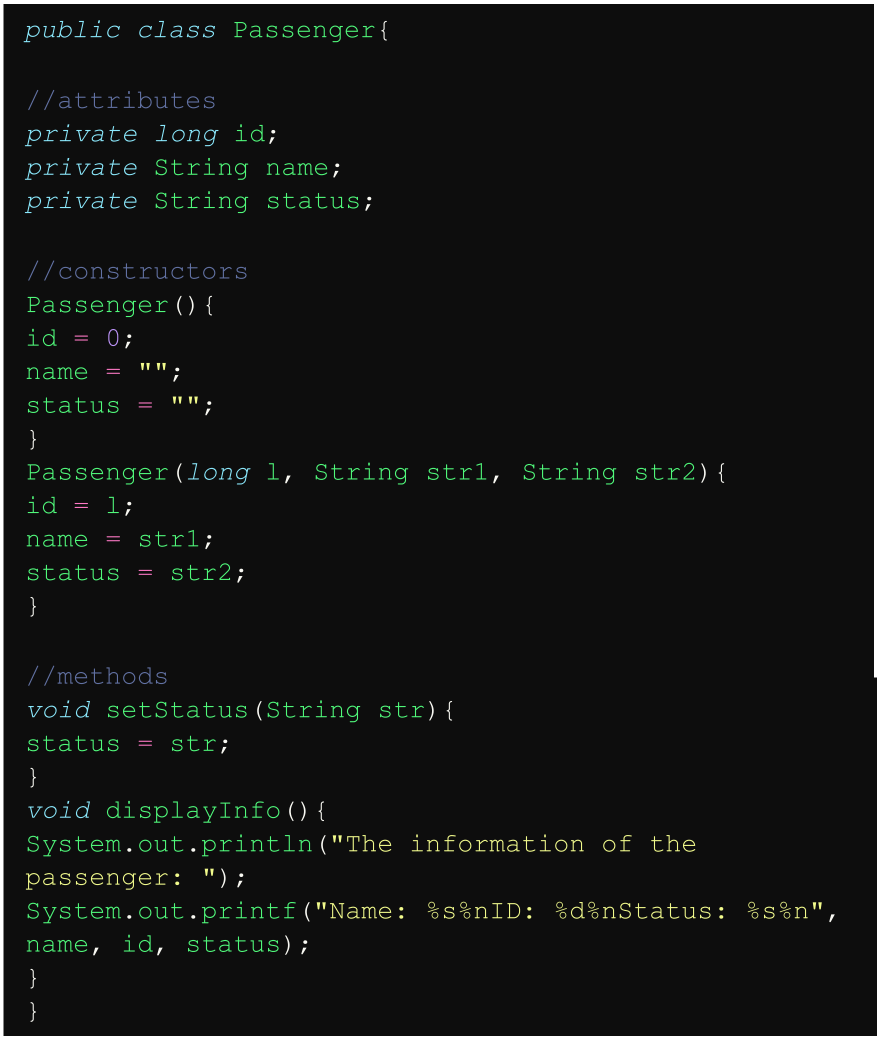
King Saud University College of Computer and Information Sciences Computer Science Department	
CSC 111 Introduction to Programming with Java	First Semester 1440-1441

Tutorial 12

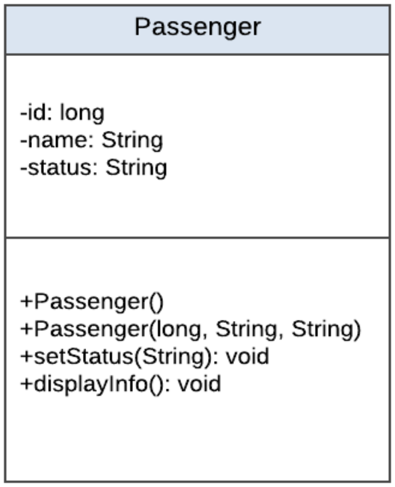
Q1 . Draw the UML and implement the corresponding class (Passenger).

Passenger has an ID, name and status. the class has a default constructor and a parameterized constructor . The two basic operations are: update status and display information of a passenger.

- in The main method , create a passenger with the following information : ID = 123 , name = "Ahmad" and status = "not confirmed", then call setStatus to set the status to "confirmed", then call DisplayInfo to print all the passengers information.



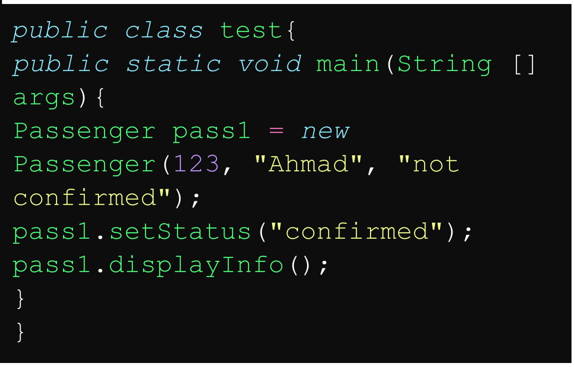
```
public class Passenger{

//attributes
private long id;
private String name;
private String status;

//constructors
Passenger(){
id = 0;
name = "";
status = "";
}
Passenger(long l, String str1, String str2){
id = l;
name = str1;
status = str2;
}

//methods
void setStatus(String str){
status = str;
}
void displayInfo(){
System.out.println("The information of the
passenger: ");
System.out.printf("Name: %s%nID: %d%nStatus: %s%n",
name, id, status);
}
}
```



Passenger

-id: long
-name: String
-status: String

+Passenger()
+Passenger(long, String, String)
+setStatus(String): void
+displayInfo(): void



```
public class test{
public static void main(String []
args){
Passenger pass1 = new
Passenger(123, "Ahmad", "not
confirmed");
pass1.setStatus("confirmed");
pass1.displayInfo();
}
}
```

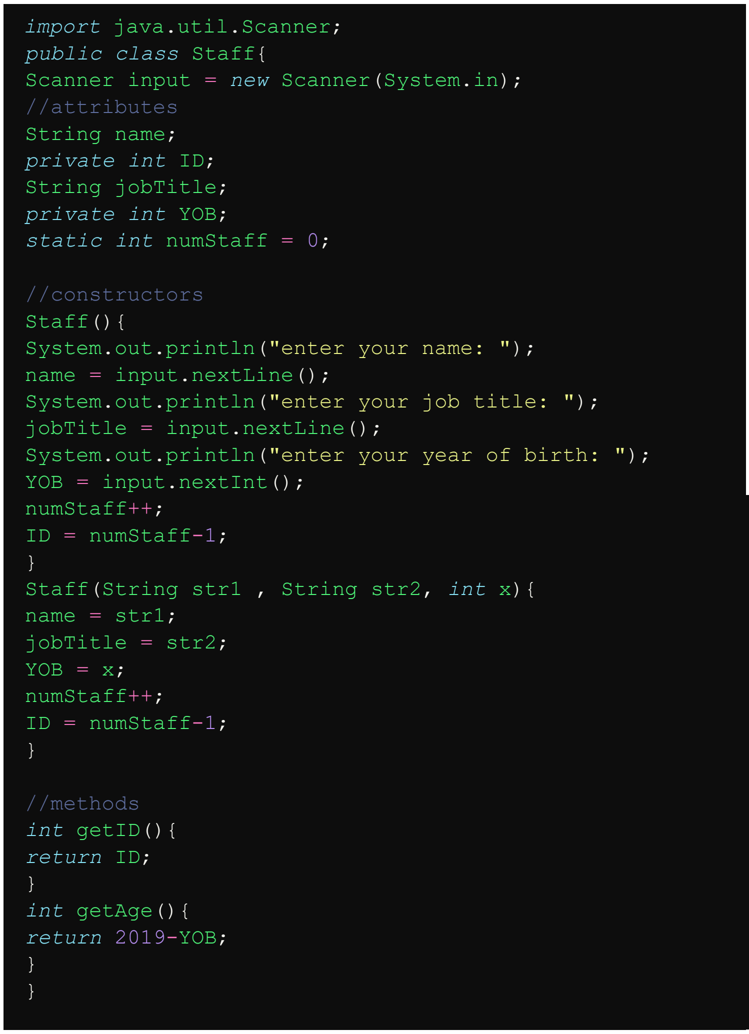
Q2: Given the following UML, implement the corresponding class. Note that ID of each staff object is generated by the system sequentially. For example: the id of the first object is zero, the ID of the second one is 1 ,… the ID of the 10th object is 9…

The method Staff() reads and fills the attributes of the current object from the user. The method Staff(String,String,int)fills the attributes of the current object from the received parameters. The method getAge calculates the staff age and returns it. The attribute numStaff stores the total number of Staff objects.

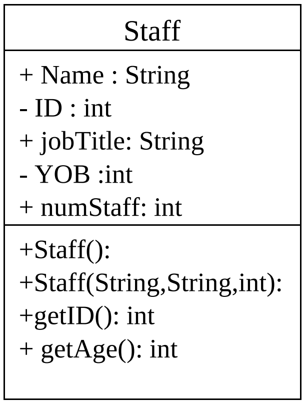
Implement the Company Class which includes the main function.

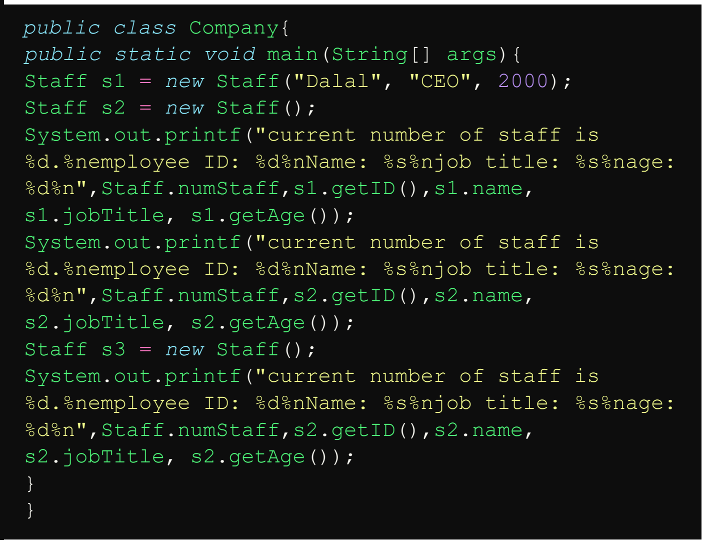
In the main function do the following in the same order:

- Create the first object using Staff(String,String,int).
- Create the second object using Staff().
- Print numStaff, and first object info:id, name, jobTitle and age.
- Print numStaff, and second object info:id, name, jobTitle and age.
- Create the third object using Staff().
- Print numStaff, and second object info:id, name, jobTitle and age.



```java
import java.util.Scanner;
public class Staff{
Scanner input = new Scanner(System.in);
//attributes
String name;
private int ID;
String jobTitle;
private int YOB;
static int numStaff = 0;

//constructors
Staff(){
System.out.println("enter your name: ");
name = input.nextLine();
System.out.println("enter your job title: ");
jobTitle = input.nextLine();
System.out.println("enter your year of birth: ");
YOB = input.nextInt();
numStaff++;
ID = numStaff-1;
}
Staff(String str1 , String str2, int x){
name = str1;
jobTitle = str2;
YOB = x;
numStaff++;
ID = numStaff-1;
}

//methods
int getID(){
return ID;
}
int getAge(){
return 2019-YOB;
}
}
```





```java
public class Company{
public static void main(String[] args){
Staff s1 = new Staff("Dalal", "CEO", 2000);
Staff s2 = new Staff();
System.out.printf("current number of staff is
%d.%nemployee ID: %d%nName: %s%njob title: %s%nage:
%d%n",Staff.numStaff,s1.getID(),s1.name,
s1.jobTitle, s1.getAge());
System.out.printf("current number of staff is
%d.%nemployee ID: %d%nName: %s%njob title: %s%nage:
%d%n",Staff.numStaff,s2.getID(),s2.name,
s2.jobTitle, s2.getAge());
Staff s3 = new Staff();
System.out.printf("current number of staff is
%d.%nemployee ID: %d%nName: %s%njob title: %s%nage:
%d%n",Staff.numStaff,s2.getID(),s2.name,
s2.jobTitle, s2.getAge());
}
}
```

Q3) Find the output of the following program:

```
class Car {
private String model;
private String color;
private double price;
public Car (String m, String c, double p) {
model = m;
color = c;
price = p;
}
public void setModel(String m){ model = m; }
public double getPrice(){ return price; }
public void display(){
System.out.println("The car model is:
"+model+ " it costs "+price+ " SR");}
}


public class CarGallery{
public static void main(String[] args) {
Car[] cars= new Car[100];
cars[0] = new Car ("Audi","black",70000);
cars[1] = new Car ("Toyota","Blue",10000);
cars[2] = new Car ("BMW","white",90000);
cars[3] = new Car ("Honda","Red",20000);
cars[1].setModel("Jeep");
System.out.println("Expensive cars:");
for (int i=0; i<=3; i++)
 if(cars[i].getPrice ()>30000)
cars[i].display();    }
}
```

Output:

```
Expensive cars:
The car model is: Audi it costs 70000.0 SR
The car model is: BMW it costs 90000.0 SR
```