Assessment: Assignment 1

Student Name: Stanley Pieda

Lab Professor Name: Stanley Pieda

Lab Section Number: ABC

Due Date: Jan XY, 2022

# Understand the Problem

I need the program to prompt the user for the diameter and the number of coats of varnish. From there the math to calculate the surface area of the table-top would be:

The radius, is one half of the diameter

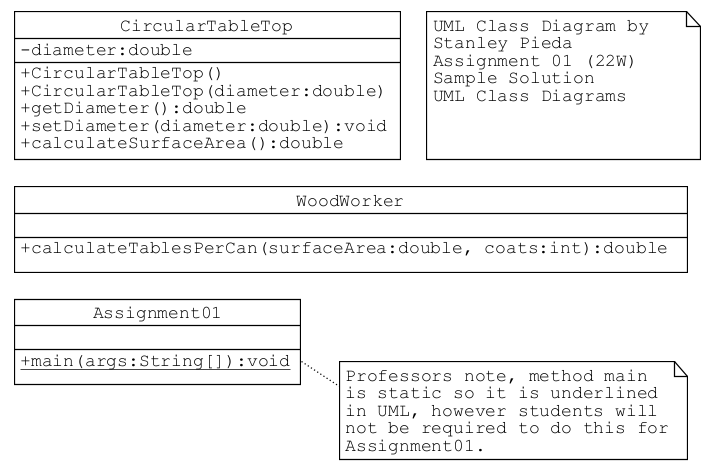
area = PI \* radius2

numberOfTables = 46800 / (area \* numberOfCoats)

Once I have the number of table-tops that can be vanished I can output this, along with my name on screen.

# UML

* The specific class names and number of classes and methods can vary from student to student, this is only a sample solution.



# Pseudocode

## Method main

start

declarations

CircularTableTop table

WoodWorker worker

num diameter

num coats

num surfaceArea

num tablesPerCan

output "Enter diameter (inches): "

input diameter

output "Enter coats needed: "

input coats

table.setDiameter(diameter)

surfaceArea = table.calculateSurfaceArea()

tablesPerCan = worker.calculateTablesPerCan(surfaceArea, coats)

output "Tables per one can: " + tablesPerCan

output "Program by Stanley Pieda"

stop

## method calculateSurfaceArea() of class CircularTableTop

// variable diameter was declared in

// class CircularTableTop, of which this

// method is also a member.

calculateSurfaceArea()

declarations

num area

num radius

radius = diameter / 2

area = Math.PI \* Math.pow(radius, 2)

return area

// Math.PI and Math.pow(num, num) are part

// of the Java API library

## Method (num surfaceArea, num coats) of class WoodWorker

// method calculateTablesPerCan(num surfaceArea, num coats)

// is a member of class WoodWorker

calculateTablesPerCan(num surfaceArea, num coats)

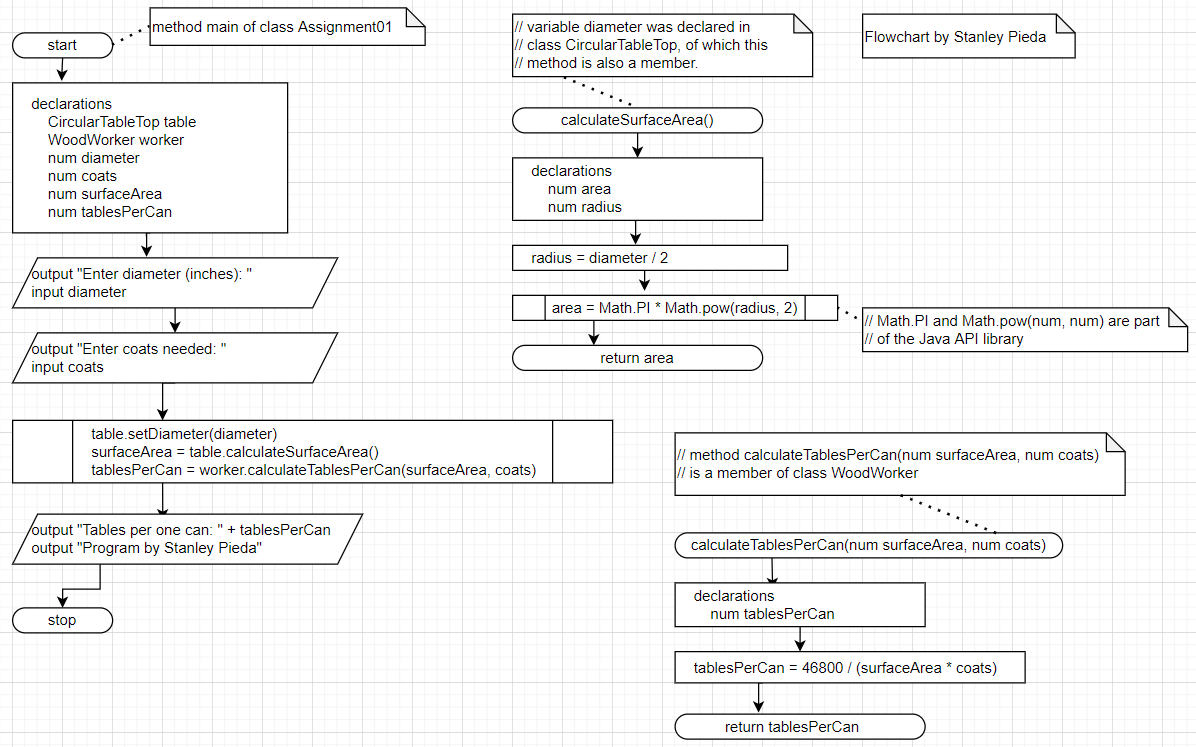
declarations

num tablesPerCan

tablesPerCan = 46800 / (surfaceArea \* coats)

return tablesPerCan

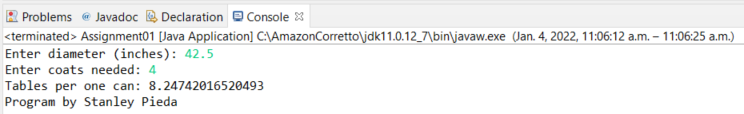
# Flowcharts



# Algorithm Test Plan

|  |  |  |  |
| --- | --- | --- | --- |
| Input | Expected Output | Actual Output | Description |
|  |  |  |  |
| 42.5  4 | Tables per one can: 8.247420165  Program by Stanley Pieda | Tables per one can: 8.247420165  Program by Stanley Pieda | Matches (used a calculator) |
| 100.0  2 | Tables per one can: 2.979380535  Program by Stanley Pieda | Tables per one can: 2.979380535  Program by Stanley Pieda | Matches (used a calculator) |
| Input | Expected Output | Actual Output | Description |
|  |  |  |  |
| 42.5  4 | Tables per one can: 8.247420165  Program by Stanley Pieda | Tables per one can: 8.247420165  Program by Stanley Pieda | Matches (used a calculator) |
| 100.0  2 | Tables per one can: 2.979380535  Program by Stanley Pieda | Tables per one can: 2.979380535  Program by Stanley Pieda | Matches (used a calculator) |

# Java Screen Shot



# Java Test Plan

|  |  |  |  |
| --- | --- | --- | --- |
| Input | Expected Output | Actual Output | Description |
| 42.5  4 | Tables per one can: 8.247420165  Program by Stanley Pieda | Tables per one can: 8.24742016520493  Program by Stanley Pieda | Similar output, Java has more decimal places |
| 100.0  2 | Tables per one can: 2.979380535  Program by Stanley Pieda | Tables per one can: 2.979380534680281  Program by Stanley Pieda | Similar output, Java has more decimal places |
| babblefish | Error? | Exception in thread "main" java.util.InputMismatchException  at java.base/java.util.Scanner.throwFor(Scanner.java:939) at java.base/java.util.Scanner.next(Scanner.java:1594) at java.base/java.util.Scanner.nextDouble(Scanner.java:2564)  at Assignment01.main(**Assignment01.java:32**) | Program crash, line number **32** in my code |
| 42  Do not panic | Error? | Exception in thread "main" java.util.InputMismatchException at java.base/java.util.Scanner.throwFor(Scanner.java:939)  at java.base/java.util.Scanner.next(Scanner.java:1594)  at java.base/java.util.Scanner.nextInt(Scanner.java:2258)  at java.base/java.util.Scanner.nextInt(Scanner.java:2212)  at Assignment01.main(**Assignment01.java:35)** | Program crash, line number **35** in my code |
| -45  -3 | Error? | Tables per one can: -9.808660196478293  Program by Stanley Pieda | No crash, but the answer does not make sense either.\*\* |

\*\* Mathematically there is no problem, however I cannot varnish tables with negative surface areas in the real-world.

# Alternative Solution(s)

* The sample solution here was created using three classes, minimally 2 classes could have been used, moving method calculateTablesPerCan into class CircularTableTop or combining the two worker methods into one longer one and so on.