

**REPORT MANUAL OF JOBSHEET**

**Practicum, Tasks and Questions**

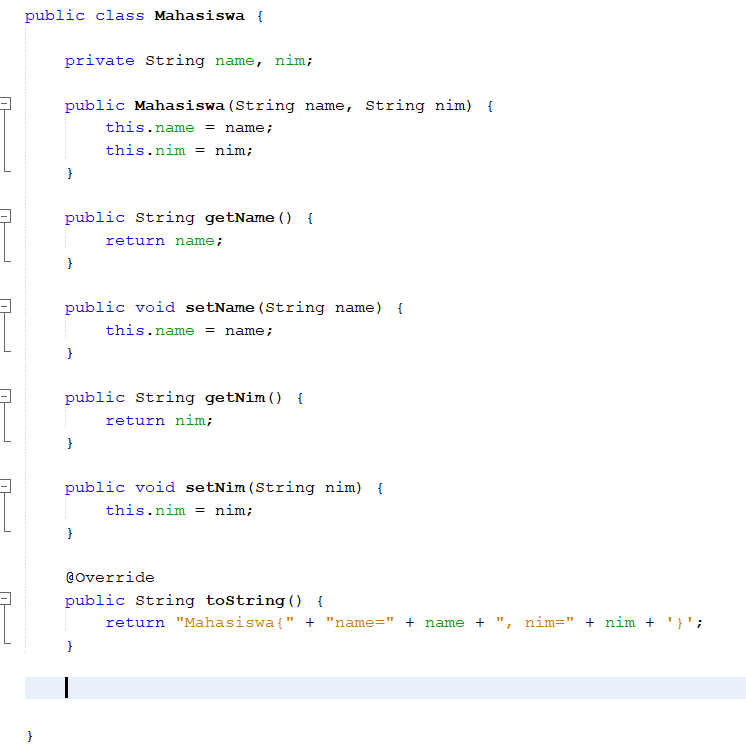
**(Pemograman jaringan)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Name** | **:** | **Brian Sayudha** |  |
|  | **Class / NIM** | **:** | **3G / 1841720158** |  |
|  | **Major** | **:** | **D-IV Informatics Enginering** |  |
|  |  |  |  |  |
|  |  |  |  |  |

Praktikum 1

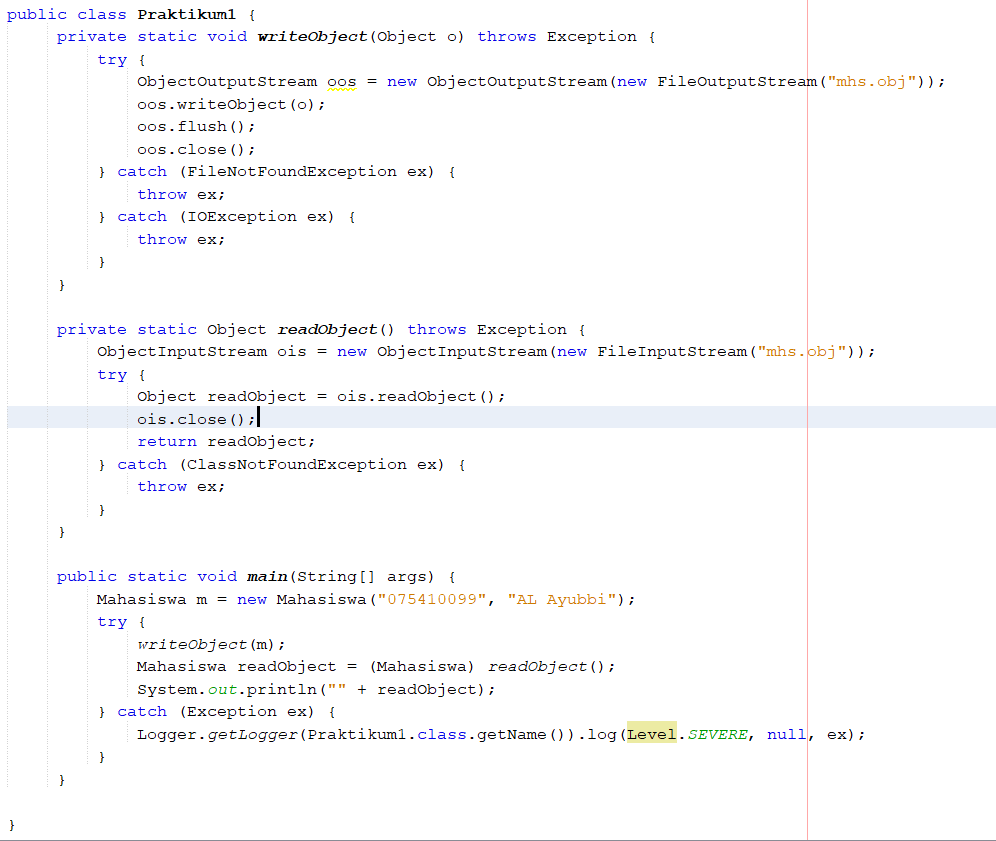
Class Mahasiswa

Code :



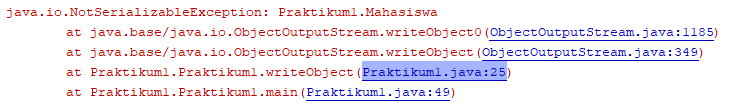
Class Praktikum1

Code :

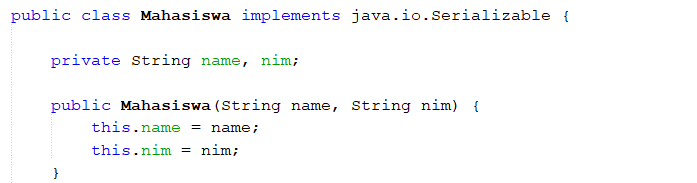


Result

(it will cause error because don’t implements the serialization)



Add implements java.io.Serializable at Mahasiswa Class



Result



Question

1. Why before when your program was run error?

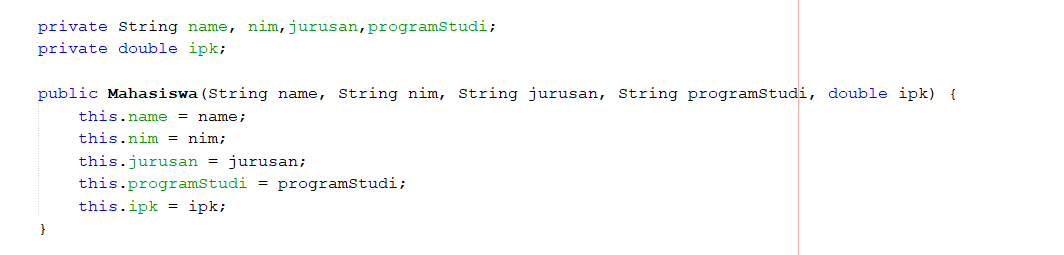
Answer : Because we didn’t call the java.io.Serializable at our model Class, Serializable is used to read and write the object .

2. What is the function of adding serializable implements?

Answer : the serializale implements is a java interface that make our class to support object serialization.

3. Add the attributes of the department, study program, and IPK to the Student class

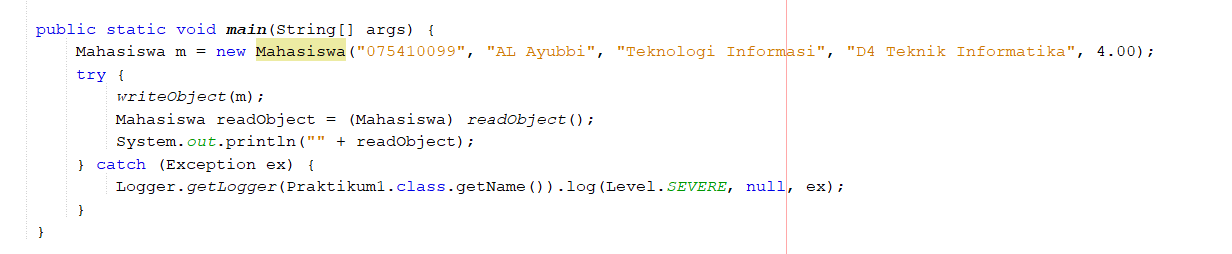
Mahasiswa Class Code





Praktikum1 at main method

Code



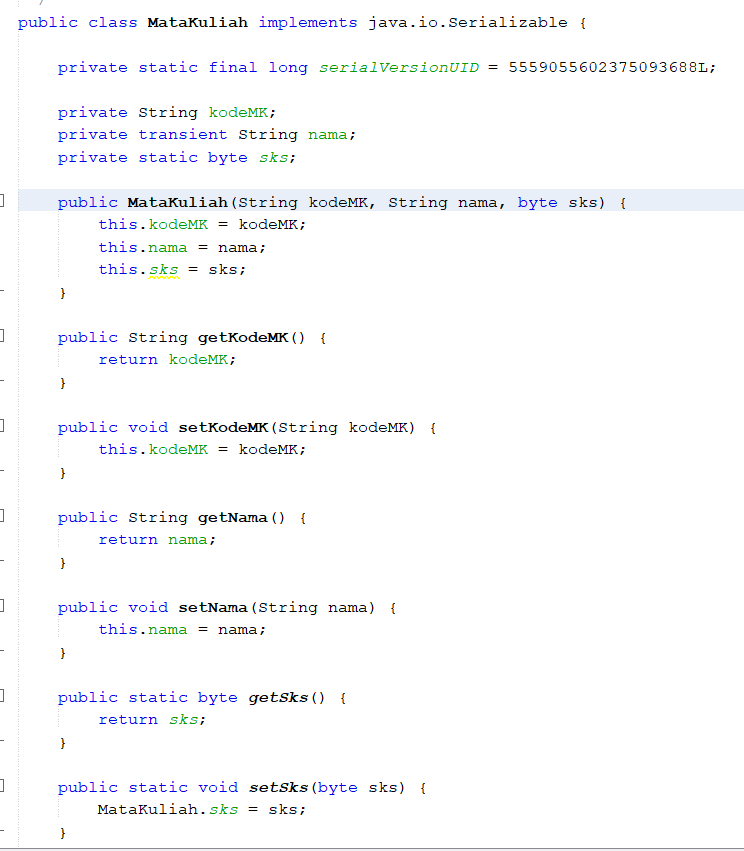
Result

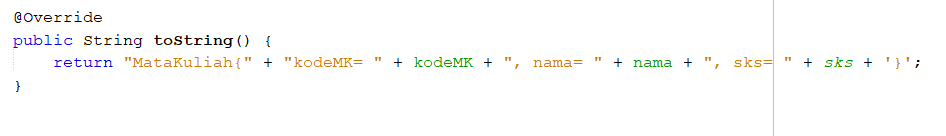


Praktikum 2

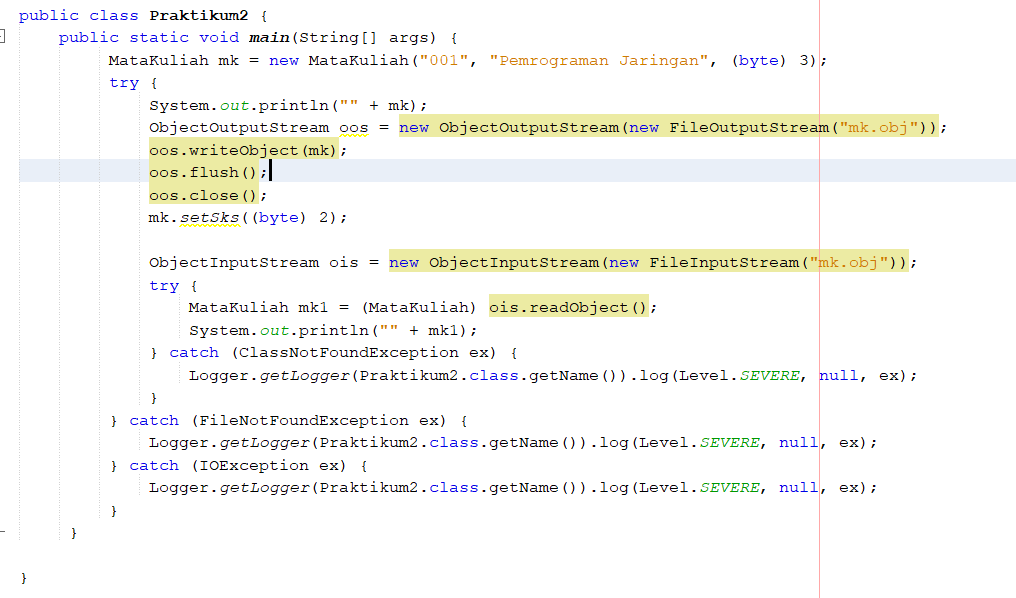
MataKuliah Class

Code

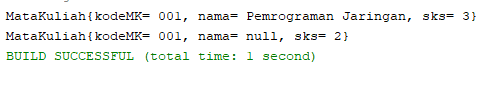




Praktikum2 Class



Result



Question

1. After running the above code, what is the difference between transient and static modifiers in the class that implements Serialize?

Answer :

Transient = The data will not serialized, so the value become null or 0.

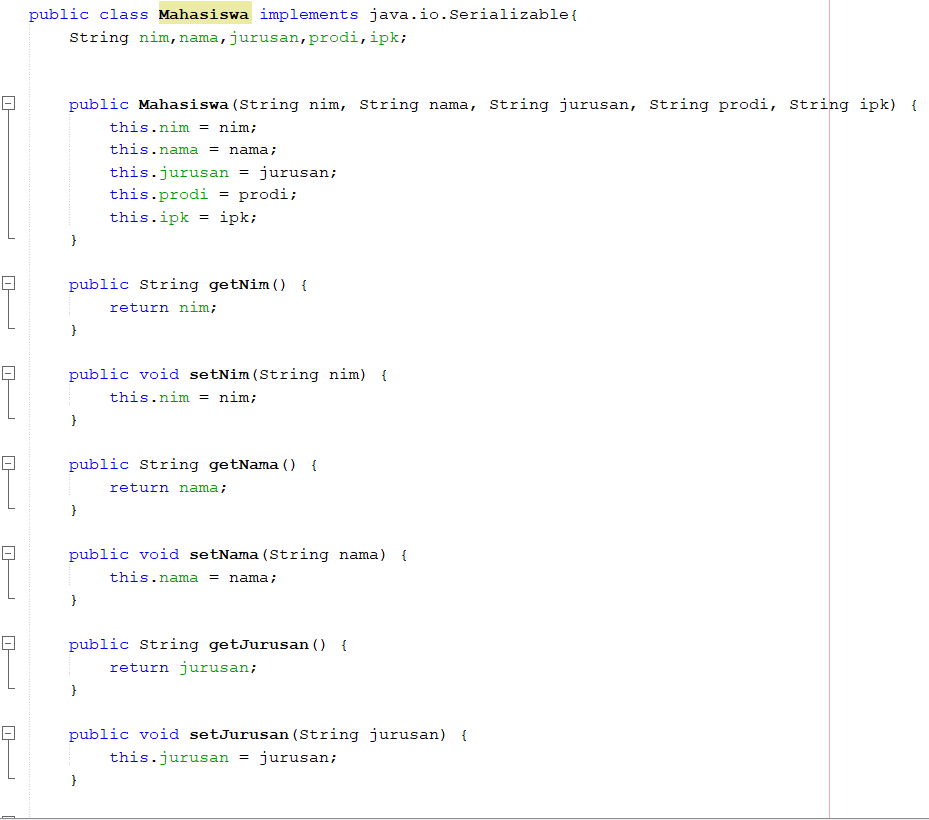
Static = this attribute when serialized or object being writed it can be changed back to the attribute (can be inputted new value).

2. What can you conclude with practicum 1 and practicum 2?

Answer : In practicum 1 we can do read and write the object serialization into our output and then in the practicum 2 we know that there are attribute to be used in serialization. We can add some attributes for the data that we don’t want to serialized.

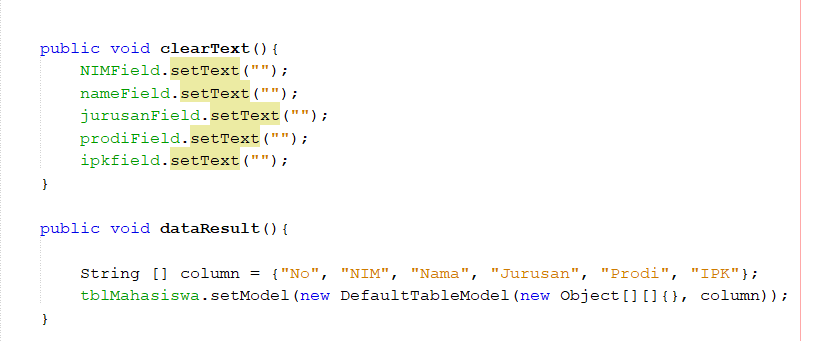
Tugas

Mahasiswa Class

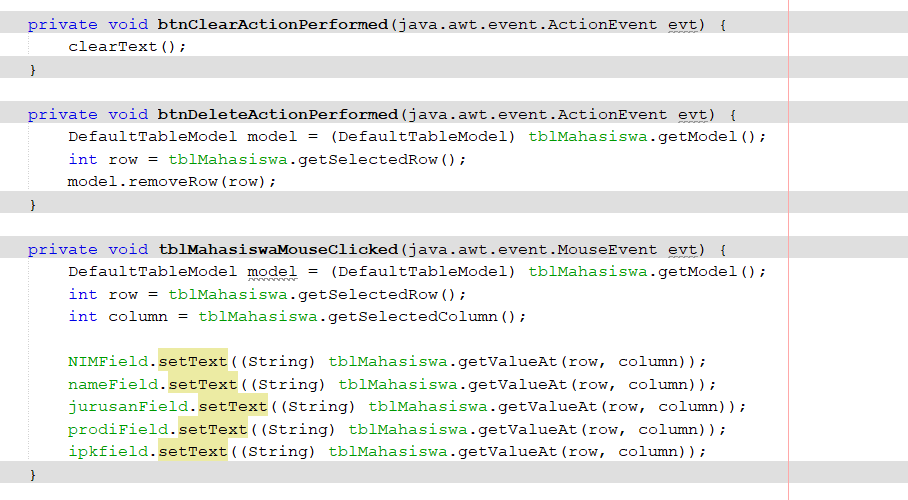


Frontend Class

Code







Design

