Polymophism #2

```
public class Person {
    protected String name;

    public Person(String name) {
        this.name = name;
    }

    public void writeOutput() {
        System.out.println("Name: " + name);
    }
}

public class Student extends Person {
    protected int studentNumber;

    public Student(String name, int studentNumber) {
        super(name);
        this.studentNumber = studentNumber;
    }

    @Override
    public void writeOutput() {
        super.writeOutput();
        System.out.println("Student Number: " + studentNumber);
    }
}
```

```
public class Undergraduate extends Student {
    private int year;

    public Undergraduate(String name, int studentNumber, int year) {
        super(name, studentNumber);
        this.year = year;
    }

    @Override
    public void writeOutput() {
        super.writeOutput();
        System.out.println("Year: " + year);
    }
}



public class University {
    public static void main(String[] args) {
        Person[] people = new Person[4];

        // Populate the array with instances of Person, Student, and Undergraduate
        people[0] = new Undergraduate("Manny Cotty", 4910, 1);
        people[1] = new Undergraduate("Anita Kick", 9931, 2);
        people[2] = new Student("Robin DeBanque", 8812);
        people[3] = new Undergraduate("June Bugg", 9901, 4);

        for (Person p : people) {
            p.writeOutput();
            System.out.println();
```

```
    }
  }
}
```

OUTPUT

Name: Manny Cotty

Student Number: 4910

Year: 1

Name: Anita Kick

Student Number: 9931

Year: 2

Name: Robin DeBanque

Student Number: 8812

Name: June Bugg

Student Number: 9901

Year: 4

Polymophism #3

```
public class Person {
   String name;
   int age;

   public void displayInfo() {
      System.out.println("Name: " + name + ", Age: " + age);
   }
}

public class Student extends Person {
   int studentID;

   public void study() {
      System.out.println(name + " is studying.");
   }
}

public class Employee extends Person {
   int employeeID;

   public void work() {
      System.out.println(name + " is working.");
   }
}



public static void main(String[] args) {
   Person p;
   p = new Student();
   p.name = "Alice";
   p.displayInfo(); // Displays: Name: Alice, Age: 0
   p.study();      // Error: The method study() is not available in Person

   p = new Employee();
   p.name = "Bob";
   p.displayInfo(); // Displays: Name: Bob, Age: 0
   p.work();       // Error: The method work() is not available in Person
}
```