

## Task -2

```
1. public class Student {  
    private String name;  
    private int age;  
    private String grade;  
  
    public Student(String name, int age, String grade) {  
        this.name = name;  
        this.age = age;  
        this.grade = grade;  
    }  
  
    @Override  
    public String toString() {  
        return "Name: " + name + ", Age: " + age + ", Grade: " + grade;  
    }  
  
    public static void main(String[] args) {  
        // Creating a student object  
        Student student1 = new Student("Chandrika", 16, "12th");  
  
        // Accessing attributes  
        System.out.println(student1.name);  
        System.out.println(student1.age);  
        System.out.println(student1.grade);  
  
        System.out.println(student1);  
    }  
}
```

```

2. class Student {
    private String name;
    private int age;
    private String grade;

    public Student(String name, int age, String grade) {
        this.name = name;
        this.age = age;
        this.grade = grade;
    }

    @Override
    public String toString() {
        return "Name: " + name + ", Age: " + age + ", Grade: " + grade;
    }
}

class UndergraduateStudent extends Student {
    private String major;

    public UndergraduateStudent(String name, int age, String grade, String major) {
        super(name, age, grade);
        this.major = major;
    }

    @Override
    public String toString() {
        return super.toString() + ", Major: " + major;
    }
}

```

```
}
```

```
class PostgraduateStudent extends Student {  
    private String researchTopic;  
  
    public PostgraduateStudent(String name, int age, String grade, String researchTopic) {  
        super(name, age, grade);  
        this.researchTopic = researchTopic;  
    }  
}
```

```
@Override
```

```
public String toString() {  
    return super.toString() + ", Research Topic: " + researchTopic;  
}  
}
```

```
public class Main {  
    public static void main(String[] args) {  
        // Creating an undergraduate student object  
        UndergraduateStudent undergradStudent = new UndergraduateStudent("Alice", 20,  
"Sophomore", "Computer Science");  
        System.out.println(undergradStudent);  
  
        // Creating a postgraduate student object  
        PostgraduateStudent postgradStudent = new PostgraduateStudent("Bob", 25,  
"Master's", "Machine Learning");  
        System.out.println(postgradStudent);  
    }  
}
```

```
3. class Student {
```

```
private String name;

private int age;

private String grade;


public Student(String name, int age, String grade) {

    this.name = name;

    this.age = age;

    this.grade = grade;

}


// Getter for name

public String getName() {

    return name;

}


// Setter for name

public void setName(String name) {

    this.name = name;

}


// Getter for age

public int getAge() {

    return age;

}


// Setter for age

public void setAge(int age) {

    this.age = age;

}
```

```
public String getGrade() {  
    return grade;  
}  
  
public void setGrade(String grade) {  
    this.grade = grade;  
}
```

@Override

```
public String toString() {  
    return "Name: " + name + ", Age: " + age + ", Grade: " + grade;  
}  
}
```

3. public class Main {

```
    public static void main(String[] args) {  
        Student student = new Student("Alice", 20, "Sophomore");
```

```
        // Using getter methods to retrieve information
```

```
        System.out.println("Name: " + student.getName());
```

```
        System.out.println("Age: " + student.getAge());
```

```
        System.out.println("Grade: " + student.getGrade());
```

```
        // Using setter methods to modify information
```

```
        student.setName("Alicia");
```

```
        student.setAge(21);
```

```
        student.setGrade("Junior");
```

```
        System.out.println(student); // Print updated student information
```

```
    }
```

```
}s
```

