

Java Lab FAT exam

Arshdeep Singh

19BCB0086

- a) Write a java program to implement the same. Create a class called king and queen that stores 8 objects of the Card class. Include methods to shuffle the deck, deal a card of your choice, and report the number of cards left in the deck. The shuffle method should assume only king and queen. Create a driver class with a main method that deals each card from a shuffled deck, printing each card that is left.

Code

```
class card{  
  
    String name;  
  
    card(String s){  
        this.name=s;  
    }  
}
```

```
class KingandQueen {  
    card c[]=new card[8];  
    int count=8;  
    KingandQueen(){  
        this.c[0]=new card("king of spades");  
        this.c[1]=new card("king of heart");  
        this.c[2]=new card("king of diamond");  
        this.c[3]=new card("king of clubs");  
        this.c[4]=new card("queen of spades");  
        this.c[5]= new card("queen of heart");  
        this.c[6]= new card("queen of diamond");  
        this.c[7]= new card("queen of clubs");  
    }  
}
```

```

public void shuffle(){
    for(int k=0;k<8;k++){
        int i= (int)(8 * Math.random());
        int j= (int)(8 * Math.random());
        card temp=c[i];
        c[i]=c[j];
        c[j]=temp;
    }
    System.out.println("Cards have been shuffled");
}

public void deal(){
    System.out.println("Dealing the card");
    System.out.println(c[0].name);
    card temp=c[0];
    c[0]=c[7];
    c[7]=temp;
    count--;
}

public void nocards(){
    System.out.println("Number of cards "+count);
}

}

public class fat {
    public static void main(String args[]){
        KingandQueen D = new KingandQueen();
        D.nocards();
        D.deal();
        D.shuffle();
        D.deal();
        D.nocards();
    }
}

```

Output

- Sample output
- Initially 8 cards
- Deals 2 random cards
- Shuffles the cards
- Number of cards after dealing

```
Output - JavaFXApplication1 (run-single) X
init:
Deleting: C:\Users\Arshdeep Singh\Documents\NetBeansProjects\JavaFXApplication1\build\build-jar.properties
deps-jar:
Updating property file: C:\Users\Arshdeep Singh\Documents\NetBeansProjects\JavaFXApplication1\build\build-jar.properties
Compiling 1 source file to C:\Users\Arshdeep Singh\Documents\NetBeansProjects\JavaFXApplication1\build\classes
compile-single:
run-single:
Number of cards 8
Dealing the card
king of spades
Cards have been shuffled
Dealing the card
queen of diamond
Number of cards 6
BUILD SUCCESSFUL (total time: 0 seconds)
```

Since shuffle function is random the output changes everytime we run.(Above queen of diamonds, below king of clubs)

```
Output - JavaFXApplication1 (run-single) X
init:
Deleting: C:\Users\Arshdeep Singh\Documents\NetBeansProjects\JavaFXApplication1\build\build-jar.properties
deps-jar:
Updating property file: C:\Users\Arshdeep Singh\Documents\NetBeansProjects\JavaFXApplication1\build\build-jar.properties
Compiling 1 source file to C:\Users\Arshdeep Singh\Documents\NetBeansProjects\JavaFXApplication1\build\classes
compile-single:
run-single:
Number of cards 8
Dealing the card
king of spades
Cards have been shuffled
Dealing the card
king of clubs
Number of cards 6
BUILD SUCCESSFUL (total time: 0 seconds)
```

```
Output - JavaFXApplication1 (run-single) X
init:
Deleting: C:\Users\Arshdeep Singh\Documents\NetBeansProjects\JavaFXApplication1\build\build-jar.properties
deps-jar:
Updating property file: C:\Users\Arshdeep Singh\Documents\NetBeansProjects\JavaFXApplication1\build\build-jar.properties
Compiling 1 source file to C:\Users\Arshdeep Singh\Documents\NetBeansProjects\JavaFXApplication1\build\classes
compile-single:
run-single:
Number of cards 8
Cards have been shuffled
Dealing the card
king of diamond
Cards have been shuffled
Dealing the card
king of clubs
Number of cards 6
BUILD SUCCESSFUL (total time: 0 seconds)
```

- b) Write a program to estimate the cost of an upcoming vacation. There are three types of expenses: food, boarding, and entertainments. Food, boarding, and entertainments are based on cost per day and estimated days for each one of them. Test the program with main class by giving you own values and print the output.

Code

```
import java.util.Scanner;

public class labfat2 {

    public static void main(String args[]) {
        int food_ex;//daily cost
        int board_ex;
        int ent_ex;
        int number_days;
        System.out.println("This is vacation expense calculator\n enter the cost of food per meal");
        Scanner sc = new Scanner(System.in);
        food_ex = sc.nextInt();
        System.out.println("Enter the daily cost of boarding");
        board_ex = sc.nextInt();
        System.out.println("Enter the daily cost of entertainment");
        ent_ex = sc.nextInt();
        System.out.println("Enter the number of days u will stay");
        number_days = sc.nextInt();
        System.out.println("The total estimated cost is ");
        int ans = (3 * food_ex + board_ex + ent_ex) * number_days;
        System.out.println("Ruppees: " + ans);

    }
}
```

Output

```
init:
Deleting: C:\Users\Arshdeep Singh\Documents\NetBeansProjects\JavaFX
deps-jar:
Updating property file: C:\Users\Arshdeep Singh\Documents\NetBeans
Compiling 1 source file to C:\Users\Arshdeep Singh\Documents\NetBe
compile-single:
run-single:
This is vacation expense calculator
  enter the cost of food per meal
300
Enter the daily cost of boarding
5000
Enter the daily cost of entertainment
700
Enter the number of days u will stay
5
The total estimated cost is
Ruppees: 33000
BUILD SUCCESSFUL (total time: 27 seconds)
```

Test run 2

```
Input - JavaFXApplication1 (run-single) #2
init:
Deleting: C:\Users\Arshdeep Singh\Documents\NetBeansProjects\JavaFXApplication1\build\build-jar
deps-jar:
Updating property file: C:\Users\Arshdeep Singh\Documents\NetBeansProjects\JavaFXApplication1\b
Compiling 1 source file to C:\Users\Arshdeep Singh\Documents\NetBeansProjects\JavaFXApplication.
compile-single:
run-single:
This is vacation expense calculator
  enter the cost of food per meal
345
Enter the daily cost of boarding
3000
Enter the daily cost of entertainment
500
Enter the number of days u will stay
4
The total estimated cost is
Ruppees: 18140
BUILD SUCCESSFUL (total time: 18 seconds)
```