Antonio Rosado

Prof. Chien

IOOP Section #4

Lab #3

**By submitting this work, I certify that this assignment is my own work, and that I have not copied in part or whole from online sources or plagiarized the work of other students.**

**Exercises**

**2.64**

The method name is getCode, and the return type is a string.

**2.65**

The method name is setCredits, and the parameter’s name is creditValue and it is of type int.

**2.66**

public class Person

{

}

**2.67**

private String name;

private int age;

private String code;

private int credits;

**2.68**

public Module(String moduleCode)

{

code = moduleCode;

}

**2.69**

public Person(String myName, int myAge)

{

name = myName;

age = myAge;

}

**2.70**

The original method had a return type of “void” but had a return statement which is false. If the return statement is void, that means it returns nothing. A return type of int would be more appropriate for a value named age.

**2.71**

public String getName()

{

return name;

}

**2.72**

public void setAge(int newAge)

{

age = newAge;

}

**2.73**

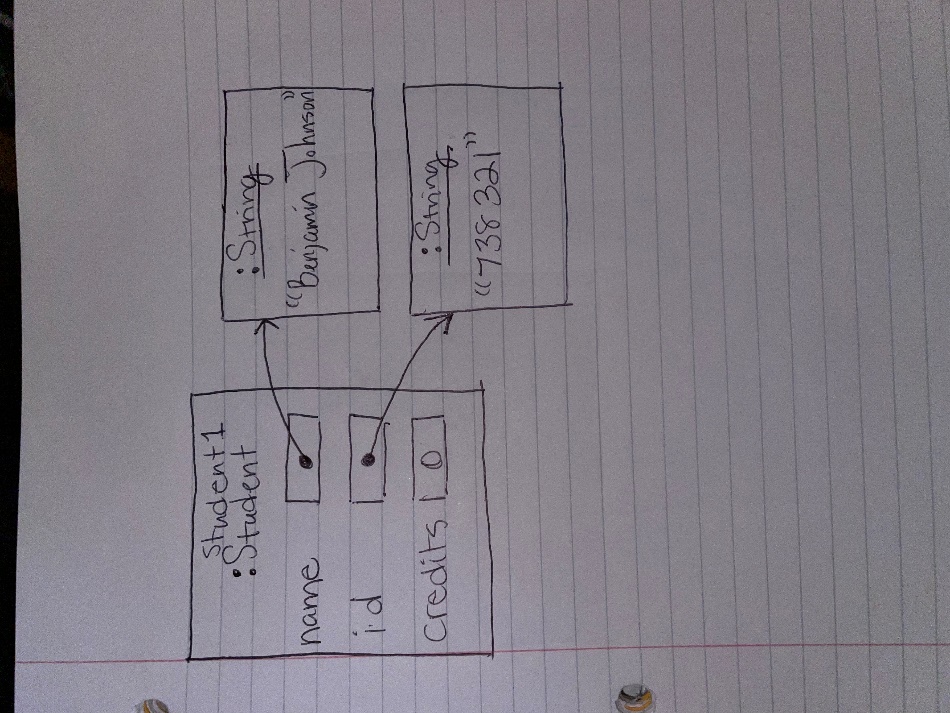
public void printDetails()

{

System.out.println(“The name of this person is ” + name);

}

**2.74**



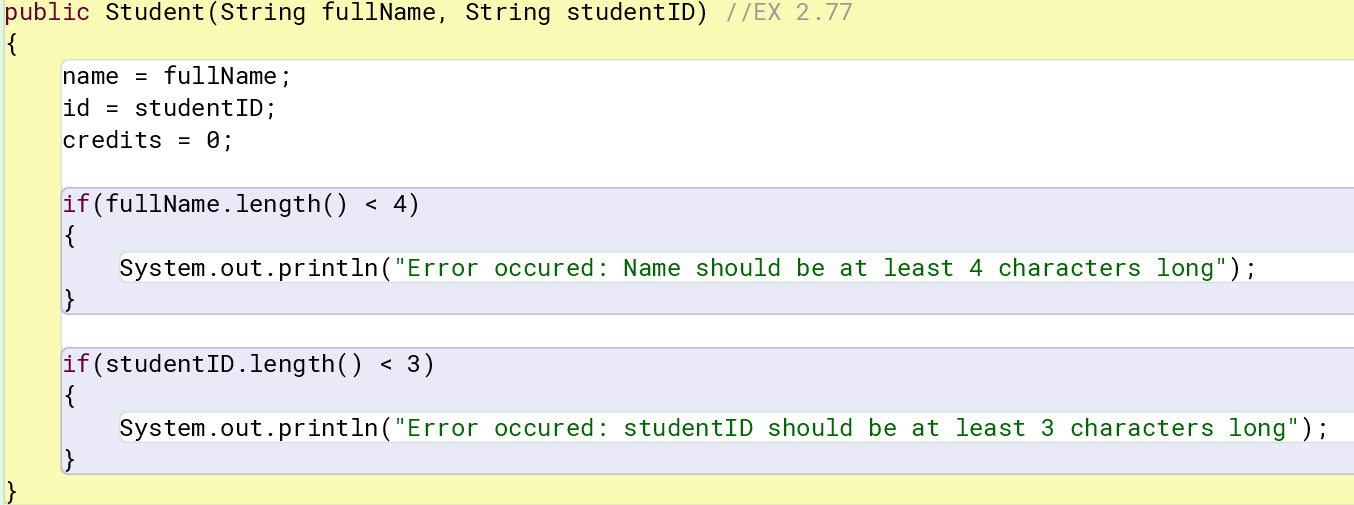
**2.75**

Henr557

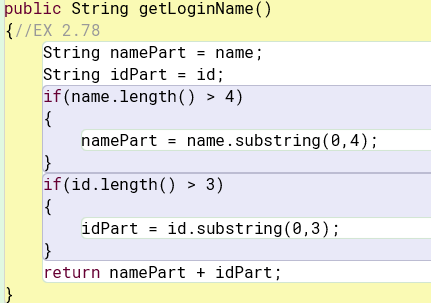
**2.76**

When getLoginName is called on the student, an error occurs stating the string is out of range. This is because the minimum character length is four characters and the student’s name does not meet said requirement so a fourth character cannot be grabbed.

**2.77**



**2.78**



**2.79**

102

“catfish”

“cat9”

“12cat”

“cat39”

“f”

Error, returns

Exception: java.lang.StringIndexOutOfBoundsException (begin 3, end 8, length 7)

**2.80**

0 (int)

500 (int)

**2.81**

I would expect a call to t2.getBalance(0 to be the same as calling the t1.getBalance() method because they hold the same values under different names.

TicketMachine t2 = t1;

t2.getBalance()

500 (int)

**2.82**

1000 (int) is what is returned and I expected that because any changes or adjustments made to t1 would reflect on t2.

**2.86**

Immutable objects are objects that cannot be changed, so our objects would be immutable because fields cannot be manipulated after they are declared.

**2.83-2.92**





****