COIS 2240 Assignment 1

Question 1 Git repository:

1- Go to the repository at https://github.com/omalam/COIS2240Assignment1.git and and click Fork. Go to the new repository that GitHub will have created in your account, which will look like https://github.com/<yourname>/COIS2240Assignment1 and click "Clone or Download" and copy URI. In terminal, clone the repository whose URI you just copied.

2- Create a branch using the command line and name it your name. The branch name should be YourFirstNameYourLastName. For example, I can create a branch with my name *OmarAlam*. You should use the command *$ git branch YourFirstNameYourLastNameBranch* for this.

3- Checkout your branch using the command *$ git checkout YourFirstNameYourLastNameBranch*

4- Add an empty text file to your branch using the command *$ git add YourFirstNameYourLastName.txt*

5- Commit the file in the previous using the command *$ git commit -m"My file for Question 1" YourFirstNameYourLastName.txt*

6- Push your branch to the remote repository using the command $ *git push -u origin YourFirstNameYourLastNameBranch*

Take screenshot showing that you executed the commands successfully.

Question 2: (a) An organization has three categories of employee: professional staff, technical staff and support staff. The organization also has departments. Each employee belongs to a department.

(b) An employee can work on any number of projects and a project can be assigned to any number of employees. However, the employee has a position (role) in the project. The position has a description.
Solve this problem with and without using an association class. Employees and Projects should have names.

Draw a class diagrams that captures the above scenarios. You may use any UML drawing tool. There are online UML drawing tools such as draw.io

Question 3:
Problem Description:

Design a class named Triangle that extends GeometricObject. The class contains:

- Three double data fields named side1, side2, and side3 with default values 1.0 to denote three sides of the triangle.
- A constructor that creates a triangle with the specified side1, side2, and side3.
- The get and set methods for all three data fields.
- A method named getArea() that returns the area of this triangle.
- A method named getPerimeter() that returns the perimeter of this triangle.
- A method named toString() that returns a string description for the triangle. The toString() method is implemented as follows:

```
return "Triangle: side1 = " + side1 + " side2 = " + side2 +
  " side3 = " + side3;
```

Coding:

```
public class Assignment1 {
  public static void main(String[] args) {
    Triangle triangle = new Triangle(1, 1.5, 1);
    triangle.setColor("yellow");
    triangle.setFilled(true);

    System.out.println(triangle);
    System.out.println("The area is " + triangle.getArea());
    System.out.println("The perimeter is "
      + triangle.getPerimeter());
    System.out.println(triangle);
  }
}

public abstract class GeometricObject {
  private String color = "white";
  private boolean filled;
  private java.util.Date dateCreated;
```

```
  /** Construct a default geometric object */
  protected GeometricObject() {
    dateCreated = new java.util.Date();
  }

  /** Construct a geometric object with color and filled value */
  protected GeometricObject(String color, boolean filled) {
    dateCreated = new java.util.Date();
    this.color = color;
    this.filled = filled;
  }

  /** Return color */
  public String getColor() {
    return color;
  }

  /** Set a new color */
  public void setColor(String color) {
    this.color = color;
  }

  /** Return filled. Since filled is boolean,
   *  the get method is named isFilled */
  public boolean isFilled() {
    return filled;
  }

  /** Set a new filled */
  public void setFilled(boolean filled) {
    this.filled = filled;
  }

  /** Get dateCreated */
  public java.util.Date getDateCreated() {
    return dateCreated;
  }

  @Override
  public String toString() {
    return "created on " + dateCreated + "\ncolor: " + color +
      " and filled: " + filled;
  }

  /** Abstract method getArea */
  public abstract double getArea();

  /** Abstract method getPerimeter */
  public abstract double getPerimeter();
}

class Triangle extends GeometricObject {
  // Implement it
}
```

Submit the following items:

1. Implement the class Triangle.
2. Run the test program that creates a Triangle object with sides 1, 1.5, 1, colour yellow and filled true, and displays the area, perimeter, colour, and whether filled or not.
3. Compile, Run, and take a screenshot of the output and submit to Blackboard (you must submit the program regardless whether it complete or incomplete, correct or incorrect).