارژ کند. اگر مقدار داده شده کوچکتر یا مساوی 0 بود باید خطای Bad Request داده شود، همینطور در صورتی که کاربر وارد حساب کاربری خود نشده بود با وارد کردن این دستور پیام Permission Denied خواهد گرفت.

ورودی	خروجی
POST wallet ? amount <amount>	OK Bad Request Permission Denied

● گرفتن چند گردش مالی آخر

کاربر می تواند چند گردش آخر حساب مالی خود را ببیند. تعداد این گردش ها براساس count محدود می شود. توجه کنید که در ابتدای کار، حساب مالی کاربر مقدار 0 دارد. خط اول، موجودی کنونی کاربر است و به ترتیب خط های پایین تر موجودی حساب کاربر در تراکنش های قبلی را نشان می دهد. همینطور اگر تعداد داده شده در ورودی، بیشتر از تعداد گردش های کاربر بود، به همان تعدادی که گردش وجود دارد، موجودی چاپ می شود.

ورودی	خروجی
GET wallet ? count <count>	<credit>
ورودی نمونه	خروجی نمونه
GET wallet ? count 4	8 20 30 0

● نمایش هتل ها

کاربر می‌تواند با دستور زیر لیست هتل‌ها را به همراه اطلاعات خلاصه شده‌ی آن‌ها مشاهده کند. در هر خط اطلاعات یک هتل می‌آید.

دقت داشته باشید در صورتی که کاربر وارد حساب کاربر خود نشده‌بود پیام `Permission Denied` و همینطور در صورتی که لیست هتل‌ها خالی بود پیام `Empty` را خروجی دهید. برای پیش نیامدن مشکل در تست های اتوماتیک ترتیب چاپ شدن هتل ها همواره به ترتیب `unique_id` هتل خواهد بود. (ترتیب چاپ به همان ترتیب حروف الفبایی است.) در بخش های بعدی فیلتر ها تعریف خواهند شد که اگر کاربر فیلتر هایی را اضافه کرده باشد، این لیست باید طبق آن فیلتر ها، ابتدا پالایش شود و سپس نتیجه نمایش داده شود. در صورتی که کاربر هیچ فیلتری ثبت نکرده باشد، لیست تمامی هتل ها نمایش داده خواهد شد.

total_num_of_rooms مجموع تعداد اتاق های هتل است که از مجموع تعداد هر کدام از چهار نوع اتاق به دست می‌آید.

مقدار **average_price** میانگین هزینه‌ی 4 نوع اتاق هتل، بدون احتساب 0 ها است. به عنوان مثال اگر چهار عدد 100، 150، 350 و 0 را داشته باشیم، میانگین هزینه می‌شود 200.00. چاپ این عدد باید تا دو رقم اعشار انجام شود. (اگر کمتر از دو رقم اعشار وجود داشت 0 چاپ شود.)

ورودی
GET hotels
خروجی
<unique_id> <name> <star_rating> <city> <total_num_of_rooms> <average_price> Empty Permission Denied
ورودی نمونه
GET hotels
خروجی نمونه
60ac1903f046575659e673d05595e452 Trident Udaipur 5 Udaipur 111 287 ba7b9e7b9073e745d38122d74045be00 Fateh Garh Resort 5 Udaipur 310 450 ...

● نمایش مشخصات یک هتل

کاربر می‌تواند با وارد کردن شناسه‌ی یک هتل، اطلاعات کامل آن را دریافت کند. همینطور در صورتی که شناسه ورودی کاربر موجود نبود و یا به عبارتی `valid` نبود نیز خروجی `Not Found` دهید. دقت داشته باشید برای حالتی که کاربر وارد حساب کاربری خود نشده باشد نیز همانند حالات گذشته خروجی `Permission`

Denied دهید.

total_num_of_rooms مجموع تعداد اتاق های هتل است که از مجموع تعداد هر کدام از چهار نوع اتاق به دست می آید.

<#comments> تعداد نظرات گذاشته شده برای این هتل است.

دو موقعیت جغرافیایی باید با دقت 2 رقم اعشار چاپ شوند. برای این کار می توانید از [این لینک](#)² استفاده کنید.

ورودی	خروجی
GET hotels ? id <id>	<unique_id> <name> star: <star_rating> overview: <hotel_overview> amenities: <property_amenities> city: <city> latitude: <latitude> longitude: <longitude> #rooms: <#standard_rooms> <#deluxe_rooms> <#luxury_rooms> <#premium_rooms> price: <standard_price> <deluxe_price> <luxury_price> <premium_price> Not Found Permission Denied
ورودی نمونه	
GET hotels ? id ba7b9e7b9073e745d38122d74045be00	
خروجی نمونه	
ba7b9e7b9073e745d38122d74045be00 Fateh Garh Resort star: 5 overview: Situated in the city of lakes, Udaipur, a city rich in cultural heritage, the historic capital of the Mewar kingdom where one can have a glimpse of the imperial Rajput era. Fateh Garh Resort is located at a distance of 30.1km away from Maharana Pratap Airport, 9.9km away from Udaipur Railway Station and there are several buses and local vehicles that connects to various parts of the city. The city is very famous for its forts and the history of Rajputana, often referred to as Venice of the East. Guest should not miss to visit the famous tourist places in and around like Jagdish Temple(6.8km), Bagore Ki Haveli(6.7km), City Palace of Udaipur(7km), Monsoon Palace(8.2km), Vintage Collection of Classic Cars Museum(8.5km) and many more. A traditionally built property with intricately designed interiors, quietly elegant and exclusive. The hotel features wide range of facilities to fulfil the varying requirement of the guest which includes front desk, swimming pool, parking facility and plenty others. Backed by a team of skilled professionals, the property ensures that all needs of guests are fulfilled timely. The sufficiently spacious rooms are perfectly designed and crafted as per the taste of guests to make their stay relaxed and comfortable. These rooms comes with amenities like television, telephone, attached bathroom, hot and cold water. Ease out and have a lovely stay at Fateh Garh Resort. amenities: Gymnasium Health Club Indoor Multi Cuisine Restaurant Lounge Outdoor Activities Outdoor Restaurants Spa Swimming Pool Telephone Bathroom Toiletries Hot & Cold Running Water Television Air Conditioning city: Udaipur latitude: 24.56 longitude: 73.64 #rooms: 0 10 100 200	

² <http://www.cplusplus.com/reference/iomanip/setprecision/>

● اضافه کردن فیلتر

هر کاربر می‌تواند تعدادی فیلتر برای خودش اضافه کند. فیلترها باعث می‌شوند لیست هتل‌هایی که با دستور نمایش هتل‌ها دیده خواهد شد پالایش شوند. انتظار می‌رود این فیلترها را به صورت کلاس پیاده‌سازی کنید و برای اعمال آنها از چندریختی³ استفاده کنید. برای آشنایی با نحوه‌ی تعریف فیلترهای مختلف به صورت کلاس، می‌توانید از این [لینک](#)⁴ استفاده کنید. البته بخش چندریختی را نیز باید به این ساختار اضافه کنید. توجه کنید که اگر یک نوع فیلتر، برای بار دوم با موفقیت اضافه شود، اطلاعات فیلتر دوم جایگزین فیلتر قبلی خواهند شد.

دقت داشته باشید فیلترهای اعمال شده کاربر با خروج از سیستم (logout) از بین می‌روند و با ورود دوباره کاربر به سیستم لیست هتل‌ها بدون اعمال هیچ‌گونه فیلتری نمایش داده خواهند شد.

○ فیلتر شهر

این فیلتر فقط شهرهایی که نامشان در دستور مشخص شده است را تایید می‌کند.

ورودی	خروجی
POST filters ? city <city>	OK Bad Request
ورودی نمونه	خروجی نمونه
POST filters ? city Tehran	OK

○ فیلتر بازه‌ی ستاره‌های هتل

این فیلتر فقط هتل‌هایی که ستاره‌شان در محدوده‌ی مشخص شده است را تایید می‌کند. (توجه داشته باشید فیلتر به صورت بسته بسته عمل می‌کند یعنی هم ابتدا و هم انتهای بازه را شامل می‌شود) اگر دستور به شکل زیر نبود، یا اعداد داده شده بین 1 تا 5 نبود یا عدد پایانی از عدد شروع کمتر بود باید خطای Bad Request داده شود. برای فیلتر کردن هتل‌ها با یک تعداد ستاره‌ی خاص، عدد شروع و پایان برابر خواهند بود.

ورودی	خروجی
POST filters ? min_star <min_star> max_star <max_star>	OK Bad Request
ورودی نمونه	خروجی نمونه
POST filters ? min_star 4 max_star 5	OK

³ polymorphism

⁴ https://www.tutorialspoint.com/design_pattern/filter_pattern.htm

○ فیلتر میانگین هزینه‌ی هتل

این فیلتر، فقط هتل هایی که میانگین هزینه‌ی 4 نوع اتاقشان در بازه‌ی خاصی است را تایید می‌کند. (دقت کنید این فیلتر نیز همانند فیلتر ستاره‌های هتل به صورت بسته بسته عمل می‌کند) مقدار میانگین هزینه‌ی هتل، میانگین هزینه‌ی 4 نوع اتاق هتل، بدون احتساب 0 ها است. به عنوان مثال اگر چهار عدد 100، 150، 350 و 0 را داشته باشیم، میانگین هزینه 200 خواهد بود. اگر دستور به شکل زیر نبود، اعداد داده شده کمتر از صفر بودند، یا عدد پایانی از عدد شروع کمتر بود باید خطای Bad Request داده شود.

ورودی	خروجی
POST filters ? min_price <min_price> max_price <max_price>	OK Bad Request
ورودی نمونه	خروجی نمونه
POST filters ? min_price 255.2 max_price 300.9	OK

○ فیلتر هتل هایی که تعدادی اتاق از یک نوع خاص در یک بازه‌ی زمانی دارند

کاربر می‌تواند با دادن تعداد اتاق، نوع اتاق، و بازه‌ی زمانی، لیست هتل هایی که در آن بازه، حداقل به تعداد خواسته شده از آن اتاق خاص را دارند فیلتر کند. دقت کنید که برای به دست آوردن این هتل‌ها باید رزروهای قبلی اتاق‌های آن هتل و تعداد اتاق های موجود را در نظر بگیرید.

ورودی	خروجی
POST filters ? type <room_type> quantity <quantity> check_in <check_in> check_out <check_out>	OK Bad Request
ورودی نمونه	خروجی نمونه
POST filters ? type premium quantity 2 check_in 21 check_out 26	OK

(تمام اطلاعات در یک خط هستند و رفتن به خط بعد به علت محدودیت عرض صفحه است.)

○ حذف تمام فیلترها

کاربر می‌تواند تمام فیلترهایی که اضافه شده است را حذف کند. خروجی برای زمانی که فیلتری وجود داشته یا هیچ فیلتری وجود نداشته فرقی ندارد و در هر دو OK چاپ می‌شود و تنها در صورتی که دستور اشتباه باشد خطا داده می‌شود.

ورودی	خروجی
DELETE filters	OK Bad Request
ورودی نمونه	خروجی نمونه
DELETE filters	OK

● رزرو اتاق در هتل

کاربر می‌تواند با دادن شناسه‌ی هتل، نوع و تعداد اتاقی که می‌خواهد و یک بازه‌ی زمانی، تعدادی اتاق از یک هتل را رزرو کند. پس از رزرو، هزینه‌ی رزرو آن اتاق‌ها از اعتبار کاربر کاسته خواهد شد. همانطور که قبلاً گفته شد، بازه‌ی زمانی رزرو اتاق‌ها یک عدد بین 1 تا 30 خواهد بود. دقت کنید لازم است که از اتاق با شناسه‌ی کوچکتر شروع کنید و در صورت امکان آن را رزرو کنید. به عنوان مثال اگر اتاق‌های s1, s3, s4 در بازه‌ی زمانی خواسته شده آزاد هستند، و کاربر 2 اتاق می‌خواهد، باید اتاق‌های s1, s3 برای او رزرو شود.

مشخص است که پس از رزرو یک اتاق، هیچ کاربر دیگری نمی‌تواند آن اتاق را در آن بازه‌ی زمانی رزرو کند. به جز خطاهای عادی که شرایط وقوع آن قبلاً گفته شد، در این بخش خروجی‌های دیگری نیز متصور است.

● در صورتی که رزرو موفقیت‌آمیز باشد، لیست شناسه‌ی اتاق‌هایی که رزرو شده با فاصله نمایش داده خواهد شد.

● اگر اعتبار کاربر برای رزرو آن اتاق‌ها کافی نباشد باید خطای Not Enough Credit چاپ شود.

● اگر به تعداد خواسته شده، اتاق برای رزرو کردن وجود نداشته باید خطای Not Enough Room چاپ شود. دقت کنید که وجود نداشتن تعداد خواسته شده در دو حالت ممکن است. اول اینکه تعداد اتاق‌های هتل کمتر از تعداد خواسته شده باشد، و دوم آنکه تعداد کافی باشد، اما اتاق‌های هتل در آن بازه پر باشند که در هر دو صورت همان خطای Not Enough Room باید داده شود.

● نوع اتاق می‌تواند یکی از مقادیر standard, deluxe, luxury, premium باشد.

ورودی
POST reserves ? hotel <hotel_id> type <room_type> quantity <quantity> check_in <check_in> check_out <check_out>
خروجی
<id1> <id2> ... Bad Request Not Found Not Enough Credit Not Enough Room
ورودی نمونه
POST reserves ? hotel ba7b9e7b9073e745d38122d74045be00 type premium quantity 2 check_in 21 check_out 26
خروجی نمونه
p3 p4

(تمام اطلاعات در یک خط هستند و رفتن به خط بعد به علت محدودیت عرض صفحه است.)

● مشاهده‌ی رزروهای انجام شده‌ی کاربر

هر کاربر می‌تواند لیست رزروهای انجام شده‌ی خود را به ترتیب زیر مشاهده کند. cost کل هزینه‌ی رزرو است. به عنوان مثال اگر 3 اتاق با قیمت 100 رزرو شده باشد، cost برابر 300 خواهد بود. بالاترین رزرو، آخرین رزرو انجام شده از نظر زمانی است. (در اینجا زمان ثبت رزرو مهم است نه بازه زمانی اقامت) اگر رزروی انجام نشده بود باید Empty چاپ شود. id شناسه‌ای یکتا برای هر رزرو کاربر است. این شناسه‌ها از 1 شروع می‌شوند، و با هر رزرو افزایش می‌یابد. شناسه‌ی رزروهای کاربران متفاوت می‌تواند یکی باشد.

خروجی	ورودی
id: <reservation_id> hotel: <hotel_id> room: <room_type> quantity: <quantity> cost: <cost> check_in <check_in> check_out <check_out> ... Empty	GET reserves
خروجی نمونه	ورودی نمونه
id: 3 hotel: ba7b9e7b9073e745d38122d74045be00 room: premium quantity: 2 cost: 2200 check_in 5 check_out 6 id: 2 hotel: ba7b9e7b9073e745d38122d74045be00 room: deluxe quantity: 4 cost: 7000 check_in 24 check_out 28	GET reserves

```
id: 1 hotel: ba7b9e7b9073e745d38122d74045be00 room: deluxe quantity:
1 cost: 350 check_in 1 check_out 1
```

● کنسل کردن رزرو

کاربر می‌تواند با وارد کردن شناسه‌ی رزرو خودش، آن رزرو را کنسل کند. در صورت موفقیت، نصف هزینه‌ی پرداخت شده به او باز می‌گردد و دیگران می‌توانند آن اتاق‌های کنسل شده را رزرو کنند. دقت کنید این شناسه فقط برای خود کاربر، یکتا است و ممکن است کاربران دیگر نیز رزروی با همین شناسه داشته باشند که طبیعتاً نباید دچار مشکل شوند. پس از کنسل کردن یک رزرو، دیگر نباید آن رزرو در لیست رزرو های کاربر نمایش داده شود اما شناسه‌ی یکتای آن رزرو، سوخته محسوب می‌شود و رزرو های جدید نمی‌توانند آن شناسه را دوباره بگیرند. اگر شناسه‌ی وارد شده وجود نداشته باشد یا قبلاً کنسل شده باشد باید خطای Not Found داده شود.

ورودی	خروجی
DELETE reserves ? id <reservation_id>	OK Bad Request Not Found
ورودی نمونه	خروجی نمونه
DELETE reserves ? id 2	OK

● اضافه کردن نظر به هتل

کاربر با دادن شناسه‌ی یک هتل می‌تواند روی آن نظر اضافه کند. comment یک رشته‌ی متنی است و اگر کاربری چندین بار با استفاده از این دستور کامنت بگذارد، همه‌ی کامنت‌ها ذخیره خواهند شد. در این فاز می‌توانید فرض کنید که کامنت، شامل کاراکتر اسپیس نیست.

ورودی	خروجی
POST comments ? hotel <hotel_id> comment <comment>	OK Bad Request Not Found
ورودی نمونه	خروجی نمونه
POST comments ? hotel ba7b9e7b9073e745d38122d74045be00 comment Great-location-for-our-planned-activities	OK

(تمام اطلاعات در یک خط هستند و رفتن به خط بعد به علت محدودیت عرض صفحه است.)

● اضافه کردن امتیاز به هتل

هتل‌ها از 6 جهت قابل امتیاز دهی هستند.

Location, Cleanliness, Staff, Facilities, Value for money, Overall rating

کاربران با استفاده از دستور زیر می‌توانند به همین ترتیب به این موارد امتیاز دهند. امتیازها باید عددی بین 1 تا 5 باشند و در غیر این صورت باید خطا داده شود. اگر کاربری دو بار از این دستور استفاده کند، امتیازهای بار دوم جایگزین امتیازهای بار اول خواهند شد. توجه کنید که overall rating نیز مانند بقیه‌ی موارد یک امتیاز مجزا است و توسط کاربر وارد می‌شود و نباید محاسباتی انجام شود.

ورودی	خروجی
POST ratings ? hotel <hotel> location <location> cleanliness <cleanliness> staff <staff> facilities <facilities> value_for_money <Value_for_money> overall_rating <overall_rating>	OK Bad Request
ورودی نمونه	خروجی نمونه
POST ratings ? hotel ba7b9e7b9073e745d38122d74045be00 location 3.22 cleanliness 4.2 staff 5 facilities 5 value_for_money 1 overall_rating 4.77	OK

(تمام اطلاعات در یک خط هستند و رفتن به خط بعد به علت محدودیت عرض صفحه است.)

● مشاهده‌ی نظرات روی یک هتل

کاربران می‌توانند لیست نظرات روی یک هتل را با دستور زیر مشاهده کنند. هر نظر در یک خط نمایش داده می‌شود و شامل نام کاربری و متن نظر او است. اگر نظری نبود چیزی چاپ نمی‌شود. بالاترین نظر جدیدترین نظر است.

ورودی	خروجی
GET comments ? hotel <hotel_id>	<username1>: <comment1> <username2>: <comment2> ... Bad Request Not Found
ورودی نمونه	خروجی نمونه
GET comments ? hotel ba7b9e7b9073e745d38122d74045be00	mohsen: Great_location zhivar: Highly_recommended mohsen: Lovely_stay

● مشاهده‌ی میانگین امتیازات یک هتل

کاربران می‌توانند میانگین امتیازات داده شده به یک هتل را ببینند. هر امتیاز برای هر بخش در یک خط نمایش داده می‌شود. این میانگین از محاسبه‌ی میانگین تمام امتیازات داده شده به هتل از طرف کاربران است و دقت نمایش نتایج تا دو رقم اعشار است. (دقت کنید اگر هیچ امتیازی توسط کاربران به هتل داده نشده باشد، مقدار No Rating نمایش داده می‌شود.)

ورودی	خروجی
GET ratings ? hotel <hotel_id>	location: <location> cleanliness: <cleanliness> staff: <staff> facilities: <facilities> value for money: <value_for_money> overall rating: <overall_rating> Bad Request Not Found No Rating
ورودی نمونه	خروجی نمونه
GET ratings ? hotel ba7b9e7b9073e745d38122d74045be00	location: 3.22 cleanliness: 4.20 staff: 5.00 facilities: 5.00 value for money: 1.00 overall rating: 4.77

● بخش امتیازی:

○ هش⁵ کردن گذرواژه

همانطور که می‌دانید گذرواژه ویژگی‌ای است که افراد تمایل دارند از دسترسی سایرین حفظ شود. به این دلیل است که هنگام ذخیره سازی گذرواژه آن را به صورت خام در حافظه نگهداری نمی‌کنند. هش کردن روشی که توسط آن می‌توان یک رشته را طی فرآیندی غیرقابل بازگشت به رشته‌ی دیگری تبدیل کرد؛ یعنی هش هر رشته یک رشته‌ی یکتا است و با هر رشته‌ی دیگر تفاوت دارد و همچنین از طریق هش یک رشته نمی‌توان به آن رشته دسترسی پیدا کرد. در این قسمت باید گذرواژه‌های افراد را به صورت هش شده در حافظه نگهداری کنیم تا از دسترسی به گذرواژه خام پرهیز شود. هش کردن الگوریتم‌های متفاوتی دارد که می‌توانید در اینترنت درباره‌ی آن‌ها جستجو کنید.

⁵ Hash

نحوه‌ی تحویل و نکات

- تمام فایل‌های خود را در قالب یک پرونده‌ی زیپ با نام `A7-<SID>.zip` در صفحه‌ی CECM درس بارگذاری کنید که SID شماره‌ی دانشجویی شماست؛ برای مثال اگر شماره‌ی دانشجویی شما ۸۱۰۱۹۸۹۹۹ است، نام پرونده‌ی شما باید `A7-810198999.zip` باشد.
- با توجه به حجم نسبتاً زیاد این فاز از تمرین توصیه می‌شود قبل از پیاده‌سازی کد طراحی اولیه‌ای برای منطق برنامه و روندهای آن مثل ثبت‌نام و رزرو و هتل‌ها و ... انجام دهید و پس از این طراحی شروع به پیاده‌سازی آن کنید. از آن جایی که در فازهای بعدی شما باید رابط کاربری برنامه‌ی خود را از `command line` به روش‌هایی دیگر تغییر دهید بهتر است تا طراحی برنامه‌ی شما طوری باشد که کمترین وابستگی میان منطق برنامه و رابط کاربری آن وجود داشته باشد.
- **دقت کنید** که پرونده زیپ آپلودی شما باید پس از Unzip شدن شامل پرونده‌های پروژه شما (از جمله Makefile) باشد و از زیپ کردن پوشه‌ای که داخل آن فایل‌های پروژه‌تان قرار دارد خودداری فرمایید.
- برنامه‌ی شما باید در سیستم عامل لینوکس و با مترجم `g++` با استاندارد `c++11` ترجمه و در زمان معقول برای ورودی‌های آزمون اجرا شود.
- **دقت کنید** که پروژه شما باید Multi-file باشد و Makefile داشته باشد. همین‌طور در Makefile خود مشخص کنید که از استاندارد `c++11` استفاده می‌کنید.
- درستی برنامه‌ی شما از طریق **آزمون‌های خودکار** سنجیده می‌شود؛ بنابراین پیشنهاد می‌شود با استفاده از ابزارهایی مانند `diff` خروجی برنامه خود را با خروجی‌هایی که در اختیارتان قرار داده شده است مطابقت دهید.
- **دقت کنید** که نام پرونده‌ی اجرایی شما باید `utrip.out` باشد.
- دقت کنید که تمام اعداد غیر صحیح که در خروجی چاپ می‌شوند باید دقت 2 رقم اعشار داشته باشند. (مثلاً عدد 2.1 باید 2.10 و عدد 4.1234 باید 4.12 نمایش داده شود.)
- طراحی درست، رعایت سبک برنامه نویسی درست و تمیز بودن کد برنامه‌ی شما در نمره‌ی تمرین تأثیر زیادی دارد.
- هدف این تمرین یادگیری شماست. لطفاً تمرین را خودتان انجام دهید. در صورت کشف تقلب مطابق قوانین درس با آن برخورد خواهد شد.
- سوالات خود را تا حد ممکن در فروم درس مطرح کنید تا سایر دانشجویان نیز از پاسخ آن‌ها بهره‌مند شوند. در صورتی که قصد مطرح کردن سوال خاص‌تری داشتید، از طریق ایمیل با طراحان این فاز پروژه ارتباط برقرار کنید. توجه داشته باشید که دیگر شبکه‌های اجتماعی مانند تلگرام راه ارتباطی رسمی با دستیاران آموزشی نیست و دستیاران آموزشی موظف به پاسخگویی در محیط‌های غیررسمی نیستند.