Assignment -4

Abhishek Shrivastava

- **Write a program in java to count the number of objects created.**

```
package countobjects;


public class CountObjects {

   static int count=0;
   CountObjects()
         {
            count++;
         }

   public static void main(String[] args) {
      CountObjects obj = new CountObjects();
      CountObjects obj1 = new CountObjects();
      CountObjects obj2 = new CountObjects();
      System.out.println("The number of objects created are: "+ count);

   }

}
```

- **Which of the following is true?**

a) The name of the constructor and name of the class need not be same. FALSE
b) We can declare return type for the constructor but it should be void. FALSE
c) We can use any modifier for the constructor. FALSE
d) Compiler will always generate default constructor. FALSE
e) The modifier of the default constructor is always default. FALSE
f) The 1st line inside every constructor should be super always. FALSE
g) The 1st line inside every constructor should be either super or this and if we are not writing anything compiler will always place this(). FALSE
h) Overloading concept is not applicable for constructor. FLASE
i) Inheritance and overriding concepts are applicable for constructors. FLASE
j) Concrete class can contain constructor but abstract class cannot. FALSE
k) Interface can contain constructor. FALSE
l) Recursive constructor call is always runtime exception. FALSE, it’s compilation error.
m) If Parent class constructor throws some un-checked exception compulsory Child class constructor should throw the same un-checked exception or it's Parent. TRUE

Assignment -4

Abhishek Shrivastava

- **Write a program java to create overloaded methods which can take single arguments and variable arguments.**

```
package methodover;
import java.util.Scanner;
public class MethodOver {

  private void meth(float a)
 {
   System.out.println("The first number is: "+a);
  }
  private void meth(float a, float b)
  {
     System.out.println("Sum of a and b: "+(a+b));
  }

  private void meth(float a, float b, char c)
  {
     System.out.println("Difference of a and b: "+(a-b));
  }

  private void meth(char c, float a, float b)
  {
     System.out.println("Multiplication of a and b: "+(a*b));
 }
  private void meth(float a, char c, float b)
  {
     System.out.println("Multiplication of a and b: "+(a/b));
  }

  public static void main(String[] args) {
     MethodOver obj = new MethodOver();
     System.out.println("Enter values of a and b:");
     float a,b;
     Scanner sn = new Scanner(System.in);
     a=sn.nextFloat();
     b=sn.nextFloat();
     obj.meth(a);
     obj.meth(a,b);
     obj.meth(a, b,'c');
     obj.meth('c',a, b);
     obj.meth(a, b,'c');

  } }
```

Assignment -4

Abhishek Shrivastava

❖ **Can we overload main() method? If yes, write a program to support the same.**

Yes, we can overload main() method in our program but when we run the program, the main method with only the String[] args array as argument will be called automatically. We can call the other versions of main() method inside our public static void main(String[] args) method. Also, it is mandatory to have public static void main(String[] args) version of main in our program. Otherwise, our program won't run and we will get an error message in our program stating, "no main classes found".

PROGRAM

```
package mainoverloading;


public class MainOverloading
 {
   public static void main(String[] args)
     {
      System.out.println("I am main with only String array as argument");
      main("abc",2);
      main(2);
     }
   public static void main(String args,int m)
      {
   System.out.println("I am main with String array and an integer variable as argument");
      }
   public static void main(int m)
     {
      System.out.println("I am main with only an integer variable as argument");
     }
}
```

OUTPUT OF OUR PROGRAM

```
run:
I am main with only String array as argument
I am main with String array and an integer variable as argument
I am main with only an integer variable as argument
BUILD SUCCESSFUL (total time: 0 seconds)
```

Assignment -4

Abhishek Shrivastava

- **Write a program in java to make use of overloaded constructors.**

```
package constover;
public class ConstOver {
   ConstOver()
   {
      // Default Constructor
      System.out.println("I am default constructor.");
   }

   ConstOver(int a)
   {
      // Parameeterized Constructor
      System.out.println("I am parameterized constructor."+ "The parameter received is
2. "+a);
   }

   public static void main(String[] args) {

      ConstOver obj=new ConstOver();
      ConstOver obj1=new ConstOver(2);
   }
```