Course Code	OCAJP 8
Section	Topic 6
Student Name	Dominic Kgoete
Student Signature	
Date	21 – 01- 2020

1. **Program Objective and Summary**

- Create methods with arguments and return values; including overloaded methods
- Apply the static keyword to methods and fields
- Create and overload constructors; including impact on default constructors
- Apply access modifiers
- Apply encapsulation principles to a class
- Determine the effect upon object references and primitive values when they are passed into methods that change the values

2. **Tools (Clearly state the IDE, Databases, languages used)**

Intellij IDEA

3. **Code**

```
/*
 **** Topic 6: Working with Methods and Encapsulation ****
 * OBJECTIVES:
 * 1 Create methods with arguments and return values; including overloaded methods
 * 2 Apply the static keyword to methods and fields
 * 3 Create and overload constructors; including impact on default constructors
 * 4 Apply access modifiers
 * 5 Apply encapsulation principles to a class
 * 6 Determine the effect upon object references and primitive values when they are
 *   passed into methods that change the values
 */

package com.company;

/**
 * @author dominic
 */

class Adder{
    static double add(double a, double b){//Method overloading
        return a / b;
    }
```

Course Code	OCAJP 8
Section	Topic 6
Student Name	Dominic Kgoete
Student Signature	
Date	21 – 01- 2020

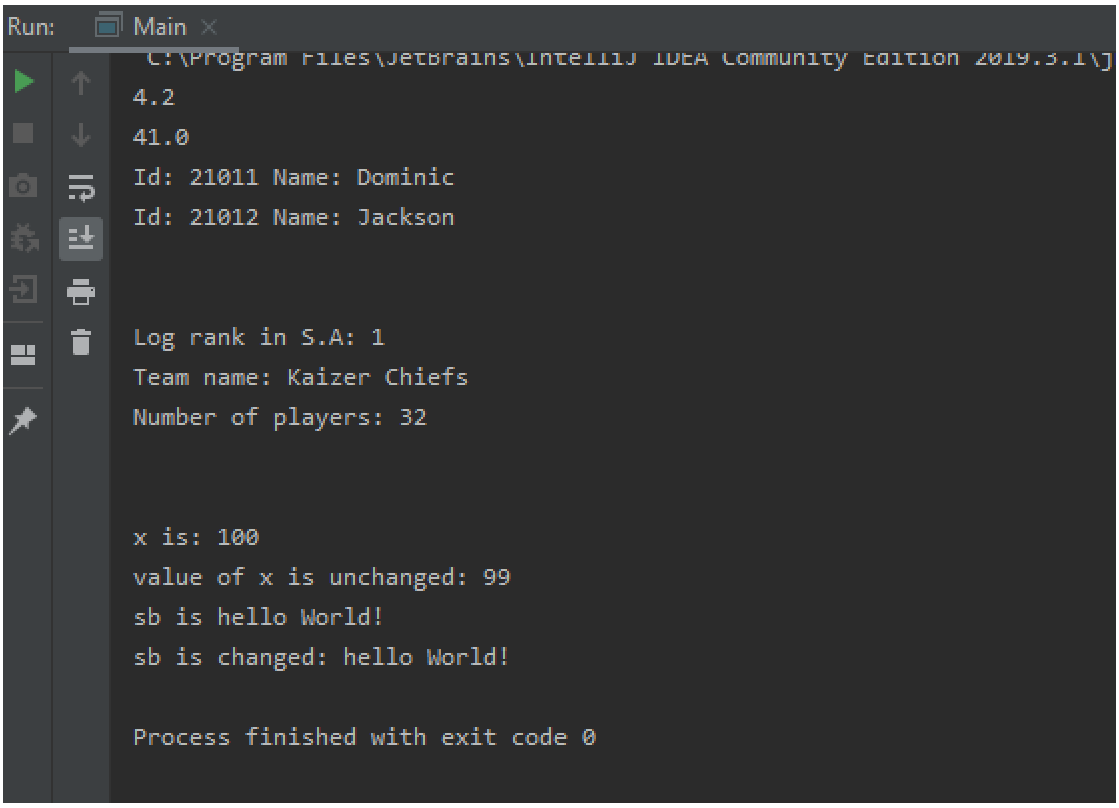
```
    static double add(double a, double b, double c) {
        return ((a + b) / c) * 100;
    }
}

class EncapsulationClass {
    private int taxNumber;
    private String teamName;
    private int numberOfPlayers;

    //Getter and setter methods
    public int getTaxNumber(){
        return taxNumber;
    }

    public String getTeamName(){
        return teamName;
    }

    public int getNumberOfPlayers(){
      return numberOfPlayers;
    }

    public void setTaxNumber(int newValue){
        taxNumber = newValue;
    }

    public void setTeamName(String newValue){
        teamName = newValue;
    }

    public void setNumberOfPlayers(int newValue){
        numberOfPlayers = newValue;
    }
}

public class Main {
    int studentId;
    String studentName;
    //parameterized constructor with two parameters
    Main(int id, String name){
        this.studentId = id;
        this.studentName = name;
    }
    void info(){
        System.out.println("Id: "+studentId+" Name: "+studentName);
    }

    public static void addOne(int x){
        x++;
```

Course Code	OCAJP 8
Section	Topic 6
Student Name	Dominic Kgoete
Student Signature	
Date	21 – 01- 2020

```
        System.out.println("x is: " + x);
    }

    public static void addWord(StringBuilder sb){
        sb.append(" World!");
        System.out.println("sb is " + sb);
    }

    public static void main(String[] args) {
        Main obj1 = new Main(21011, "Dominic");
        Main obj2 = new Main(21012, "Jackson");
        System.out.println(Adder.add(10.5, 2.5));
        System.out.println(Adder.add(30.5, 10.5, 100));
        obj1.info();
        obj2.info();

        //Accessing the encapsulation class with its methods that have private fields.
        EncapsulationClass obj3 = new EncapsulationClass();
        obj3.setTaxNumber(1);
        obj3.setTeamName("Kaizer Chiefs");
        obj3.setNumberOfPlayers(32);
        System.out.println("\n");
        System.out.println("Log rank in S.A: " + obj3.getTaxNumber());
        System.out.println("Team name: " + obj3.getTeamName());
        System.out.println("Number of players: " + obj3.getNumberOfPlayers());

        /*
            Determine the effect upon object references and
            primitive values when they are passed into methods that change the values
         */
        System.out.println("\n");
        int x = 99;
        addOne(x);
        System.out.println("value of x is unchanged: " + x);
        /*
            pass a new, different reference to the same object
         */
        StringBuilder sb = new StringBuilder("hello");
        addWord(sb);
        System.out.println("sb is changed: " + sb);
    }
}
```

Course Code	OCAJP 8
Section	Topic 6
Student Name	Dominic Kgoete
Student Signature	
Date	21 – 01- 2020

4. Output (Please attach screen output as evidence)



```
 C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2019.3.1\j
4.2
41.0
Id: 21011 Name: Dominic
Id: 21012 Name: Jackson


Log rank in S.A: 1
Team name: Kaizer Chiefs
Number of players: 32


x is: 100
value of x is unchanged: 99
sb is hello World!
sb is changed: hello World!

Process finished with exit code 0
```

Clone: https://github.com/lorthbroth/OCA-java-se8-programmer-1.git for output

5. Completed Practical Assignment Template
Completed Practical Assignment Template and attach it on LMS as a pdf file