NAME: MUTHUWANA M.A.N.R

INDEX:210400N

Formative assessment - 1

1. You are required to develop a software program that will simulate a swimming competition. The simulation contains a swimming pool that has 5 swim lanes. It also has a pavilion. There are swimmers, spectators, judges and supporting staff. Swimmers should be distinguished as male and female – males wear blue and females wear red. Every person in the simulation has a unique name. When the judge blows the whistle, the competition is started and the players start swimming. Once they reach the other end of the pool, they have finished the game. They need to touch a touch pad at this finishing end. Each touch pad then notifies the finishing time to a score board. The score board compares these finishing times, and identifies the order of finishing. This information is then displayed in the score board.

    - Identify objects, their states, and behavior in this scenario.

Objects	states	Behavior in this scenario
Swimming pool	no of swimming lanes, length	
swimmer	name, gender, clothes color	swim, touch touchpad
Spectator	name	watch, cheer
Judge	name	blow whistle
Supporting staff	name	support

Touchpad		notifies finishing time
Score board	scorecard	compare finishing times, identifies the order of finishing display scoreboard

- Advanced task – identify relationships among these objects.
  - ✓ When **judge** blow whistle ->**swimmer** start swim
  - ✓ **Swimmer** touchtouchpad->**touchpad** notifies finishing time to **scoreboard**->**scoreboard** compare finishing time & identifies the order of finishing & display scoreboard

2. You are required to develop a software program that will simulate a newspaper. The newspaper consists of a set of news articles. Each news article contains one or more paragraphs of text. Some news articles may contain one or more images. Each news article has a headline (for example, "Learning OOP is banned in Sri Lanka"), a byline (for example, "By Prince of Wales, Weekly News Academic Reporter"), a tagline (for example, "OOP considered a health hazard!") and a date-stamp (for example, "03 May 2010").
   Each line of text used in the newspaper has the following set of attributes associated with it: font type, font size, font color, background color, bold type face, italic type face, underlined type face.
   Each headline, byline, tagline or a date-stamp is considered a single line of text. A paragraph of text may contain one or more lines of text.

   - Identify the “Objects” and their attributes from the above scenario description.

objects	attributes
newspaper	name
article	paragraphs of text, images, headline, byline, tagline, date stamp
images	size
paragraph	Lines of text
Line of text	front type, front size, front color, background color, bold type face, italic type face, underlined type face

- Decide on objects that can form the basis of an "abstract" object.

- Decide on the “characteristics” of those objects that would represent their internal state.

objects	characteristics
article	create an article
paragraph	create a text
Line of text	create a text

3. Given below is a simplified version of the swimming competition scenario:

   You are required to develop a software program that will simulate a swimming competition. In the simulation, there are swimmers, spectators, judges and supporting staff. They all have a name, and an id number. They all can check the scoreboard. In addition, swimmers can swim, and judges can blow the whistle. Number of objects to be created for swimmers, spectators, etc should be able to be given in as command-line arguments.

   - Write a simple program to capture this information.
   - Copy all your Java code into your word doc or pdf and submit.

```
import java.util.*;

public class Main {
    public static void main(String[] args) {
        Scanner input=new Scanner(System.in);
        System.out.println("||||||||||||||Swimming
Competetion||||||||||||||");
        System.out.print("Enter number of Swimmers : ");
        int no_of_swimmers= input.nextInt();
        System.out.print("Enter number of Judges : ");
        int no_of_judges=input.nextInt();
        System.out.print("Enter number of Spectators : ");
        int no_of_spectators=input.nextInt();
        System.out.print("Enter number of Supporting staff : ");
        int no_of_supporting_staff=input.nextInt();

        Swimmer[] swimmers=new Swimmer[no_of_swimmers];
```

```
        Judge[] judges=new Judge[no_of_judges];
        Spectator[] spectators=new Spectator[no_of_spectators];
        SupportingStaff[] supportingStaffs=new
SupportingStaff[no_of_supporting_staff];

        for(int i=0;i<no_of_swimmers;i++){
            System.out.print("Enter the name of the swimmer : ");
            String name= input.next();
            System.out.print("Enter the id of the swimmer : ");
            String id= input.next();
            swimmers[i]=new Swimmer(name,id);
        }
        for(int i=0;i<no_of_judges;i++){
            System.out.print("Enter the name of the judge : ");
            String name= input.next();
            System.out.print("Enter the id of the judge : ");
            String id= input.next();
            judges[i]=new Judge(name,id);
        }
        for(int i=0;i<no_of_spectators;i++){
            System.out.print("Enter the name of the swimmer : ");
            String name= input.next();
            System.out.print("Enter the id of the swimmer : ");
            String id= input.next();
            spectators[i]=new Spectator(name,id);
        }
        for(int i=0;i<no_of_supporting_staff;i++){
            System.out.print("Enter the name of the swimmer : ");
            String name= input.next();
            System.out.print("Enter the id of the swimmer : ");
            String id= input.next();
            supportingStaffs[i]=new SupportingStaff(name,id);
        }
////////////////////////////////////////////////////////////////////////

        for(int j=0;j<no_of_judges;j++){
            judges[j].blowWhistle();
        }
        for(int i=0;i<no_of_swimmers;i++){
            swimmers[i].swim();
        }

    }
}
```

```
public class Character {
    private String name;
    private String id;

    Character(String name,String id){
        this.name=name;
        this.id=id;
    }
    public void chekScoreBoard(Scoreboard scoreboard){
```

```
        scoreboard.displayScorecard();
    }
    public String getName(){
        return this.name;
    }
}
public class Swimmer extends Character {
    private float time=0;
    private String clothescolor;
    private String gender;

    Swimmer(String name,String id){
        super(name,id);
    }
    public void swim(){
        System.out.println("Swimming");
    }
    public void touchTouchpad(Touchpad touchpad){
        touchpad.notifyy();
    }
}
public class Judge extends Character{

    public Judge(String name, String id){
        super(name,id);
    }
    public void blowWhistle(){
        System.out.println("whistle! competition starts");
    }
}
public class SupportingStaff extends Character{

    public SupportingStaff(String name, String id){
        super(name,id);
    }
    public void support(){
        System.out.println("Support!");
    }
    public void support1(Scoreboard scoreboard){
       scoreboard.displayScorecard();
    }
}
public class Spectator extends Character {

    public Spectator(String name, String id){
        super(name,id);
    }
    void cheer(){
        System.out.println("Cheer!");
    }
}
public class Touchpad {
    void notifyy(){
        System.out.println("Notify to the Scoreboard");
    }
}
```

```java
public class Scoreboard{
    float[] scorecard;

    public void displayScorecard(){
        System.out.println("Score board");
    }
    public void identifiesOrderFinishing(){
        System.out.println("Identifying");
    }
    public void compareFinishingTimes(){
        System.out.println("Comparing!");
    }
}
```