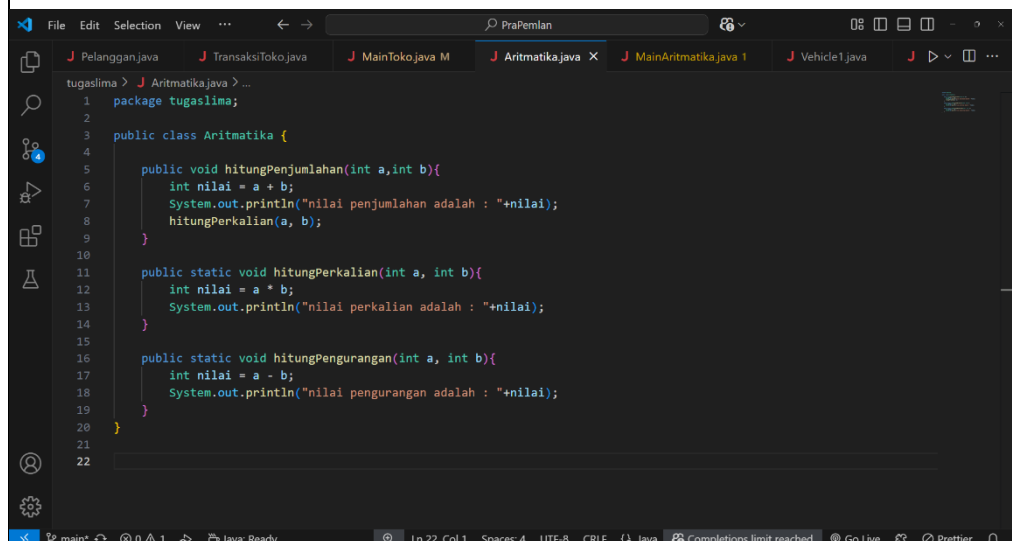


NAMA : Sefina Ayudia Syauqi  
NIM : 245150701111023  
KELAS : TI-D  
TUGAS : 5

#### A. Static Method

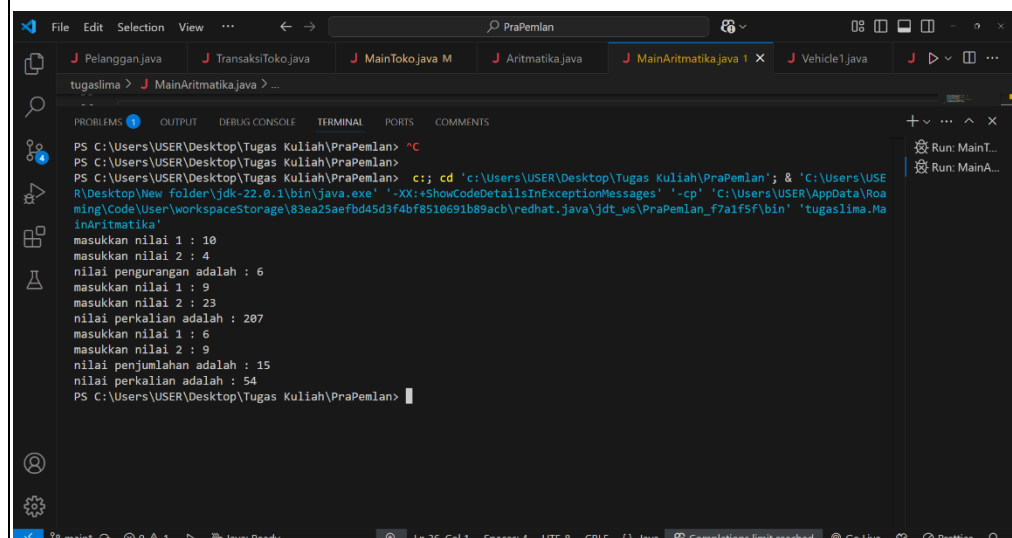
1. Apakah yang disebut dengan *static variable*? Dan apa fungsi dari *static variable* serta kapan kita dapat menggunakan *static variabel*?
  - *Static variable* adalah *variable* yang dideklarasikan dengan kata kunci *static* didalam *class* diluar *method*, *variable* ini bukan milik objek atau *instance* tetapi milik *class*.
  - Fungsi :
    1. Dapat diakses tanpa membuat objek terlebih dahulu.
    2. Penghematan waktu.
    3. Penghematan memori.
  - Digunakan ketika membutuhkan *variable* yang nilainya sama untuk semua objek dari suatu kelas. Selain itu juga dapat digunakan ketika ingin membuat counter.
2. Mengapa pada main method harus dituliskan *static*? Jelaskan jawaban anda beserta dengan alasan!
  - Karena *method* ini adalah titik awal eksekusi program, dengan demikian memungkinkan JVM (*Java Virtual Machine*) untuk memanggil tanpa harus membuat *instance* dari kelas yang berisi method *main()*.
3. Lakukan percobaan diatas dan benahi jika menemukan kesalahan!
  - Tidak ada kesalahan kode yang menyebabkan error.
4. Jika pada tubuh method hitungPenjumlahan ditambahkan syntax *hitungPerkalian* (a,b) apa yang terjadi? Jelaskan?
  - Tidak ada masalah atau error pada program, karena *method non static* dapat mengakses *member static* atau *method non static* dapat memanggil *method static* secara langsung. *Method static* dapat dipanggil di mana saja tanpa *instance*.

## Kode



```
tugaslima > J Aritmatika.java > ...
1 package tugaslima;
2
3 public class Aritmatika {
4
5     public void hitungPenjumlahan(int a,int b){
6         int nilai = a + b;
7         System.out.println("nilai penjumlahan adalah : "+nilai);
8         hitungPerkalian(a, b);
9     }
10
11     public static void hitungPerkalian(int a, int b){
12         int nilai = a * b;
13         System.out.println("nilai perkalian adalah : "+nilai);
14     }
15
16     public static void hitungPengurangan(int a, int b){
17         int nilai = a - b;
18         System.out.println("nilai pengurangan adalah : "+nilai);
19     }
20 }
21
22
```

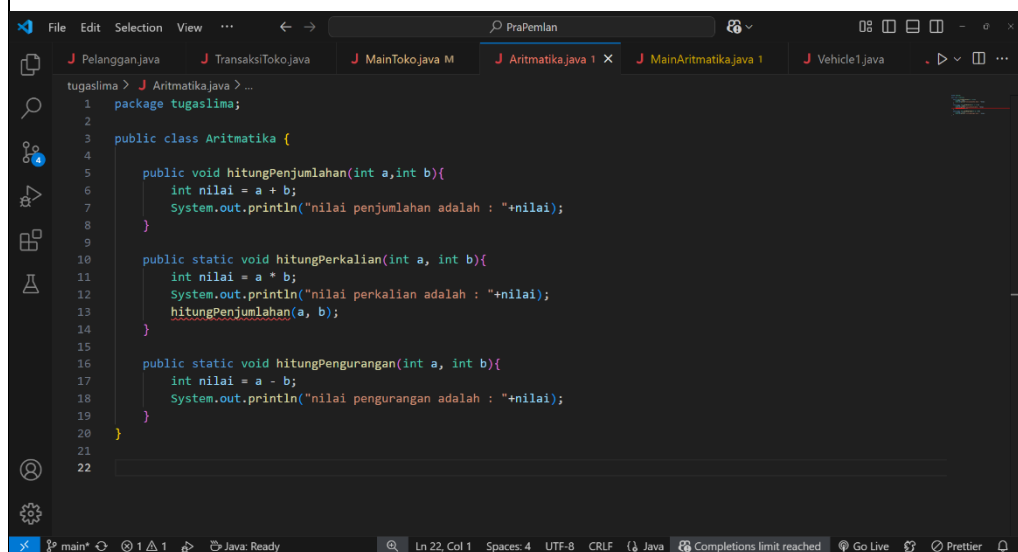
## Hasil



```
PS C:\Users\USER\Desktop\Tugas Kuliah\PraPemlan> ^C
PS C:\Users\USER\Desktop\Tugas Kuliah\PraPemlan>
PS C:\Users\USER\Desktop\Tugas Kuliah\PraPemlan> cd 'c:\Users\USER\Desktop\Tugas Kuliah\PraPemlan'; & 'C:\Users\USER\Desktop\New folder\jdk-22.0.1\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'c:\Users\USER\AppData\Roaming\Code\User\workspaceStorage\83ea25aefbd45d3f4bf8510691b89acb\redhat.java\jdt_ws\PraPemlan_f7a1f5f\bin' 'tugaslima.MainAritmatika'
masukkan nilai 1 : 10
masukkan nilai 2 : 4
nilai pengurangan adalah : 6
masukkan nilai 1 : 9
masukkan nilai 2 : 23
nilai perkalian adalah : 207
masukkan nilai 1 : 6
masukkan nilai 2 : 9
nilai penjumlahan adalah : 15
nilai perkalian adalah : 54
PS C:\Users\USER\Desktop\Tugas Kuliah\PraPemlan>
```

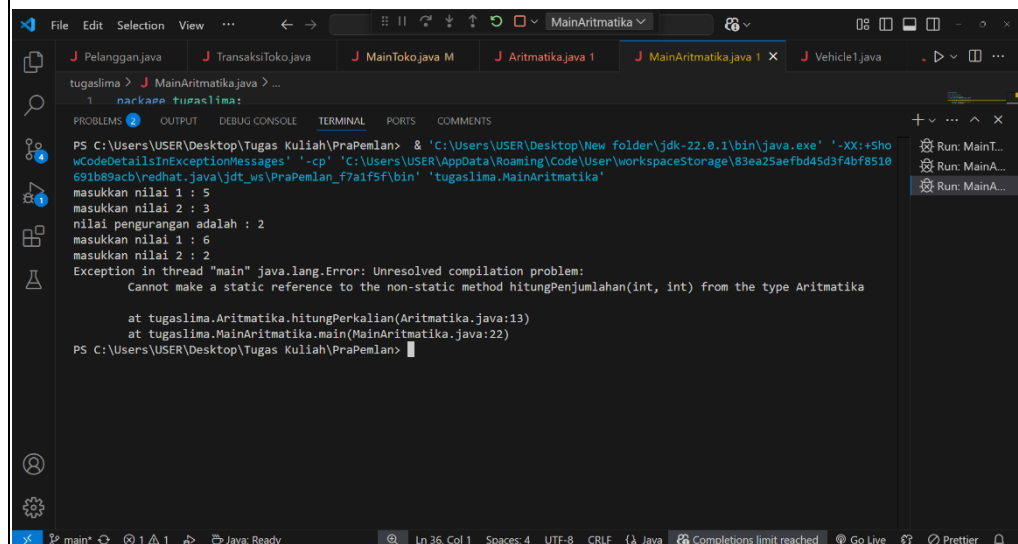
5. Jika pada tubuh method `hitungPerkalian` ditambahkan syntax `hitungPenjumlahan(a,b)` apa yang terjadi? Jelaskan?
- Akan terjadi eror, karena *method static* memerlukan objek untuk mengakses *member non static* atau *method static* tidak bisa memanggil *method non static* secara langsung tanpa *instance*. Sehingga diperlukan untuk membuat instance sebelum memanggil method non static. Contoh :  
`aritmatika.hitungPenjumlahan(a, b);`

## Kode



```
1 package tugaslima;
2
3 public class Aritmatika {
4
5     public void hitungPenjumlahan(int a,int b){
6         int nilai = a + b;
7         System.out.println("nilai penjumlahan adalah : "+nilai);
8     }
9
10    public static void hitungPerkalian(int a, int b){
11        int nilai = a * b;
12        System.out.println("nilai perkalian adalah : "+nilai);
13        hitungPenjumlahan(a, b);
14    }
15
16    public static void hitungPengurangan(int a, int b){
17        int nilai = a - b;
18        System.out.println("nilai pengurangan adalah : "+nilai);
19    }
20 }
21
22
```

## Hasil



```
PS C:\Users\USER\Desktop\Tugas Kuliah\PraPemlan> & 'C:\Users\USER\Desktop\New folder\jdk-22.0.1\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\USER\AppData\Roaming\Code\User\workspaceStorage\83ea25aefbd45d3f4bf8510691b89acb\redhat.java\jdt_ws\PraPemlan_f7a1f5f\bin' 'tugaslima.MainAritmatika'
masukkan nilai 1 : 5
masukkan nilai 2 : 3
nilai pengurangan adalah : 2
masukkan nilai 1 : 6
masukkan nilai 2 : 2
Exception in thread "main" java.lang.Error: Unresolved compilation problem:
    Cannot make a static reference to the non-static method hitungPenjumlahan(int, int) from the type Aritmatika
    at tugaslima.Aritmatika.hitungPerkalian(Aritmatika.java:13)
    at tugaslima.MainAritmatika.main(MainAritmatika.java:22)
PS C:\Users\USER\Desktop\Tugas Kuliah\PraPemlan>
```

6. Tambahkan *method non static* dengan nilai balikan *double* untuk menghitung pembagian dengan parameter String nil dan String nil2, dan panggil method tersebut pada method main

## Kode

```

File Edit Selection View Go Run ... ← → PraPeman
J Pelanggan.java J TransaksiToko.java J MainToko.java M J Aritmatika.java X J MainAritmatika.java J Vehicle1.java 1 J TestVehicle1.java
tugaslima > J Aritmatika.java > ...
1 package tugaslima;
2
3 public class Aritmatika {
4
5     public void hitungPenjumlahan(int a,int b){
6         int nilai = a + b;
7         System.out.println("nilai penjumlahan adalah : "+nilai);
8     }
9
10    public static void hitungPerkalian(int a, int b){
11        int nilai = a * b;
12        System.out.println("nilai perkalian adalah : "+nilai);
13    }
14
15    public static void hitungPengurangan(int a, int b){
16        int nilai = a - b;
17        System.out.println("nilai pengurangan adalah : "+nilai);
18    }
19
20    public double hitungPembagian(String nilai1, String nilai2){
21        double a = Double.parseDouble(nilai1);
22        double b = Double.parseDouble(nilai2);
23
24        if(b == 0){
25            System.out.println("Tidak bisa membagi dengan 0");
26            return 0;
27        }
28        return a / b;
29    }
30
31 }
32
main 0 0 A 1 Java Ready
Ln 32, Col 1 Spaces: 4 UTF-8 CRLF {} Java Go Live Preflight

```

```

File Edit Selection View Go Run Terminal Help ← → PraPeman
tugaslima > J MainAritmatika.java > ...
1 package tugaslima;
2 import java.util.Scanner;
3
4 public class MainAritmatika {
5     Run(Demo)
6     public static void main(String[] args) {
7
8         Scanner in = new Scanner(System.in);
9
10        System.out.print("masukkan nilai 1 : ");
11        int nil1 = in.nextInt();
12        System.out.print("masukkan nilai 2 : ");
13        int nil2 = in.nextInt();
14        //menganggil method static
15        Aritmatika.hitungPengurangan(nil1, nil2);
16
17        System.out.print("masukkan nilai 1 : ");
18        nil1 = in.nextInt();
19        System.out.print("masukkan nilai 2 : ");
20        nil2 = in.nextInt();
21        //menganggil method static
22        Aritmatika.hitungPerkalian(nil1, nil2);
23
24        System.out.print("masukkan nilai 1 : ");
25        int value1 = in.nextInt();
26        System.out.print("masukkan nilai 2 : ");
27        int value2 = in.nextInt();
28        //menganggil method objek harus melalui objek
29        Aritmatika a = new Aritmatika();
30        a.hitungPenjumlahan(value1, value2);
31
32        System.out.print("masukkan nilai 1 : ");
33        String nilai1 = in.next();
34        System.out.print("masukkan nilai 2 : ");
35        String nilai2 = in.next();
36
37        Aritmatika aritmatika = new Aritmatika();
38        double nilai = aritmatika.hitungPembagian(nilai1, nilai2);
39        System.out.println("nilai pembagian adalah : "+ nilai);
40
41        in.close();
42    }
43
44 }
45
main 0 0 A 1 Java Ready
Ln 45, Col 1 Spaces: 4 UTF-8 CRLF {} Java Go Live Preflight

```

## Hasil

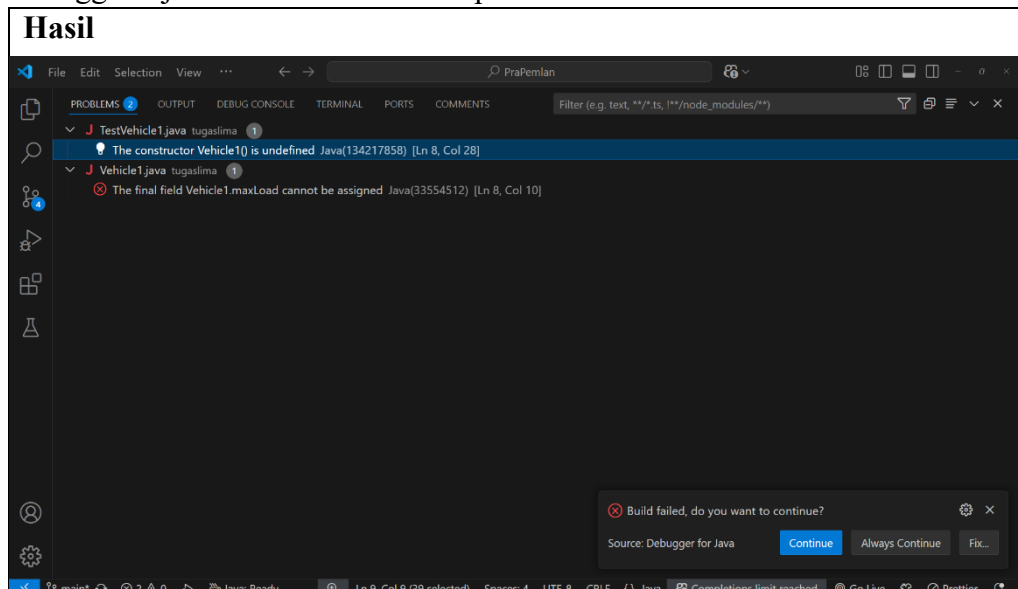
```

File Edit Selection View Go Run ... ← → MainAritmatika
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS COMMENTS
masukkan nilai 2 : 3
nilai pengurangan adalah : 3
masukkan nilai 1 : 9
masukkan nilai 2 : 8
nilai perkalian adalah : 72
nilai penjumlahan adalah : 12
masukkan nilai 1: 10
nilai penjumlahan adalah : 12
masukkan nilai 1: 10
masukkan nilai 2: 5
nilai pembagian adalah: 2.0
PS C:\Users\USER\Desktop\Tugas Kuliah\PraPeman> ^C
PS C:\Users\USER\Desktop\Tugas Kuliah\PraPeman>
PS C:\Users\USER\Desktop\Tugas Kuliah\PraPeman> cd 'c:\Users\USER\Desktop\Tugas Kuliah\PraPeman'; & 'C:\Users\USER\Desktop\New folder\jdk-22.0.1\bin\java.exe' -XX:+ShowCodeDetailsInExceptionMessages -cp 'C:\Users\USER\AppData\Roaming\Code\User\workspaceStorage\83ea25aefbd45d3f4bf8510691b89ac3\redhat_java\jdk_ws\PraPeman_f7a1f5f\bin' 'tugaslima.MainAritmatika'
masukkan nilai 1 : 9
masukkan nilai 2 : 5
nilai pengurangan adalah : 4
masukkan nilai 1 : 4
masukkan nilai 2 : 8
nilai perkalian adalah : 32
masukkan nilai 1 : 4
masukkan nilai 2 : 9
nilai penjumlahan adalah : 13
masukkan nilai 1: 9
masukkan nilai 2: 4
nilai pembagian adalah: 2.25
PS C:\Users\USER\Desktop\Tugas Kuliah\PraPeman> ^C
PS C:\Users\USER\Desktop\Tugas Kuliah\PraPeman>
PS C:\Users\USER\Desktop\Tugas Kuliah\PraPeman> cd 'c:\Users\USER\Desktop\Tugas Kuliah\PraPeman'; & 'C:\Users\USER\Desktop\New folder\jdk-22.0.1\bin\java.exe' -XX:+ShowCodeDetailsInExceptionMessages -cp 'C:\Users\USER\AppData\Roaming\Code\User\workspaceStorage\83ea25aefbd45d3f4bf8510691b89ac3\redhat_java\jdk_ws\Pr

```

## B. Konstanta Final

1. Benahi kode `Vehicle1` dan `TestVehicle1` dan perbaiki jika menemui kesalahan!
  - Tidak ada kesalahan kode yang menyebabkan eror.
2. Hapus separator “/” pada file `Vehicle1.java` pada baris 4-6 serta pada file `TestVehicle1.java` pada baris 6, apa yang terjadi dan jelaskan!
  - Pada kelas `vehicle1` akan terjadi eror pada pemrograman. Eror disebabkan karena `maxLoad` dideklarasikan sebagai *final* yang dimana `maxLoad` hanya bisa diinisialisasi sekali saat deklarasi atau saat di konstruktor tidak dapat diinisialisasi dikedunya, sedangkan konstruktor `Vehicle1` mencoba mengubah `maxLoad` dengan `this.maxLoad = max;`.
  - Pada kelas `TestVehicle1` akan terjadi eror pada pemrograman karena pada kelas `Vehicle1` tidak ada konstruktor yang memiliki atau menerima parameter sehingga terjadi kesalahan atau eror pada kelas `TestVehicle1`.



3. Pada file `Vehicle1.java` variable `load` ubah menjadi konstanta *final*, apa yang terjadi, jelaskan!
  - Kelas `Vehicle1` akan eror karena `load` dideskripsikan secara *final* tidak bisa diubah lagi setelah diinisialisasi, sedangkan pada konstruktor `addBox` mencoba untuk mengubah `load`. Seharusnya atribut `load` tidak dalam *final* karena `load` seharusnya bersifat dinamis menyesuaikan penambahan `box`.

# Hasil

```
PS C:\Users\USER\Desktop\Tugas Kuliah\PraPeman> & 'C:\Users\USER\Desktop\New folder\jdk-22.0.1\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\USER\AppData\Roaming\Code\User\workspaceStorage\83ea25aefbd45d3f4b78510691b89acb\redhat.java\jdt_ws\PraPeman_f7a1f5f\bin' 'tugaslima.TestVehicle1'
Creating a vehicle with a 10,000 kg maximumload.
Add box #1 (500kg) : true
Add box #2 (2650kg) : true
Add box #3 (5000kg) : true
Add box #4 (4000kg) : true
Add box #5 (300kg) : false
Vehicle load is 9750.0 kg
PS C:\Users\USER\Desktop\Tugas Kuliah\PraPeman> ^C
PS C:\Users\USER\Desktop\Tugas Kuliah\PraPeman>
PS C:\Users\USER\Desktop\Tugas Kuliah\PraPeman> cd 'C:\Users\USER\Desktop\Tugas Kuliah\PraPeman'; & 'C:\Users\USER\Desktop\New folder\jdk-22.0.1\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\USER\AppData\Roaming\Code\User\workspaceStorage\83ea25aefbd45d3f4b78510691b89acb\redhat.java\jdt_ws\PraPeman_f7a1f5f\bin' 'tugaslima.TestVehicle1'
Creating a vehicle with a 10,000 kg maximumload.
Exception in thread "main" java.lang.Error: Unresolved compilation problems:
  The blank final field load may not have been initialized
  The final field Vehicle1.load cannot be assigned

    at tugaslima.Vehicle1.<init>(Vehicle1.java:4)
    at tugaslima.TestVehicle1.main(TestVehicle1.java:8)
PS C:\Users\USER\Desktop\Tugas Kuliah\PraPeman>
```

Build failed, do you want to continue?

Source: Debugger for Java

Continue Always Continue Fix...

4. Tambahkan keyword “static” pada file Vehicle1.java variabel `maxLoad`, apa yang terjadi dan jelaskan!
  - Tidak akan terjadi error, karena *variable maxLoad* memiliki nilai awal 10000, tetapi *maxLoad* tidak lagi bersifat dinamis dan menjadi atribut kelas bukan objek yang memiliki nilai sama untuk semua objek *Vehicle1*. Sehingga semua kendaraan memiliki kapasitas *maximum* yang sama.