

Name:	Debjit Ghosal
UID:	2023300065
Experiment No.	6a

AIM:	Method Overriding
Program 1	
PROBLEM STATEMENT :	<p>Define class Production that has attributes String title, String director, String writer. Production class has 3 argument constructor that sets the values. It also has getter and setter methods and Overridden toString() of object class to display details of class.</p> <p>class Play is a sub class of Production with getter and setter methods and has an attribute int performances that is incremented every time a play happens. Add Overridden toString() of object class to display details of class</p> <p>class Musical is a Play with songs. Musical object has all attributes of Play as well as String composer and String lyricist along with getter and setter methods. Override toString display all attributes of Musical object</p> <p>In main create 3 objects of Play and 2 objects of Musical. Every time an object of Play or Musical is created, performances get incremented. Also add the number of seats booked for each play or musical.</p> <p>Find the total box office collection, provided cost of 1 seat for Play is Rs 500(can be variable) and cost of 1 seat for Musical is Rs 800(can be variable)</p> <p>Display total No. of performances as 5 and display the box office collection.</p> <p>Create class Tester with main method.</p>

PROGRAM:

```
import java.util.Scanner;

class Production {
    String title;
    String director;
    String writer;

    public Production(String title, String director, String writer) {
        this.title = title;
        this.director = director;
        this.writer = writer;
    }

    public String getTitle() {
        return title;
    }

    public void setTitle(String title) {
        this.title = title;
    }

    public String getDirector() {
        return director;
    }

    public void setDirector(String director) {
        this.director = director;
    }

    public String getWriter() {
        return writer;
    }

    public void setWriter(String writer) {
        this.writer = writer;
    }

    @Override
    public String toString() {
        return "Production{" +
```

```

        "title=" + title + "\" +
        ", director=" + director + "\" +
        ", writer=" + writer + "\" +
        '>';
    }
}

class Play extends Production {
    static int performances;

    public Play(String title, String director, String writer) {
        super(title, director, writer);
        performances++;
    }

    public int getPerformances() {
        return performances;
    }

    @Override
    public String toString() {
        return "Play{" +
            "performances=" + performances +
            '}';
    }
}

class Musical extends Play {
    String composer;
    String lyricist;

    public Musical(String title, String director, String writer, String
composer, String lyricist) {
        super(title, director, writer);
        this.composer = composer;
        this.lyricist = lyricist;
    }

    public String getComposer() {
        return composer;
    }
}

```

```

    }

    public void setComposer(String composer) {
        this.composer = composer;
    }

    public String getLyricist() {
        return lyricist;
    }

    public void setLyricist(String lyricist) {
        this.lyricist = lyricist;
    }
    @Override
    public String toString() {
        return "Musical{ " +
            "composer=" + composer + "\" +
            ", lyricist=" + lyricist + "\" +
            '}'";
    }
}

public class Tester {
    public static void main(String[] args) {
        // 3 different objects of Play
        Play p1 = new Play("The Lion King", "Disney", "Disney");
        Play p2 = new Play("Chotta Bheem", "Disney", "Disney");
        Play p3 = new Play("Tom and Jerry", "Disney", "Disney");

        // 2 objects of Musical
        Musical m1 = new Musical("Mast Malang Jhoom", "T Series", "T
Series", "T Series", "T Series");
        Musical m2 = new Musical("Ghungroo", "T Series", "T Series", "T
Series", "T Series");
        // printing the number of performances
        System.out.println("The number of performances: " +
Play.performances);

    }
}

```

RESULT:

```
lenovo@lenovo-ThinkCentre-neo-50s-Gen-3:~/Desktop/2023300065$ javac Tester.java
lenovo@lenovo-ThinkCentre-neo-50s-Gen-3:~/Desktop/2023300065$ java Tester
The number of performances: 5
lenovo@lenovo-ThinkCentre-neo-50s-Gen-3:~/Desktop/2023300065$ █
```

CONCLUSION:

I have learnt about method overriding of polymorphism.