TRAVEL CASE BASED REASONING

HOMEWORK 3

INDIANA UNIVERSITY
BLOOMINGTON

Raksha Rank
Abhishek Sharma

03.30.2022
EXPLAINABLE AI

CASE BASED REASONING MODEL FOR TRAVEL PACKAGE PRICE PREDICTION:

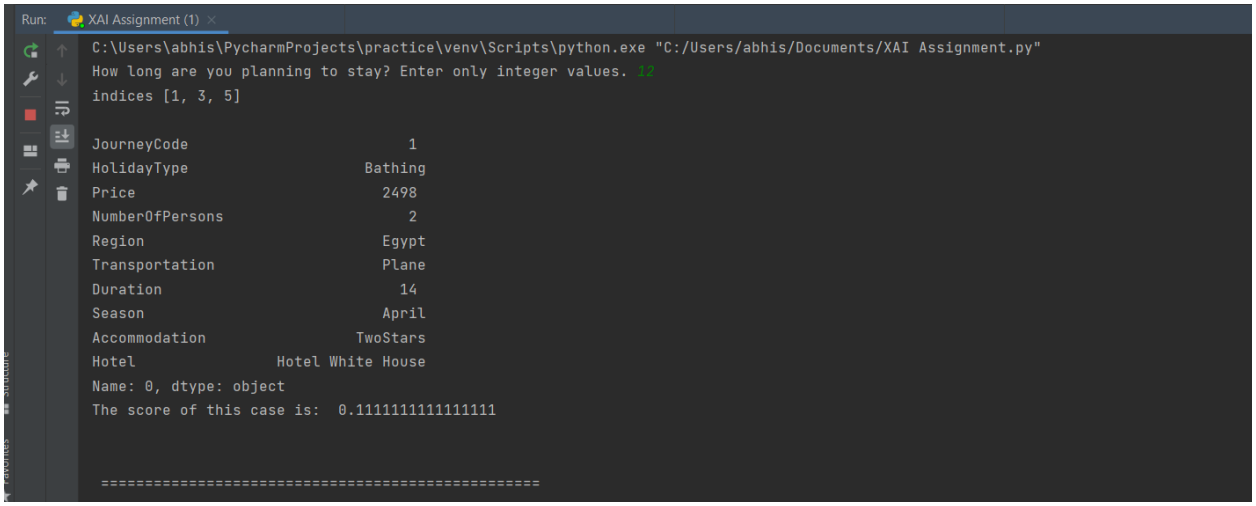
- The user will be asked certain questions in a particular order on the basis of which similarity scores will be calculated. The more similar the query is with the cases, the lower the score will be. On the basis of the similarity score, top 3 cases with the least similarity scores will be presented to the user to choose from.
- We are assuming that the user would enter a valid value from the range of the cases provided. We are handling the case wherein the user provides an invaid value, the system would ask to input the feature value once again.
- The user will have the liberty to choose from those top 3 cases or move ahead and answer more questions to converge with a more similar package.
- One the user is satisfied with the case, he/she would select the preferred package and price for that package would be adapted based on the duration, number of persons and peak/off seasons.
- For each package, an explanation would be provided on what basis the price has been calculated.

Example 1 : (*Case of adapted price)*

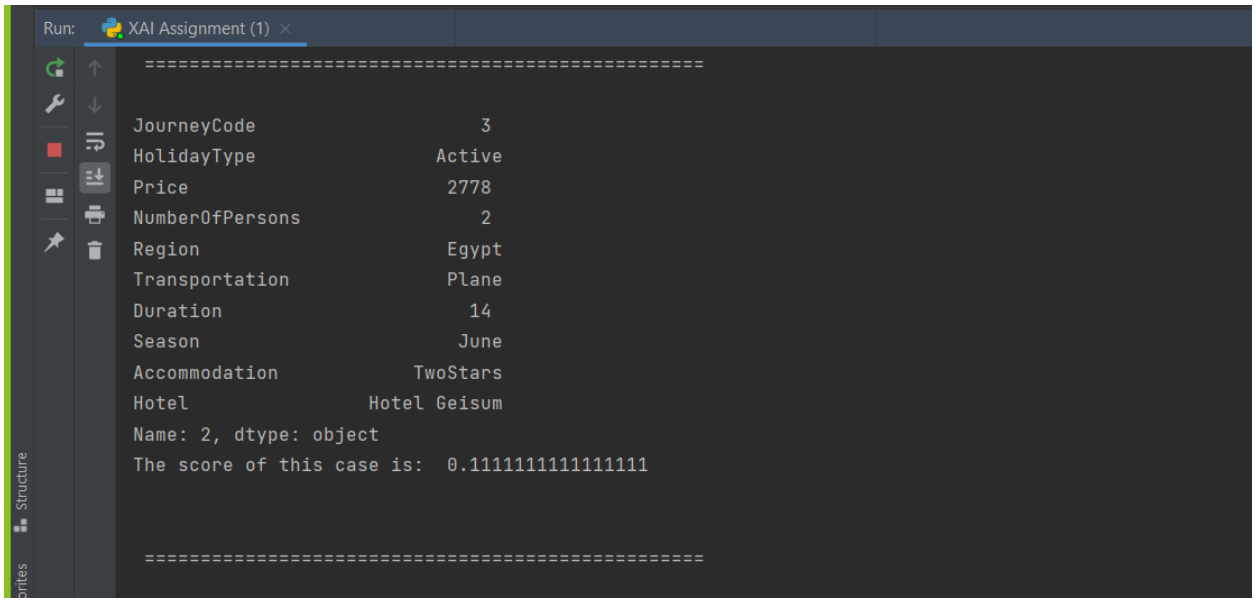
Q1. User is asked the duration of the stay.

The duration entered is 12. Top 3 cases which have the least score, that is maximum similarity in the case base would be returned. Here, Cases with Journey Codes 1, 3 and 5 are available for the user to choose from.

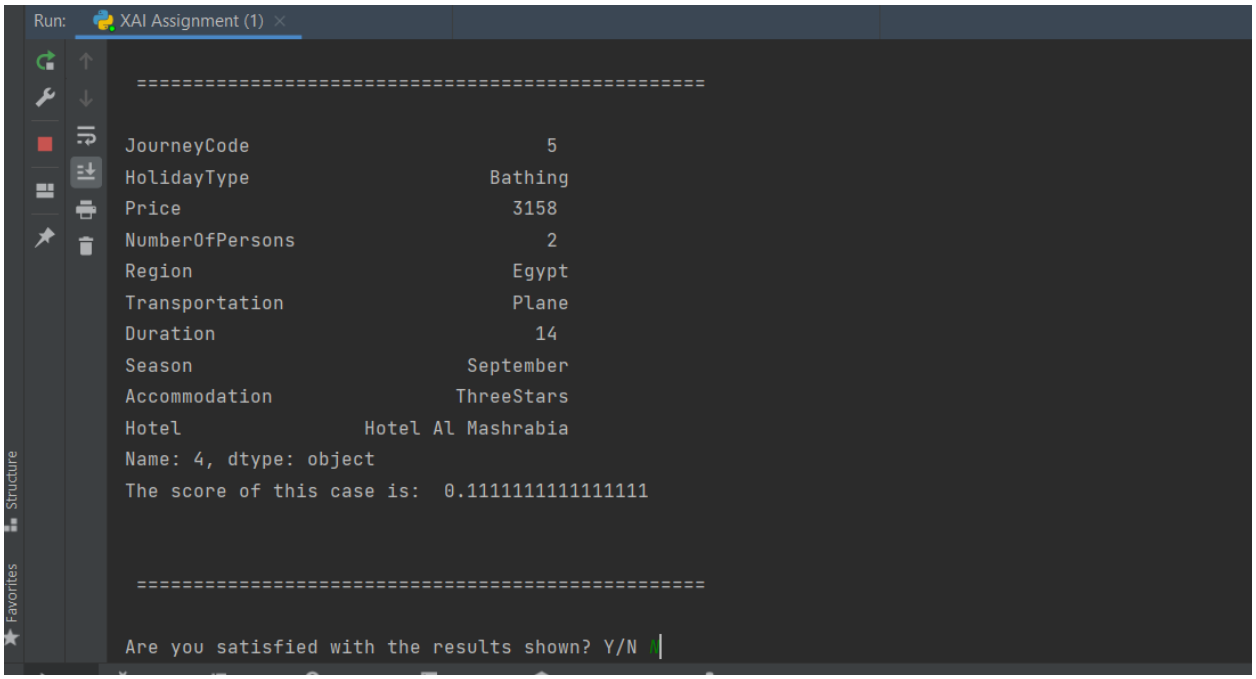
Case 1:


Run:
XAI Assignment (1)
C:\Users\abhis\PycharmProjects\practice\venv\Scripts\python.exe "C:/Users/abhis/Documents/XAI Assignment.py"
How long are you planning to stay? Enter only integer values. 12
indices [1, 3, 5]
JourneyCode 1
HolidayType Bathing
Price 2498
NumberOfPersons 2
Region Egypt
Transportation Plane
Duration 14
Season April
Accommodation TwoStars
Hotel Hotel White House
Name: 0, dtype: object
The score of this case is: 0.1111111111111111
==========

Case 2:


Run:
XAI Assignment (1)
==========
JourneyCode 3
HolidayType Active
Price 2778
NumberOfPersons 2
Region Egypt
Transportation Plane
Duration 14
Season June
Accommodation TwoStars
Hotel Hotel Geisum
Name: 2, dtype: object
The score of this case is: 0.1111111111111111
==========

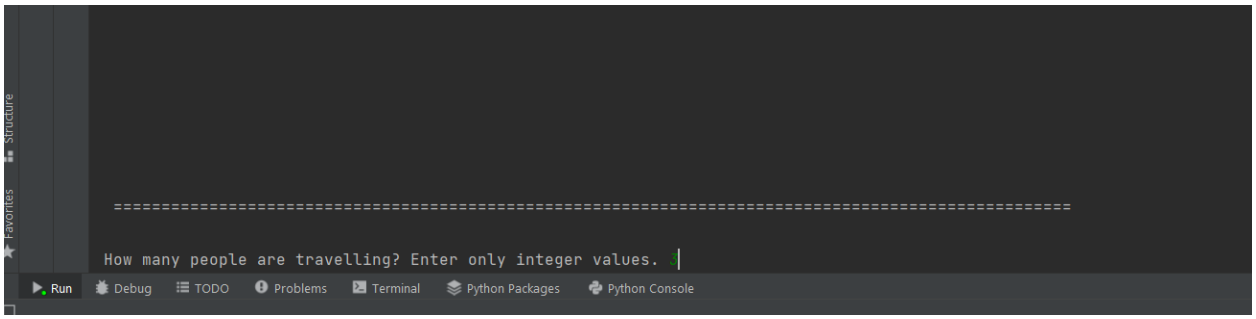
Case 3:



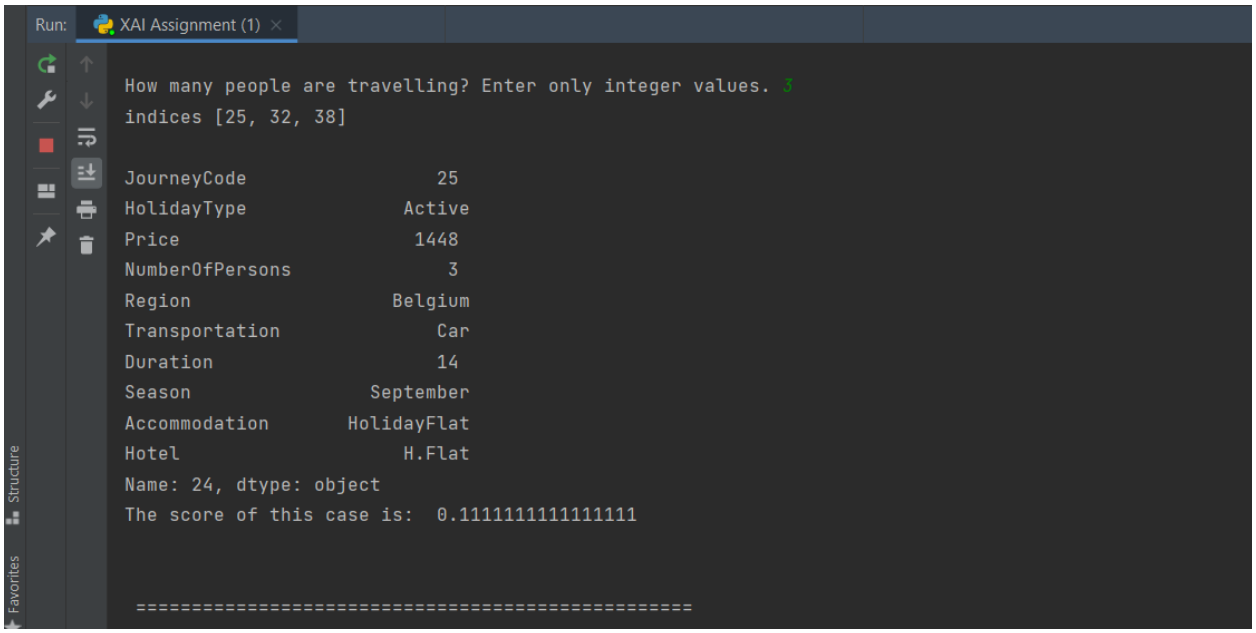
As the user is not satisfied with the cases yet, he selects no and moves ahead with answering other questions.

Q2. The number of people traveling.

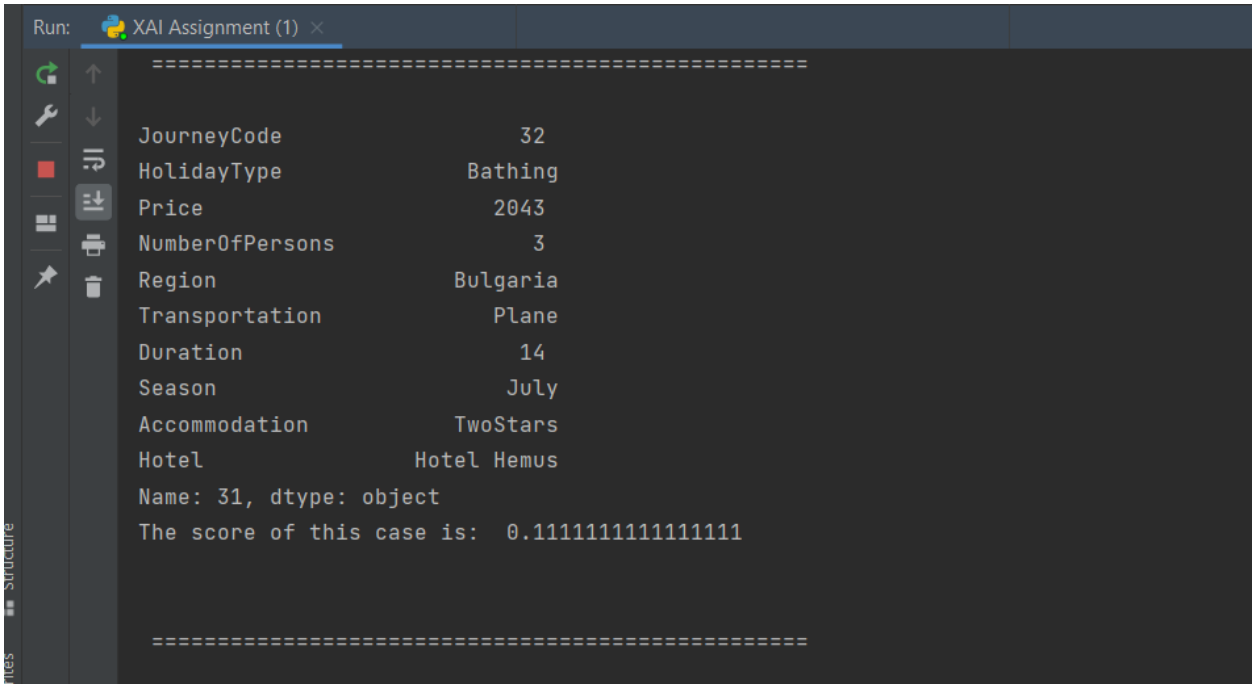
The user answers 3 and similarly again top 3 questions with taking into consideration both the question's answers would be shown to the user to choose from.



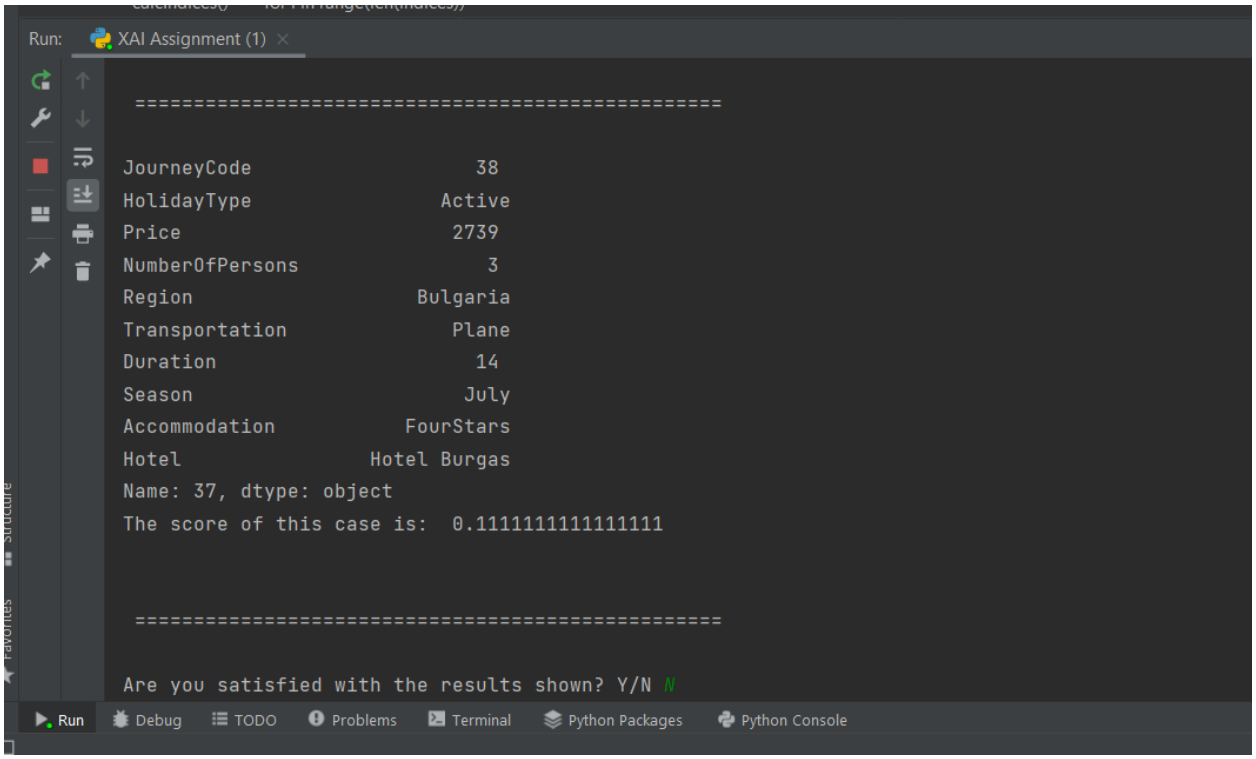
Case 1:


Run:
XAI Assignment (1)
How many people are travelling? Enter only integer values. 3
indices [25, 32, 38]
JourneyCode 25
HolidayType Active
Price 1448
NumberOfPersons 3
Region Belgium
Transportation Car
Duration 14
Season September
Accommodation HolidayFlat
Hotel H.Flat
Name: 24, dtype: object
The score of this case is: 0.1111111111111111
==========
Structure
Favorites

Case 2:

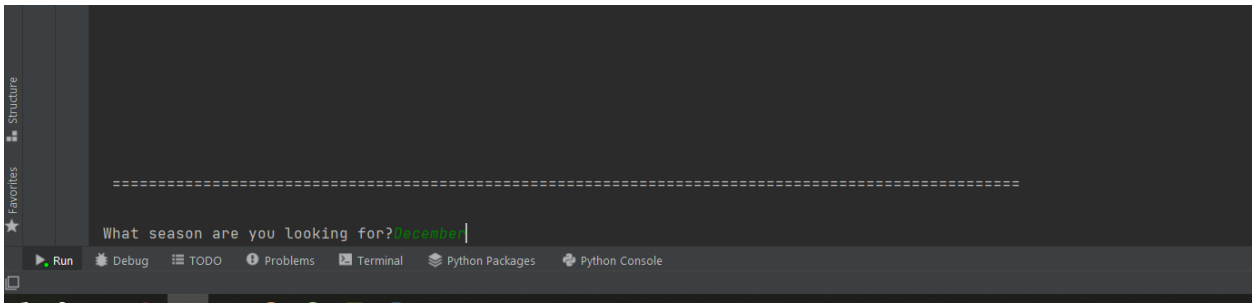

Run:
XAI Assignment (1)
==========
JourneyCode 32
HolidayType Bathing
Price 2043
NumberOfPersons 3
Region Bulgaria
Transportation Plane
Duration 14
Season July
Accommodation TwoStars
Hotel Hotel Hemus
Name: 31, dtype: object
The score of this case is: 0.1111111111111111
==========
Structure

Case 3:

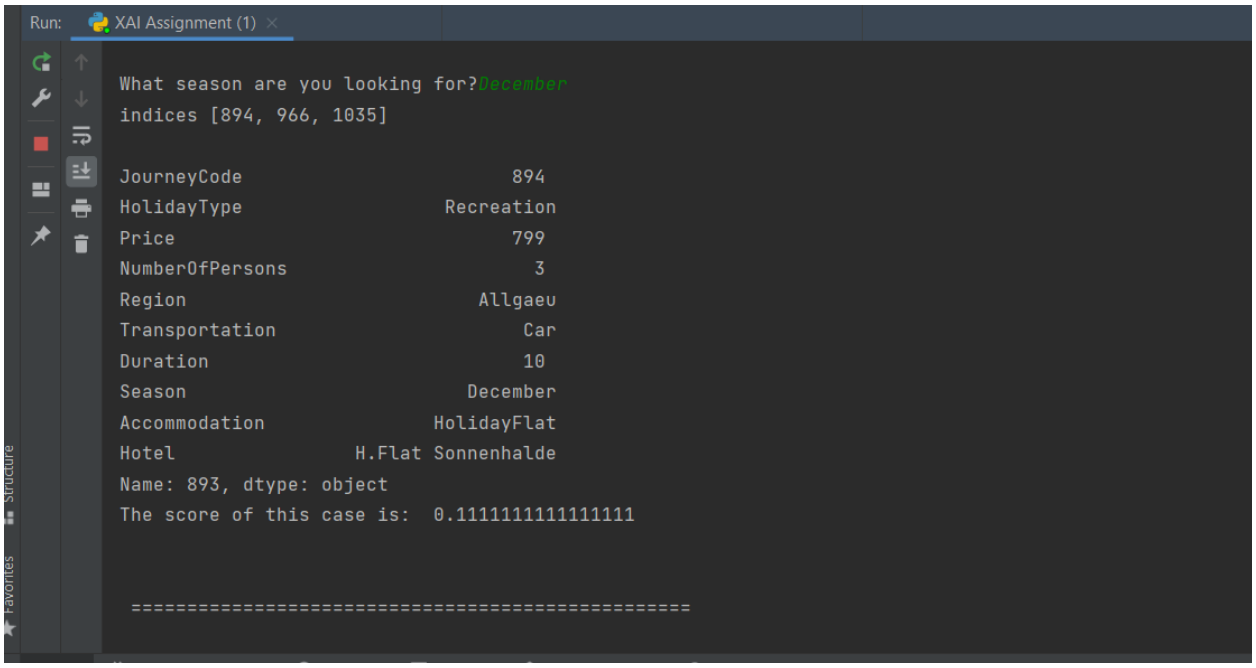


Q3. Season looking to travel in.

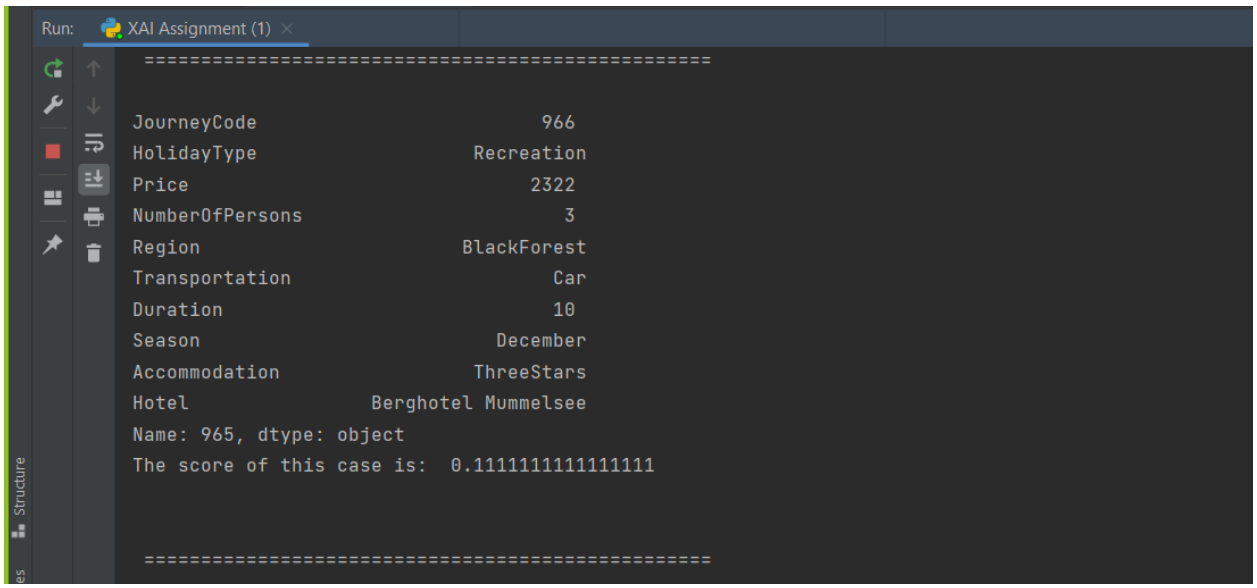
The user answers December and relevant cases would be populated.



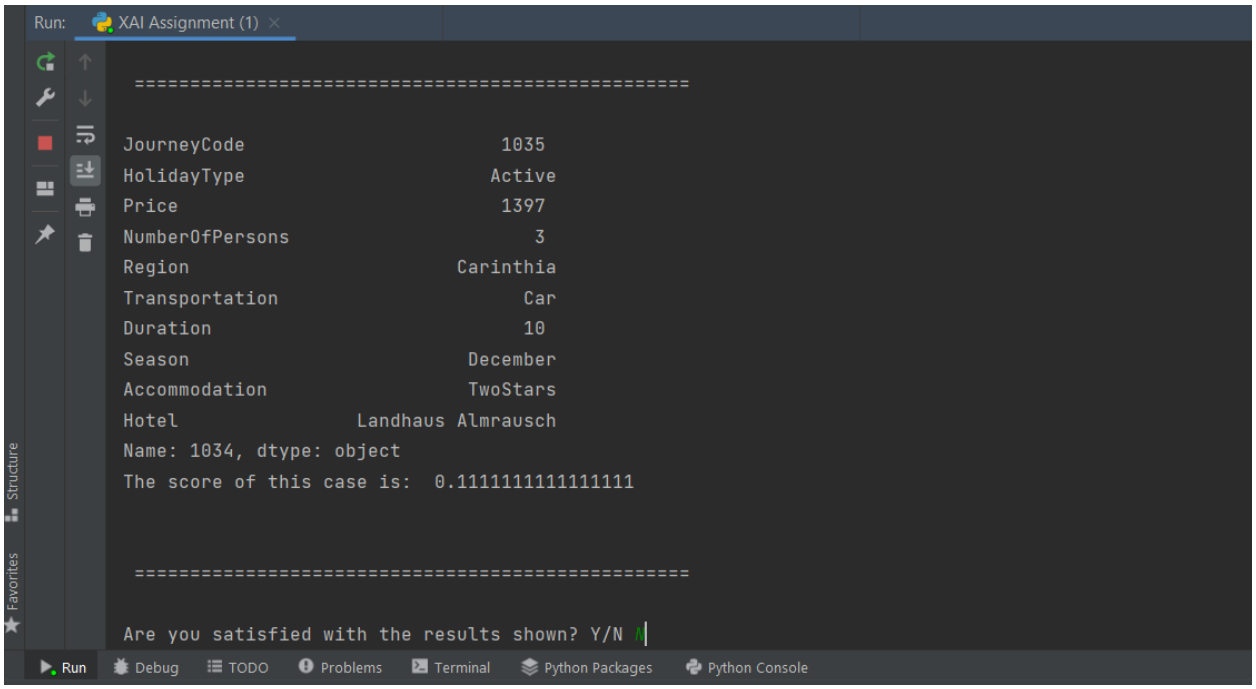
Case 1:


Run:
XAI Assignment (1)
What season are you looking for?December
indices [894, 966, 1035]
JourneyCode 894
HolidayType Recreation
Price 799
NumberOfPersons 3
Region Allgaeu
Transportation Car
Duration 10
Season December
Accommodation HolidayFlat
Hotel H.Flat Sonnenhalde
Name: 893, dtype: object
The score of this case is: 0.1111111111111111
==========
Structure
Favorites

Case 2:

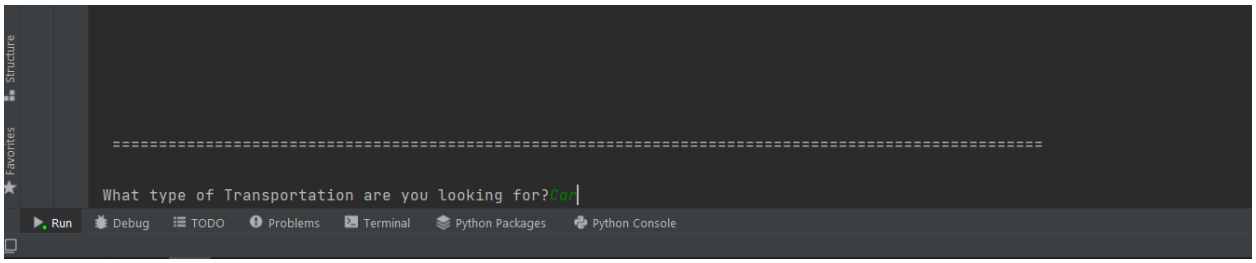

Run:
XAI Assignment (1)
==========
JourneyCode 966
HolidayType Recreation
Price 2322
NumberOfPersons 3
Region BlackForest
Transportation Car
Duration 10
Season December
Accommodation ThreeStars
Hotel Berghotel Mummelsee
Name: 965, dtype: object
The score of this case is: 0.1111111111111111
==========
Structure

Case 3:

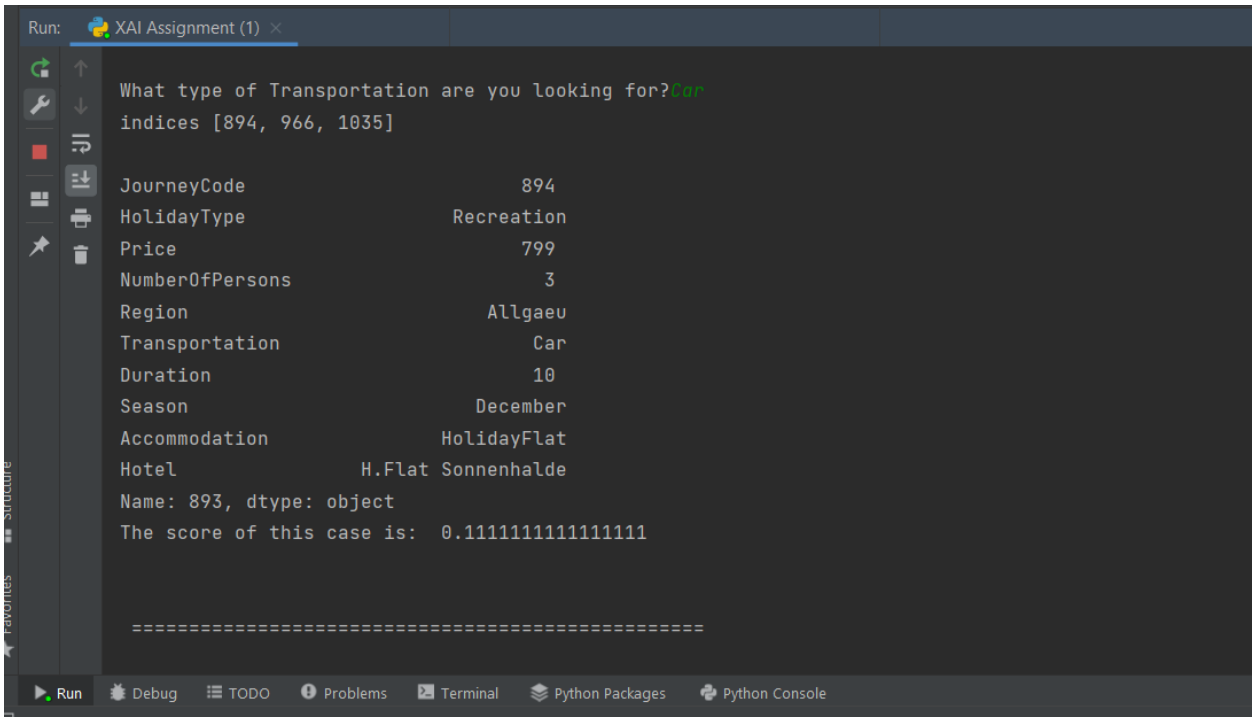


Q4. Type of Transportation.

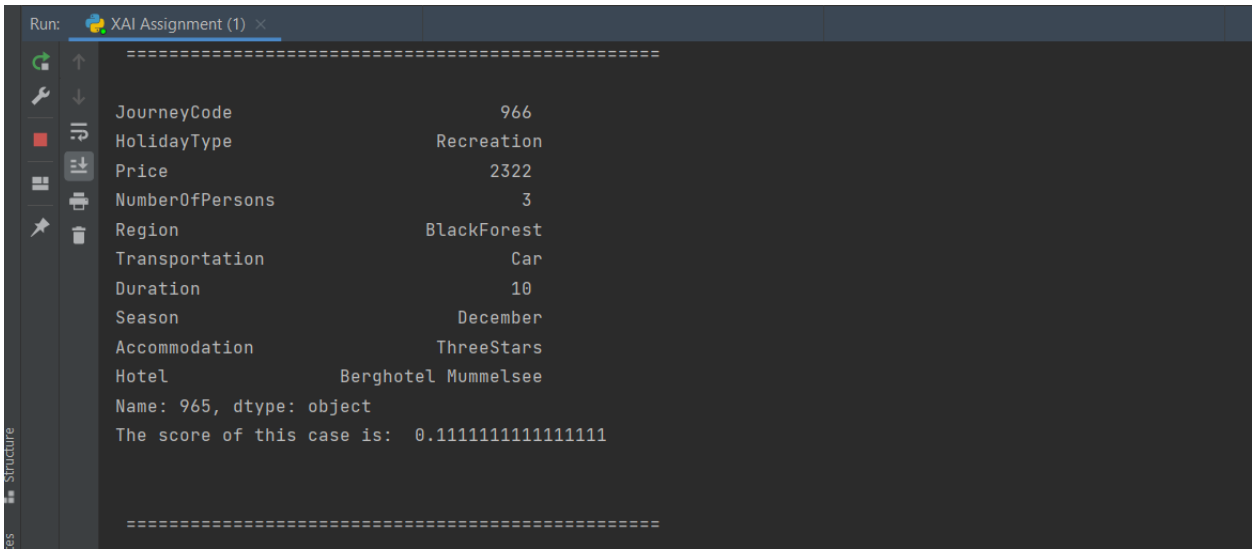
User answers Car



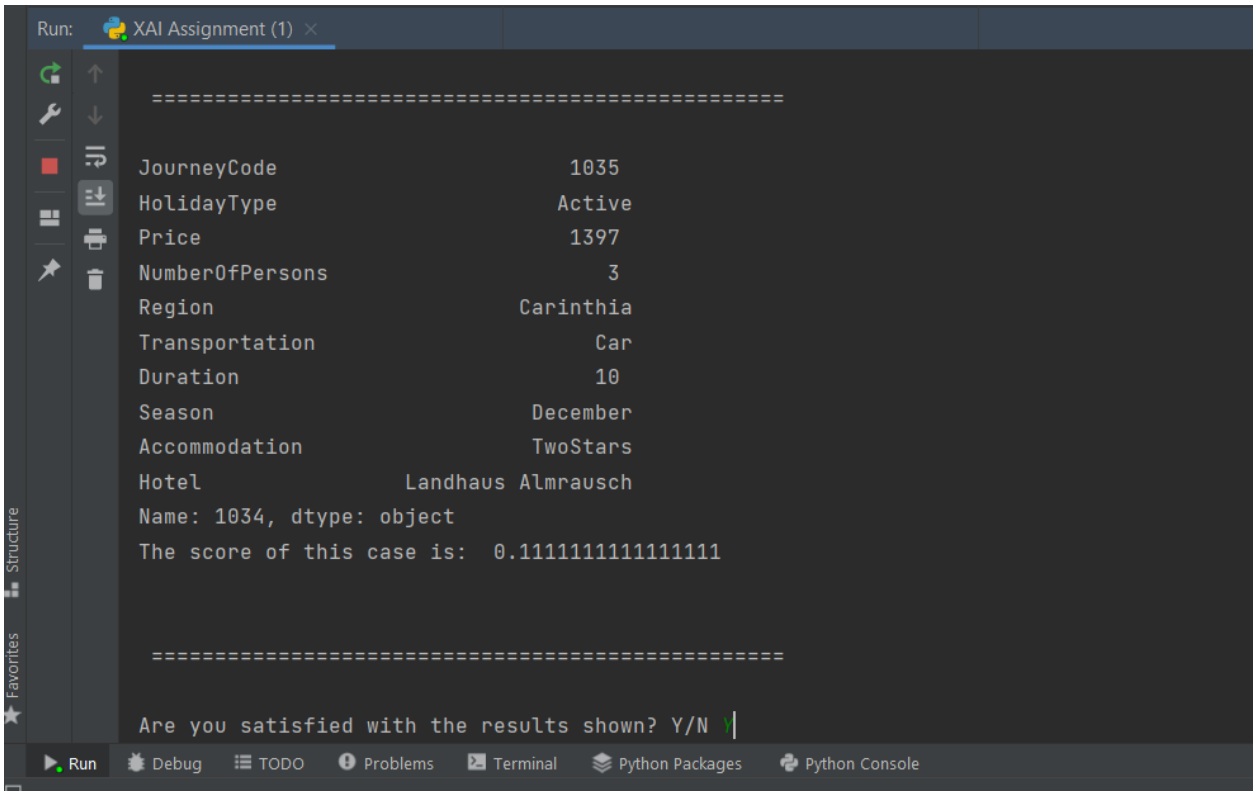
Case 1:


Run:
XAI Assignment (1)
What type of Transportation are you looking for?Car
indices [894, 966, 1035]
JourneyCode 894
HolidayType Recreation
Price 799
NumberOfPersons 3
Region Allgaeu
Transportation Car
Duration 10
Season December
Accommodation HolidayFlat
Hotel H.Flat Sonnenhalde
Name: 893, dtype: object
The score of this case is: 0.1111111111111111
==========
Run
Debug
TODO
Problems
Terminal
Python Packages
Python Console

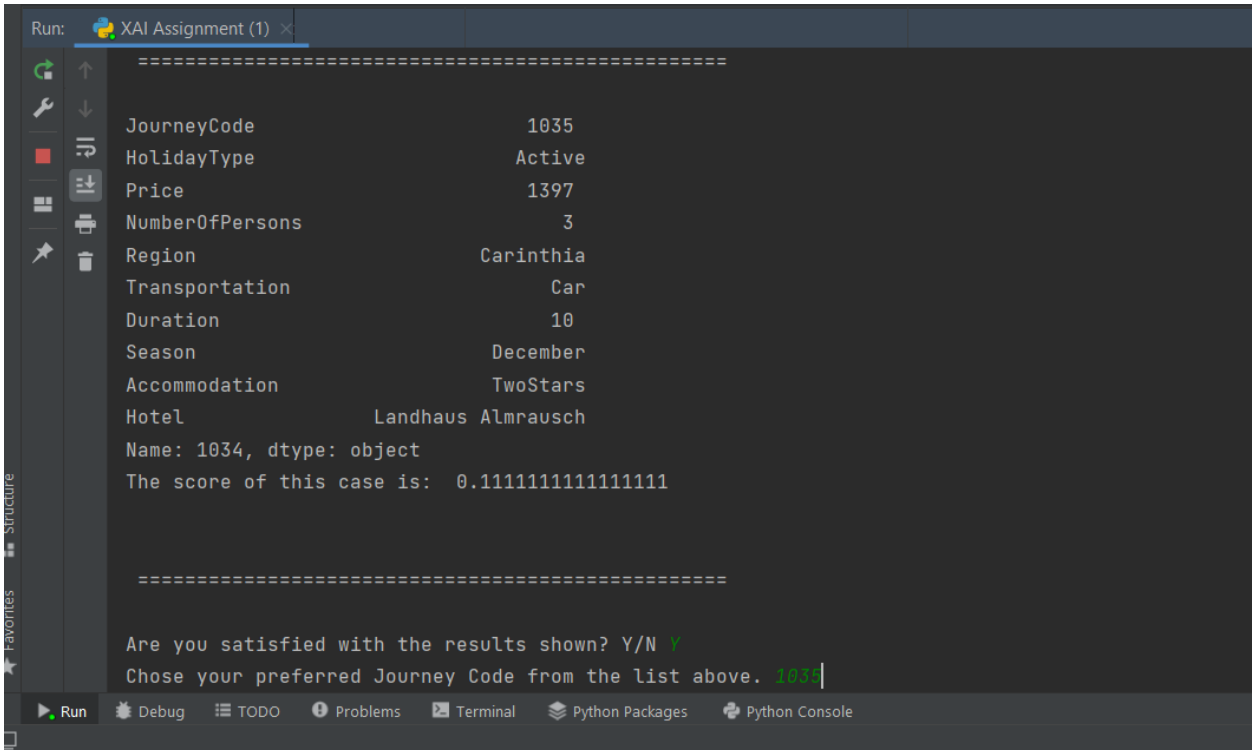
Case 2:


Run:
XAI Assignment (1)
==========
JourneyCode 966
HolidayType Recreation
Price 2322
NumberOfPersons 3
Region BlackForest
Transportation Car
Duration 10
Season December
Accommodation ThreeStars
Hotel Berghotel Mummelsee
Name: 965, dtype: object
The score of this case is: 0.1111111111111111
==========

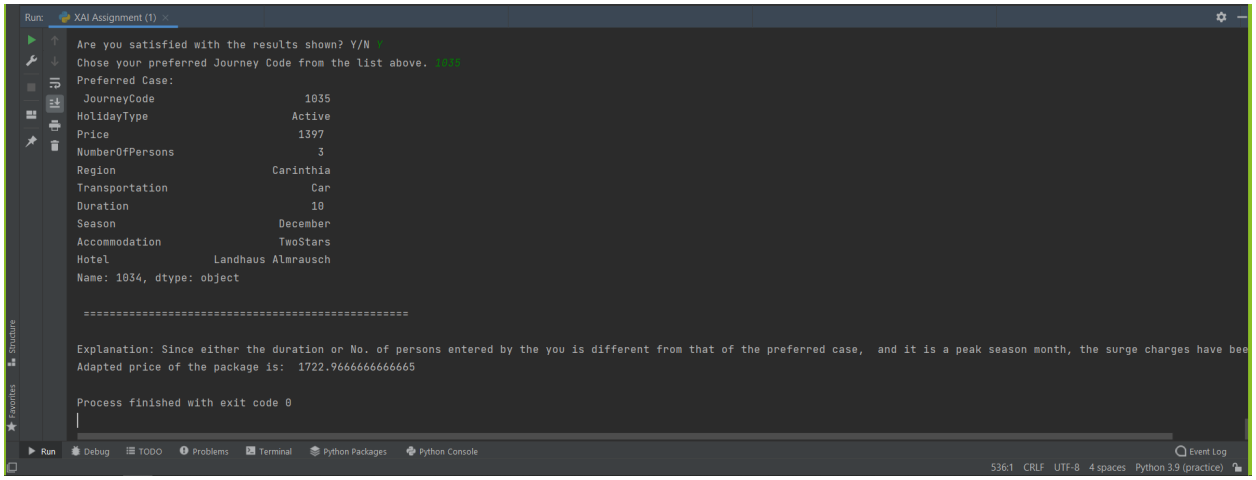
Case 3:



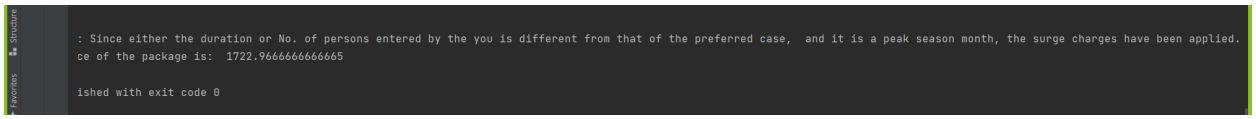
Now the user is satisfied with the case and chooses case 1035 as the preferred case.



The preferred case would be returned to the user with the adapted price. The price is increased from 1035 to 1722.9 as the duration is adapted to 12 days and surge charges for December are applied.



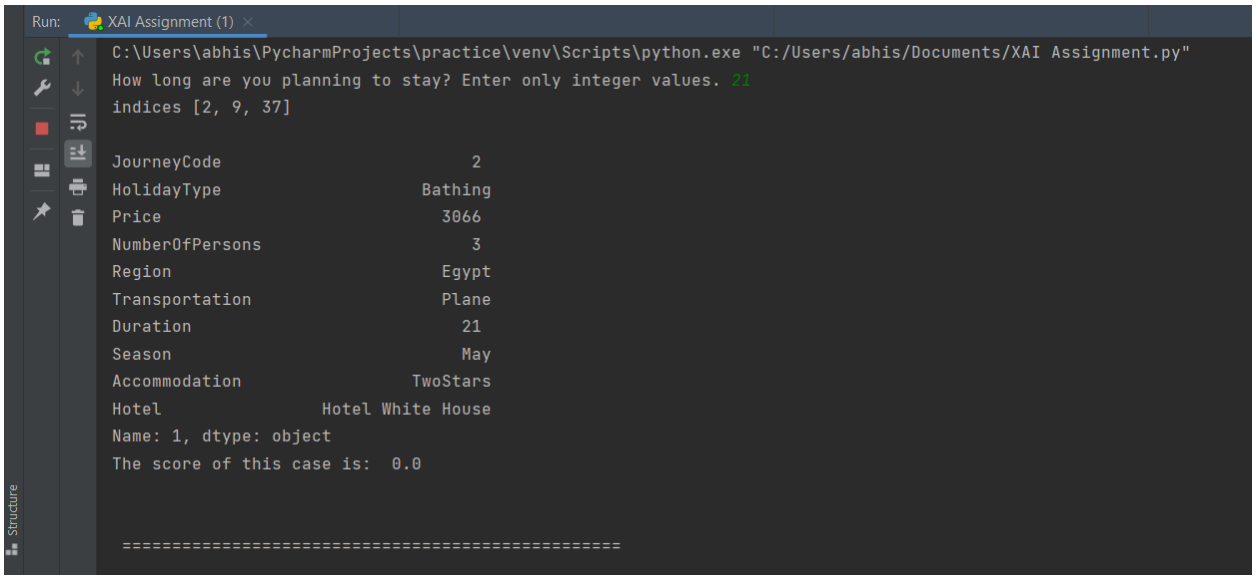
Explanation for the predicted price is provided to the user.



Example 2: *(When there is a perfect match with the preferred case)*

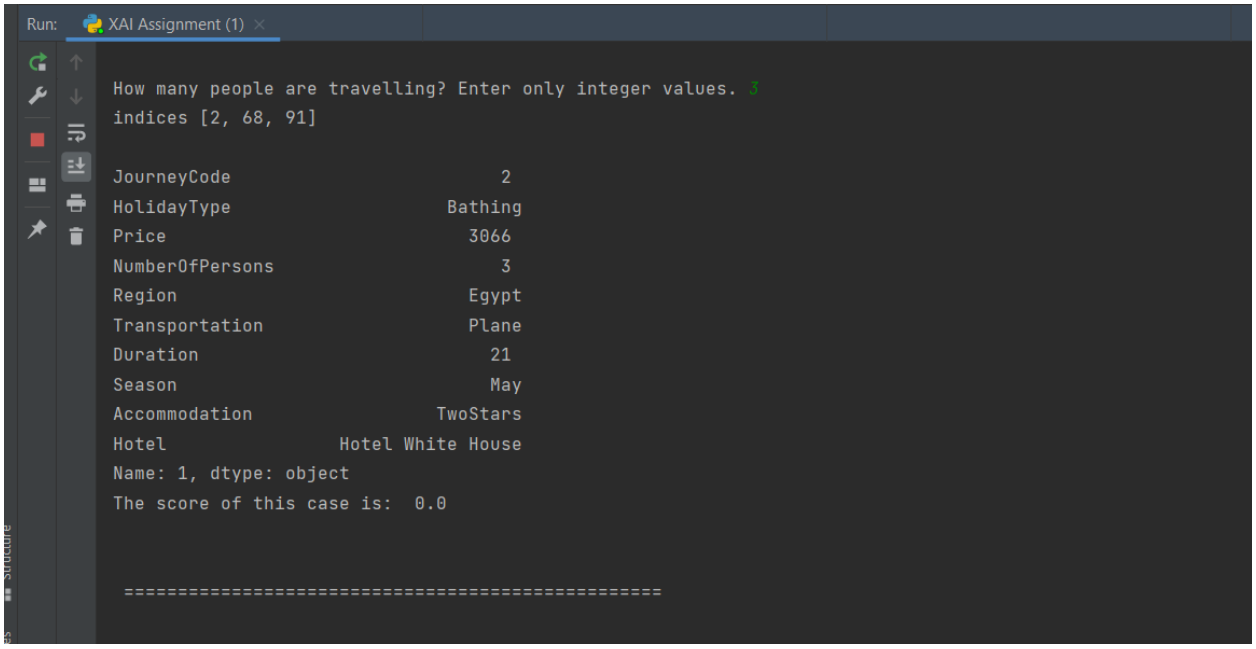
Q1. Duration of stay

User entered 21 and the top 3 relevant cases are populated.

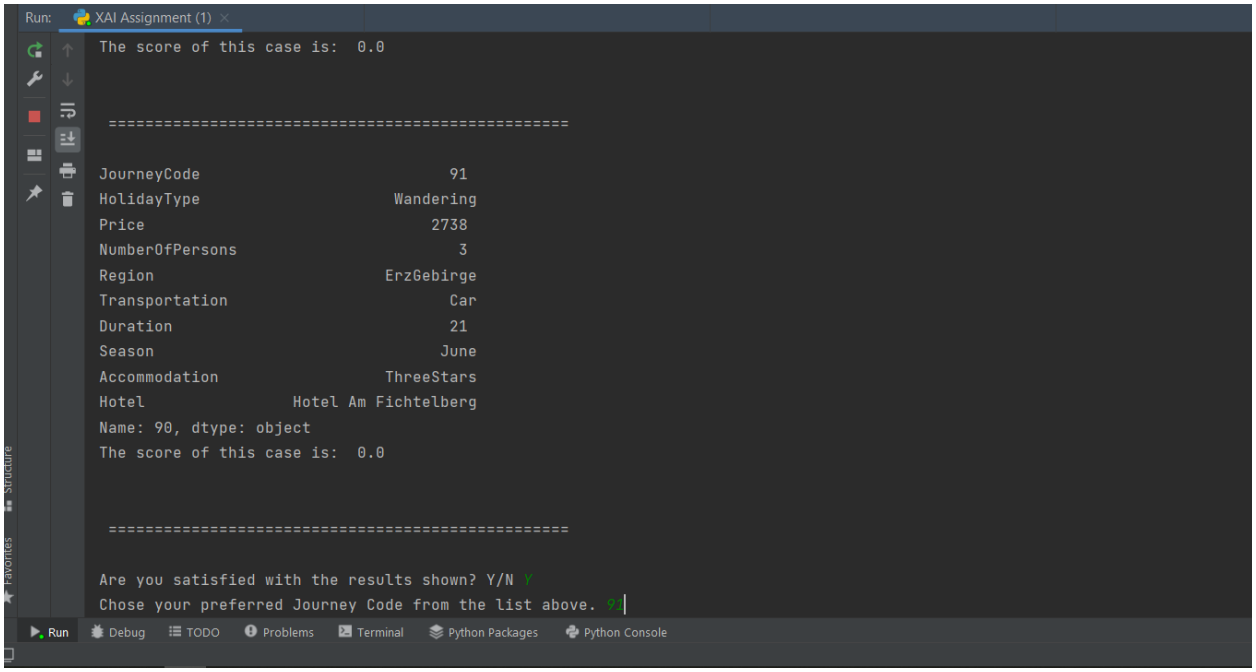


Q2. Number of people traveling.

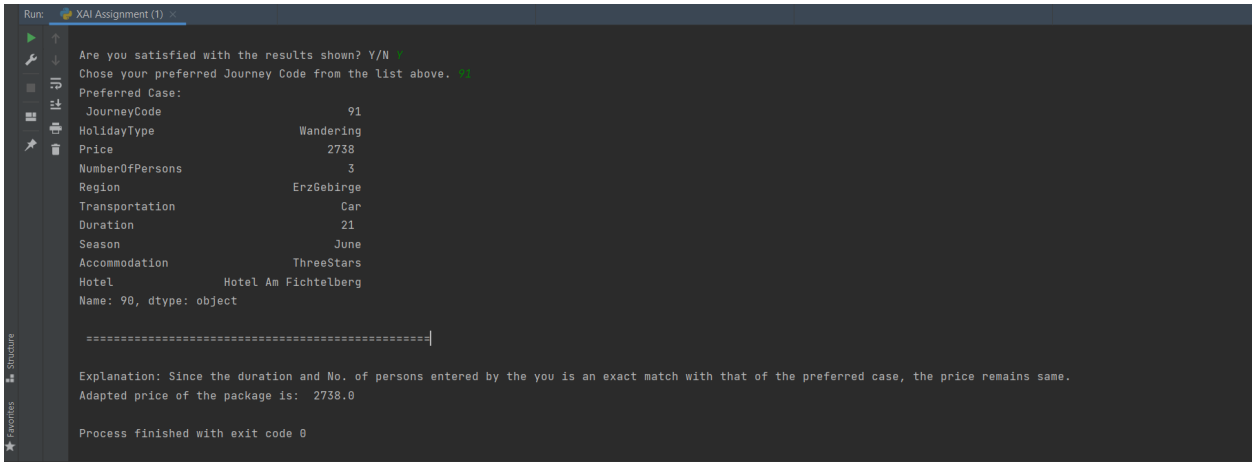
The user answered 3 and as there is an exact match, top 3 relevant cases will be populated again.



The user is satisfied with the results and chooses a preferred case as one with the Journey Code 91.



The preferred case is retrieved and as there is an exact match, predicted price is same as that of the preferred case, i.e. 2738. Explanation for the predicted price is provided for user's clarity.

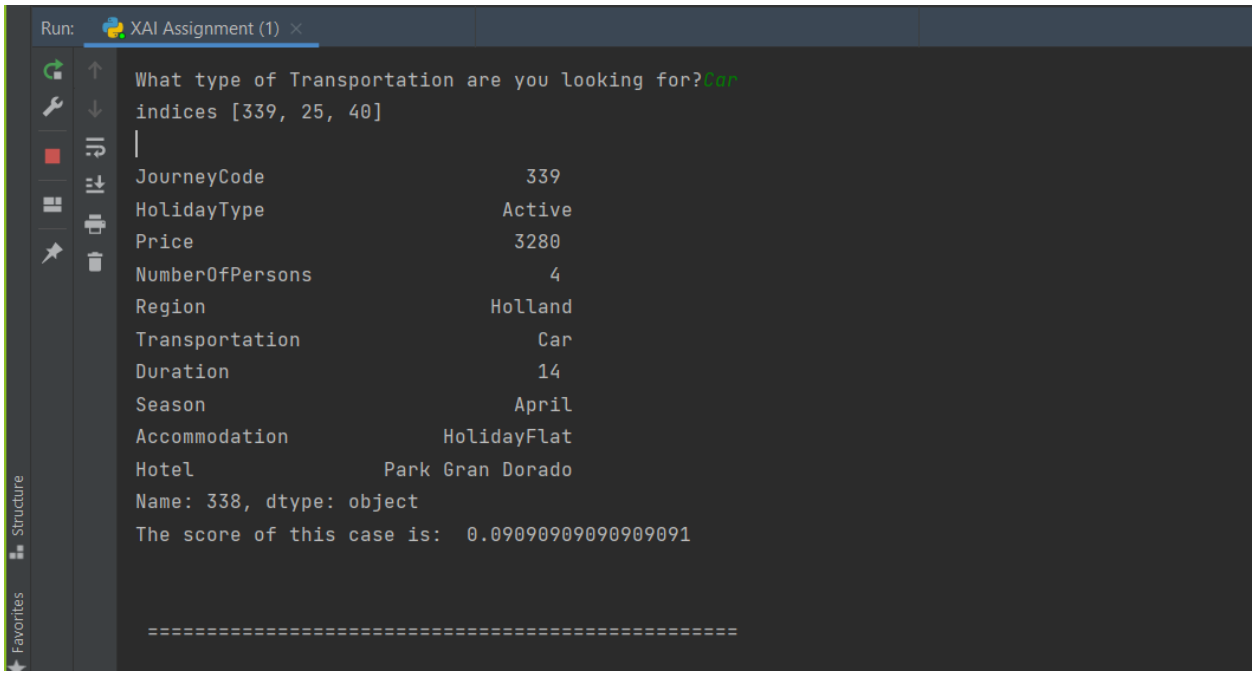


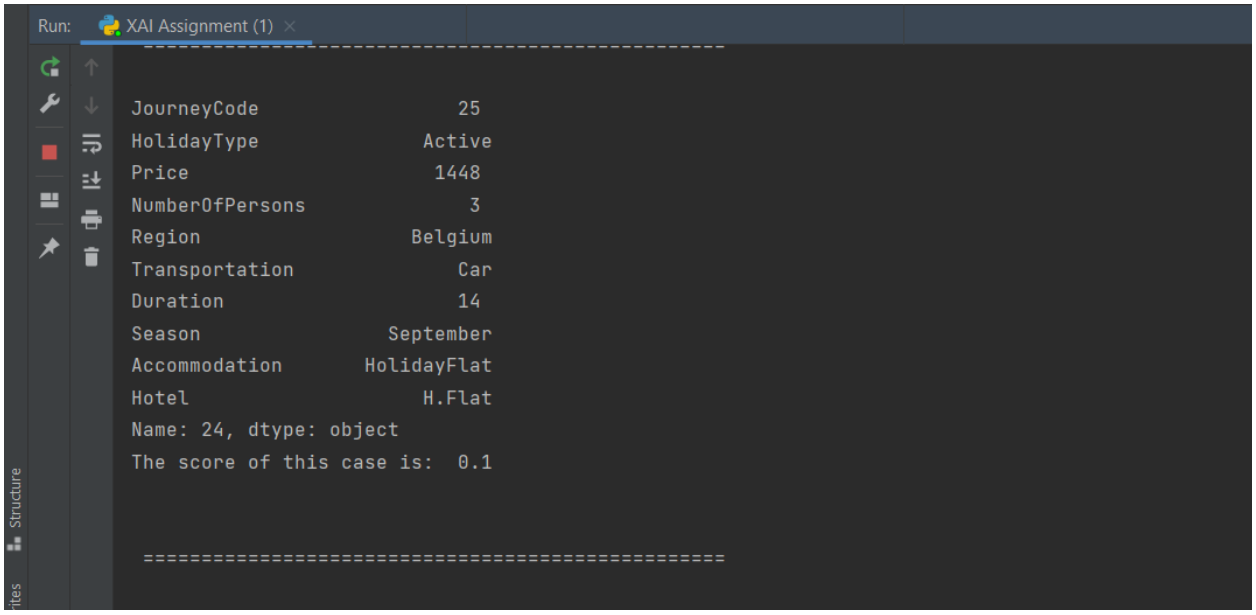
Example 3: *(To show difference between retrievals in unweighted and weighted features)*

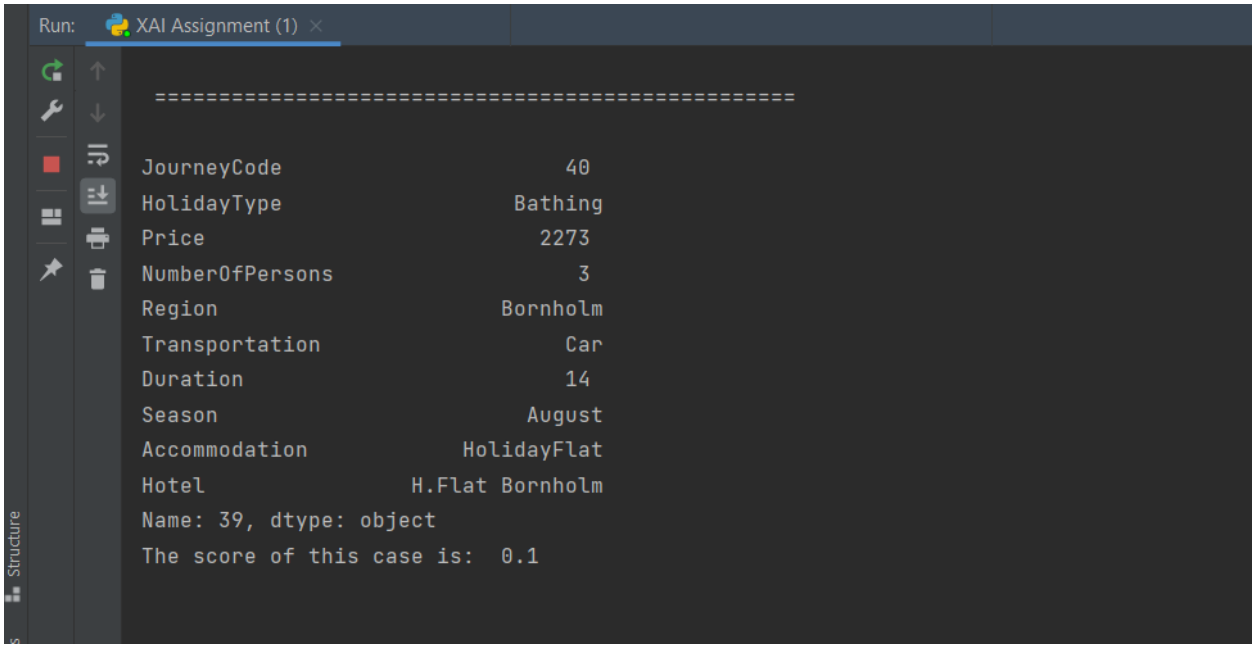
Values entered for the features:

Duration: 14, NoOfPersons: 3, season: April, transportation: Car

→ Unweighted, i.e. all the features are uniformly weighted

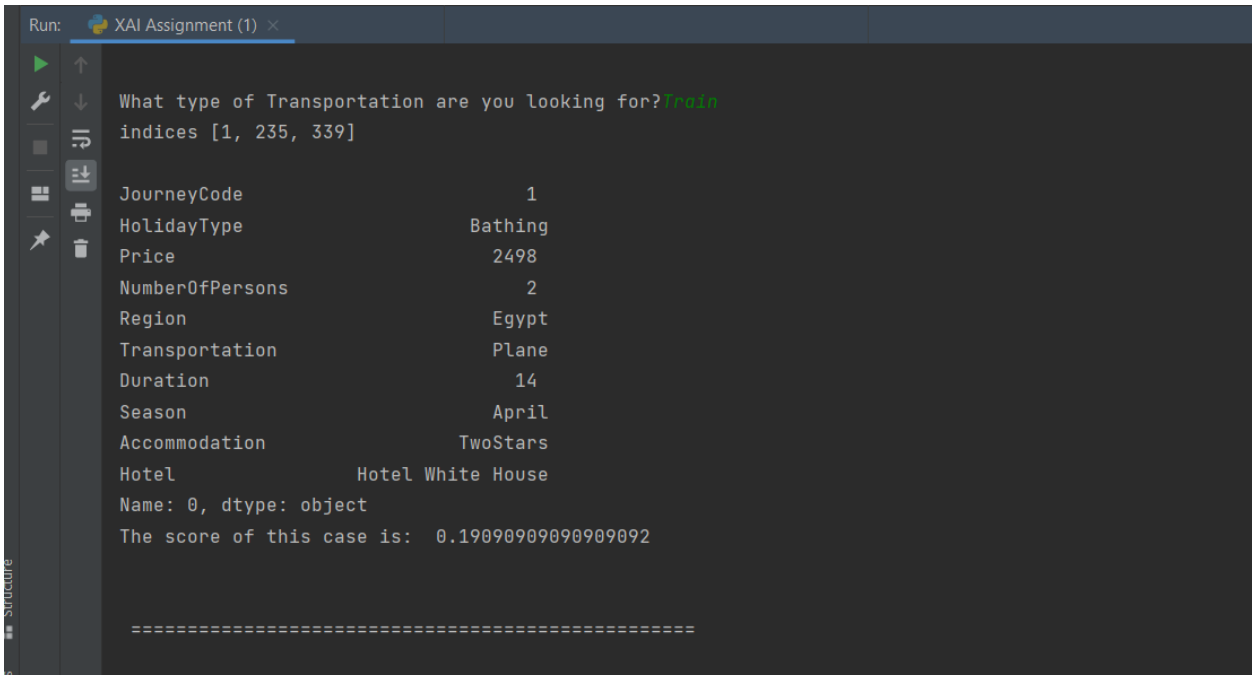


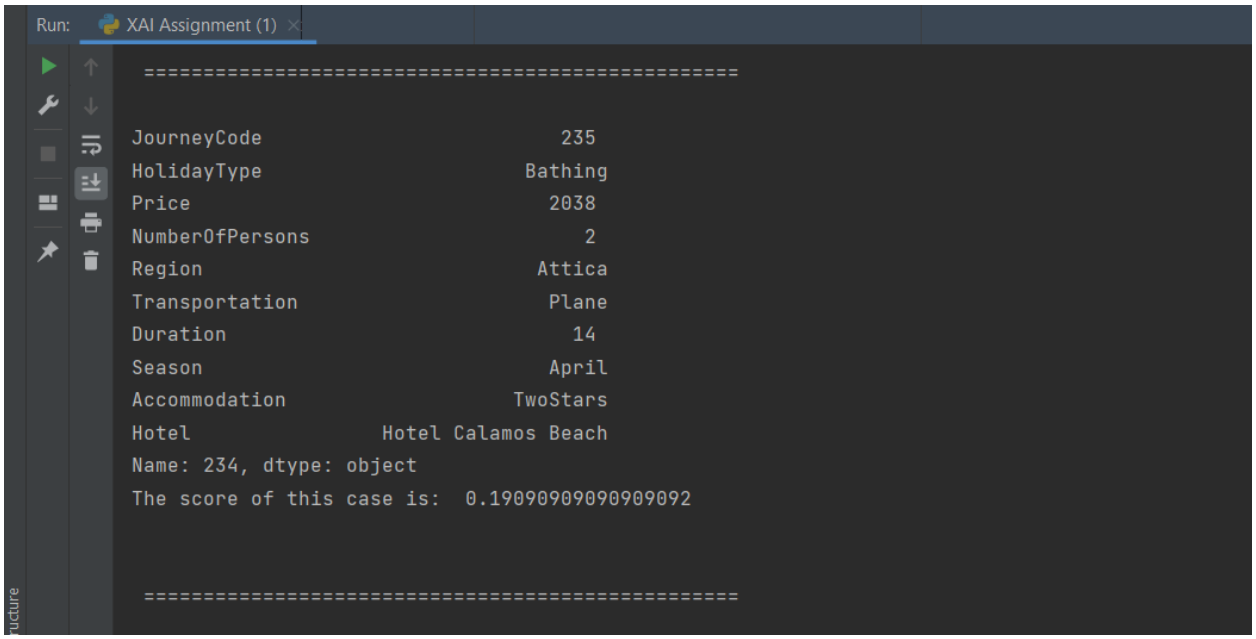


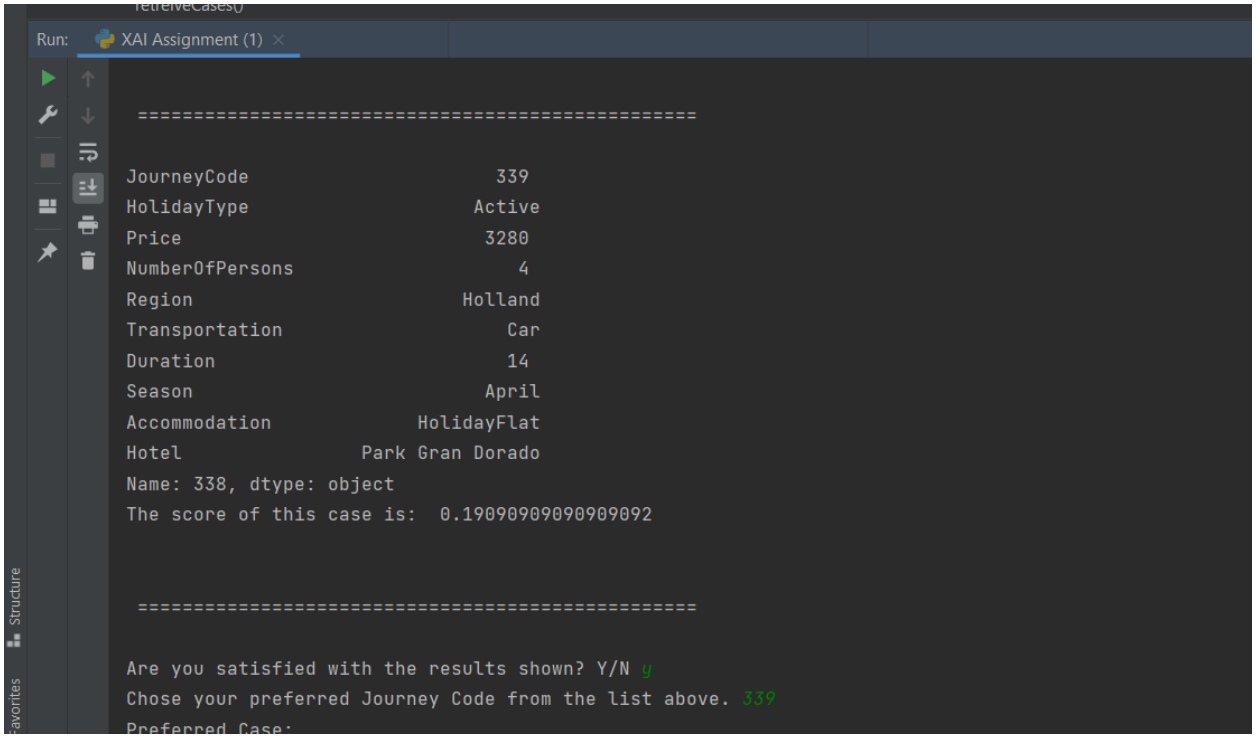


As we can see in the final result, if the features are unweighted, the results need not necessarily be matched with them. For example, here, the season entered by the user was April, but only one of the cases retrieved contains April. However, other features were matched with the closest results.

→ **Weighted, i.e. season is heavily weighted and multiplied with a factor of 5**







As we can see in the above three screenshots, since the season is now a weighted factor, all the results match the season entered by the user, i.e April. Thus, weighting a feature increases its priority and the results are matched according to that feature first then considering the other features as per their weights.

QUESTIONS:

1. **If you refined the method above for calculating similarity, describe what you did (otherwise you can simply say that you followed the assignment statement). Describe your system's approach for selecting questions to ask.**

   We followed the given method to calculate similarity scores.

   We played around with the cases and their influence on the price parameter and based on our intuition came up with the order in which questions should be asked for the quickest convergence.

   We believe that the most important factor while planning a trip is the duration of the stay followed by the number of people we would be traveling with. Also, season plays a crucial role in determining the prices as prices surge during the peak season and are quite low during the off season. Transportation is an independent factor which determines the price based on the mode, for instance, Plane is costlier compared to a car and so on.

   The order in which the questions were asked:

   1.Duration 2.NoOfPersons 3.Season 4.Transportation 5.Accommodation 6.Region 7.HolidayType

2. **Test the system on the data, with equally weighted features (i.e., all weights = 1). What do you observe about the quality of the selection of packages? Are the solutions intuitively reasonable? Why or why not?**

   Yes, when only 2 or 3 features are requested, we get a good match but as more and more features are requested, the results start deteriorating and the minimum distance comes into play as every feature is equally weighted here.

   For instance, when the user enters following features:

   Duration: 14, NoOfPersons: 3, season: April, transportation: Car

   Amongst the top 3 cases retrieved, only one of the cases match their season feature with April as shown in Example 3.

3. Tune the system by defining unequal feature weights, based on your sense of feature importance. Report the weighting you chose and your rationale.

Our model works on the intuition that once the weight of the feature is increased, more importance would be given to that particular feature and the model would try its level best to match that feature first followed by the rest.

Rationale behind giving higher importance to a feature is the intuition that number of people and duration play a crucial role in determining the price of the package.

4. What do you observe now for the quality of retrievals? How does it compare to the version with equal weights?

When the features are weighted, the quality of retrievals improve. Our algorithm makes sure to return the case which has maximum matching with the features with higher weights.

For instance, the same query when carried out with giving the season feature a weight of 5, all the 3 cases retrieved at the end would have the season matched with April in them. Reference: Example 3.

5. State in English plausible adaptation rules for adapting the price of a package based on NumberOfPersons, Duration, and Season. The rules should describe a possible difference between a retrieved case and a current problem, and the associated change to be made in the solution of the retrieved case. For example, if the domain were predicting travel times, a rule might be expressed as: "If the old trip was not during rush hour and the new prediction is for rush hour, increase travel time by 20%". Add code to adapt the solutions of the retrieved cases using your rules. The rules may be hard coded. Test the revised system to generate sample output.

We would be using a unitary method to adapt to the price for both Number Of Persons and Duration. That is, if the price for 4 people is given, we would calculate the price for 1 person and then calculate that for the number of people required. Similar calculations can be done for duration as well. For the season, we are keeping in mind the surge rates that can be applied based on the type of rush. As

we know from the real world, months like June July are high season months due to summer vacations, November December are again high season due to Winter breaks, we would apply a surge of 1.5% with respect to the current month's price. Similarly, even the months September October and January February are close to the holiday season and hence, we applied a surge charge of 1.3%.

6. What are the strengths and limitations of your approach for this domain (both similarity and question selection)?

Strength:

For queries perfectly matching the case base, we get predicted price value precisely. For difference in Number of Persons, Duration and Season, easy adaptation rules can be applied to predict price as accurately as possible. Here, as we are applying surge rates based on intuition and real world conditions, they are likely to change and results might not be as accurate as possible.

Limitations:

In some specific cases, the feature values entered by the user might not match at all with the cases present in the case base. In such a situation, our system would try to match all the features as closely as possible, however, it might still not be able to provide closely matching cases which could eventually compromise user trust and satisfaction.

Also, the process that we have followed to assign the weights to the features and the order in which we ask the questions is purely our intuition and might not resonate properly with that of the user's.

7. This system provides explanations of why cases are chosen and how they are adapted. What are the weaknesses of both types of explanations here, and how could they be improved?

The price calculation works on the basis of extracting information from the case base as well as applying adaptation to the features if required.

The explanation for the cases that are chosen based on user inputs is self explanatory and is simple to understand. The cases are chosen depending on their distance from the user's inputs. This has a small limitation that a user might not sometimes understand why a particular case did not exactly match with the feature values entered by him/her.

The limitation for the explanation for the price adaptation is that since it consists of hardcoded rules, it might sometimes fail to adapt to the changing circumstances like month popularity and so on.

This can be improved by taking into account even more finer details about the individual factors at play behind the possible changes of the circumstances.

8. A statement of relative contributions of the group members.

Equivalent