8186_Rohit Gupta_Introduction to Java - 2

Q.1 Write a program to display values of enums using a constructor & getPrice() method (Example display house & their prices).

```
enum house {  2 usages
    house1( price: 11234),  no usages
    house2( price: 534433),  no usages
    house3( price: 353335),  no usages
    house4( price: 583855);  no usages
 final int price;  2 usages
    house(int price) {  8 usages
        this.price = price;
    }
    public int getprice() {  1 usage
        return price;
    }
}

// Main class to display house names and their prices
public class hello {
    public static void main(String[] args) {
        System.out.println("house types and their prices:");
        for (house housee : house.values()) {
            System.out.println(housee + " costs Rs. " + housee.getprice());
        }
    }
}
```

```
/usr/lib/jvm/java-1.8.0-openjdk-amd64/bi
House Types and Their Prices:
house1 costs Rs. 11234
house2 costs Rs. 534433
house3 costs Rs. 353335
house4 costs Rs. 583855

Process finished with exit code 0
```

Q.2 Create a User class with fields: firstname, lastname, age, phonenumber. Write a program which accepts values of user fields from commandline, create object and append that to a text file. After every user creation the program should prompt: "Do you want to continue creating users? (Type QUIT to exit)" and keep on accepting values and writing to file unitl user quits.

```
import java.util.*;
import java.io.*;


class User {
   private String firstName;
   private String lastName;
   private int age;
   private String phoneNumber;

   public User(String firstName, String lastName, int age, String phoneNumber)
{
       this.firstName = firstName;
       this.lastName = lastName;
       this.age = age;
       this.phoneNumber = phoneNumber;
   }

   @Override
   public String toString() {
       return firstName + "," + lastName + "," + age + "," + phoneNumber;
   }
}

public class hello {
   public static void main(String[] args) {
       Scanner scanner = new Scanner(System.in);

       while (true) {
           System.out.print("Enter First Name: ");
           String firstName = scanner.nextLine();

           System.out.print("Enter Last Name: ");
           String lastName = scanner.nextLine();

           System.out.print("Enter Age: ");
           int age = Integer.parseInt(scanner.nextLine());

           System.out.print("Enter Phone Number: ");
           String phoneNumber = scanner.nextLine();

           User user = new User(firstName, lastName, age, phoneNumber);
```

```
            try (FileWriter fw = new FileWriter("example.txt", true)) {
                fw.write(user.toString() + "\n");
            } catch (IOException e) {
                System.out.println("An error occurred while writing to the
file.");

            }

            System.out.print("Do you want to continue creating users? (Type QUIT
to exit): ");
            String response = scanner.nextLine();
            if (response.equalsIgnoreCase("QUIT")) {
                break;
            }
        }
        scanner.close();
    }
}
```

```
/usr/lib/jvm/java-1.8.0-openjdk-amd64/bin/java ...
Enter First Name: rohit
Enter Last Name: gypta
Enter Age: 21
Enter Phone Number: 86788784844
Do you want to continue creating users? (Type QUIT to exit):
Enter First Name: kunal
Enter Last Name: rawat
Enter Age: 22
Enter Phone Number: 537959375533
Do you want to continue creating users? (Type QUIT to exit): QUIT
```

```
rohit@TTNPL-rohitgupta:~/IdeaProjects/bootcampp$ cat example.txt
rohit,gypta,21,86788784844
kunal,rawat,22,537959375533
```

Q.3 Write a program to count number of occurrences of a word in a file. The file name and word should be supplied through commandline.

```java
import java.io.*;
import java.util.Scanner;
public class hello {
    public static void main(String[] args) {
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter filename: ");
        String fileName=sc.nextLine();
        System.out.println("Enter wordtofind: ");
        String wordToFind =sc.nextLine();
        int wordCount = 0;
        try (BufferedReader br = new BufferedReader(new FileReader(fileName))) {
            String line;
            while ((line = br.readLine()) != null) {
                String[] words = line.split( s: "\\s+");
                for (String word : words) {
                    if (word.equals(wordToFind)) {
                        wordCount++;
                    }
                }
            }
            System.out.println("The word '" + wordToFind + "' occurred " + wordCount + " times.");
        } catch (IOException e) {
            System.out.println("An error occurred while reading the file: " + e.getMessage());
        }
    }
}
```

```
rohit@TTNPL-rohitgupta:~/IdeaProjects/bootcampp$ cat example.txt
rohit
kunal
vijay
vikas
rohit
rohit
```

```
Enter filename:
example.txt
Enter wordtofind:
rohit
The word 'rohit' occurred 3 times.

Process finished with exit code 0
```

Q.4 Write a program to show application of Factory Design Pattern.

```java
interface employee {
   void hi();
}
```

```java
class developer implements employee {
   @Override
   public void hi() {
       System.out.println("hi,I am developer");
   }
}

class manager implements employee {
   @Override
   public void hi() {
       System.out.println("hi,I am manager");
   }
}
class tester implements employee {
   @Override
   public void hi() {
       System.out.println("hi,I am tester");
   }
}
class employeefactory {
   public static employee getemployee(String emp) {
       if (emp == null) {
           return null;
       }
       if (emp.equals("developer")) {
           return new developer();
       } else if (emp.equals("manager")) {
           return new manager();
       } else if (emp.equals("tester")) {
           return new tester();
       }
       return null;
   }
}
public class rohit {
   public static void main(String[] args) {
       employee emp1 = employeefactory.getemployee("manager");
       emp1.hi();  // Output: Driving a Car


   }
}
```

```
hi,I am manager

Process finished with exit code 0
```

Q.5 Write a program to show application of Singleton Design Pattern.

```java
// Singleton Class
class hello{  7 usages
    private static hello instance=null;  3 usages
    private hello() {  1 usage
    }
    public static hello getInstance() {  2 usages
        if (instance == null) {
            instance = new hello();
        }
        return instance;
    }
}
public class rohit{
    public static void main(String[] args) {
        hello hello1 = hello.getInstance();
        hello hello2 = hello.getInstance();
        System.out.println("Are both instances same? " + (hello1==hello2));  // Should print 'true'
    }
}
```

```
Are both instances same? true

Process finished with exit code 0
```