CS261 LAB9

NAME: DIPEAN DASGUPTA **ID:202151188**

SECTION: 2B **DATE: 16/11/22**

TASK 1: WRITE A PROGRAM TO DEMONSTRATE THE DIFFERENCE BETWEEN STATIC KEYWORD AND FINAL KEYWORD.

Key Difference between Static and final keyword in OOP are:

Static	Final
1. It's variable can be re-initialized or over-ridden.	1.Its value, once declared, can't be changed or re-initialized or over-ridden.
2. Static method can be called only by other static methods.	2. Final methods can't be inherited by any class.
3. It is necessary to initialize the static variable at the time of its declaration.	3. It is not necessary to initialize the final variable at the time of its declaration.

Solution Code:

Code to demonstrate differences between static and final methods.

```
import java.util.*;

class P {
    public final void Name(){
        System.out.println(x: "Dipean Dasgupta");
    }

    public static  void Course(){
        System.out.println(x: "BTECH");
    }


    public final void Section(){
        System.out.println(x: "2B");
    }
}
```

Here class P contains Name() and Section which are final and Course is static.

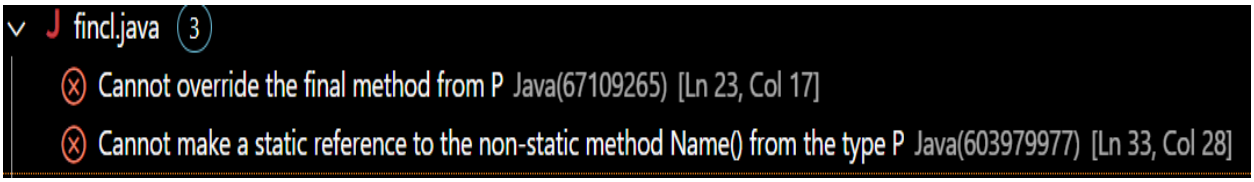
```java
class C extends P {

    // Method 1
    // Trying to override the last name
    public void Section(){
        // Display surname
        System.out.println(x: "2A");
    }

    // Method 2
    // Main driver method
    Run | Debug
    public static void main(String[] args)
    {

        System.out.println(Name() + " "+ "is a "+ Course()+"student");
    }
}
```

Here **Output** Shows the following Problems:



Here, we can see Final method can't be over-ridden. Besides, on the other side static reference or call can made by the other static methods. Non static methods can't refer or call static method.

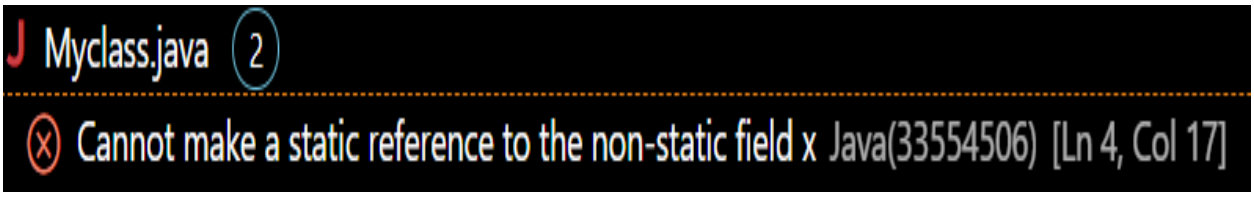
That is why there is red mark under "Name" function as it was final but "Course" function is working as it was static.

TASK 2: WILL THE FOLLOWING CODE SNIPPET COMPILE SUCCESSFULLY? IF YES, WHAT IS THE OUTPUT OF THE FOLLOWING PROGRAM?

Solution Code:

```
public class Myclass
{   private int x =
10;   static int m1()
{      int y = x;
return y;
   }
public static void main(String[] args) {
m1();
  }
}
```

OUTPUT:



Here x is not a static variable. But it was being called or referenced in a method which is static. As per rule, Static reference can be done only by static methods.

Output: Error.

Task 3: Is there any error in the below code snippet? If not, what will be the output of the following program?

```
public class Myclass {
private final int x = 10;
  void m1()
  {     final int x = 20;
System.out.println(x);
  }
public static void main(String[] args) {
Myclass obj = new Myclass();
obj.m1();
  }
}
```

OUTPUT:

Output is compilation error. It is because the code tries to declare the final variable again in method m1 which is not valid. As per rules, final variable once assigned can never be changed.