Java OOP Inheritance Maya's Candles

Time required: 90 minutes

- Comment each line of code as shown in the tutorials and other code examples.

- Follow all directions carefully and accurately.

- Think of the directions as minimum requirements.

Pseudocode

1. Write pseudocode for the exercise

2. Save it in a document

3. Submit with the assignment

Public members can be accessed from anywhere in the program, regardless of package or class. Protected members can be accessed from within the same package, as well as from subclasses in other packages. Private members can only be accessed from within the class in which they are declared.

Requirements

Maya's Candles makes candles in various sizes. she would like a program to help keep track of her candle prices.

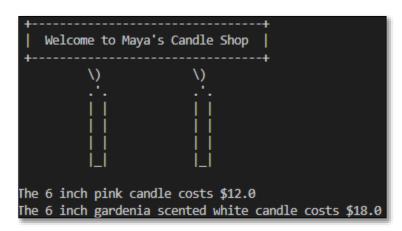Use named constants for any numbers we know before the program runs.

1. Create a class for the business named **Candle.java**

    a. Create private data fields for color, height, and price.

    b. Create public get methods for all three fields.

    c. Create public set methods for color and height, but not for price. Price will be calculated based on height.

    d. In the **setHeight()** method, when height is set, calculate the price field as $2 per inch. Price is a calculated field based on height.

    e. **@Overide** the **toString()** method that returns a string that contains information about the class, including the fields.

f. Write an application named DemoCandles.java that instantiates a Candle object and displays the details of a transaction.

2. Create a child class named **ScentedCandle.java.**

   a. Create an additional data field named **scent** and create methods to get and set it.

   b. In the Candle class → set price as protected, instead of private. It can then be accessed from the ScentedCandle class.

   c. In the child class, override the parent's **setHeight()** method to set the price of a **ScentedCandle** object at $3 per inch.

   d. Override the parent **toString()** method. Include information about the subclass, like the parent class display, that also includes the new field.

3. Modify **DemoCandles** to instantiate an object from each class. You do not have to have input from the user.

4. Display the details of each transaction.

Save the files as **Candle.java**, **ScentedCandle.java**, and **DemoCandles.java**.

Example run:

Assignment Submission

Assignment Submission

1. Use pseudocode or TODO.

2. Comment your code to show evidence of understanding.

3. Attach the program files.

4. Attach screenshots showing the successful operation of the program.

5. Submit in Blackboard.

Revised: 4/3/2025