Command Prompt

```
D:\workspace\HW8\src>javac Driver.java

D:\workspace\HW8\src>java Driver
Shape A Area 314.0
Shape B Area 100.0
Shape A Circumferencee 62.800000000000004
Shape B Circumferencee 40.0
5.0
Divide by zero excpetion
Divide by zero excpetion
0.25
Divide by zero excpetion
C A E B D F
Normal Race Card Speed: 200
Super Race Card Speed: 500

D:\workspace\HW8\src>
```


Java - HW8/src/Driver.java - Eclipse

```java
import java.util.ArrayList;


public class Driver {

    public static void main(String[] args) {
        //Test Shape Interface
        Shape shapeA = new Circle();
        Shape shapeB = new Square();

        shapeA.setSide(10);
        shapeB.setSide(10);

        System.out.println("Shape A Area " + shapeA.getArea());
        System.out.println("Shape B Area " + shapeB.getArea());
        System.out.println("Shape A Circumferencee " + shapeA.getCircumference());
        System.out.println("Shape B Circumferencee " + shapeB.getCircumference());

        //Test DivideByZero Exception
        Divider divider = new Divider();

        try
        {
            System.out.println(divider.divide(15, 3));
            System.out.println(divider.divide(15, 0));

        }
        catch(DivideByZeroException de)
        {
            System.out.println(de.getMessage());
        }
        catch(Exception e)
        {
            e.printStackTrace();
        }

        Ratio ratio = new Ratio();
        try
        {
            ratio.getRatio(0, 0);
        }
        catch (DivideByZeroException e)
        {
            System.out.println(e.getMessage());
        }

        try
        {
            System.out.println(ratio.getRatio(1, 4));
            System.out.println(ratio.getRatio(0, 0));
        }
        catch (DivideByZeroException e)
        {
            System.out.println(e.getMessage());
        }

        ArrayList<String> arrayList = new ArrayList<String>();

        arrayList.add("C");
```

```java
        System.out.println(divider.divide(25, 0));

    }
    catch(DivideByZeroException de)
    {
        System.out.println(de.getMessage());
    }
    catch(Exception e)
    {
        e.printStackTrace();
    }

    Ratio ratio = new Ratio();
    try
    {
        ratio.getRatio(0, 0);
    }
    catch (DivideByZeroException e)
    {
        System.out.println(e.getMessage());
    }

    try
    {
        System.out.println(ratio.getRatio(1, 4));
        System.out.println(ratio.getRatio(0, 0));
    }
    catch (DivideByZeroException e)
    {
        System.out.println(e.getMessage());
    }

    ArrayList<String> arrayList = new ArrayList<String>();

    arrayList.add("C");
    arrayList.add("A");
    arrayList.add("E");
    arrayList.add("B");
    arrayList.add("D");
    arrayList.add("F");

    java.util.Iterator<String> itr = arrayList.iterator();
    while(itr.hasNext())
    {
        String element = itr.next();
        System.out.print(element + " ");
    }
    System.out.println();

    Vehicle normalRaceCar = new RaceCar();
    Vehicle superRaceCar = new SuperRaceCar();

    System.out.println("Normal Race Card Speed: " + normalRaceCar.getTopSpeed());
    System.out.println("Super Race Card Speed: " + superRaceCar.getTopSpeed());

    }
}
```

```java
public class Car extends Vehicle {
    int topSpeed = 100;
    public int getTopSpeed()
    {
        return topSpeed;
    }
}
```

```java
public class Circle implements Shape {

    public double radius;
    @Override
    public double getArea() {
        return PI * Math.pow(radius, 2);
    }

    @Override
    public double getCircumference() {
        return PI * 2 * radius;
    }

    public void setSide(double input)
    {
        radius = input;
    }
}
```

```java
@SuppressWarnings("serial")
public class DivideByZeroException extends Exception {
    DivideByZeroException()
    {
        super("Divide by zero excpetion");
    }
}
```

```java
public class Divider {
    public float divide(float Num, float Den) throws DivideByZeroException
    {
        if (Den == 0)
        {
            throw new DivideByZeroException();
        }

        return Num/Den;
    }
}
```



```java
public class RaceCar extends Car {
    int numberOfSeats = 1;

    public int getNumberOfSeats()
    {
        return numberOfSeats;
    }

    public int getTopSpeed()
    {
        return 200;
    }
}
```

```java
public class Ratio {
    float getRatio(float X, float Y) throws DivideByZeroException
    {
        Divider divider = new Divider();
        return divider.divide(X, Y);
    }
}
```

```java
public interface Shape {
    public double getArea();

    public double getCircumference();

    public void setSide(double input);

    public static final double PI = 3.14;
}
```

```java
public class Square implements Shape {

    public double side;

    @Override
    public double getArea() {
        return Math.pow(side, 2);
    }

    @Override
    public double getCircumference() {
        return 4 * side;
    }

    @Override
    public void setSide(double input)
    {
        side = input;
    }
}
```

```java
public class SuperRaceCar extends VirtualCar {

    public int getNumberOfSeats() {
        // TODO Auto-generated method stub
        return 2;
    }

    @Override
    public int getTopSpeed() {
        // TODO Auto-generated method stub
        return 500;
    }
}
```

Java - HW8/src/Vehicle.java - Eclipse

```java
public abstract class Vehicle {

    String fuelType = "Gas";
    int numberOfWheels = 4;

    public String getFuleType()
    {
        return fuelType;
    }

    public int getNumberOfwheels()
    {
        return numberOfWheels;
    }

    public abstract int getTopSpeed();
}
```


Java - HW8/src/VirtualCar.java - Eclipse

```java
public abstract class VirtualCar extends Vehicle {
    public abstract int getNumberOfSeats();
}
```