

**Computer Science or Information Technology**

Instructor: **Dawei Li, Ph.D.**

Day, Month, Year

Day

CSIT 501

Department of CSIT

Assessment

Module-7

Hidalgo, Rafael

Exercise 7.1,

Write a method called product that accepts four integer parameters and returns their product as an integer.

// static method multiplying four ints and returning one int

**public** **static** **int** product(**int** a, **int** b, **int** c, **int** d)

{

**int** e = a\*b\*c\*d;

**return** e;

}

Exercise 7.4,

Write a method called doubleConcat that takes a String and an integer as parameters. Return a String that consists of the string parameter concatenated with itself twice the number times, where number is an integer parameter. For example, if the parameter values are "hello" and 1, the return value should be "hellohello". Return the original string if the integer parameter is less than 1.

**public** **static** String doubleConcat(String a, **int** b) {

String concate = "";

**if** (b < 1) {

**return** a;

}

**else** {

**for** (**int** i = 0; i < (2 \* b); i++) {

concate += a;

}

**return** concate;

}

}

Exercise 7.6,

Write a method called drawCircle that draws a circle based on the method’s parameters: a Graphics object through which to draw the circle, two integer values representing the (x, y) coordinates of the center of the circle, another integer that represents the circle’s radius, and a Color object that defines the circle’s color. The method does not return anything.

**public** **void** drawCircle(Graphics page, **int** ux, **int** uy, **int** r, Color shade) {

page.setColor(shade);

page.fillOval(ux, uy, r\*2, r\*2);

}

Exercise7.7,

Overload the drawCircle method of Exercise 7.6 such that if the Color parameter is not provided, the circle’s color will default to black.

**public** **void** drawCircle(Graphics page, **int** ux, **int** uy, **int** r, Color shade) {

page.setColor(shade);

page.fillOval(ux, uy, r\*2, r\*2);

}

**public** **void** drawCircle(Graphics page, **int** ux, **int** uy, **int** r) {

page.setColor(Color.***black***);

page.fillOval(ux, uy, r\*2, r\*2);

}

Exercise 7.11,

Explain why a static method cannot refer to an instance variable.

A static method does not require to instantiate the class that it belongs to. Therefore, a static method can be called by the class name followed by the method name. This means that no objects are created and thus, no instance variables exist.