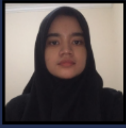


# **TUGAS PENDAHULUAN**

## **MODUL 3**

Penanganan Eksepsi dan  
Multithreading

# ASISTEN



**ADIS**



**AFHM**



**ARTI**



**CACA**



**DAPA**



**DEYA**



**DIKA**



**DRFZ**



**FATH**



**FDIL**



**FRQI**



**HILM**



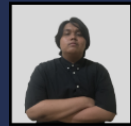
**NICA**



**OLLA**



**RAIN**



**SKUY**



**TITY**



**WANN**



**WISH**



**XASH**

## Peraturan Studi Kasus

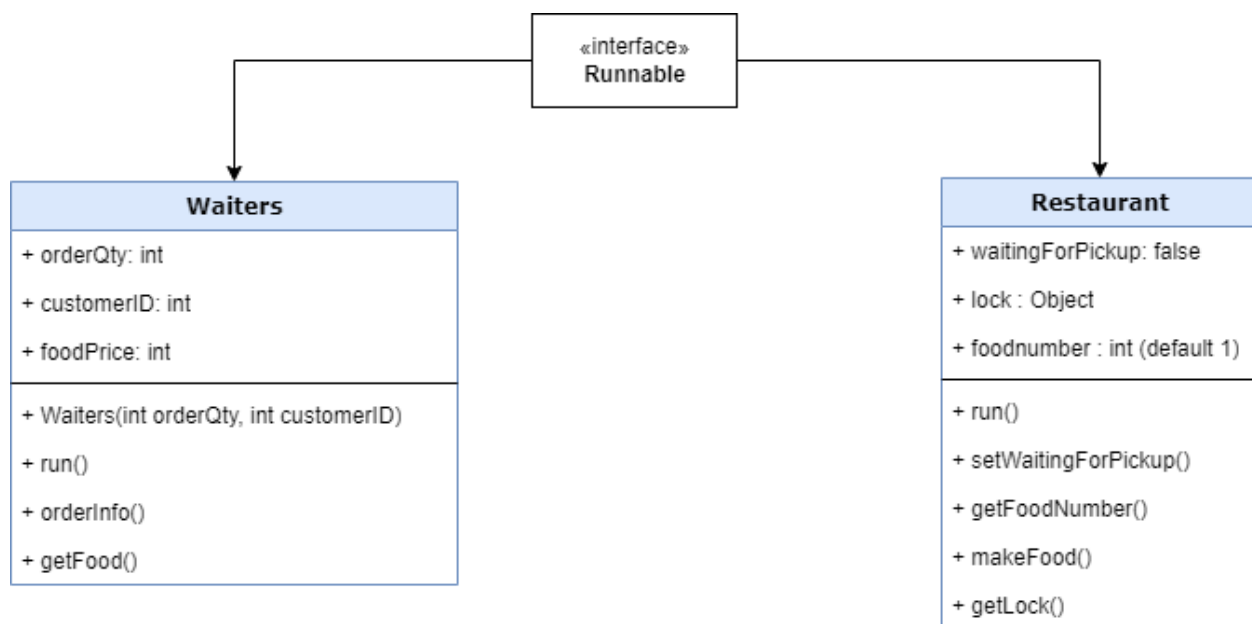
1. Studi Kasus dikerjakan secara individu.
2. Jawaban tidak boleh sama dengan setiap individu.
3. Hasil pengerjaan di push ke repository github tiap individu
4. Submit hasil pengerjaan berupa file format PDF dan link github ke LMS tiap individu
5. Format Pengumpulan Hasil Pengerjaan
  - Format nama project/folder (penamaan project/folder yang di push ke repository github) :  
MODULX\_NAMA PENDEK  
Contoh :  
MODUL2\_ADHISTY
  - Format nama file PDF (Dikumpulkan di LMS dalam format .pdf yang berisi screenshot source code dan hasil output dengan format penamaan file) :  
OOP\_KODEASISTEN\_NAMAPENDEK\_NIM\_SSMODULX  
Contoh :  
OOP\_ADIS\_ADHISTY\_1202204087\_SSMODUL2
6. Salah penamaan pada file pengerjaan nilai modul akan dipotong sebesar 10%
7. Terlambat mengumpulkan file pengerjaan nilai modul akan dipotong sebesar 20%
8. Segala alat komunikasi dikumpulkan di loker yang tersedia dalam ruangan.

## Tugas Pendahuluan Modul 3

ESD Restaurant adalah restoran cabang ke-100 dari ESD Laboratory, Sebagai Programmer Handal Buatlah sebuah program java dengan menggunakan konsep Threading untuk restoran ESD agar semakin mantap.

Buatlah program java dengan Class **Main** sebagai program utama beserta 2 Class dengan rincian di bawah menggunakan konsep Threading dan Error Handling

Terdapat 2 Class, yaitu **Restaurant**, **Waiters**. Berikut adalah rincian dari keduaClass tersebut:



## Class Restaurant :

Implementasikan lah interface **Runnable** pada class Restaurant

- Terdapat atribut private waitingForPickup dengan tipe data boolean dan nilai defaultnya adalah false;
- Terdapat atribut private static final lock dengan tipe Object hint: **private static final Object lock = new Object();**
- Terdapat atribut private static foodNumber dengan tipe data int dan nilai defaultnya adalah 1
- Terdapat method public void run() yaitu @Override dari interface Runnable Lalu isi lah dengan perulangan while pada class ini dan panggil method makeFood(), lalu buatlah try-catch dengan mengisi Thread.sleep pada bagian try dan catch isilah dengan default error message.
- Terdapat Getter/Setter yaitu setWaitingForPickup(), getFoodNumber(), getLock()
- Terdapat method public void makeFood(). Pada class ini buatlah logika Threading dengan menggunakan synchronized(lock)

## Class Waiters:

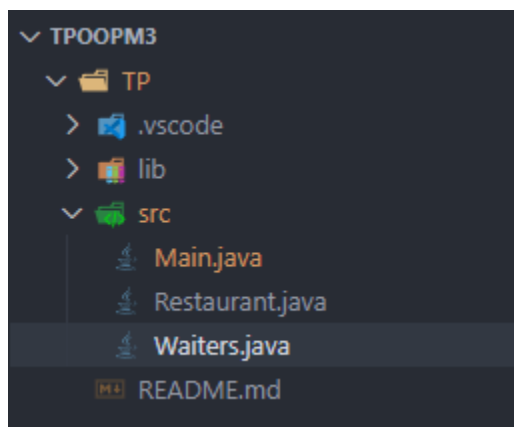
- Terdapat atribut private final orderQty dengan tipe data int
- Terdapat atribut private final customerID dengan tipe data int

- Terdapat atribut static foodPrice dengan tipe data int dengan nilai default =25000 ( kamu bebas membuat harga food)
- Terdapat 1 Constructor Waiters(int orderQty, int customerID)
- Terdapat method public void run() yaitu @Override dari interface Runnable Lalu isi lah dengan perulangan while pada class ini dan panggil method getFood(), lalu buatlah try-catch dengan mengisi Thread.sleep pada bagian try dan catch isilah dengan default error message.
- Terdapat method public void orderInfo() yaitu print informasi Customer ID, Number Of Food, Total Price (orderQty + foodPrice)
- Terdapat method public void makeFood(). Pada class ini buatlah logika Threading dengan menggunakan synchronized(lock)

## Class Main:

- Pada class Main kita diminta untuk menerima inputan user

## Contoh Struktur File :



Contoh Output Ketika program dijalankan:

```
Enter Customer ID: 303
Enter how much food to made: 2
Restaurant: Making Food Number 1
Restaurant: Telling the waiter to get the coffee

Waiter: Food is ready to deliver
Waiter: Tell the restaurant to make another food

Restaurant: Waiting for the Waiter to deliver the food
Waiter: Food is ready to deliver
Waiter: Tell the restaurant to make another food

Restaurant: Making Food Number 2
Restaurant: Telling the waiter to get the coffee

Restaurant: Waiting for the Waiter to deliver the food
Waiter: Food is ready to deliver
=====
Customer ID: 303
Number of Food: 2
Total Price: 50000
=====
```

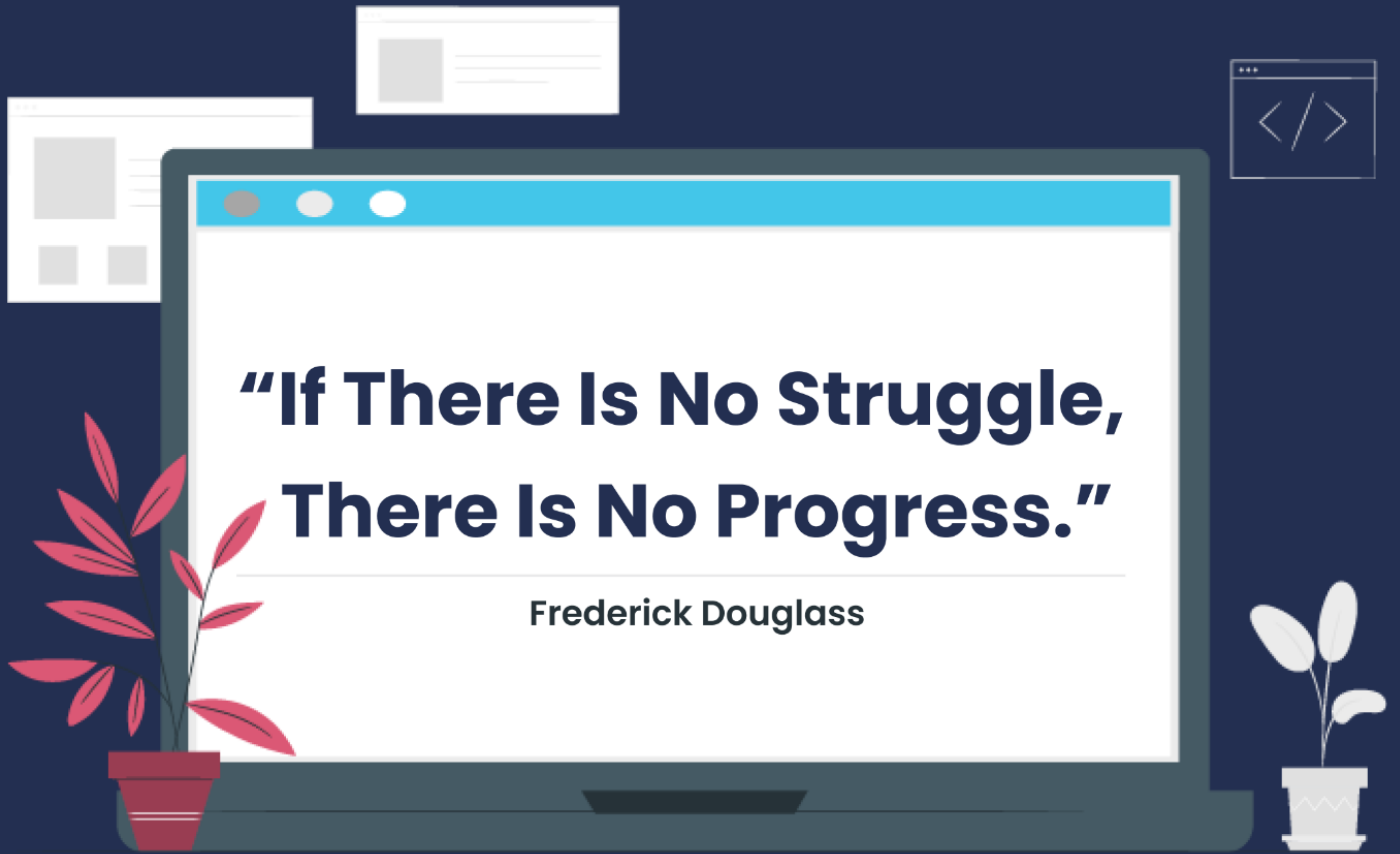
If Wrong Input:

```
Enter Customer ID: one
Input must be Integer
```

\*Output ini hanya contoh, boleh dikreasikan

Karena ini modul terakhir kalian bisa melihat github <https://github.com/faruqii> ^\_^

Semangat Menyelam guys. Usahakan Mengerti ya dengan codingannya



Find us on :

 @eadlaboratory

 @XXX