Swapnanil Dutta Roll: 12

Question: Create a class First, make instance variable [int x], method [void show()] and also put main method inside that class and use the instance variable and method from main.

Code:

```
public class First{
    int x;
    First(int x) {
        this.x = x;
    }
    void show(){
        System.out.println("Printing");
    }
    public static void main(String[] args) {
        First obj = new First(12);
        obj.show();
        System.out.println(obj.x);
    }
}
```

Output:

```
PS D:\OOPS-PCC-CS593\Day 6 (24.09.2020)> javac First.java
PS D:\OOPS-PCC-CS593\Day 6 (24.09.2020)> java First
Printing
12
```

Question: Create a class; make its instance variable and method. Use them from main, declared in different class.

Code:

```
class First{
    int x;
    First(int x) {
        this.x = x;
    }
    void show(){
        System.out.println("Printing");
    }
}
public class DifferentClass{
    public static void main(String[] args) {
        First obj = new First(12);
        obj.show();
        System.out.println(obj.x);
    }
}
```

Output:

```
PS D:\OOPS-PCC-CS593\Day 6 (24.09.2020)> javac DifferentClass.java
PS D:\OOPS-PCC-CS593\Day 6 (24.09.2020)> java DifferentClass
Printing
12
```

Question: Create a class named Test, make method inside it and pass object as parameter of this method using (a) pass same class's object, (b) pass different class's object.

Code:

```
class Test{
    int x;
    Test(int x) {
        this.x = x;
    }
    void show(Test obj){
        System.out.println("Printing X: "+ obj.x);
    }
    public static void main(String[] args) {
        Test obj = new Test(12);
        obj.show(obj);
    }
}
public class TestEx{
    public static void main(String[] args) {
        Test obj = new Test(24);
        obj.show(obj);
    }
}
```

Output:

```
PS D:\OOPS-PCC-CS593\Day 6 (24.09.2020)> javac TestEx.java
PS D:\OOPS-PCC-CS593\Day 6 (24.09.2020)> java TestEx
Printing X: 24
```

Question: Create a class; put a method inside this class which will return a class reference return same class and/or different class object.

Code:

```
class Test{
    int x;
    Test(int x) {
        this.x = x;
    }
}
public class TestExample{
    public static Test getObj(int x) {
        return new Test(x);
    }
    public static void main(String[] args) {
        Test obj = getObj(24);
        System.out.println("Value of X: "+obj.x);
    }
}
```

Output:

```
PS D:\OOPS-PCC-CS593\Day 6 (24.09.2020)> javac TestExample.java
PS D:\OOPS-PCC-CS593\Day 6 (24.09.2020)> java TestExample
Value of X: 24
```

Question: Garbage collector.

Code:

```java
public class GarbageCollector {
    protected void finalize() throws Throwable {
        System.out.println("finalize() is called");
    }

    public static void main(String[] args) {
        // object creation
        GarbageCollector obj1 = new GarbageCollector();
        GarbageCollector obj2 = new GarbageCollector();

        // making object ready for garbage collection
        obj1 = null;
        obj2 = null;

        // explicit call of garbage collector
        System.gc();
    }
}
```

Output:

```
PS D:\OOPS-PCC-CS593\Day 6 (24.09.2020)> javac GarbageCollector.java
Note: GarbageCollector.java uses or overrides a deprecated API.
Note: Recompile with -Xlint:deprecation for details.
PS D:\OOPS-PCC-CS593\Day 6 (24.09.2020)> java GarbageCollector
finalize() is called
```

Question: Method returns object.

Code:

```
public class TestMethod
{
    int a;
    TestMethod(int a){
        this.a=a;
    }
    TestMethod returnObj(TestMethod obj){
        TestMethod t= new TestMethod(obj.a+10);
        return t;
    }
    public static void main(String[] args){
        TestMethod t1=new TestMethod(10);
        System.out.println("t1.a= "+t1.a);
        TestMethod t2=t1.returnObj(t1);
        System.out.println("t2.a= "+t2.a);
        TestMethod t3=t2.returnObj(t2);
        System.out.println("t3.a= "+t3.a);
    }
}
```

Output:

```
PS D:\OOPS-PCC-CS593\Day 6 (24.09.2020)> javac TestMethod.java
PS D:\OOPS-PCC-CS593\Day 6 (24.09.2020)> java TestMethod
t1.a= 10
t2.a= 20
t3.a= 30
```

Question: Pass by reference

Code:

```
import java.util.Scanner;
public class Testing
{
    private int a, b;
    public Testing(int a, int b){
        this.a=a;
        this.b=b;
    }
    public static void swap(Testing ob){
        int t=ob.a;
        ob.a=ob.b;
        ob.b=t;
    }
    public static void main(String[] args){
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter two integers");
        int x=sc.nextInt();
        int y=sc.nextInt();
        Testing ob=new Testing(x, y);
        System.out.println("Before Swapping");
        System.out.format("Value of a=%d b=%d\n", ob.a, ob.b);
        swap(ob);
        System.out.println("After Swapping");
        System.out.format("Value of a=%d b=%d", ob.a, ob.b);
    }
}
```

Output:

```
PS D:\OOPS-PCC-CS593\Day 6 (24.09.2020)> javac Testing.java
PS D:\OOPS-PCC-CS593\Day 6 (24.09.2020)> java Testing
Enter two integers
12
14
Before Swapping
Value of a=12 b=14
After Swapping
Value of a=14 b=12
```