

TP INFO 4067

Nom : Tomdieu TchadieuKo Ivan Gottfried

Matricule : 20U2947

Exercice

Singleton Pattern : Arithmetic Class

- Diagram UML

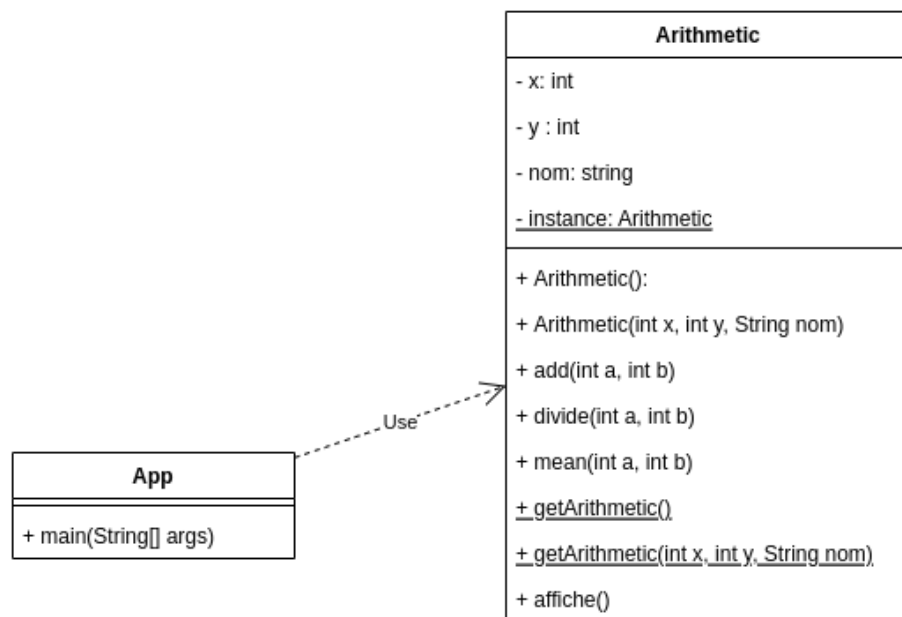


Figure 1: Image Du Diagram UML

- Code

- App.java

```
import com.singleton.Arithmetic;

public class App {
    public static void main(String[] args) throws Exception {

        int sum = Arithmetic.getArithmetic().add(5, 1);

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

```

        Arithmetic a1 = Arithmetic.getArithmetic(5,8,"ivantom");
        a1.affiche();

        Arithmetic b1 = Arithmetic.getArithmetic(-52, 25, "Tomdieu");
        b1.affiche();
    }
}

```



```

package com.singleton;

public final class Arithmetic {
    private static Arithmetic instance = null;
    private int x;
    private int y;
    private String nom;

    private Arithmetic() {
        super();
    }

    private Arithmetic(int x, int y, String nom) {
        this.x = x;
        this.y = y;
        this.nom = nom;
    }

    public int add(int a, int b) {
        return a + b;
    }

    public float divide(int a, int b) {
        return (float) a / b;
    }

    public float mean(int a, int b) {
        return add(a, b) / 2;
    }

    public static Arithmetic getArithmetic() {
        if (instance == null) {
            instance = new Arithmetic();
        }
        return instance;
    }

    public static Arithmetic getArithmetic(int x, int y, String nom) {
        if (instance == null) {
            instance = new Arithmetic(x, y, nom);
        }
        else {
            instance.x = x;
            instance.y = y;
            instance.nom = nom;
        }
        return instance;
    }

    public void affiche() {
        System.out.println("Je suis une instance avec les valeurs \n: x = " + this.x + " y = " + this.y + " nom = " + this.nom);
    }

    @Override
    public Object clone() throws CloneNotSupportedException {
        throw new CloneNotSupportedException();
    }
}

```

Figure 2: Image Du Code