**4.3 – Understanding Object**

**Oriented Programming Theory**

For this assignment we will be using A Guide to Programming in JAVA by Beth Brown. Please type your answers in this document. When you are done, upload the file to your GitHub account in a repo called “Assignment 4-3” available at:

<https://bbarrettchs.weebly.com/uploads/3/7/7/8/37782575/lvp_java_text.pdf>

**Who are you?**

0. What is your name? Connor McDougall

**What is an Object?**

Read page 179-180 and answer the following questions:

1. The textbook describes an object as a collection of state and behaviour. What is meant by state and behaviour?

State of an Object: Defines the data or info of the object. Should only be changed by behavior

Ex: State of a Circle object is radius, because a circle is defined by its radius

Behavior of an Object: The actions or communication that the object gives/does

Ex: Circle object changing radius or telling us what the radius is (Methods)

2. Define Encapsulation / Information Hiding.

Encapsulation is the object hiding some data from places outside the class, aka from the client code

3. Define client code.

A body of code or application that uses one or more class(es)

Client code can access methods of classes, but not variables (directly)

**Designing and Writing a Class**

Read page 180-182 and answer the following questions:

4. Define Functional Decomposition.

The process of making clear and concise methods by breaking them down until they cannot be simplified further.

5. What three things does the class declaration contain?

It contains the: Accesses Level (Public or Private)

class keyword

and the return type (void is no return type)

6. What three things does the class body contain?

Constructer

Variables

Methods?

7. Access levels: what does it mean to make a variable or method public? What does it mean to make a variable or method private?

It means that you are making that method/variable only accessible by the class it is from.

8. What is an interface?

It is the public method/(s) of a class, as that is how you interact with the class.

9. Define accessor method, modifier method, and helper method. Which one of these types of methods is NOT part of the interface?

Accessor Method: Called to determine the value of a class variable

Modifier Method: Called to change the value of a class variable.

Helper Method: Called within the class by other methods. Have access level private.

-Helper Methods are not part of the interface

10. Do the problem "Review: Circle - part 1 of 4" on page 182

public static void main(String[] args) {

Circle spot = new Circle();

spot.setRadius(3);

System.out.println("Circle radius: " + spot.getRadius());

System.out.println("Circle circumference: " + spot.circumference())

}

Public double circumference(int r){

return (2\*Math.PI()\*r);

}

**OR**

Public double circumference(Circle C){

Return (2\*Math.PI() \* C.getRadiu());

}

**Writing Constructors**

Read page 183 and answer the following questions:

11. What does it mean for an object to be instantiated?

Instantiation means creating. Ex: I created the object; I instantiated the object.

12. What is a constructor method and what does it do?

It is added to a class so that you can create the class object. You can add parameters to specify the objects variables values during creation.

13. What two things are always true about constructor methods?

Always has the same name as the class

They have no return Type

13. What does it mean to "overload" a constructor method?

It means assigning variables values during the creation of the class.

14. Do the problem "Review: Circle - part 2 of 4" on page 184

Public Circle (double r){

radius = r;

}

**Instance and Class Members**

Read page 184-185 and answer the following questions:

15. What is the difference between an instance variable and a class variable? How do you declare a variable as an instance variable? How do you declare a variable as a class variable? Give an example of each from the Circle class. **(16 was the same as 15)**

**Instance Variable**: Is a variable specific to an object

*public Circle(double r) { radius = r; }*

**Class Variable**: Is a variable general (the same) to all objects in the class. Uses static keyword.

*public static void displayAreaFormula() {*

*System.out.println("The formula for the area of a circle is a=Pi\*r\*r");*

*}*

17. Do the problem "Review: Circle - Part 3 of 4" on page 185.

Public static displayAreaFormula() {

System.out.Println(“The Area Formula is A = PI\*r^2”)

}

public static void main(String[] args) {

Circle spot = new Circle();

spot.setRadius(3);

System.out.println("Circle radius: " + spot.getRadius());

System.out.println("Circle circumference: " + spot.circumference())

**displayAreaFormula();**

}