Practice problems to do on your own for final

You do not need to submit them

Polymorphism

Define Polymorphism in your own words. Provide your own complete example. Comment your code.

1. Exercise 11.39 page 442
2. Exercise 11.40 page 442

Protected, Private and Public Data Variables

1. Define in your own words the difference between protected, private and public variable.
2. Give a comprehensive example in Java using (Java code) on using protected, private and public on data members. Comment and explain your code.

Inheritance

1. Define Inheritance in your own words. Provide your own complete example. Comment and explain your code.
2. (BONUS) Exercise 11.2 page 445 (the person, student, employee…)

Interfaces in Java (20pts)

1. What is an "interface" in Java? Provide your own example.
2. What is an "abstract class" in Java? Provide your own example.
3. Create a Java program that uses Polymorphism to model 3Dimentional-shapes.

3.1 Create an **<u>interface</u>** named: ShapeThreeDimentional
The <u>interface</u> should the following methods:

public double getShapeArea();

public double getShapeVolume();

public void printShapeInformation();

public void printShapeInformation(); // prints the area, the volume and the shape type (circle, square…) of the shape object.

3.2 Build the following classes:

Class SquarePyramidShape

The class should implements ShapeThreeDimentional

This class represents a pyramid that has a square as its base.

Add the necessary data fields and provide the implementation for overridden methods.

The **SquarePyramidShape** class has a length,width, and height;

Class SphereShape

The class should implement ShapeThreeDimentional interface.

Add the necessary data fields and provide the implementation for the overridden methods.

Create a class called ShapeRectangle that implements ShapeThreeDimentional Interface.

Create class called ShapeRectangularPrism – **the ShapeRectangularPrism class should inherit from the ShapeRectangle.**

Create the necessary data fields and implement the necessary methods.
Use the super keyword to utilize the parents' constructor.

Create a class called Circle (implement ShapeThreeDimentional Interface ShapeThreeDimentional Interface.

Create a class called **Cylinder** that inherits from the circle class.

Create a class called ShapeApplication that has an array of 3D-shapes.

```
ShapeThreeDimentional [] shapes = new  ShapeThreeDimentional [4];
ShapeThreeDimentional [0]//make this one as SquarePyramidShape


ShapeThreeDimentional [1]// make this one as SphereShape


ShapeThreeDimentional [2]//make this one as ShapeRectangularPrism

ShapeThreeDimentional [3]// make this one as a Cylinder
```

Print all the data fields of the above objects using the appropriate methods.

Make your own assumptions. Use comments to documents your assumptions.