Swapnanil Dutta Roll: 12

Assignment 4

Question 2:

Code:

```
class Outer {
    String name = "Miss Dawson Homenick";

    void show() {
        System.out.println("This is the outer class");
        System.out.println("Name : " + name);
    }

    public class Inner {
        String occupation = "Executive";

        void display() {
            System.out.println("This is the inner class");
            show();
            System.out.println("Occupation :" + occupation);
        }
    }

    public static void main(String[] args) {
        new Outer().show();
        new Outer().new Inner().display();
        // new Outer().display(); ! Shows error
    }
}
```

Output:

```
PS D:\OOPS-PCC-CS593\Day-16-(04.11.2020)> javac Outer.java
PS D:\OOPS-PCC-CS593\Day-16-(04.11.2020)> java Outer
This is the outer class
Name : Miss Dawson Homenick
This is the inner class
This is the outer class
Name : Miss Dawson Homenick
Occupation :Executive
```

Question 4:

Code:

```
class AnonymousClass {
    void show() {
        System.out.println("Super class method");
    }
}

interface A1 {
    public AnonymousClass get();
}

class B1 implements A1 {
    public AnonymousClass get() {
        return new AnonymousClass() {
            void show() {
                System.out.println("Overridden method");
            }
        };
    }
}

class Ass4Q4 {
    public static void main(String[] args) {
        new B1().get().show();
    }
}
```

Output:

```
PS D:\OOPS-PCC-CS593\Day-16-(04.11.2020)> javac Ass4Q4.java
PS D:\OOPS-PCC-CS593\Day-16-(04.11.2020)> java Ass4Q4
Overridden method
```

Question 5:

Code:

```
class Block1 {
    public Block1() {
        System.out.println("Block1 Constructor");
    }
    // Block1 Initializer block
    {
        System.out.println("Block1 Initializer Block1");
    }
    {
        System.out.println("Block1 Initializer Block2");
    }
}
class Block2 extends Block1 {
    public Block2() {
        System.out.println("Block2 Constructor");
    }

    {
        System.out.println("Block2 Initializer Block1");
    }
    {
        System.out.println("Block2 Initializer Block2");
    }
}
public class BlockTest1 {
    public static void main(String[] args) {
        new Block2();
    }
}
```

Output:

```
PS D:\OOPS-PCC-CS593\Day-16-(04.11.2020)> javac BlockTest1.java
PS D:\OOPS-PCC-CS593\Day-16-(04.11.2020)> java BlockTest1
Block1 Initializer Block1
Block1 Initializer Block2
Block1 Constructor
Block2 Initializer Block1
Block2 Initializer Block2
Block2 Constructor
```

Question 6:

Code:

```
class A {
    A() {
        System.out.println("Super constructor");
    }{
        System.out.println("Super Initializer block");
    }// Super Initializer block
    static {
        System.out.println("Super static block");
    }
}
// static blocks are called when a class is loaded into jvm
class B extends A {
    B() {
        System.out.println("Sub constructor");
    }
    // Sub Initializer block
    {
        System.out.println("Sub Initializer block");
    }
    static {
        System.out.println("Sub static block");
    }
}
public class BlockTest2 {
    public static void main(String[] args) {
        new B();
    }
}
```

Output:

```
PS D:\OOPS-PCC-CS593\Day-16-(04.11.2020)> javac BlockTest2.java
PS D:\OOPS-PCC-CS593\Day-16-(04.11.2020)> java BlockTest2
Super static block
Sub static block
Super Initializer block
Super constructor
Sub Initializer block
Sub constructor
```

Question 7:

Code:

```
public class BlockTest3 {
    public static void main(String[] args) {
        new Block2();
    }
}

class Block2 {
    static int a;
    int b;

    Block2() {
        System.out.println("Constructor Block");
    }

    static {// only static fields can be accessed inside static blocks
        a = 40;
        System.out.println("a=" + a);
    }
    {
        // both static and non-static fields can be accessed inside
intializer blocks
        a = 30;
        b = 50;
        System.out.println("a=" + a + " b=" + b);
    }
}
```

Output:

```
PS D:\OOPS-PCC-CS593\Day-16-(04.11.2020)> javac BlockTest3.java
PS D:\OOPS-PCC-CS593\Day-16-(04.11.2020)> java BlockTest3
a=40
a=30 b=50
Constructor Block
```

Question 8:
Code:

```
class A1 {
    A1() {
        System.out.println("Super constructor");
    }
    {
        System.out.println("Super Initializer block");
    }
    static {
        System.out.println("Super static block");
    }
}
// static blocks are called when a class is loaded into jvm
class B1 extends A1 {
    B1() {
        System.out.println("Sub constructor");
    }
    {
        System.out.println("Sub Initializer block");
    }
    static {
        System.out.println("Sub static block");
    }
}
public class BlockTest4 {
    public static void main(String[] args) {
        new B1();
    }
}
```

Output:

```
PS D:\OOPS-PCC-CS593\Day-16-(04.11.2020)> javac BlockTest4.java
PS D:\OOPS-PCC-CS593\Day-16-(04.11.2020)> java BlockTest4
Super static block
Sub static block
Super Initializer block
Super constructor
Sub Initializer block
Sub constructor
```