School of Computer Science Engineering and Technology

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

Tutorial-6

Tutorial No.	Name	CO1	CO2	CO3
1	Basics	✓	--	--

Objective: The main objective of this tutorial is to learn about the basics of Java language.

6.1 What will be the Output of the below code:

```
import java.io.*;
class Employee {
        int salary = 60000;
}
class Engineer extends Employee {
        int benefits = 10000;
}
class Main {
        public static void main(String args[])
        {
                Engineer E1 = new Engineer();
                System.out.println("Salary : " + E1.salary+ "\nBenefits : " + E1.benefits);
        }
}
```

6.2 What will be the Output of the below code:

```
package test;
import java.io.*;
import java.lang.*;
import java.util.*;
class one {
public void print_bu()
{
System.out.println("BU");
}
}
class two extends one {
public void print_for() { System.out.println("for"); }
}
class three extends two {
public void print_edu()
{
System.out.println("Education");
```

```
}
}
public class Main {
public static void main(String[] args)
{
three g = new three();
g.print_bu();
g.print_for();
g.print_edu();
}
}
```

6.3. What will be the output of the following program?

```
class P
{
int a = 30;
}
class Q extends P
{
int a = 50;
}
public class Main extends Q {
public static void main(String[] args)
{
Q q = new Q();
System.out.println(" Value of a: " +q.a);
P p = new Q();
System.out.println("Value of a: " +p.a);
}
}
```

6.4. What is the result of the following code?

```
package test;
class Test {
int i = 3;
}
class Main extends Test{
int i=5;
public static void main(String[] args)
{
for (int i = 1; i < 10; i++) {
i = i + 3;
System.out.print(i + " ");
}
}
}
```

6.5 What is the output of following code?

```
class Test {
final int x = 10;
int y=10;
void my(){
System.out.println(x+" "+ ++y);
}
}
class Main extends Test{
void my(){
int y=22;
System.out.println(x+" "+ ++y);
}
public static void main(String[] args)
{
Test t1 = new Main();
t1.my();
}
}
```

6.6 What will be the output of following code?

```
class A {
void show(){
int x=21;
System.out.println(x);
}
}
class B extends A{
void show(){
int x=22;
System.out.println(x);
}
}
class Main extends B{
void show(){
int x=23;
System.out.println(x);
}
public static void main(String[] args){
Main m=new Main();
m.show();
}
}
```

6.7 What would be the result of the following code?

```
class Test
{
   static int a;
```

```
    static
    {
        a = 4;
        System.out.println ("inside static block");
        System.out.println ("a = " + a);
    }
    void show()
    {
        System.out.println ("inside constructor");
        a = 10;
    }
    public void func()
    {
        a = a + 1;
        System.out.println ("a = " + a);
    }
}
class Main extends Test{
    public static void main(String[] args)
    {
        Main obj = new Main();
        obj.func();
        obj.show();
    }
}
```

6.8 What will be the output of the following program?

```
class Profile {
void tell(int w) {
System.out.println(w);
}
public Profile(int x) {
System.out.println(x);
}
}
class Test extends Profile{
public Test(int x) {
super(x);
}
}
class Main {
public static void main(String args[]) {
Profile t = new Test(25);
}
}
```

6.9 What will be the Output of the below code:

```
class A {
```

```
void A(){
int x=23;
++x;
x--;
System.out.println(x);
}
}
class B extends A{
void B() {
}
}
class Main extends A{
public static void main(String args[]) {
new B().A();
}
}
```

6.10 What will be the Output of the below code:

```
package test;
class Baseclass
{
int x = 20;
void msg()
{
System.out.println("Base class method");
}
}
class Childclass extends Baseclass
{
int x = 50;
int y = 100;
void msg()
{
System.out.println("Child class first method");
}
void msg2()
{
System.out.println("Child class second method");
}
}
public class Main extends Childclass {
public static void main(String[] args)
{
Childclass obj = new Childclass();
System.out.println("Value of x: " +obj.x);
obj.msg();
obj.msg2();
Baseclass obj2 = new Childclass();
System.out.println("Value of x: " +obj2.x);
```

```
obj2.msg();
}
}
```