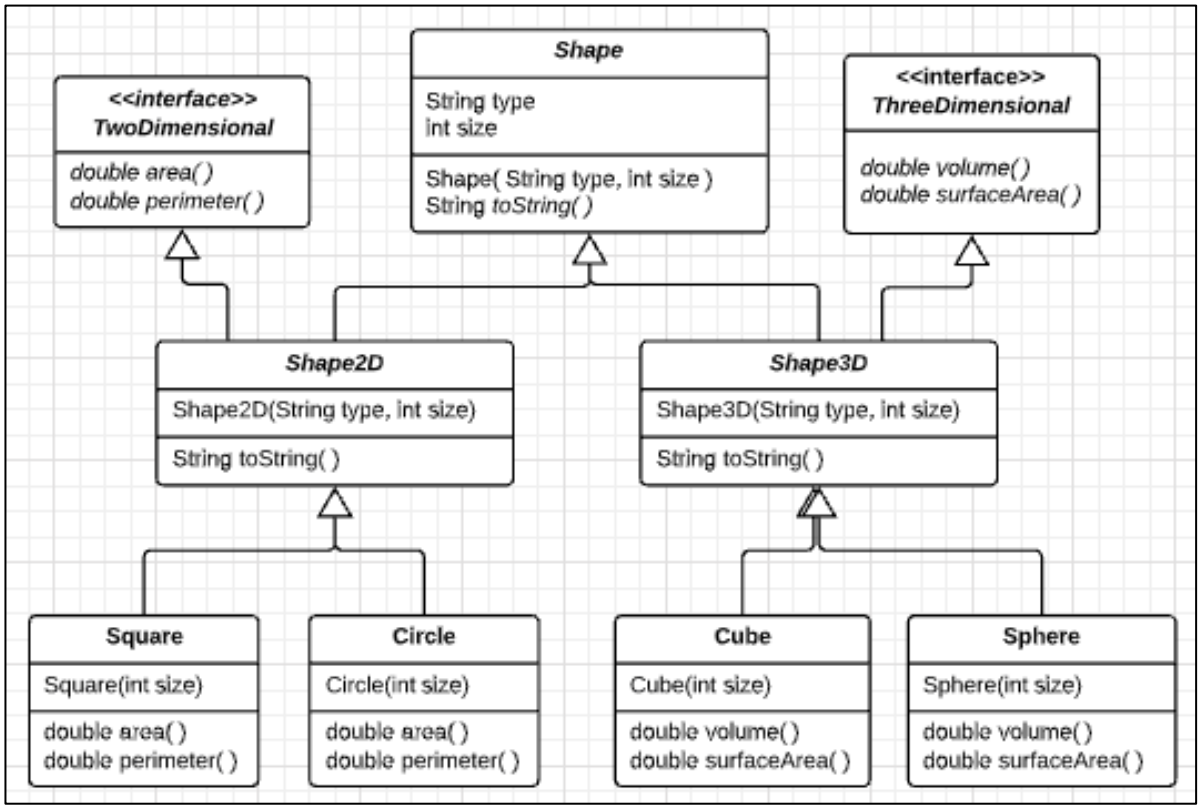
Homework 04: Polymorphic Shapes
CS 412

For this assignment, your task is to implement several shapes using inheritance, abstract classes, interfaces, and polymorphism. Specifically, you must implement a class hierarchy that matches the UML diagram below. Your code must have **three** *abstract* classes, **two** *interfaces*, and **four** *concrete* classes. *Your code MUST work with the Main class (and main method) provided*. Here's the class hierarchy that your code must implement:



Here's how to interpret this UML diagram:

- **Three** abstract classes: Shape; Shape2D; Shape3D
- **Two** interfaces: TwoDimensional; ThreeDimensional
- **Four** concrete classes: Square; Circle; Cube; Sphere

Abstract **Shape** class:

- Has two instance variables: **String type** and **int size**
- Has one *non-abstract* constructor: **Shape(String type, int size)**
- Has one *abstract* method: **String toString( )**

TwoDimensional interface:

- Defines abstract method **double area( )**
- Defines abstract method **double perimeter( )**

ThreeDimensional interface:

- Defines abstract method **double volume( )**
- Defines abstract method **double surfaceArea( )**

Abstract **Shape2D** class:

- *Extends* **Shape** class
- *Implements* **TwoDimensional** interface
- Provides code for non-abstract constructor **Shape2D(String type, int size)**
- Provides code for non-abstract method **String toString( )**

Abstract **Shape3D** class:

- *Extends* **Shape** class
- *Implements* **ThreeDimensional** interface
- Provides code for non-abstract constructor **Shape3D(String type, int size)**
- Provides code for non-abstract method **String toString( )**

Concrete **Square** class:

- *Extends* **Shape2D** class
- Provides code for non-abstract constructor **Square(int size)**
- Provides code for non-abstract method **double area( )**
- Provides code for non-abstract method **double perimeter( )**

Concrete **Circle** class:

- *Extends* **Shape2D** class
- Provides code for non-abstract constructor **Circle(int size)**
- Provides code for non-abstract method **double area( )**
- Provides code for non-abstract method **double perimeter( )**

Concrete **Cube** class:

- *Extends* **Shape3D** class
- Provides code for non-abstract constructor **Cube(int size)**
- Provides code for non-abstract method **double volume( )**
- Provides code for non-abstract method **double surfaceArea( )**

Concrete **Sphere** class:

- *Extends* **Shape3D** class
- Provides code for non-abstract constructor **Sphere(int size)**
- Provides code for non-abstract method **double volume( )**
- Provides code for non-abstract method **double surfaceArea( )**

EXAMPLE:

- Just a small sampling of output provided by main( )

```
3D shape   cube: size = 10, volume = 1000.00, surface area = 600.00
2D shape square: size =  4, area = 16.00, perimeter = 16.00
2D shape circle: size =  9, area = 254.47, perimeter = 56.55
3D shape sphere: size =  7, volume = 1436.76, surface area = 615.75
```

HINTS:

- `class`
- `public`
- `abstract`
- `extends`
- `interface`
- `implements`
- `@Override`
- `super( .. )`
- `String.format("%5s %4d %.2f", "cube", 42, 123.456)`