

# CSE2062

# Object Oriented Programming

# Instructure : Şakir Bingöl

|  |  |  |
| --- | --- | --- |
|  | Student ID Number | Name & Surname |
| 1 | 150717052 | Tahir Temizkan |
| 2 | 150718003 | Mehmet Temmuz Boyraz |
| 3 | 150718021 | Uygar Gündüz |
| 4 | 150718039 | Şerife Küçükköşker |
| 5 | 150718050 | Mehmet Karaca |
| 6 | 150720842 | Talha Yavçın |

HOSPITAL ABOVE:

*Hospital Appointment System*

This project’s aim in general is allowing patients to take appointments in an easy manner. While doing that helps patients to learn about the doctors in the hospital and give them an opportunity to see their own BMI and minimal calorie requirement. As an assignment we had to take atleast 5 different type of inputs and give atleast different type of 3 outputs and use necessary concepts such as polymorphism, overload etc. . To achive this we implemented 9 different classes with each has its own purpose. These classes are:

*Doctor Class*

In this class we take care of the doctor’s available days and other attributes such as experiences and departments. If a user of our system would like to see the doctor’s informations this class allows us to proccess the info’s about our doctors and shows them to the user as an output.

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

*Department Classes(Neurology,Cardiology,Dermatology,Plastic)*

These classes are subclasses of the class “Doctor”. In these classes we separate the doctors from each other as giving each of them a department. These classes simply gives the doctors their final identity.

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu metin içeren bir resim

Açıklama otomatik olarak oluşturuldu metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

*Interface BMI*

In our Interface we introduced a bodyless method “bmiCalculate”. We are going to use this method to calculate the Body Mass Index of the user.

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

*Abstract Class MinimumCal*

In the abstract class of our project we introduce 2 methods to take height and weight of the user as inputs. These methods called as “enterHeight” and “enterWeight”. After that we introduce the bodyless method “MinCal”. We are going to use this method to calculate the minimal requirement of calories for the user.

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

*We are going to use these two methods if user wants to see these information about themselves.*

*Patient Class*

In this class we process the taken inputs from user about themselves. In the upcoming main class we are going to take the patients name, mail adress, telephone number, age and if they so chose we take their height and weight with the help of our abstract class and interface class. The “Patient” Class is a subclass extends to our abstract class and implements to our interface. And in final we calculate the cost of the appointment.

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

*Our Main Class HospitalAbove*

In the main class we have two methods which are “appointment” and “main”. In the “appointment” method we are going to use the entered department and appointment day to create the appointment. In the “main” method we create our doctors than start our process of taking appointment. First we introduce our hospital and ask the user if they want to see about our doctors and their availability. If they say yes we display the doctors informations. If they say no we go to the next step right away. After that we ask the user about the department and the day they would like to take an appointment. With these steps done we create the appointments first step and start to ask about the patients informations like their name age etc. . After their necessary informations we ask the patient if they would like to know their minimal requirement of calories a day and their BMI. If they say yes we ask the patient their height and weight. If they say no we go to the next step right away. After all these we display the patients informations and appointment informations and ask the user if they have a health insurance. If they say yes we give a discount and print the discounted cost and if they say no we display the raw cost. With all these done patient got their appointment and get the informations about whatever they might want to know.

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

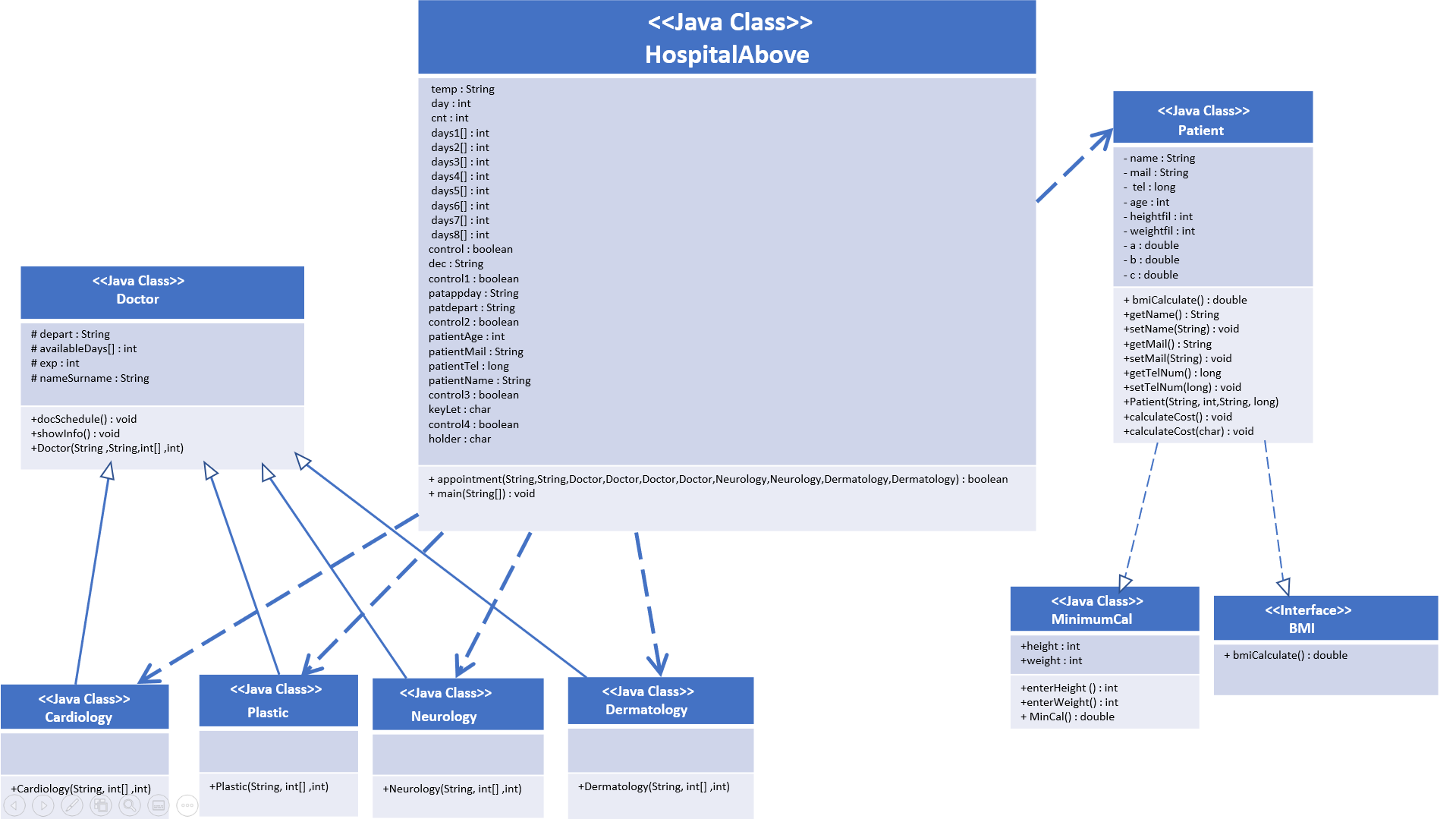
metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

metin içeren bir resim

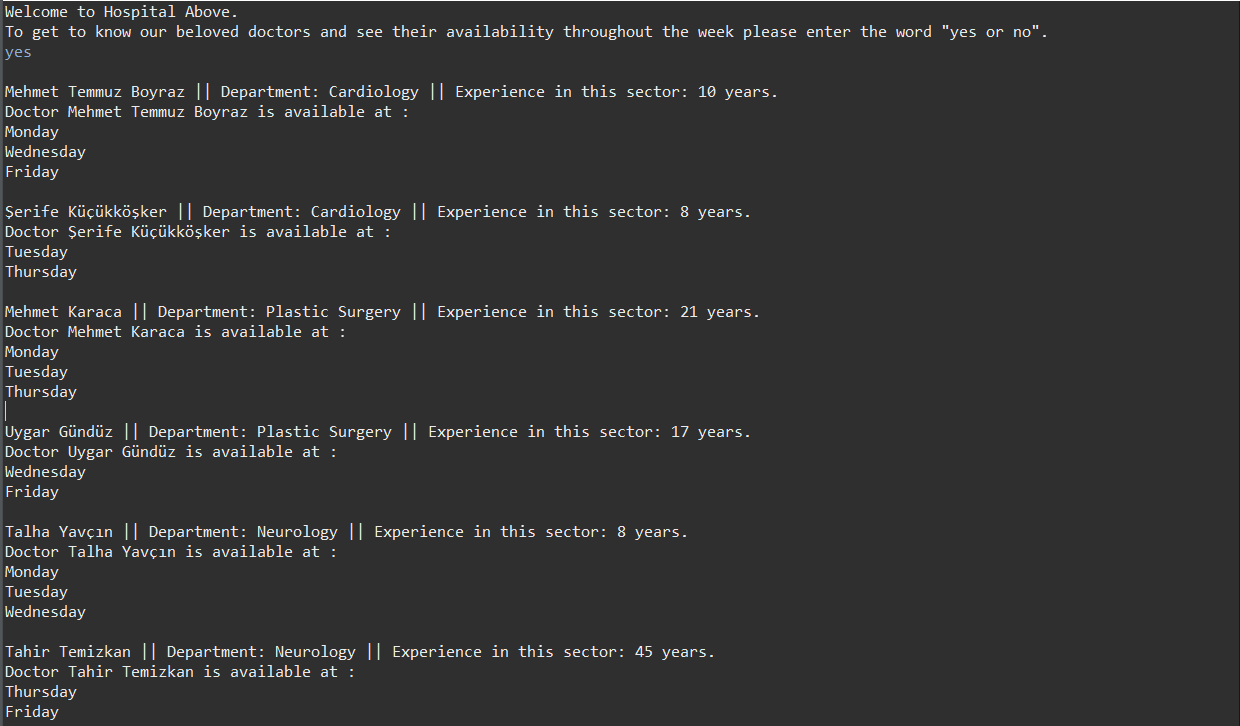
Açıklama otomatik olarak oluşturuldu

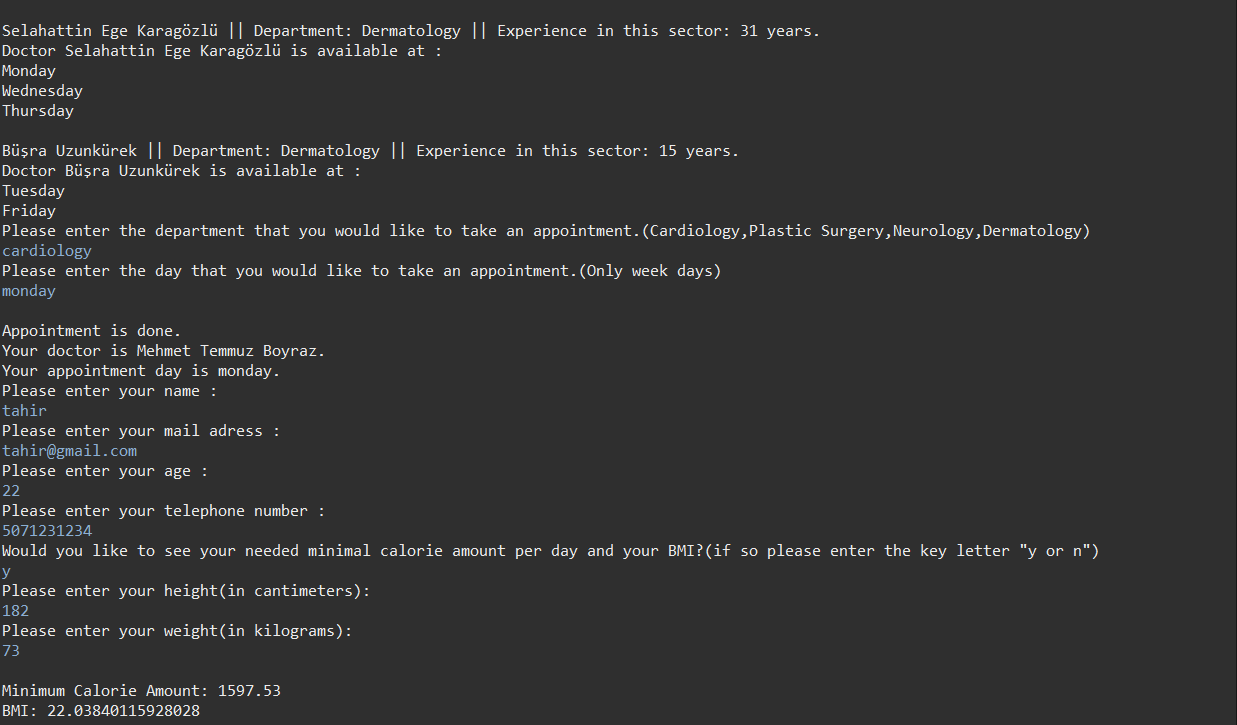
*UML Diagram of Our Project*

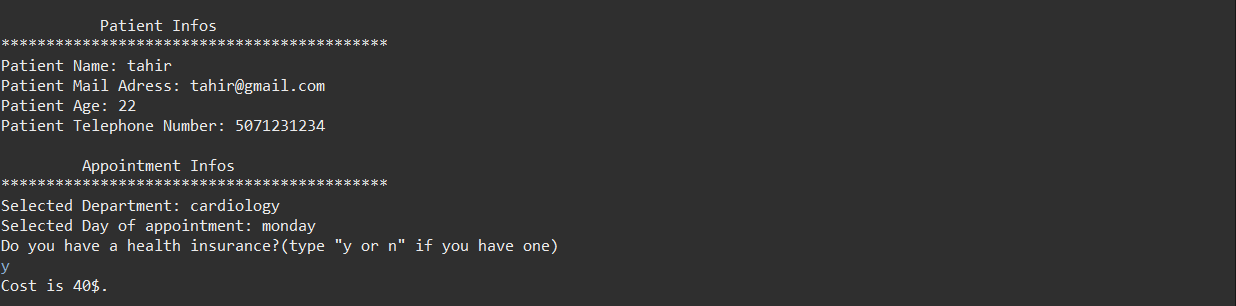


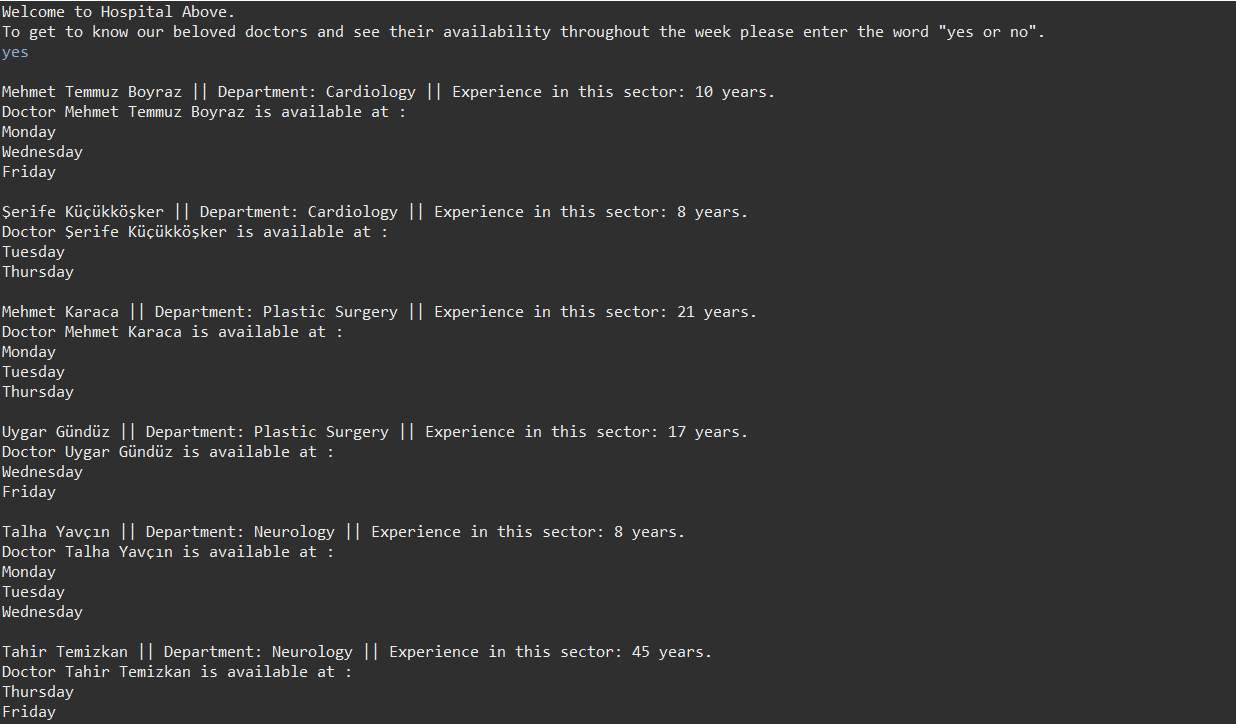
We draw our UML Diagram ourselves because we wanted to learn about UML Diagrams better.

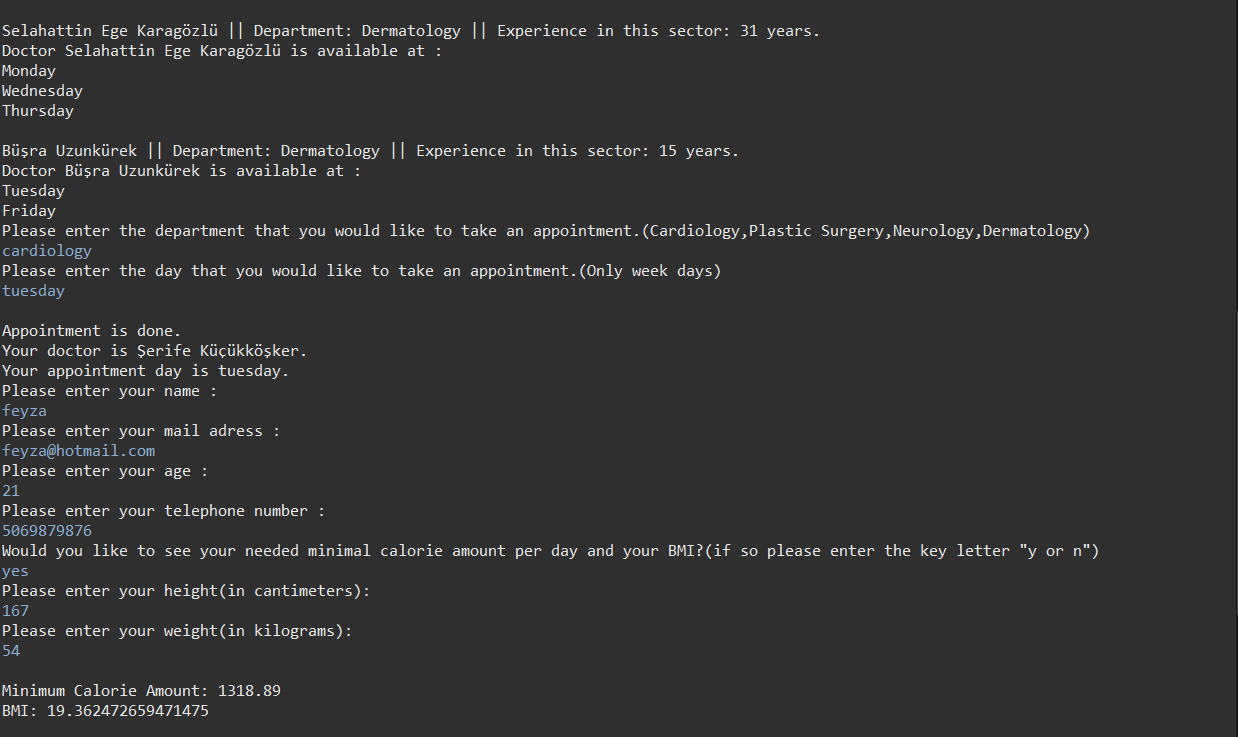
*Outputs of Our Project*

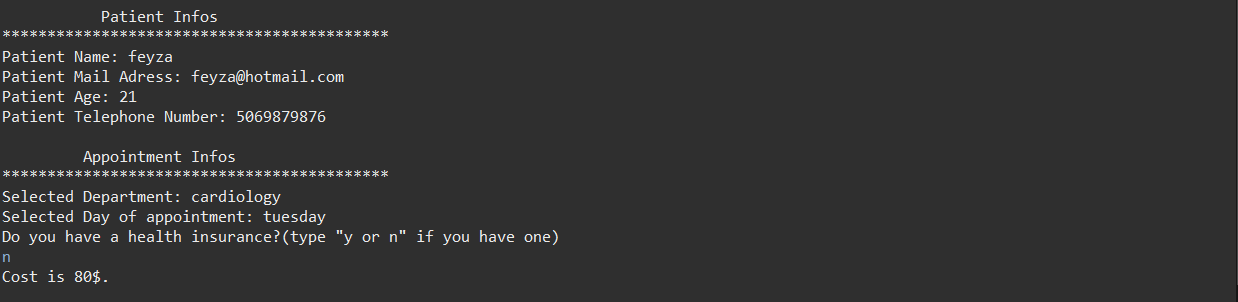
*Output 1:*

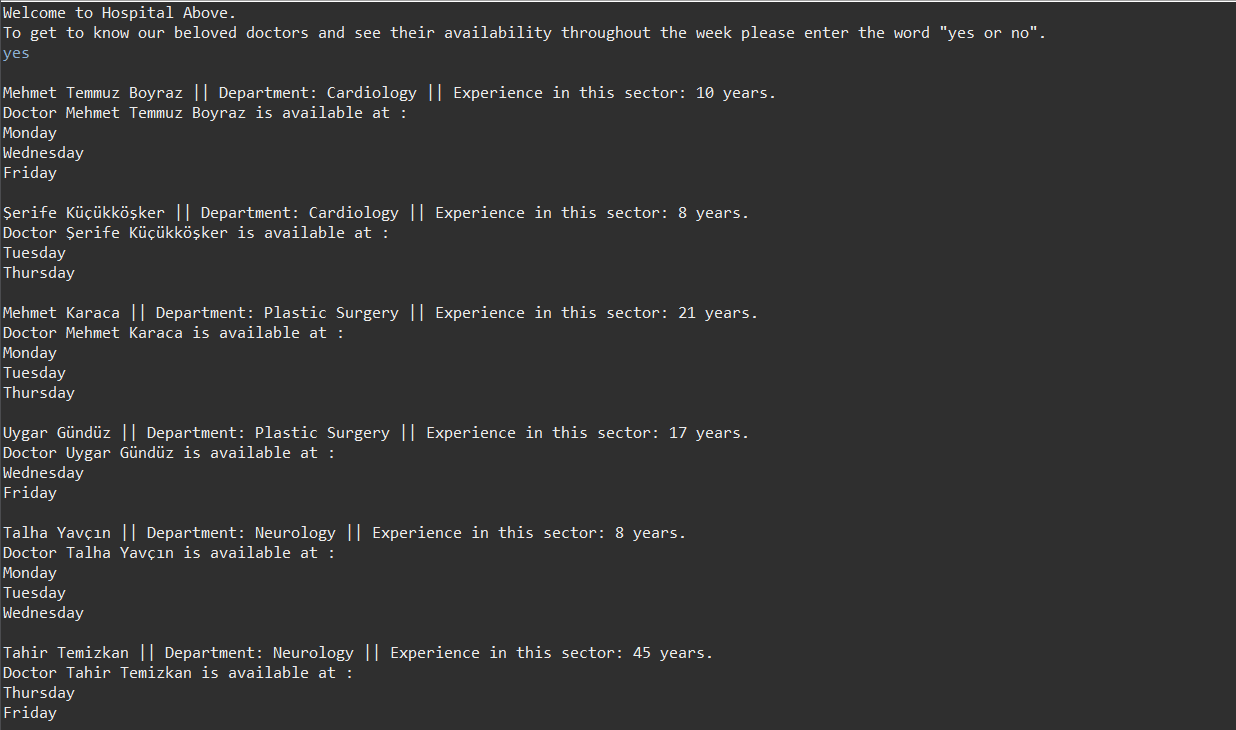


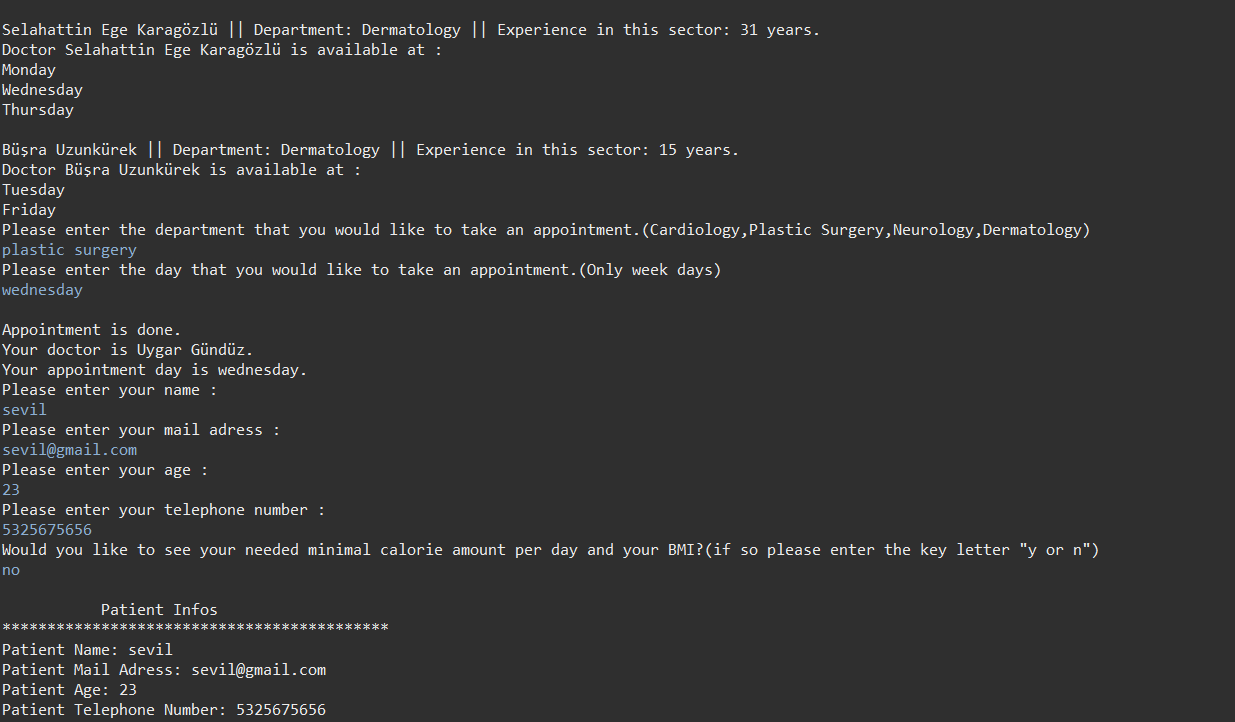


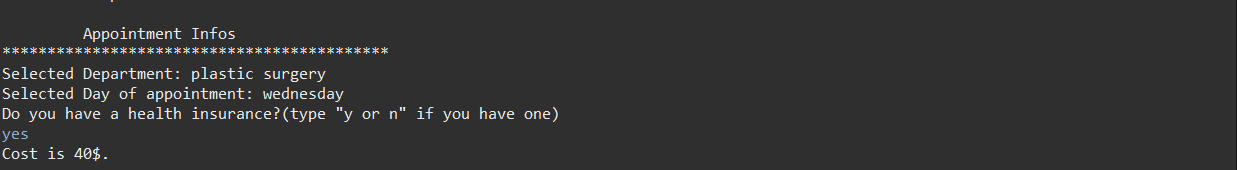
*Output 2:*

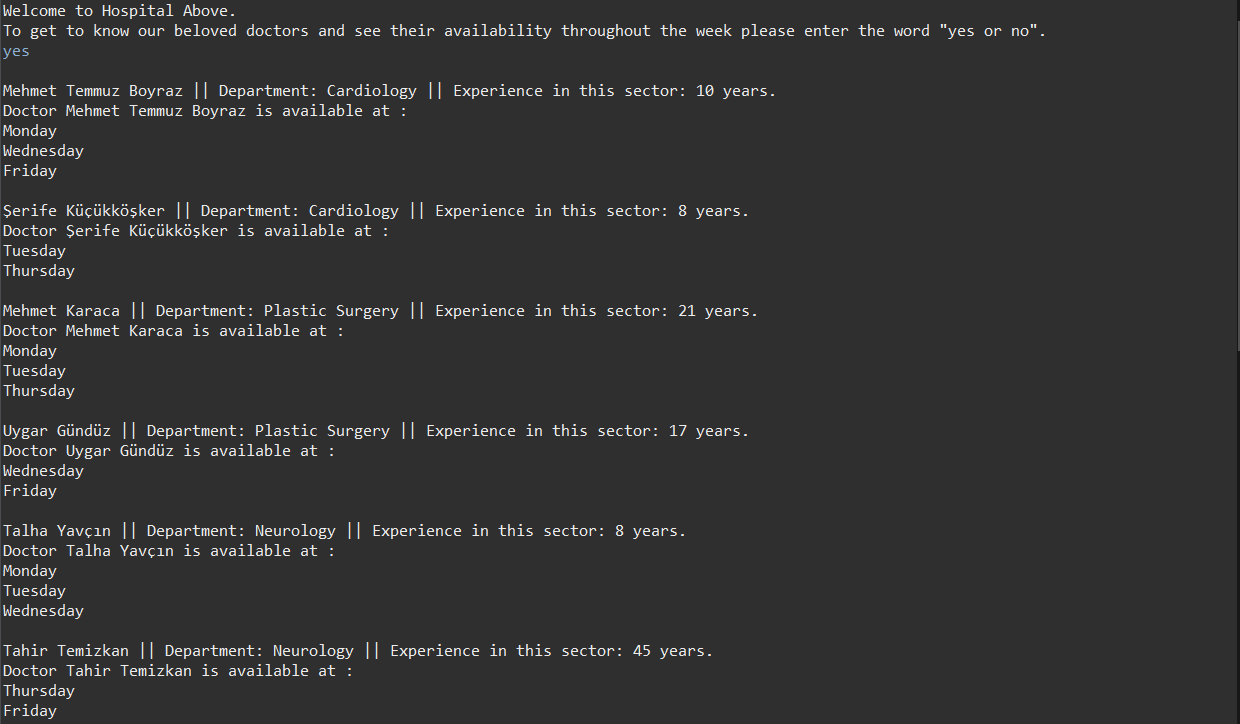


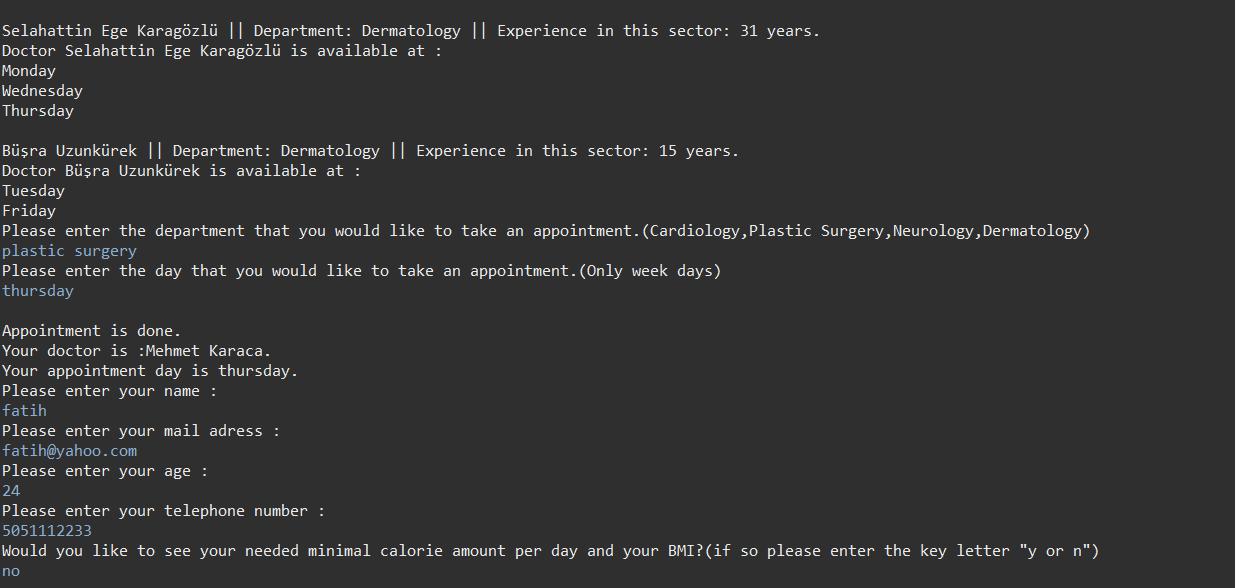


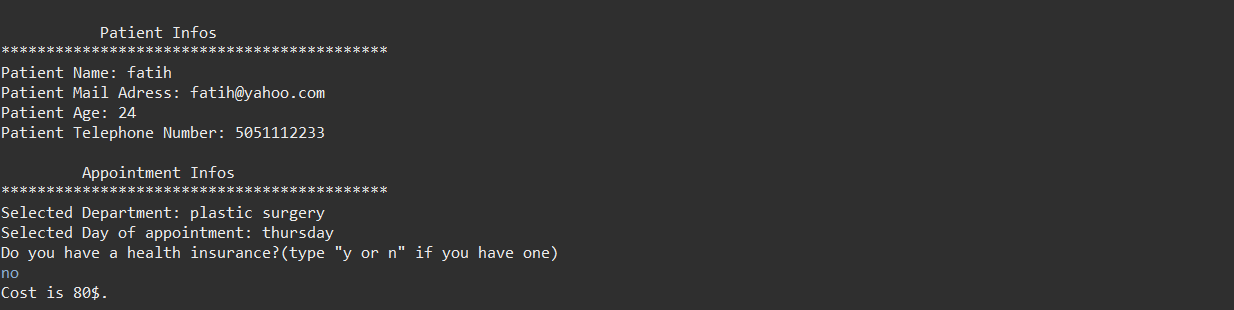
*Output 3:*



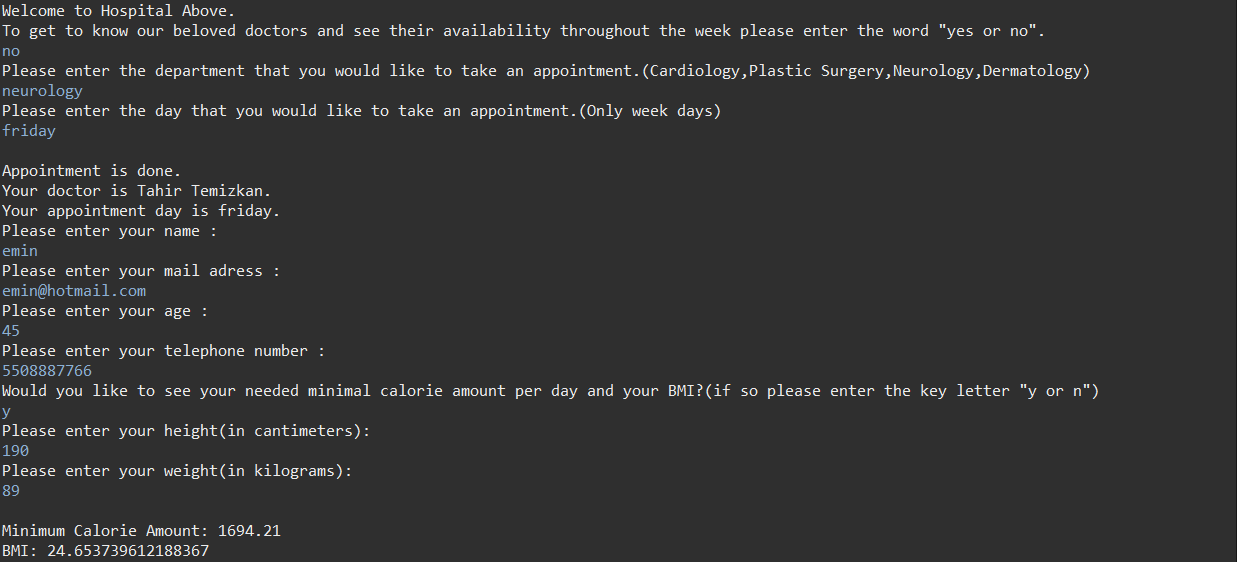
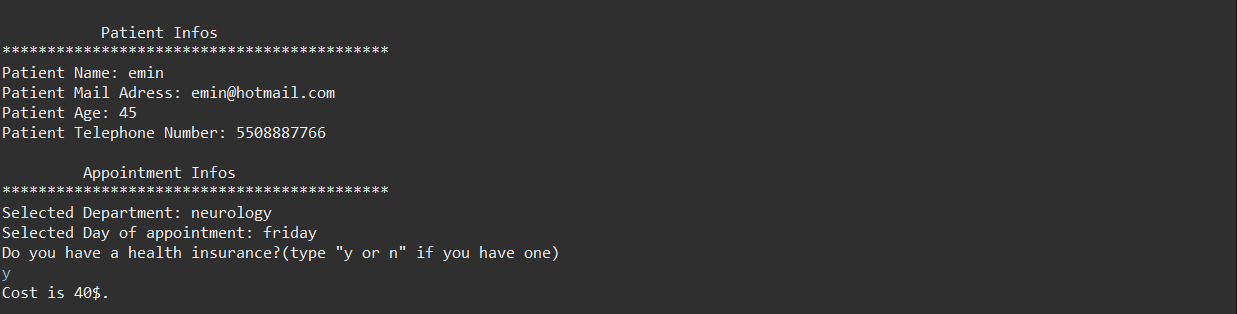


*Output 4:*

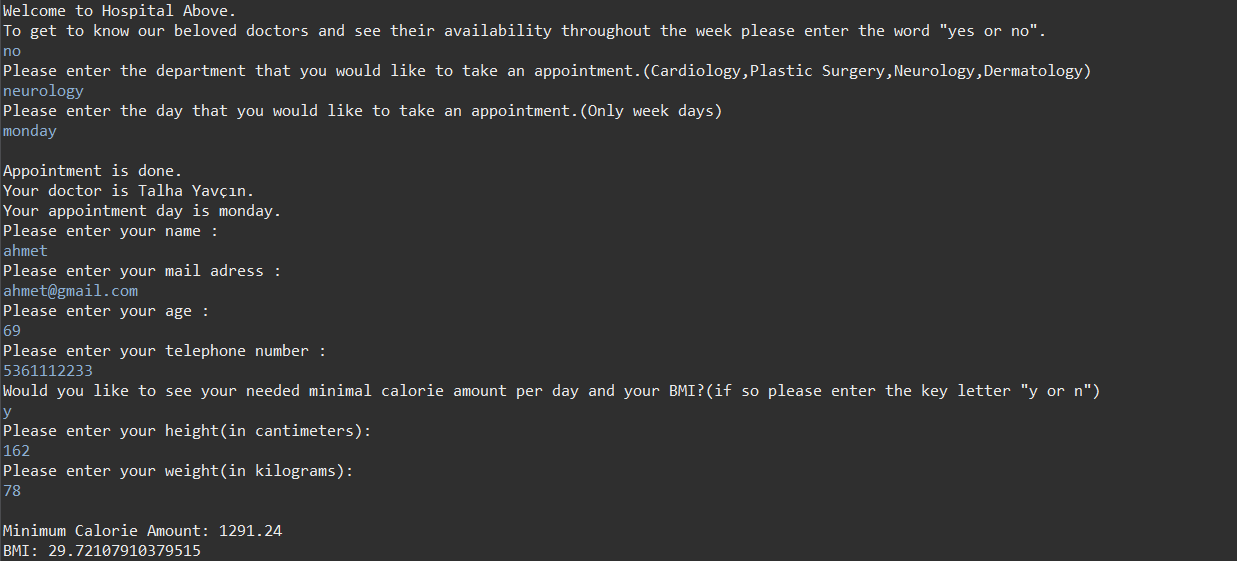
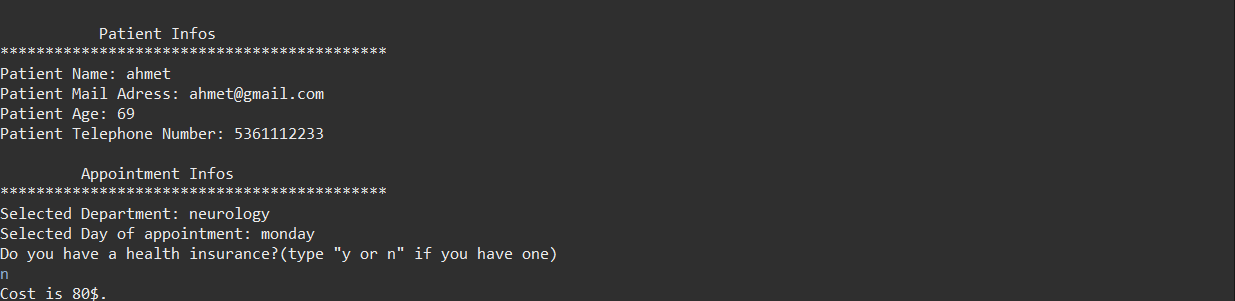




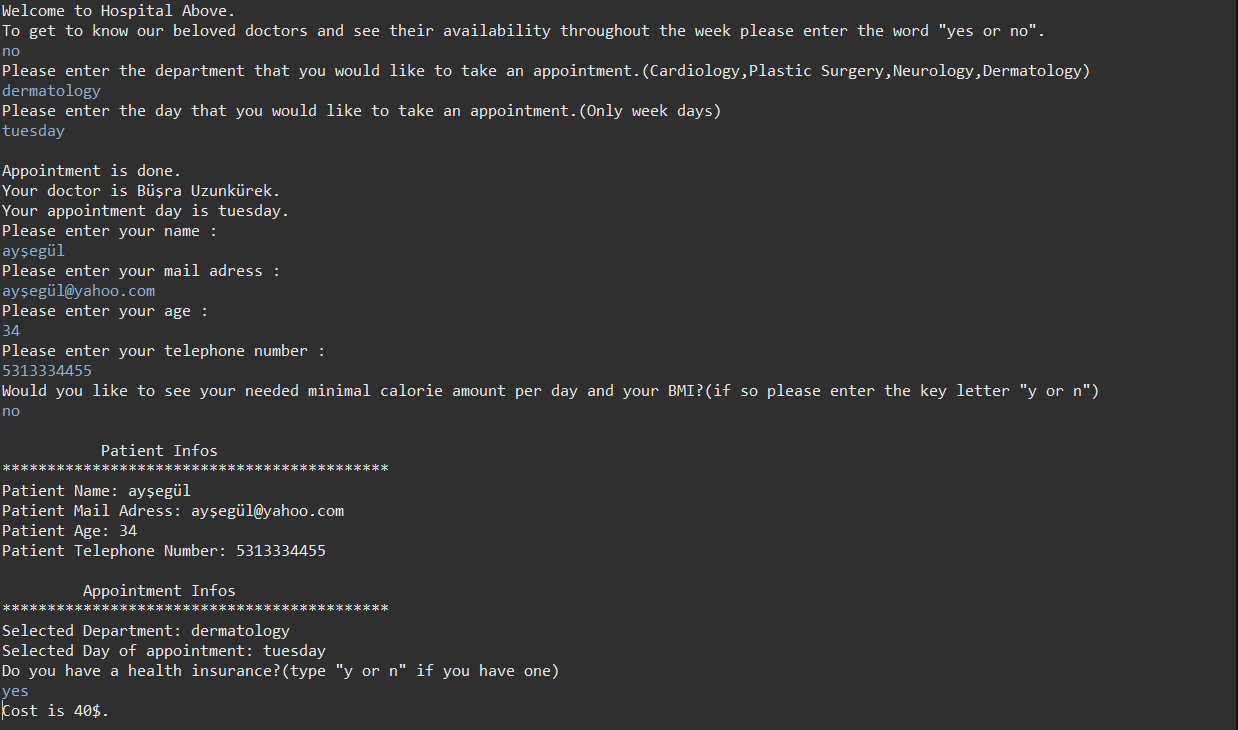
*Output 5:*



*Output 6:*



*Output 7:*



*Output 8:*

