Inheritance assignment thing

**Problem**: find the surface area and volume of 3d shapes

Potential shapes:

* Cube
* Rectangular Prism
* Cylinder
* Triangular Prism
* Sphere
* Triangular pyramid
* Cone
* Pentagonal Prism
* Dodecahedron
* Triangular based pyramid

Algorithm:

1. Find out what shape the person wants
2. Instantiate that object
3. Take user input for any side lengths required
   1. 1 side for a **cube**
   2. 3 sides for rectangular **prism**
   3. Height and radius for **cylinder**
   4. Get the 3 sides and the height **triangular prism**
   5. Get the radius for a **sphere**
   6. Get the base side length and the h for **square based pyramid**
   7. Get the radius and height for a **cone**
   8. A side length and the height for a **pentagonal prism**
   9. A side length for a **dodecahedron**
   10. A side length for a **tetrahedron**
4. Display the surface area and volume
   1. Using object behaviours

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Classes | Shapes  (superclass) | Cube (extends Shapes) | Rectangular Prism(extends Shapes) | Cylinder (extends from chapes) |
| States | Side1  Side2  Side3  Decimal format instance (to format all calculations) | Inherit from Shapes | Inherit from Shapes | Inherit from Shapes |
| Behaviours | - get choice of shape  -check if sides are negative  -check if sides are not numbers | -implements an interface and overrides the methods  -gets volume  x3  -gets surface area  6(x2)  -prints surface area and volume | -implements an interface and overrides the methods  -gets volume  (L) x(w) x (h)  -gets surface area  2(wl+hl+hw)  -prints surface area and volume | -implements an interface and overrides the methods  -gets volume  πr2h  -gets surface area  2πrh+2πr2  -prints surface area and volume |

|  |  |  |
| --- | --- | --- |
| Classes | Sphere | Tetrahedron |
| States | Inherit from Shapes | Inherit from Shapes |
| Behaviours | -implements an interface and overrides the methods  -gets volume  -gets surface area  4πr2  -prints surface area and volume | -implements an interface and overrides the methods  -gets volume  -gets surface area  -prints surface area and volume |

Pseudocode:

1. Getting the shape of choice
   1. Instantiate new scanner
   2. Print out shape options with number options
   3. Get user input using scanner above
   4. Switch case on that input
      1. Case 1 Tell user to enter dimensions
         1. instantiate a cube object
      2. Case 2 Tell user to enter dimensions
         1. instantiate a rectangular prism object
      3. Case 3 Tell user to enter dimensions
         1. instantiate a Cylinder object
      4. Case 4 Tell user to enter dimensions
         1. instantiate a Sphere object
      5. Case 5 Tell user to enter dimensions
         1. instantiate a Tetrahedron

Check if the length is negative

1. Create a boolean method
   1. If the input is < 0
      1. Return false
   2. Else
      1. Return true

Checking method

1. Try the input statement
2. Catch an inputmismatch exception
   1. Tell them to input a number
   2. Request another input
3. while check() is false
   1. Print please enter a positive number
   2. Take another input

Volume method

1. Return (formulae above using states from superclass)

Surface Area method

1. Return (formulae above using states from superclass)

Helper method

1. Print the surface area method
2. Print the volume method