Exercise - 1

1. Expected Behaviour: => To Create the Instance of Example Class
2. Actual Behaviour:=> Unable to Create the Instance - Debug the code

```
package com.training;

public class Example {

	protected int x, y;

		protected Example(int x, int y) {
			super();
			this.x = x;
			this.y = y;
		}

		public int getX() {
			return x;
		}

		public void setX(int x) {
			this.x = x;
		}

		public int getY() {
			return y;
		}

		public void setY(int y) {
			this.y = y;
		}


}
```

```
package example.demo;

import com.training.Example;

public class Application {

	public static void main(String[] args) {

		Example example = new Example(5,6);

		System.out.println(example);
	}

}
```

Exercise - 2

1. Expected Behaviour: => To Print the message Hello World on the console
2. Actual Behaviour:=> Not Printing any Messages

```
package com.training;

public class Example {


   private String args[] ;

        public Example(String[] args) {
                super();
                this.args = args;
        }


public String[] getArgs() {
                return args;
        }


public static void main(String[] args) {

   String values[] = { "1", "2" };

   Example example = new Example(values);

   if (example.getArgs().length < 0)
      System.out.println("Hello World");

}
}
```

Exercise - 3

1. Expected Behaviour: => To Print the message ODD
2. Actual Behaviour:=> Not able to compile the code

```
public class Application {



      public static void main(String[] args) {

         int odd = 1;
   if (odd) {
      System.out.println("odd");
   } else {
      System.out.println("even");
   }
}
```

```
}
```

Exercise - 4

1. Expected Behaviour: => To Print the Number 15
2. Actual Behaviour:=> Not able to compile the code

```
package com.training;

public class Example {


        int add(int i, int j){
                return i+j;
                }

}


package com.training;

public class AnotherExample extends Example {

        public static void main(String[] args) {
                short s = 9;
                System.out.println(add(s,6));

        }
}
```

Exercise - 5

1. Expected Behaviour: => To Print the String Value-A,Value-B and number 10
2. Actual Behaviour:=> Not able to compile the code and hence the expected output is not displayed

```
public class Example {

                int i = 10;

        public void printValue() {
        System.out.println("Value-A");
        }

}


public class AnotherExample extends Example {

         int i = 12;
        public void printValue() {
        System.out.print("Value-B");
        }

}

package example.demo;
```

```
import com.training.AnotherExample;
import com.training.Example;

public class Application {

	public static void main(String[] args) {

		Example example = new AnotherExample();
		example.printValue();
		System.out.println(example.i);

	}

}
```

Exercise - 6

1. Expected Behaviour: => To Print numbers 14 and 21
2. Actual Behaviour:=> Not able to compile the code and hence the expected output is not displayed
    a. Can Provide More than One Solution to fix

```
public class Application {

	public static void main(String[] args) {

		byte b = 6;
		b+=8;
		System.out.println(b);
		b = b+7;
		System.out.println(b);
		}


}
```

Exercise - 7

1. Expected Behaviour: => To Print String Welcome
2. Actual Behaviour:=> Prints the String Hello

```
public class Application {

	public static void main(String[] args) {

		String value = "Hello";
		changeValue(value);
		System.out.println(value);
		}
		public static void changeValue(String value){
		value = "Welcome";
		}


}
```

Exercise - 8

1. Expected Behaviour: => To Print String Name-B Name-B
2. Actual Behaviour:=> Code is Not Compiling

```
public class Example {

	private void printName(){
		System.out.println("Value-A");
		}

}

public class AnotherExample extends Example {

	public void printName(){
		System.out.println("Name-B");
		}

}
public class Application {

	public static void main(String[] args) {

		AnotherExample example1 = new AnotherExample();
		example1.printName();

		Example example2 = new AnotherExample();
		example2.printName();

	}

}
```

Exercise - 9

1. Expected Behaviour: => To Print Number 5 and 8
2. Actual Behaviour:=> Printing A B 8

```
package example.demo;

class Parent {
public Parent(){
System.out.println("A");
}
public Parent(int i){
```

```
System.out.println(i);
}
}
class Child extends Parent{
public Child (){
System.out.println("B");
}
public Child (int i){
        this();
System.out.println(i+3);
}
}

public class Application {

        public static void main(String[] args) {

 new Child(5);

        }

}
```

Exercise - 10

1. Expected Behaviour: => To Print String Father Done
2. Actual Behaviour:=> Code is not compiling

```
package example.demo;
import java.io.*;

class Father {
        public Father() throws IOException {
        System.out.print("Father");
        throw new IOException();
        }
        }

        class Son extends Father {
                public Son()  {
                        System.out.print("Son");
                        }
        }
public class Application {

        public static void main(String[] args) {

                new Son();


        }

}
```