Assignment

Printing Star Patterns:

```
package patterns;

import java.util.Scanner;

public class Printingstars {

	public static void main(String[] args) {

			int opt=0;
			try (Scanner sc = new Scanner(System.in)) {
				System.out.println("Choose the pattern to print");
				System.out.println("1: Full Star");
				System.out.println("2: Left star");
				System.out.println("3: right star");
		loop: while(true) {
				int i,j,row;
				System.out.println("Enter the opt..");
				opt =sc.nextInt();
				switch(opt)
					{
					case 1:

				System.out.print("Enter the number of rows: ");

					row = sc.nextInt();

					for ( i=0; i<row; i++)
					{
					 for (j=row-i; j>1; j--)
					   {
					    System.out.print(" ");
					   }
					for (j=0; j<=i; j++ )
					{
					    System.out.print("* ");
					}
					  System.out.println();
					}
				 break loop;
				case 2:
				System.out.print("Enter the number of rows: ");

					row = sc.nextInt();

					for (i=0; i<row; i++)
					{
					    for (j=2*(row-i); j>=0; j--)
```

```
                                        {
                                                System.out.print(" ");
                                        }

                                        for (j=0; j<=i; j++ )
                                        {

                                                System.out.print("* ");
                                        }

                                        System.out.println();
                                        }
                                        break loop;
                                case 3:
                              System.out.print("Enter the number of rows: ");

                                        row = sc.nextInt();
                                        for(i=0; i<row; i++)
                                        {
                                          for(j=0; j<=i; j++)
                                                                {

                                        System.out.print("* ");
                                                                }

                                        System.out.println();
                                        }
                                        break loop;

                                default:
                         System.out.println("Invalid opt!  Choose the correct option");
                                         }
                                }
                        }

                }

        }
```

Polymorphism:

```java
package oops.capg;

public class Animal {
	public void animalSound() {
	    System.out.println("The animal makes a sound");
	  }
	}

	class Pig extends Animal {
	  public void animalSound() {
	    System.out.println("The pig says: wee wee");
	  }
	}

	class Dog extends Animal {
	  public void animalSound() {
	    System.out.println("The dog says: bow wow");
	  }
	}
```

```java
package oops.capg;

public class Main {

	public static void main(String[] args) {

	   Animal myAnimal = new Animal();  // Create Animal object
	    Animal myPig = new Pig();  // Create a Pig object
	    Animal myDog = new Dog();  // Create a Dog object
	    myAnimal.animalSound();
	    myPig.animalSound();
	    myDog.animalSound();

	}

}
```

Inheritance:

```
package oops.capg;

class Animal1 {
		void eat()
		{
			System.out.println("eating...");
		}
	}

class dog extends Animal1
		{
		void bark()
		{
			System.out.println("barking...");
		}
	}
class BabyDog extends dog
		{
		void weep()
		{
			System.out.println("weeping...");
		}
	}

package oops.capg;

public class Inheritance {

	public static void main(String[] args) {

		BabyDog d=new BabyDog();
		d.weep();
		d.bark();
		d.eat();

	}

}
```

Abstraction:

```
package oops.capg;

public abstract class MotorBike {

	abstract void brake();

}

class SportsBike extends MotorBike {

	  // implementation of abstract method
	  public void brake() {
	    System.out.println("SportsBike Brake");
	  }
	}

	class MountainBike extends MotorBike {

	  // implementation of abstract method
	  public void brake() {
	    System.out.println("MountainBike Brake");
	  }
	}


package oops.capg;

public class Test {

	public static void main(String[] args) {

		MountainBike m1 = new MountainBike();
	    m1.brake();
	    SportsBike s1 = new SportsBike();
	    s1.brake();

	}

}
```

Encapsulation:

```java
package com.mds.encapsulation;

public class Employee {

	private int id;
	private String name;
	private int salary;


	public int getId() {
		return id;
	}
	public void setId(int id) {
		this.id = id;
	}
	public String getName() {
		return name;
	}
	public void setName(String name) {
		this.name = name;
	}
	public int getSalary() {
		return salary;
	}
	public void setSalary(int salary) {
		this.salary = salary;
	}
}
```

```java
package com.mds.encapsulation;

public class Test {

	public static void main(String[] args) {

	 Employee e =new Employee();

	 e.setId(350);
	 e.setName("Harika");
	 e.setSalary(50000);

	 System.out.println(e.getId());
	 System.out.println(e.getName());
	 System.out.println(e.getSalary());
}
}
```

ALLA HARIKA DEVI
CAPG48LSRB350