School of Computer Science Engineering and Technology

Course- BTech
Course Code- **CSET109**
Year- First

Type- Core
Course Name- **Object Oriented Programming Using Java**
Semester- Even Batch- BTech 2nd Semester

Tutorial-7

Tutorial No.	Name	CO1	CO2	CO3
7	OOPs concepts	✓	✓	--

Objective: The main objective of this tutorial is to learn about the object-oriented concepts of Java language.

7.1 Analyse the given program segment and predict the output:

```
public class Test
{
	private int data = 5;

	public int getData()
	{
		return this.data;
	}
	public int getData(int value)
	{
		return (data+1);
	}
	public int getData(int... value)
	{
		return (data+2);
	}

	public static void main(String[] args)
	{
		Test temp = new Test();
		System.out.println(temp.getData(7, 8, 12));
	}
}
```

7.2 What will be the output of the following program?

```
public class Base
{
	private int multiplier(int data)
	{
		return data*5;
	}
}
```

```
class Derived extends Base
{
	private static int data;
	public Derived()
	{
		data = 25;
	}
	public static void main(String[] args)
	{
		Base temp = new Derived();
		System.out.println(temp.multiplier(data));
	}
}
```

7.3 Predict the output of the following program:

```
import java.io.IOException;
import java.util.EmptyStackException;

public class newclass
{
	public static void main(String[] args)
	{
		try
		{
			System.out.printf("%d", 1);
			throw(new Exception());
		}
		catch(IOException e)
		{
			System.out.printf("%d", 2);
		}
		catch(EmptyStackException e)
		{
			System.out.printf("%d", 3);
		}
		catch(Exception e)
		{
			System.out.printf("%d", 4);
		}
		finally
		{
			System.out.printf("%d", 5);
		}
	}
}
```

7.4 Predict the output of the following program:

```
package overridingPrograms;
public class One
```

```
{
void m1() throws Throwable
{
 System.out.println("m1-One");
 }
}
public class Two extends One
{
@Override
protected void m1() throws Exception
{
 System.out.println("m1-Two");
 }
}
import java.io.IOException;
public class Three extends Two
{
@Override
public final void m1() throws IOException
{
 System.out.println("m1-Three");
 }
}
public class MyTest
{
public static void main(String[] args) throws Throwable
{
 One o = new Two();
  o.m1();
 Two t = new Three();
  t.m1();
 Three th = new Three();
  th.m1();
 }
}
```

7.5 What will be the output of the following program?

```
class superClass
{
        final public int calc(int a, int b)
        {
                return 0;
        }
}
class subClass extends superClass
{
        public int calc(int a, int b)
        {
                return 1;
```

```
        }
}
public class main
{
        public static void main(String args[])
        {
                subClass get = new subClass();
                System.out.println("x = " + get.calc(0, 1));
        }
}
```

7.6 What will be the output of this program?

```
package overridingPrograms;
public class X
{
void draw(int a, float b) throws Throwable
{
 System.out.println("Circle");
 }
}
public class Y extends X
{
@Override
void draw(int a, float b)
{
 System.out.println("Rectangle");
 }
}
public class Z extends Y
{
@Override
void draw(int a, float b) throws ArithmeticException
{
 System.out.println("Square");
 }
}
public class Test
{
public static void main(String[] args) throws Throwable
{
 X x = new Y();
 x.draw(20, 30.5f);
 Y y = (Y)x;
 y.draw(10,2.9f);
 Z z = (Z)y;
 z.draw(20, 30f);
 }
}
```

7.7 Analyse the given program segment and predict the output:

```
public class P
{
void m1(Number n)
{
 System.out.println("m1-P");
 }
}
public class Q extends P
{
void m1(double d)
{
 System.out.println("m2-Q");
}
public static void main(String[] args)
{
 Q q = new Q();
  q.m1(1);
 q.m1(null);
 }
}
```

7.8 What will be the output of the following program?

```
public class Final
{
   final int assign = 30;

   public static void main(String[] args)
   {
      final int result = 20;
      final int assign;
      Final f = new Final();
      assign = 20;
      System.out.println(assign);
      System.out.println(f.assign);
      System.out.println(f.process(result));
   }
   int process(int a)
   {
      return a + 5;
   }
}
```

7.9 What will be the output of the following program?

```
public class Final
{
   final int assign = 35;
   Final()
```

```
    {
        final int assign = 10;
        print(assign);
    }
    public static void main(String[] args)
    {
        final int result = 20;
        final int assign;
        Final f = new Final();
        assign = 30;
        f.print(result);
        System.out.println(process(result) + ", " + assign + ", " + f.assign);
    }

    static int process(int a)
    {
        return a + 5;
    }

    void print(int assign)
    {
        System.out.print(assign + ", ");
    }
}
```

7.10 Analyse the given program segment and predict the output:

```
public class Test
{
        public int getData() //getdata() 1
        {
                return 0;
        }
        public long getData() //getdata 2
        {
                return 1;
        }
        public static void main(String[] args)
        {
                Test obj = new Test();
                System.out.println(obj.getData());
        }
}
```