**JavaBasics Assignment 501**

**Objective:**

* Understand and practise with Classes, Object, Access Modifier, Constructors, Is A Super, this keyword.
* Understand and practise with Control-of-flow.

**Business needs:**

* TBD

**Working requirements:**

* Working environment: Eclipse IDE.
* Delivery: Source code, deployment and testing, reviewing evident packaged in a compress archive.

**Product architecture:**

* N/A

**Technologies:**

The product implements one or more technology:

* Java basics
* Control of Flows
* OOP

**Stored Data:**

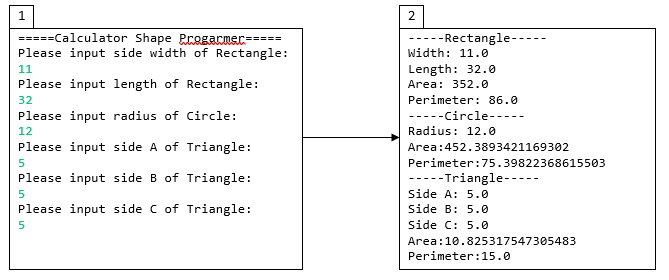
* N/A

**Project Descriptions:**

**1. Exercise 1**

Thiết kế ứng dụng tính chu vi, diện tích hình Tròn, Chữ nhật, Tam giác.

**Yêu cầu giao diện:**

****

**Yêu cầu chức năng:**

**Điều kiện tiền đề:**

* Người dùng chạy ứng dụng, chương trình yêu cầu người dùng nhập vào thông tin các cạnh của hình Vuông, Tròn, Tam giác.
* In ra thông tin các cạnh và diện tích và chu vi của các hình Vuông, Tròn, Tam giác.

**Chức năng 1:** Tính chu vi

* Chương trình tính chu vi của hình.
* Tên hàm trong code: public double getPerimeter()
  + Giá trị trả về: chu vi của hình.

**Chức năng 2:** Tính diện tích.

* Chương trình tính diện tích của hình.
* Tên hàm trong code: public double getArea()
  + Giá trị trả về: Diện tích của hình.

**Chức năng 3:** in thông tin hình.

* Chương trình in ra thông tin các cạnh, chu vi và diện tích.
* Tên hàm trong code: public void printResult()
  + Giá trị trả về: void.

**Hướng dẫn cho học viên:**

**Gợi ý:**

* Thiết kế 1 abstract class Shape chứa ba phương thức printResult, getPerimeter và getArea.
* Các class Tam giác, Chữ nhật, Tròn kế thừa Shape.
* Xây dựng các hình gồm các thuộc tính của hình tròn(radius), chữ nhật(width, length), tam giác(sideA, sideB, sideC) và tạo thêm các phương thức getter và setter.
* Override lại các phương thức của lớp Shape.
* Tính diện tích hình tròn bằng công thức Heron:

https://codelearn.io/Media/Default/Users/AnhDQ14/AnhDQ14/test2.PNG

Sử dụng Math.sqrt()

* Tính diện tích hình tròn với Pi = Math.PI

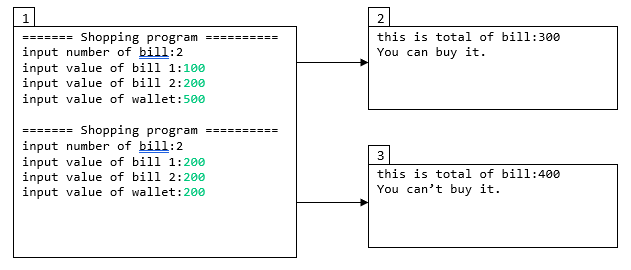
**Thời gian ước tính:**  90 phút

**2. Exercise 2**

Thiết kế bài toán nhận đầu vào là các hóa đơn của một người. Tính số tiền mà người này đã sử dụng. So sánh số tiền đã sử dụng với số tiền trong ví của người này, xem giao dịch có thực hiện được không?

* Thiết kế một class Person thể hiện cho người mua.
* Thiết kế một class Wallet thể hiện ví tiền của người dùng nằm trong class Person.

**Yêu cầu giao diện:**

****

**Yêu cầu chức năng:**

**Điều kiện tiền đề:**

* Người dùng chạy ứng dụng, chương trình hiển thị màn hình yêu cầu người dùng nhập vào số lượng hóa đơn.
* Tiếp đó, chương trình yêu cầu người dùng nhập giá trị của từng hóa đơn.
* Nhập số tiền có trong ví.

**Chức năng 1:** Tính tổng giá trị của các hóa đơn

* Chương trình tính tổng giá trị của các hóa đơn.
* Tên hàm trong code: public int calcTotal (int[] bills)
  + Đầu vào:
    - bills: các hóa đơn.
  + Giá trị trả về: Tổng giá trị của các hóa đơn.

**Chức năng 2:** Kiểm tra xem trong ví con đủ số tiền để thanh toán hay không.

* Chương trình so sánh số lượng tiền trong ví với tổng giá trị các hóa đơn.
* Tên hàm trong code: public boolean payMoney(int total)
  + Đầu vào:
    - total: tổng giá trị các hóa đơn.
  + Giá trị trả về: trạng thái thanh toán.

**Hướng dẫn cho học viên**

**Gợi ý:**

Class Wallet sẽ chứa thông tin về số tiền trong ví và hàm so sánh số tiền trong ví với giá trị hóa đơn.

**Thời gian ước tính**: 90 mins

**--THE END--**