OBJECT ORIENTED PROGRAMMING WITH JAVA 8– LAB 4

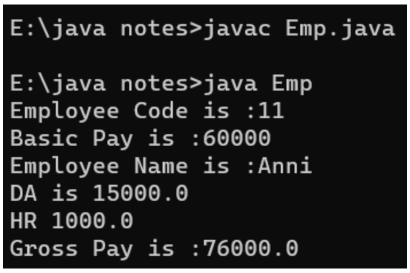
Q1. Build a class Emp which contain the following details about the employee and make an object in the main function and call the functions to set values and print values of the object.

a. Data Members
i. int empCode
ii. String name
iii. int basicPay
iv. float DA
v. float HRA
vi. float grossPay

Program:

```
 class Emp
   {
         int empCode, basicPay;
         String name;
         float DA, HRA, grossPay;
  void setempCode(int empCode){
         this.empCode = empCode;
         }
  void setbasicPay(int basicPay) {
         this.basicPay = basicPay;
         }
  void setname(String name) {
         this.name = name;
         }
  void setDA(float DA) {
         this.DA = DA;
         }
  void setHRA(float HRA) {
         this.HRA = HRA;
         }
  void grossPay() {
         grossPay = basicPay + DA + HRA;
         }
  void printValues() {
         System.out.println("Employee Code is :" + empCode);
         System.out.println("Basic Pay is :" + basicPay);
         System.out.println("Employee Name is :" + name);
         System.out.println("DA is " + DA);
         System.out.println("HR " + HRA );
         System.out.println("Gross Pay is :" + grossPay);
     }
   public static void main(String args[]){
         Emp e = new Emp();
         e.setempCode(11);
         e.setbasicPay (60000);
         e.setname("Anni");
         e.setDA(15000);
         e.setHRA(1000);
         e.grossPay();
         e.printValues();    }}
```

Output:



```
E:\java notes>javac Emp.java

E:\java notes>java Emp
Employee Code is :11
Basic Pay is :60000
Employee Name is :Anni
DA is 15000.0
HR 1000.0
Gross Pay is :76000.0
```

b. Function members

i. to accept and set values for empCode, name, and basicPay

ii. to calculate grossPay

iii. to print values of the employee object

If basicPay < 3500 then DA is 45% of basicPay and HRA is 20% of basicPay.

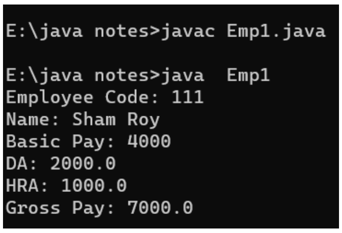
If basicPay >=3500, then DA = 50% of basicPay and HRA is Rs 1000.

Find and print the grossPay = basicPay +DA+HRA.

Program:

```
class Emp1
  {
               int empCode,basicPay;
               String name;
               float DA, HRA, grossPay;
    void setEmpDetails(int empCode, String name, int basicPay) {
        this.empCode = empCode;
        this.name = name;
        this.basicPay = basicPay;
     }
    void grossPay() {
        if (basicPay < 3500) {
           DA = 0.45f * basicPay;
           HRA = 0.20f * basicPay;
        } else {
           DA = 0.50f * basicPay;
           HRA = 1000;
        }
        grossPay = basicPay + DA + HRA;
     }
     void printDetails() {
        System.out.println("Employee Code: " + empCode);
        System.out.println("Name: " + name);
        System.out.println("Basic Pay: " + basicPay);
        System.out.println("DA: " + DA);
        System.out.println("HRA: " + HRA);
        System.out.println("Gross Pay: " + grossPay);
     }
     public static void main(String[] args) {
        Emp1 emp = new Emp1();
        emp.setEmpDetails(111, "Sham Roy", 4000);
        emp.grossPay();
        emp.printDetails();
  } }
```

Output :



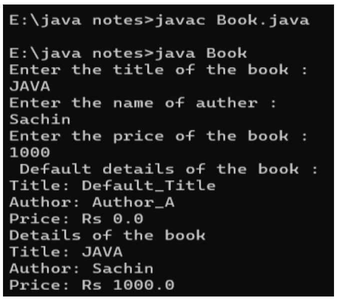
```
E:\java notes>javac Emp1.java

E:\java notes>java  Emp1
Employee Code: 111
Name: Sham Roy
Basic Pay: 4000
DA: 2000.0
HRA: 1000.0
Gross Pay: 7000.0
```

Q 2 . Create a class named Book with data members title, author and price. Write a no argument constructor which initializes the instance variables with some default values and a parameterized constructor also to initialize variables with user input. Write a function for displaying the details. Create two objects for the class using two constructors and invoke the display function.

Program :

```
        import java.util.Scanner;
             class Book
             {
                       String title,author;
                  double price;
               Book() {
                  title = "Default_Title";
                  author = "Author_A";
                 price = 0.0;
                 }
               Book(String title, String author, double price) {
                  this.title = title;
                  this.author = author;
                  this.price = price;
                 }
               void display() {
                  System.out.println("Title: " + title);
                  System.out.println("Author: " + author);
                  System.out.println("Price: Rs " + price);
                 }
               public static void main(String[] args) {

                 Book book = new Book();

                 Scanner s = new Scanner(System.in);
                 System.out.println("Enter the title of the book :");
                 String title = s.nextLine();
                 System.out.println("Enter the name of auther :");
                 String auther = s.nextLine();
                 System.out.println("Enter the price of the book :");
                 double price = s.nextLine();

                 Book b = new Book(title,auther,price);

                 System.out.println(" Default details of the book :");
                 book.display();

                 System.out.println("Details of the book");
                 b.display();
                }}
```

Output:



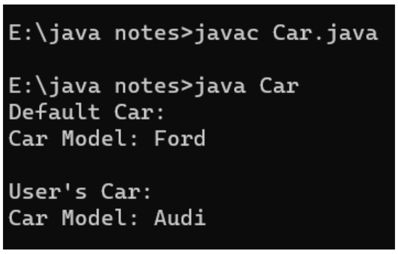
```
E:\java notes>javac Book.java

E:\java notes>java Book
Enter the title of the book :
JAVA
Enter the name of auther :
Sachin
Enter the price of the book :
1000
 Default details of the book :
Title: Default_Title
Author: Author_A
Price: Rs 0.0
Details of the book
Title: JAVA
Author: Sachin
Price: Rs 1000.0
```

Q3 Create a class named Car with a default constructor which initializes the instance variable model with the value “Ford”. Write a parametrized constructor also to initialize model. Write a getModel() method to print the value of model. Create two objects for the class using two constructors and invoke the getModel()method.

Program :

```
import java.util.Scanner;
class Car
{
   String model;
   Car() {              // Default constructor with a default model value
      model = "Ford";
   }
   Car(String model) {    // Parameterized constructor to set the model
      this.model = model;
   }
   void getModel() {     // Method to get and print the model
      System.out.println("Car Model: " + model);
   }

   public static void main(String[] args) {
      Car defaultCar = new Car();      // Create object using default constructor

      Car userCar = new Car("Audi");  // Create object using parameterized constructor

      System.out.println("Default Car:");  // Print the model for both cars
      defaultCar.getModel();

      System.out.println("\nUser's Car:");
      userCar.getModel();
   }
}
```

Output:



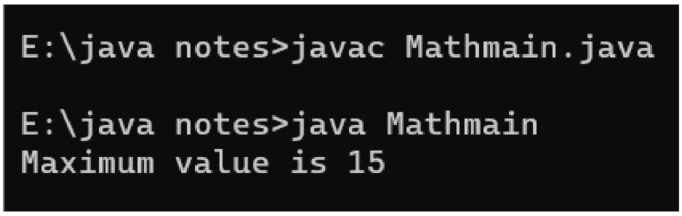
```
E:\java notes>javac Car.java

E:\java notes>java Car
Default Car:
Car Model: Ford

User's Car:
Car Model: Audi
```

Q4. Write a class with a static method that returns the maximum value of three given integers. Write another class with main method and call the static method to print the maximum value for a set of integer values.

Program:

```
class MathUtil
{
  static int maxvalue(int a, int b, int c) {     // to find and return max value
  return Math.max(Math.max(a,b) ,c);
  }
}
class Mathmain {           // main method
  public static void main(String[] args) {
    int n1 = 5;
    int n2 = 10;
    int n3 = 15;
    int max = MathUtil.maxvalue(n1,n2,n3);   // calling the maxvalue by using class
name(Math)
  System.out.println("Maximum value is " + max);
    }
}
```

Output :



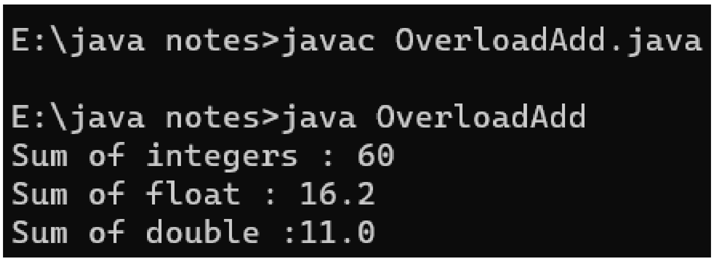
```
E:\java notes>javac Mathmain.java

E:\java notes>java Mathmain
Maximum value is 15
```

Q5.

Program :

```
class OverloadAdd{
int add(int a, int b){
  return a + b;
 }
float add (float a, float b){
  return a + b;
 }
double add(double a, double b) {
  return a + b;
 }
public static void main(String args[]) {
  OverloadAdd add = new OverloadAdd();
  int intSum = add.add(10 ,50);
  float floatSum = add.add(5.6f ,10.6f);
  double doubleSum = add.add(6.5 , 4.5);

  System.out.println("Sum of integers : "  + intSum);
  System.out.println("Sum of float : " + floatSum);
  System.out.println("Sum of double :" +doubleSum);
 }
}
```

Output:



```
E:\java notes>javac OverloadAdd.java

E:\java notes>java OverloadAdd
Sum of integers : 60
Sum of float : 16.2
Sum of double :11.0
```