

**Don Bosco Institute of Technology, Kurla**  
**Academic Year 2023-24**

**EXPERIMENT NO. 4**

**SEMESTER:III**

**DATE OF PERFORMANCE:**

**SUBJECT:** Skill based Lab Course: Object Oriented Programming with Java

**DATE OF SUBMISSION:**

**NAME OF THE STUDENT:**

**ROLL NO.:**

<b>AIM</b>	Program on method and constructor overloading. 1. Write a program to calculate areas of various shapes such as rectangle, square and circle using method overloading.
<b>LEARNING OBJECTIVE</b>	Implement programs to demonstrate method overloading.
<b>LEARNING OUTCOME</b>	The student will be able to write a program with input and output
<b>COURSE OUTCOME</b>	Students will be able to explain the various Java constructs and will be able to compare classes, objects, packages, arrays and strings.
<b>PROGRAM OUTCOME</b>	PO1,PO2,PO3,PO4,PO5,PO9,PO10,PSO1,PSO2,PSO3.
<b>BLOOM'S TAXONOMY LEVEL</b>	Apply.
<b>THEORY</b>	<ul style="list-style-type: none"><li>• What is a Constructor and explain their types?</li><li>• Explain Overloading.</li><li>• Explain overriding</li></ul>

**Don Bosco Institute of Technology, Kurla**  
**Academic Year 2023-24**

**LAB EXERCISE**

- Every student will execute the Program as per the instruction and attach output as per the requirements.

**Append all snapshots here.**

**\*Note: correct the program before execution.**

```
import java.util.*;
```

```
class Area{  
void Area(float x)  
{  
System.out.println((x*x));  
}
```

```
-----  
-----  
-----  
-----
```

```
void Area(double x, double y)  
{  
System.out.println(x*y);  
}
```

```
public static void main(String[] args) {  
Scanner in = new Scanner(System.in);  
int n, a,b;  
System.out.println("Enter your choice : ");  
Area p = new Area();  
n=in.nextInt();  
switch(n)  
{
```

```
-----  
System.out.println("Square");  
break;  
case 2: p.Area(5.0);  
System.out.println("Circle");  
-----
```

```
case 3: p.Area(5,6);
```

**Don Bosco Institute of Technology, Kurla**  
**Academic Year 2023-24**

	<pre>System.out.println("Rectangle"); ----- } } -----</pre>
<b>REFERENCES</b>	<ol style="list-style-type: none"><li>1.<a href="https://www.javatpoint.com/java-oops-concepts">https://www.javatpoint.com/java-oops-concepts</a></li><li>2.<a href="https://www.geeksforgeeks.org/object-oriented-programming-oops-concept-in-java/">https://www.geeksforgeeks.org/object-oriented-programming-oops-concept-in-java/</a></li><li>3.<a href="https://www.javatpoint.com/java-oops-concepts">https://www.javatpoint.com/java-oops-concepts</a></li></ol>