

# Laporan Praktikum Dasar Pemrograman



Nama: Rizqi Bagus Andrean

Kelas: TI-1D

Absen: 25

## Daftar Isi

<b>Jobsheet 1</b> .....	4
Percobaan 1 .....	4
Percobaan 2 .....	4
Percobaan 3 .....	4
Tugas .....	5
<b>Jobsheet 2</b> .....	6
Percobaan 1: Menggunakan Github .....	6
Percobaan 2: Dasar Kolaborasi Github .....	9
Percobaan 3: Menggunakan Trello .....	10
Tugas .....	13
<b>Jobsheet 3</b> .....	17
Percobaan 1: Penggunaan Variable .....	17
Percobaan 2: Penggunaan Tipe Data .....	18
Percobaan 3: Penggunaan Operator .....	18
Percobaan 4: Studi Kasus .....	18
Tugas Kelompok .....	19
<b>Jobsheet 4</b> .....	21
Percobaan 1 .....	21
Percobaan 2 .....	23
Percobaan 3 .....	24
<b>Jobsheet 5</b> .....	28
Percobaan 1 .....	28
Percobaan 2 .....	29
Percobaan 3 .....	30
Tugas .....	31
<b>Jobsheet 6</b> .....	34
Percobaan 1 .....	34
Percobaan 2 .....	35
Percobaan 3 .....	35
Tugas .....	36
<b>Jobsheet 7</b> .....	37
Percobaan 1 .....	37
Percobaan 2 .....	39
Percobaan 3 .....	41

<b>Tugas</b>	43
<b>Jobsheet 9</b>	44
<b>Percobaan 1</b>	44
<b>Percobaan 2</b>	46
<b>Percobaan 3</b>	48
<b>Percobaan 4 (Searching)</b>	51
<b>Tugas</b>	53
<b>Jobsheet 10</b>	55
<b>Percobaan 1</b>	55
<b>Percobaan 2</b>	63
<b>Percobaan 3</b>	67
<b>Tugas Kelompok</b>	68
<b>Jobsheet 11</b>	69
<b>Percobaan 1</b>	69
<b>Percobaan 2</b>	69
<b>Percobaan 3</b>	70
<b>Percobaan 4</b>	70
<b>Percobaan 5</b>	71
<b>Tugas Individu dan Kelompok</b>	73
<b>Jobsheet 13</b>	78
<b>Percobaan 1: Fungsi tanpa parameter</b>	78
<b>Percobaan 3: Membuat fungsi kembalian</b>	79
<b>Percobaan 4: Fungsi yang mengcalling fungsi lain</b>	79
<b>Percobaan 5: Fugsi varargs</b>	80
<b>Percobaan 6: Membuat program dengan fungsi vs tanpa fungsi</b>	81
<b>Tugas</b>	82
<b>Jobsheet 14</b>	85
<b>Percobaan 1</b>	85
<b>Percobaan 2</b>	85
<b>Percobaan 3</b>	86
<b>Tugas</b>	87

## Jobsheet 1

### Percobaan 1

1. Jelaskan apa kegunaan memasukkan lokasi folder bin dari Java ke dalam variabel PATH!  
Memasukkan folder bin ke dalam environment variable digunakan agar command seperti **javac**, **java** bisa dieksekusi secara global, misal menggunakan cmd ataupun terminal lainnya.
2. Jelaskan Kegunaan perintah javac ketika masuk di command prompt!  
Perintah **javac** digunakan untuk mengcompile kode sumber .java menjadi byte (akan membuat file baru yang berekstensi .class) dimana kode ini yang akan dieksekusi oleh JVM (Java Virtual Machine)

### Percobaan 2

1. Jelaskan apa kegunaan memasukkan lokasi folder bin dari Java ke dalam variabel PATH!  
Memasukkan folder bin ke dalam environment variable digunakan agar command seperti **javac**, **java** bisa dieksekusi secara global, misal menggunakan cmd ataupun terminal lainnya.
2. Jelaskan Kegunaan perintah javac ketika masuk di command prompt!  
Perintah **javac** digunakan untuk mengcompile kode sumber .java menjadi byte (akan membuat file baru yang berekstensi .class) dimana kode ini yang akan dieksekusi oleh JVM (Java Virtual Machine)

### Percobaan 3

1. Gantilah perintah **System.out.println** menjadi **System.out.print** pada baris ke 7 dan 8 kemudian jalankan program!
2. Apakah perbedaan **System.out.println(".....");** dan **System.out.print(".....");**, Jelaskan!

Perbedaan println dan print adalah jika kita mennggunakan println maka akan ditambahkan karakter new line setelah mencetak teks, sedangkan print tidak akan ditambahkan karakter new line.

3. Gantilah perintah pada baris ke 8 menjadi seperti berikut ini :

```
System.out.print("\nsaya adalah Mahasiswa Jurusan Teknologi Informasi -  
POLINEMA");
```

4. Jelaskan fungsi perintah \n pada baris ke 8!

Fungsi perintah \n adalah digunakan untuk pindah baris ke bawah

5. Pada baris ke 8 tambahkan karakter // sehingga baris tersebut menjadi

```
//System.out.print("\nsaya adalah Mahasiswa Prodi Teknik Informatika");
```

6. Jalankan program dan amati apakah yang terjadi! Jelaskan fungsi karakter //

Fungsi karakter // digunakan untuk komentar, sehingga kode yang diberi tanda komentar tidak akan dieksekusi.

### Tugas

Buatlah program java dengan hasil output sebagai berikut, kemudian jelaskan jalankan program yang telah anda buat!

```
***** IDENTITAS DIRI *****  
Nama : XXXXX  
Absen : XXXXX  
Kelas : XXXXX  
Prodi : XXXXX  
Jurusan : XXXXX
```

```
Tugas1.java > Tugas1 > main(String[])
1  public class Tugas1 {
2      Run | Debug
3      public static void main(String[] args) {
4          System.out.println("**** IDENTITAS DIRI ****");
5          System.out.println("Nama Lengkap\t\t: Rizqi Bagus Andrean");
6          System.out.println("Absen\t\t\t: 25");
7          System.out.println("Kelas\t\t\t: TI-1D");
8          System.out.println("Prodi\t\t\t: Teknik Informatika");
9          System.out.println("Jurusan\t\t\t: Teknologi Informasi");
10     }
11 }
```

```
11\b1n\java.exe' '-cp' 'C:\Users\Acer\AppData\Roaming
va\jdt_ws\jobsheet1_3ae0be6e\bin' 'Tugas1'
**** IDENTITAS DIRI ****
Nama Lengkap : Rizqi Bagus Andrean
Absen : 25
Kelas : TI-1D
Prodi : Teknik Informatika
Jurusan : Teknologi Informasi
PS C:\Users\Acer\Project\Daspro\jobsheet1>
```

Kode program diatas digunakan untuk mencetak identitas diri di terminal.

Class Tugas1 adalah nama kelasnya, dimana didalam kelas Tugas1 terdapat sebuah fungsi main yang mana fungsi main merupakan fungsi utama yang akan dijalankan saat program dimulai.

Didalam fungsi main terdapat perintah **System.out.println** yang digunakan untuk mencetak output teks, sedangkan di beberapa perintah tersebut ada yang menggunakan \t , dimana \t digunakan untuk membuat tab saat dicetak ke terminal.

## Jobsheet 2

### Percobaan 1: Menggunakan Github

1. Jelaskan perbedaan perintah **git commit** dan **git push**?

#### Git commit

Digunakan untuk menyimpan perubahan file atau kode yang dibuat ke local dan memberikannya pesan , misal git commit -m “ada perubahan kode”

### Git push

Digunakan untuk mempush atau mengirim file tadi ke repository online, misal github, gitlab

2. Apakah bisa alurnya dibalik, membuat folder atau projek terlebih dahulu kemudian upload (push) ke Github? Jika bisa, buktikan!

**Bisa**, ini saya sudah punya folder dari tugas dari jobsheet 1, namun folder ini belum ada di github

Name	Date modified	Type
Latihan25	05/09/2023 10:58	Java
MyFirstJava25.class	05/09/2023 10:36	CLASS
MyFirstJava25	05/09/2023 10:36	Java
Tugas1	05/09/2023 11:04	Java

- b. Buat repository githubnya dulu.

Repository template

No template ▾

Start your repository with a template repository's contents.

---

Owner \*      Repository name \*

 bagusok / Tugas Jobsheet 1

⚠ Your new repository will be created as **Tugas-Jobsheet-1**.  
The repository name can only contain ASCII letters, digits, and the characters ., -, and \_.

Great repository names are short and memorable. Need inspiration? How about [upgraded-funicular](#) ?

Description (optional)

Ini adalah tugas jobsheet 1 daspro

---

 Public  
Anyone on the internet can see this repository. You choose who can commit.

 Private  
You choose who can see and commit to this repository.

Initialize this repository with:

Add a README file  
This is where you can write a long description for your project. [Learn more about READMEs](#).

Add .gitignore

...  
...

b. lalu buka folder jobsheet tadi dan ketik **git init**

```
PS C:\Users\Acer\Project\Daspro\jobsheet1> git init
Initialized empty Git repository in C:/Users/Acer/Project/Daspro/jobsheet1/.git/
```

c. tambahkan semua file yang mau di commit dengan perintah **git add .**

d. lalu commit perubahannya dengan **git commit -m "first commit"**

```
PS C:\Users\Acer\Project\Daspro\jobsheet1> git commit
[main (root-commit) 220779c] first commit
4 files changed, 21 insertions(+)
create mode 100644 Latihan25.java
create mode 100644 MyFirstJava25.class
create mode 100644 MyFirstJava25.java
create mode 100644 Tugas1.java
PS C:\Users\Acer\Project\Daspro\jobsheet1>
```

f. lalu buat branch baru Bernama **main** dengan perintah **git branch main**

g. lalu atur remote repositorynya dengan perintah **git remote add origin <https://github.com/bagusok/Tugas-Jobsheet-1.git>**

```
PS C:\Users\Acer\Project\Daspro\jobsheet1> git remote add origin https://github.com/bagusok/Tugas-Jobsheet-1.git
```

h. Lalu push kode nya ke github dengan perintah **git push -u origin main**

```
PS C:\Users\Acer\Project\Daspro\jobsheet1> git push -u origin main
Enumerating objects: 6, done.
Counting objects: 100% (6/6), done.
Delta compression using up to 8 threads
Compressing objects: 100% (6/6), done.
Writing objects: 100% (6/6), 1.04 KiB | 1.04 MiB/s, done.
Total 6 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), done.
To https://github.com/bagusok/Tugas-Jobsheet-1.git
 * [new branch]      main -> main
branch 'main' set up to track 'origin/main'.
```

i. Lalu buka githubnya lagi dan cek, tugas jobsheet 1 sudah terupload di github

The screenshot shows a GitHub repository named 'Tugas-Jobsheet-1'. The repository has 1 branch ('main') and 0 tags. The 'Code' tab is selected. The commit history shows four commits from user 'bagusok': 'Latihan25.java' (first commit, 15 minutes ago), 'MyFirstJava25.class' (first commit, 15 minutes ago), 'MyFirstJava25.java' (first commit, 15 minutes ago), and 'Tugas1.java' (first commit, 15 minutes ago). On the right side, there is an 'About' section with the text 'Ini adalah tugas jobsheet 1 daspro', an 'Activity' section showing 0 stars, 1 watching, and 0 forks, a 'Releases' section indicating no releases have been published, and a 'Packages' section.

## Percobaan 2: Dasar Kolaborasi Github

1. Jelaskan fungsi dari **Pull requests!**

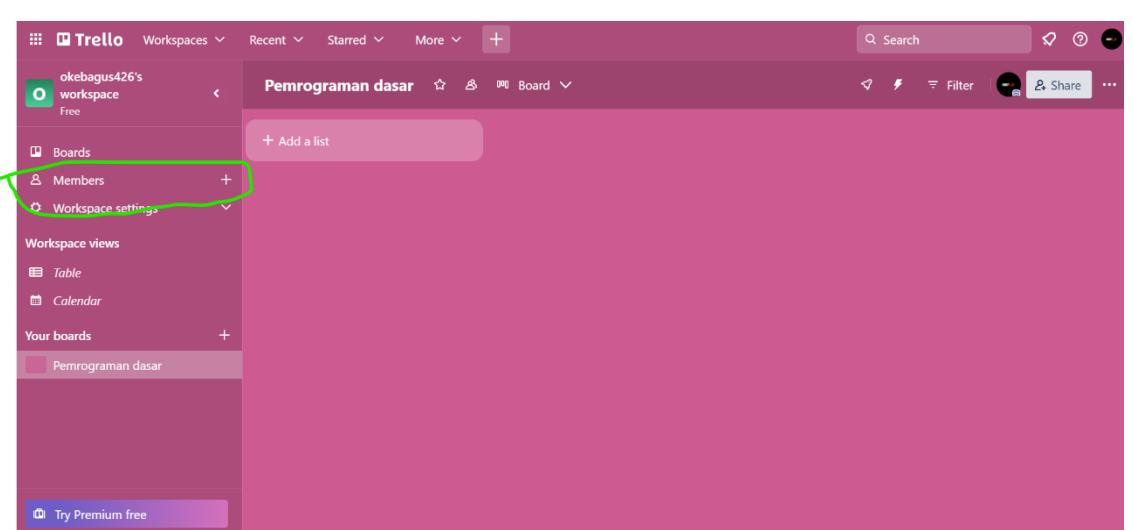
Pull request digunakan jika kita ingin berkontribusi ke suatu project atau pun repository, setelah kita melakukan perubahan kode, kita bisa melakukan pull request

2. Mengapa kita perlu membuat sebuah **branch**, manfaatnya apa?

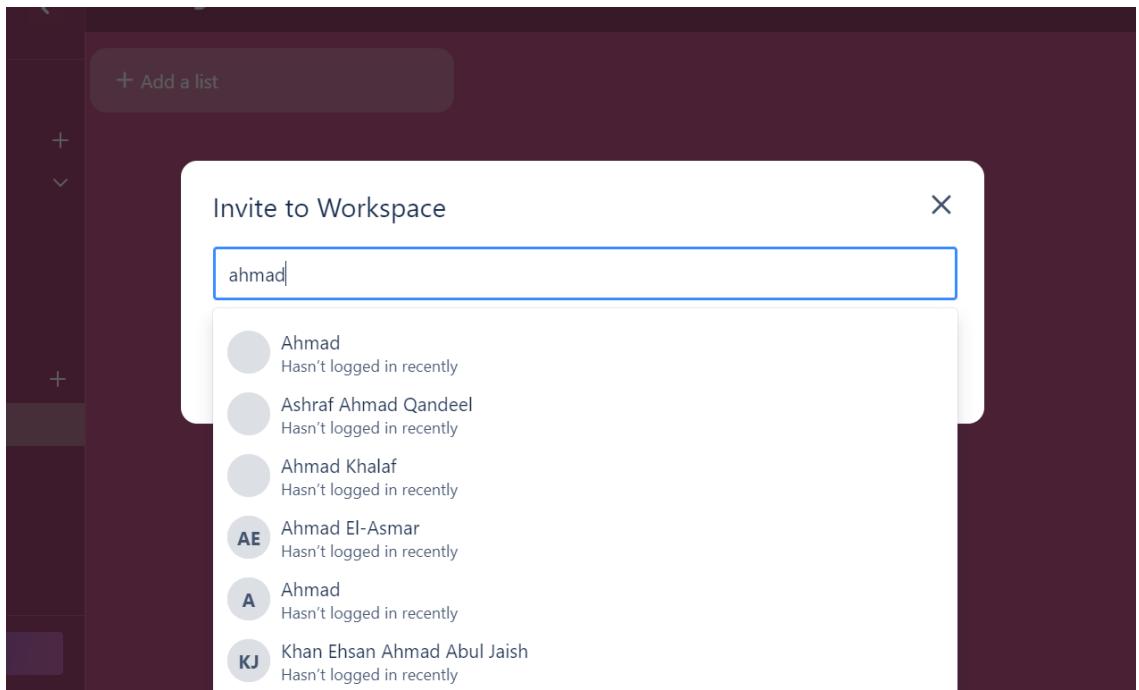
Karena dengan membuat branch kita lebih mudah untuk melakukan perubahan kode, ataupun bereksperimen. Misalnya kita punya sebuah branch main dimana di dalam branch itu ada kode yang sudah stabil, namun kita ingin bereksperimen menambah fitur baru ataupun melakukan eksperimen tertentu, kita bisa membuat branch baru agar tidak mengganggu kode kita yang sudah stabil tadi. Lalu jika fitur baru/eksperimen tadi sudah stabil dan mau di terapkan ke kode utama, kita tinggal merge aja branch nya ke branch utama.

### Percobaan 3: Menggunakan Trello

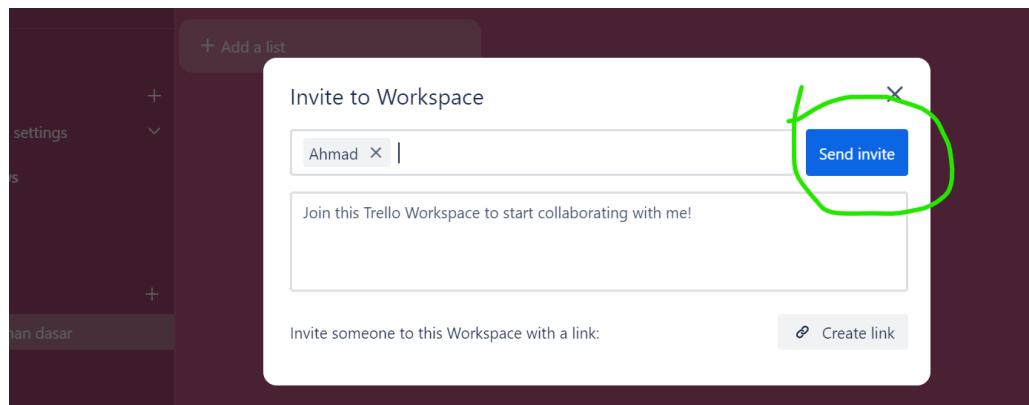
1. Bagaimana langkah yang dilakukan untuk mengundang anggota tim untuk bergabung ke dalam board Anda?
  - a. Klik icon + yang ada disebelah menu members di sidebar



- b. Masukkan email atau nama user yang mau diundang

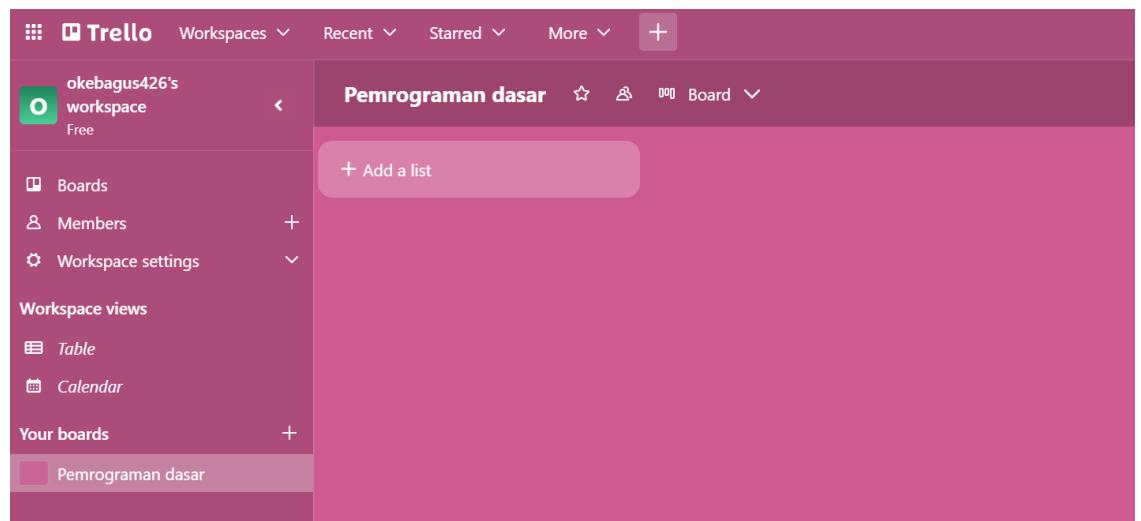


### c. Klik Send Invite

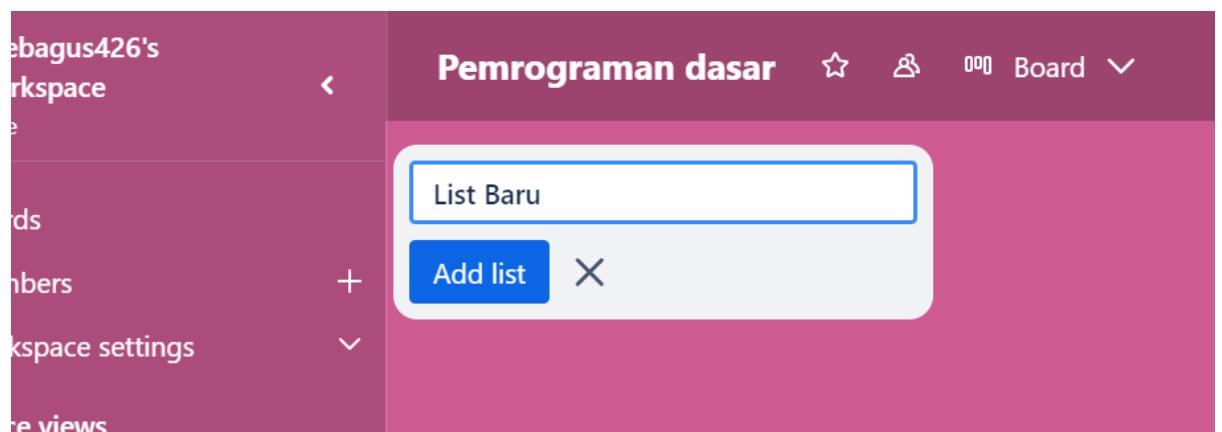


2. Apakah memungkinkan untuk membuat Board tidak dari awal (menambahkan list satu per satu), jika mungkin bagaimana caranya? **Bisa**

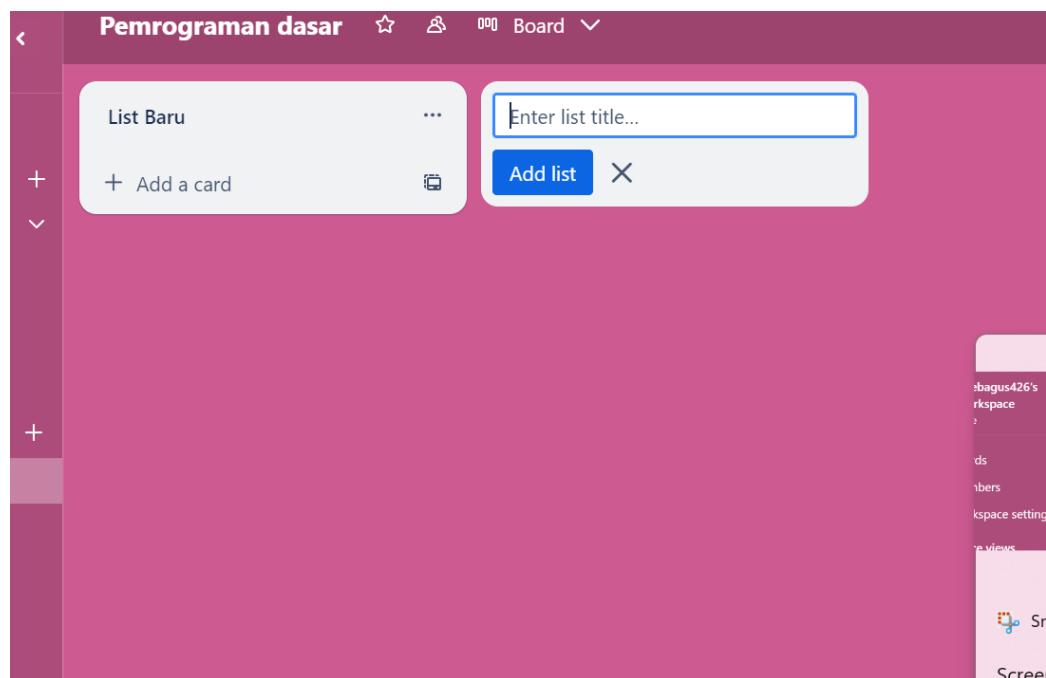
#### a. Klik add a list



b. Masukkan Nama list yang akan dibuat, Lalu klik add list



c. List Berhasil dibuat



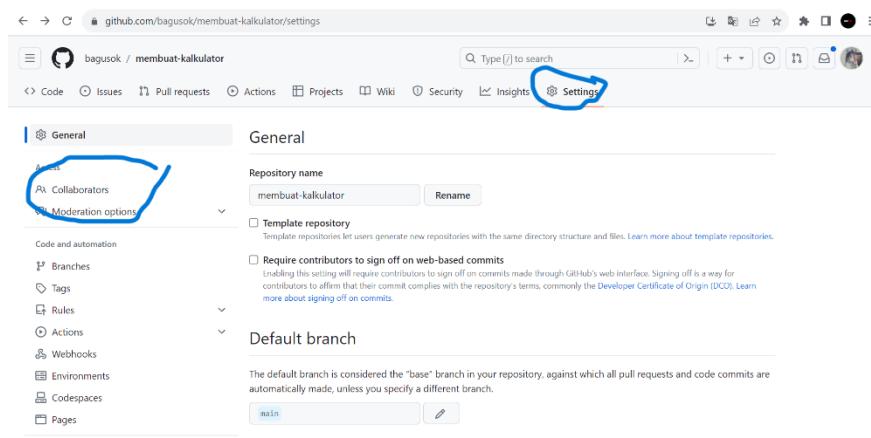
## Tugas

**Waktu pengerojan : 60 menit**

1. Buatlah kelompok yang terdiri dari 3 orang!
2. Buatlah repository berdasarkan topik yang dipilih oleh kelompok Anda, undang semua anggota tim ke dalam repository tersebut.

<https://github.com/bagusok/membuat-kalkulator>

- a. a. Buat Repositorynya dulu, setelah membuat repositorynya, pergi ke setting lalu collaborators



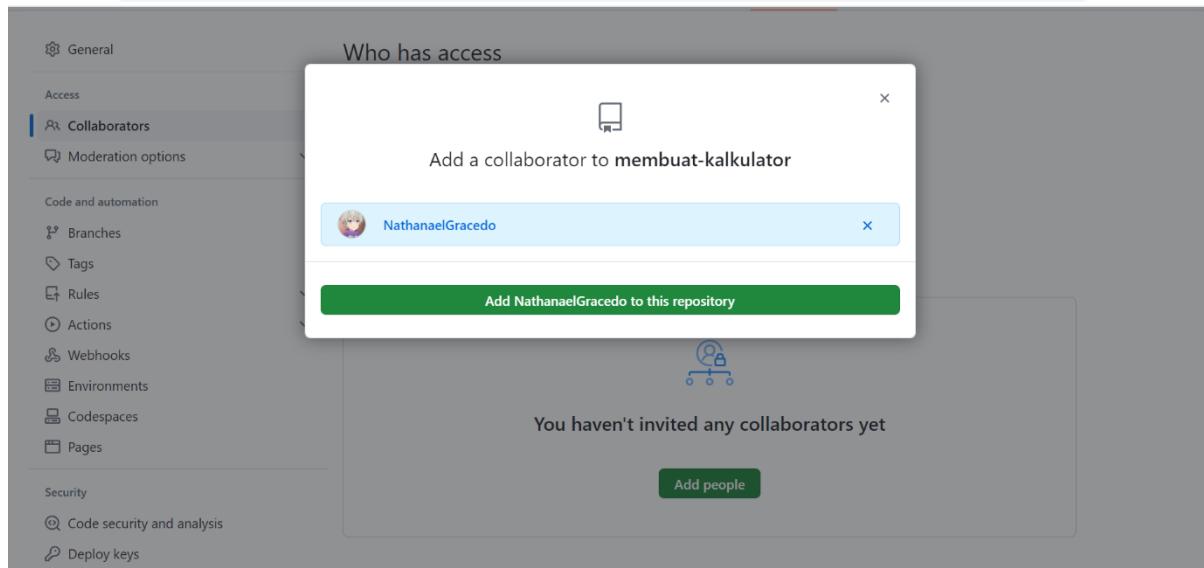
- b. Lalu klik add people

The screenshot shows the 'Who has access' section of a GitHub repository's settings. On the left, there's a sidebar with sections like General, Access (Collaborators selected), Code and automation, and Security. The main area shows two options: 'PUBLIC REPOSITORY' (selected) and 'DIRECT ACCESS'. It states '0 collaborators have access to this repository. Only you can contribute to this repository.' A 'Manage' button is present. Below this is a 'Manage access' section with a message 'You haven't invited any collaborators yet' and a 'Add people' button.

c. Lalu cari username github temanmu dan klik nama temanmu

The screenshot shows a modal window titled 'Add a collaborator to membuat-kalkulator'. It contains a search bar with 'NathanaelGracedo' and a result card for 'NathanaelGracedo' with the subtext 'Invite collaborator'. A green button at the bottom says 'Select a collaborator above'. The background shows the same 'Who has access' interface as the previous screenshot.

d. Lalu klik "Select a collaborate above"



e. Selesai, tinggal menunggu teman menerima invitanya yang dikirim ke email

The screenshot shows the 'Manage access' section of the GitHub repository settings. It lists two pending invites: 'fieraziadattunisa' and 'NathanaelGracedo', both with 'Pending invite' status and 'Remove' options. The interface includes a search bar and a 'Select all' checkbox.

3. Buatlah sebuah Kanban Board dengan judul board lengkap dengan card sesuai dengan topik proyek yang kelompok Anda pilih, undang semua anggota tim ke dalam board tersebut.

<https://trello.com/b/Dy93HpGD>

**DODRUS**

- ♥ Highlights
- Views
- Members
- Settings

**GUEST WORKSPACES** ⓘ

**Trello Workspace**

**Membuat Kalkulator**

Try Trello Premium

Get unlimited boards, all the views, unlimited automation, and more.

[Start free trial](#)

[View all closed boards](#)

**Membuat Kalkulator**

To Do

Menentukan Problem

Membuat Flowchart

Membuat Kode menggunakan java

Menambahkan Penjumlahan

Menambahkan Pengurangan

Menambahkan Perkalian

Menambahkan Pembagian

+ Add a card

Doing

+ Add a card

Done

+ Add a card

Board

Filter

NJ FN

Share

...

View all closed boards

Snipping Tool

Screenshot copied to clipboard and saved  
Select here to mark up and share the image

Share board

Email address or name

Member

Share

Share this board with a link  
[Create link](#)

Member

Admin

Member

o kebagus426 (you)  
@bagusok • Workspace guest

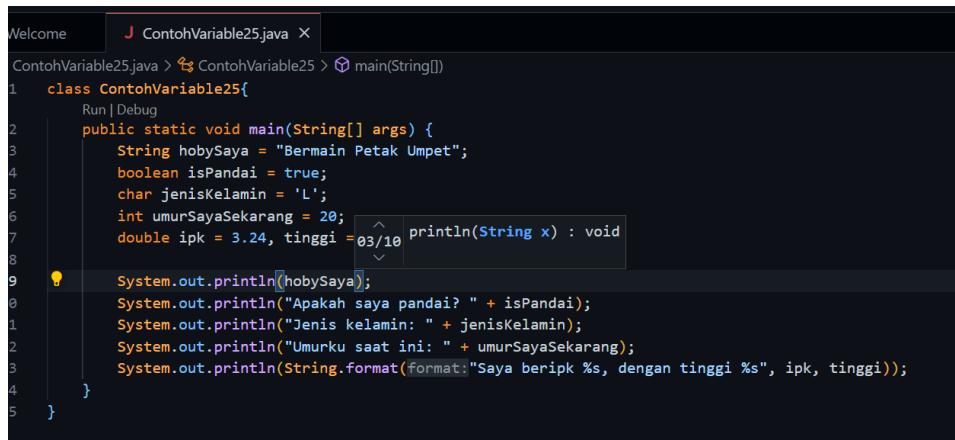
Nathan Juan  
@nathanjuan • Workspace admin

Fiera Ziadattun Nisa'  
@fieraziadattunnisa • Workspace guest

## Jobsheet 3

### Percobaan 1: Penggunaan Variable

1. Silakan Anda ubah nama variabel sehingga model penamaan variabel menjadi baik dan benar!



```
Welcome J ContohVariable25.java X
ContohVariable25.java > ContohVariable25 > main(String[])
1 class ContohVariable25{
    Run | Debug
2     public static void main(String[] args) {
3         String hobySaya = "Bermain Petak Umpet";
4         boolean isPandai = true;
5         char jenisKelamin = 'L';
6         int umurSayaSekarang = 20;
7         double ipk = 3.24, tinggi = 03/10
8         System.out.println(hobySaya);
9         System.out.println("Apakah saya pandai? " + isPandai);
10        System.out.println("Jenis kelamin: " + jenisKelamin);
11        System.out.println("Umurku saat ini: " + umurSayaSekarang);
12        System.out.println(String.format("Saya beripk %s, dengan tinggi %s", ipk, tinggi));
13    }
14 }
```

2. Untuk apakah %s pada statement dibawah ini?

```
System.out.println(String.format("Saya beripk %s, dengan tinggi badan %s", $ipk, tinggi));
```

Apakah ada yang bisa digunakan selain %s? Sebut dan jelaskan!

%s digunakan untuk tempat penampungan nilai yang akan dimasukkan ke dalam string. String format merupakan fungsi yang digunakan untuk mengganti %s dengan nilai yang diberikan dalam argument setelah string format.

Ada selain %s yaitu:

%d untuk integer

%f untuk decimal

%c untuk char

%b untuk Boolean

%x untuk heksadesimal

%t untuk tanggal dan waktu

## Percobaan 2: Penggunaan Tipe Data

1. Mengapa ketika menampilkan nilai `golonganDarah` hasilnya bukan A ?

Karena variable `golonganDarah` telah di ubah tipe datanya yang semula char ke byte, dimana byte itu hanya menampung nilai -128 sampai 127, dan setiap karakter itu telah ada nilai ascii decimal, contohnya karakter A itu memiliki nilai 65.

2. Apa maksud sintak `byte jarak = (byte) 130` ? kemudian mengapa ketika ditampilkan hasilnya berubah?

Maksud syntax tersebut adalah mencoba mengubah 130 menjadi byte, sedangkan rentang nilai byte kurang hanya -128 sampai 130. Maka 130 akan mengalami underflow dan akan Kembali ke batas bawah dan nilainya menjadi -126

3. Pada `float suhu = 60.50F;`, silakan hilangkan `F` kemudian jalankan kembali. Apa yang terjadi?

Eror, karena kalo float tanpa f maka java akan menganggap sebagai double

4. Mengapa ketika menampilkan nilai berat , hasilnya berubah?

Karena berat mengalami tipe casting dari double ke float, dimana float memiliki presisi yang lebih rendah dari double atau bisa disebut narrowing type casting. Atau simple nya tipe data float hanya mampu manampung angka dibelakang koma lebih sedikit dari float, maka angka dibelakang komanya dikonversi agar bisa ditampung oleh float

5. Maksud inisialisasi `0x10` pada variabel `angkaDesimal` digunakan untuk apa?

`0x10` merupakan bilangan heksadesimal basis 16, `0x10` berarti 16 dalam desimal

## Percobaan 3: Penggunaan Operator

1. Jelaskan menurut pendapat Anda perbedaan antara `x++` dan `++x` ?

Jika `x++` variable x akan digunakan dulu lalu ditambah 1, sedangkan `++x` langsung menambahkan 1 ke variable xnya

2. Berapa hasil dari `int z = x ^ y;` , silakan dilakukan perhitungan secara manual!

Kita ubah dulu x dan y ke biner

`x = 1011`

`y = 1100`

lalu kita lakukan operasi xor menjadi 0111 atau kalo dalam decimal menjadi 71

## Percobaan 4: Studi Kasus

1. Jelaskan mengapa harus melakukan deklarasi Scanner di praktikum percobaan 4 diatas?

Deklarasi scanner digunakan untuk mengambil input dari pengguna

2. Jelaskan apa kegunaan potongan program dibawah ini!

```
alas = sc.nextInt();
tinggi = sc.nextInt();
```

Kode diatas digunakan untuk menentukan tipe data dari inputan pengguna (integer), sehingga menghindari kesalahan tipe data.

## Tugas Kelompok

### a) Menentukan input, output, dan proses

**Input :** Nama karyawan, Jumlah jam kerja, Bonus.

**Output :** Gaji total

**Proses :**

1. Input Nama karyawan, Jumlah jam kerja, Bonus.
2. Gaji total = bonus + (gaji per jam x jumlah jam kerja)
3. Output gaji total

### b) Mengidentifikasi variable dan jenis data

Variabel	Tipe Data
Nama karyawan	String
Jumlah jam kerja	Int
Bonus	Double
Gaji total	Double

### c) Implementasikan soal a dan b ke dalam kode program java

1. Buat file nama **GajiKaryawan.java** :

# J GajiKaryawan.java 1

2. Buat kelas Gaji Karyawan :

```
3 public class GajiKaryawan {
```

3. Buat inisialisasi scanner dan jangan lupa import scanner :

```
5 Scanner sc = new Scanner(System.in);
6 System.out.println("===== Hitung Gaji Karyawan =====");
7
```

4. Buat variabel dan jangan lupa gaji per jam diberi nilai default :

```
8     double bonus, gajiPerJam = 2000;
9     int jumlahJam;
10    String namaKaryawan;
11
```

5. Buat input untuk nama karyawan, jumlah jam kerja, dan juga bonus :

```
12    System.out.print("Masukkan nama karyawan: ");
13    namaKaryawan = sc.nextLine();
14    System.out.print("Masukkan jumlah jam kerja: ");
15    jumlahJam = sc.nextInt();
16    System.out.print("Masukkan Bonus dalam rupiah (isi 0 jika tidak ada): ");
17    bonus = sc.nextInt();
18
```

6. Hitung gaji karyawan :

```
21    double gajiTotal = jumlahJam * gajiPerJam + bonus;
```

7. Print output gaji karyawan :

```
21    double gajiTotal = jumlahJam * gajiPerJam + bonus;
22    System.out.println("\nNama karyawan: " + namaKaryawan);
23    System.out.println("Gaji per jam: Rp. " + gajiPerJam);
24    System.out.println("Jumlah jam kerja: " + jumlahJam);
25    System.out.println("Bonus: Rp. " + jumlahJam);
26    System.out.println("Gaji total: Rp. " + gajiTotal + "\n");
27
28
29 }
```

Hasil Output :

```
ccadbc6a8f\redhat.java\jdt_ws\jobsheet3_3ae0be70\bin' 'GajiKaryawan
===== Hitung Gaji Karyawan =====
Masukkan nama karyawan: Anton
Masukkan jumlah jam kerja: 40
Masukkan Bonus dalam rupiah (isi 0 jika tidak ada): 12000
=====
Nama karyawan: Anton
Gaji per jam: Rp. 20000.0
Jumlah jam kerja: 40
Bonus: Rp. 40
Gaji total: Rp. 812000.0

PS C:\Users\Acer\Project\Daspro\jobsheet3> █
```

## Jobsheet 4

### Percobaan 1

1. Pada percobaan 1 yang telah dilakukan modifikasi pseudocode, dengan nilai phi 3.14 yang dimasukkan langsung di proses diubah menjadi sebuah variable **phi** yang diberikan nilai awal 3.14!

Algoritma: LingkaranNoAbsen

{dibaca jari-jari lingkaran dari piranti masukan. Hitunglah keliling dan luas lingkaran tersebut}

Deklarasi:

r : int

keliling, luas, phi : double

phi: 3.14

Deskripsi:

```

1. print "masukkan jari-jari lingkaran!"

2. read r

3. keliling = 2*3.14*r

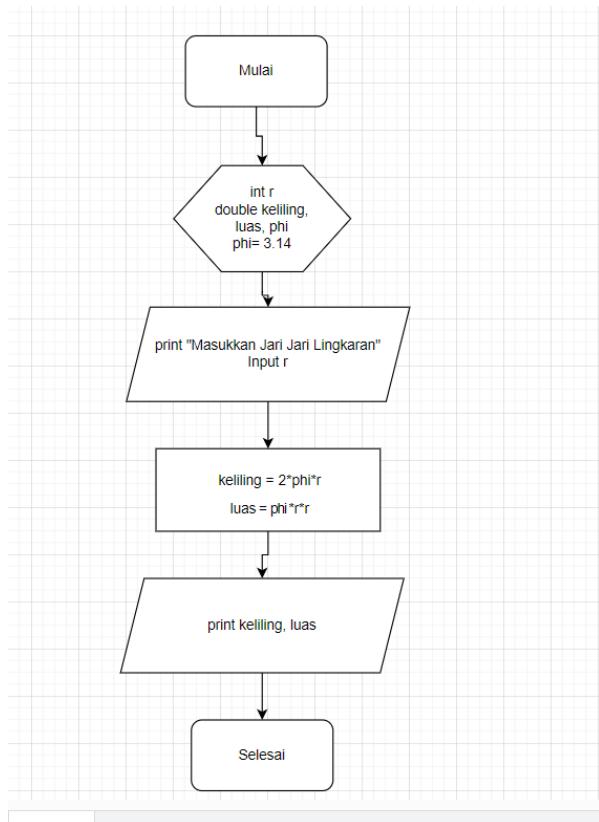
4. luas = 3.14 *r*r

5. print keliling

print luas

```

2. Buatlah flowchart berdasarkan pseudocode yang telah dimodifikasi pada soal no 1!



3. Implementasikan hasil modifikasi pseudocode yang telah dilakukan pada soal no 1 kedalam kode program (modifikasi kode program sesuai pseudocode pada soal no 1)!

```
J Welcome J Lingkaran25.java 1 X
J Lingkaran25.java > ↗ Lingkaran25
1 import java.util.Scanner;
2
3 class Lingkaran25{
4     Run | Debug
5     public static void main(String[] args){
6
7         Scanner input = new Scanner(System.in);
8
9         int r;
10        double keliling, luas, phi = 3.14;
11        System.out.println("Masukkan jari-jari lingkaran : ");
12        r = input.nextInt();
13        keliling = 2 * phi * r;
14        luas = phi * r * r;
15
16        System.out.println(keliling);
17        System.out.println(luas);
18    }
}
```

## Percobaan 2

1. Buatlah pseudocode pada berdasarkan flowchart pada percobaan 2, kemudian lakukan modifikasi dimana besaran gaji dan potongan gaji adalah inputan juga!

Algoritma: Gaji

Deklarasi

Int jmlMasuk, jmlTdkMasuk, totGaji, gaji, potGaji

Deskripsi

Read jmlMasuk, jmlTdkMasuk, gaji, potGaji

Totgaji=(jmlMasuk \* gaji) – (jmlTdkMasuk \* potGaji)

## Print totalGaji

2. Implementasikan pseudocode pada soal no 1 menjadi sebuah kode program (modifikasi program percobaan 2 sesuai pseudocode pada soal no 1)!

```
Gaji25.java > Gaji25 > main(String[])
1 import java.util.Scanner;
2
3 class Gaji25{
4     Run | Debug
5     public static void main(String[] args) {
6         int jmlMasuk, jmlTdkMasuk, totGaji, gaji, potGaji;
7
8         Scanner scanner = new Scanner(System.in);
9
10        System.out.println("Masukkan jumlah masuk : ");
11        jmlMasuk = scanner.nextInt();
12        System.out.println("Masukkan jumlah tidak masuk : ");
13        jmlTdkMasuk = scanner.nextInt();
14        System.out.println("Masukkan gaji : ");
15        gaji = scanner.nextInt();
16        System.out.println("Masukkan potongan gaji : ");
17        potGaji = scanner.nextInt();
18
19        totGaji = (jmlMasuk * gaji) - (jmlTdkMasuk * potGaji);
20
21        System.out.println("Total gaji : " + totGaji);
22    }
}
```

## Percobaan 3

1. Buatlah pseudocode berdasarkan project kelompok masing-masing, dimana pseudocode yang dibuat dapat diidentifikasi dari proses aritmatika kelompok masing-masing yang sudah dilakukan di minggu sebelumnya!

Algoritma: GajiKaryawan

Deklarasi

bonus, gajiPerjam = 20000 : double

jumlahJam : int

namaKaryawan : StringS

Deskripsi

Print “Masukkan nama karyawan”

Read namaKaryawan

Print "Masukkan Jumlah Jam Kerja"

Read jumlahJam

Print "Masukkan Bonus dalam rupiah"

Read bonus

Print "=====

GajiTotal = jumlahJam \* gajiPerjam + bonus

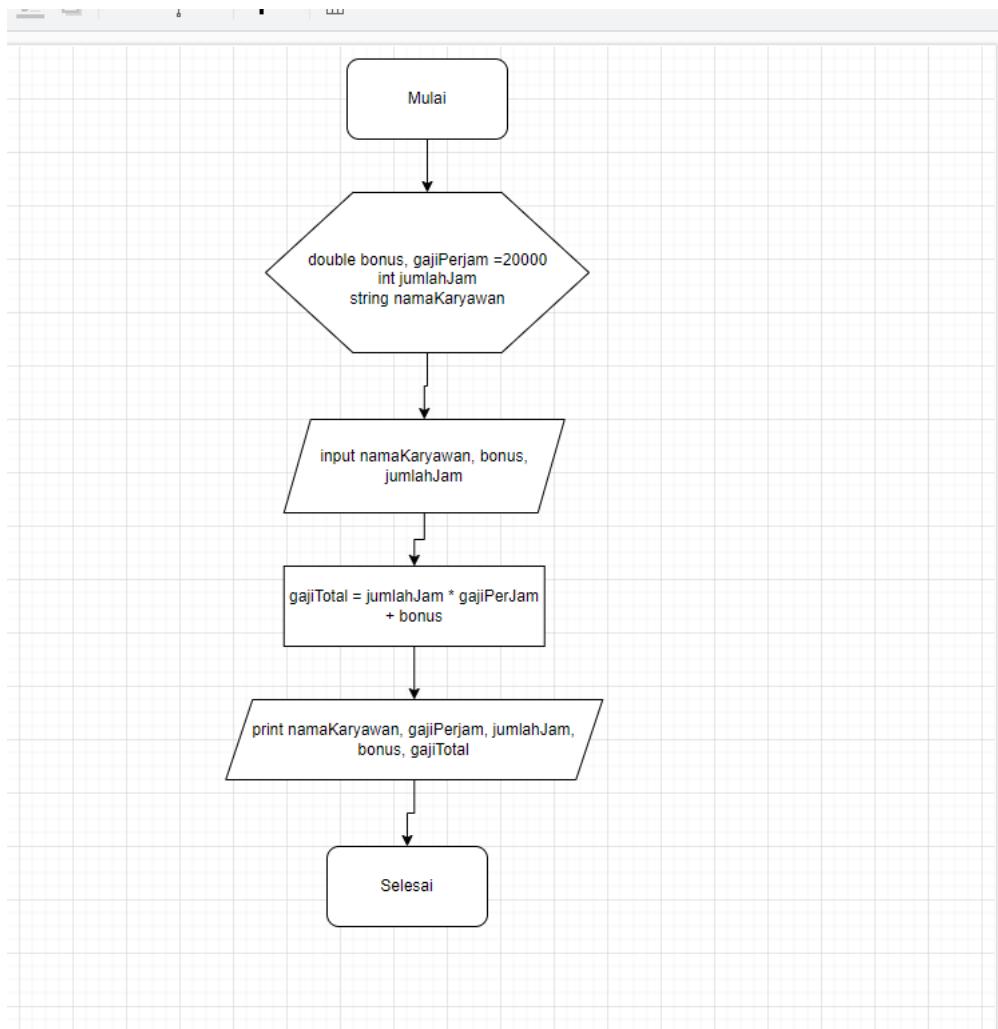
Print "namaKaryawan"

Print "gajiPerJam"

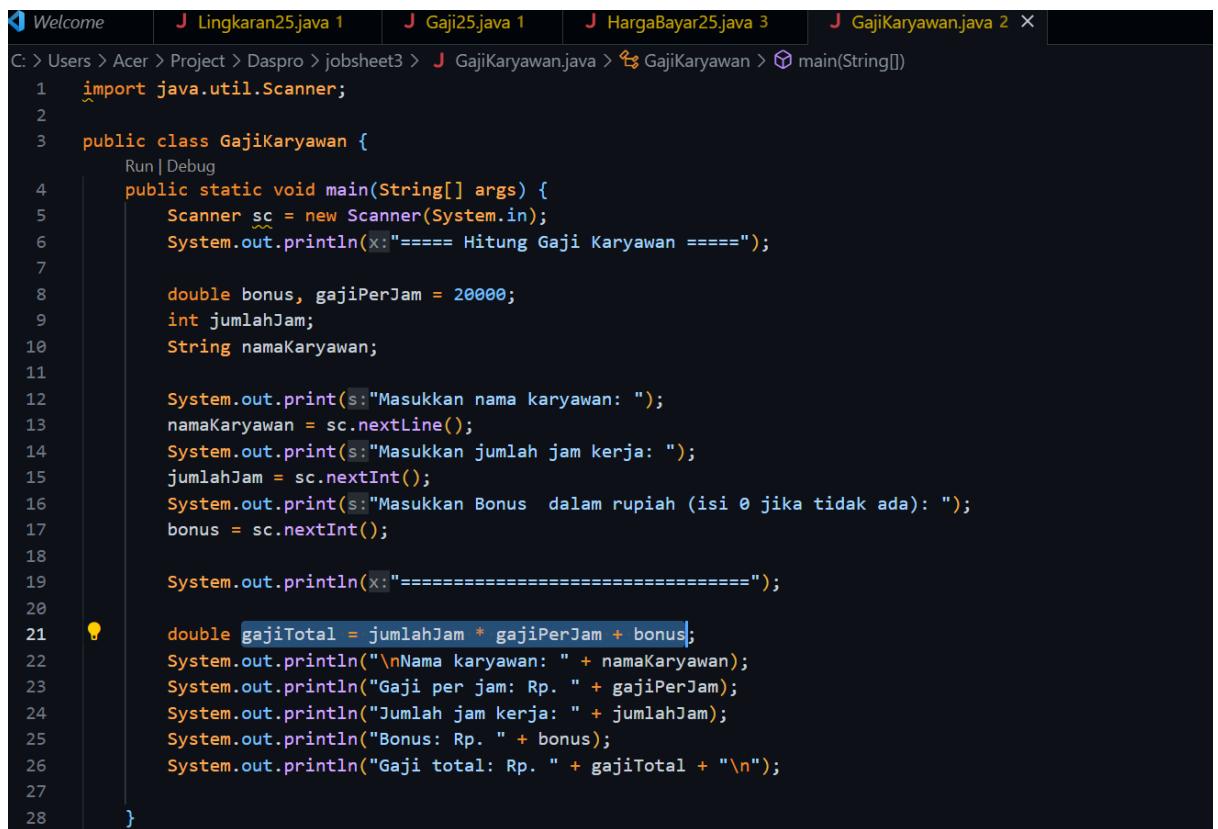
Print "jumlahJam"

Print "gajiTotal"

2. Buatlah flowchart berdasarkan project kelompok masing-masing, dimana pseudocode yang dibuat dapat diidentifikasi dari proses arimatika kelompok masing-masing yang sudah dilakukan di minggu sebelumnya!



3. Implementasikan pseudocode atau flowchart pada tugas no 1 dan 2 ke dalam kode program!



```
1 import java.util.Scanner;
2
3 public class GajiKaryawan {
4     Run | Debug
5     public static void main(String[] args) {
6         Scanner sc = new Scanner(System.in);
7         System.out.println("===== Hitung Gaji Karyawan =====");
8
9         double bonus, gajiPerJam = 20000;
10        int jumlahJam;
11        String namaKaryawan;
12
13        System.out.print("Masukkan nama karyawan: ");
14        namaKaryawan = sc.nextLine();
15        System.out.print("Masukkan jumlah jam kerja: ");
16        jumlahJam = sc.nextInt();
17        System.out.print("Masukkan Bonus dalam rupiah (isi 0 jika tidak ada): ");
18        bonus = sc.nextInt();
19
20        System.out.println("=====");
21        double gajiTotal = jumlahJam * gajiPerJam + bonus;
22        System.out.println("\nNama karyawan: " + namaKaryawan);
23        System.out.println("Gaji per jam: Rp. " + gajiPerJam);
24        System.out.println("Jumlah jam kerja: " + jumlahJam);
25        System.out.println("Bonus: Rp. " + bonus);
26        System.out.println("Gaji total: Rp. " + gajiTotal + "\n");
27
28    }
}
```

# Jobsheet 5

## Percobaan 1

1. Modifikasi program diatas pada bagian struktur pemilihannya dengan memanfaatkan Ternary Operator!

```
1 import java.util.Scanner;
2
3 class PemilihanPercobaan25{
4     Run | Debug
5     public static void main(String[] args) {
6
7         Scanner input = new Scanner(System.in);
8         System.out.println("Masukkan angka: ");
9         int angka = input.nextInt();
10
11         String output = (angka % 2 == 0) ? "Genap" : "Ganjil";
12         System.out.println("Angka " + angka + " adalah " + output + "");
13     }
14 }
```

2. Jalankan dan amati lah hasilnya!

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL FORTS
baan25'
Masukkan angka:
10
Angka 10 adalah Genap
PS C:\Users\Acer\Project\Daspro\jobsheet5>
```

3. Push dan commit hasil modifikasi anda ke repository!

The screenshot shows a GitHub repository interface. At the top, there are buttons for 'main' (with 2 branches and 0 tags), 'Code' (with a dropdown menu), and 'Add file'. Below this is a list of recent commits by user 'bagusok':

Commit	File	Message	Date
12f48e8	HitungAkarKuadrat.java	add close scanner	42 minutes ago
12f48e8	HitungJarakTembak.java	add close scanner	42 minutes ago
12f48e8	Login.java	add close scanner	42 minutes ago
12f48e8	PemilihanPercobaan225.java	add close scanner	42 minutes ago
12f48e8	PemilihanPercobaan25.java	add close scanner	42 minutes ago
12f48e8	PemilihanPercobaan325.java	add close scanner	42 minutes ago

At the bottom, there is a note to 'Add a README with an overview of your project.' and a green 'Add a README' button.

4. Jelaskan mengapa output program yang dimodifikasi sama dengan output program sebelum dimodifikasi!

Karena sama saja, ternary operator merupakan if else versi singkatnya.

## Percobaan 2

1. Modifikasi program diatas sehingga dapat menampilkan nilai huruf sesuai aturan berikut ini!

Nilai Angka	Nilai Mutu		
	Nilai Huruf	Nilai Setara	Kualifikasi
80 < N ≤ 100	A	4	Sangat Baik
73 < N ≤ 80	B+	3,5	Lebih dari Baik
65 < N ≤ 73	B	3	Baik
60 < N ≤ 65	C+	2,5	Lebih dari Cukup
50 < N ≤ 60	C	2	Cukup
39 < N ≤ 50	D	1	Kurang
N ≤ 39	E	0	Gagal

2. Push dan commit hasil modifikasi anda ke repository

```
import java.util.Scanner;

public class PemilihanPercobaan225 {

    public static void main(String[] args) {
        Scanner input00 = new Scanner(System.in);
        System.out.println("Nilai uas: ");
        float uas = input00.nextFloat();
        System.out.println("Nilai uts: ");
        float uts = input00.nextFloat();
        System.out.println("Nilai kuis: ");
        float kuis = input00.nextFloat();
        System.out.println("Nilai tugas: ");
        float tugas = input00.nextFloat();

        float total = (uas * 0.4f) + (uts * 0.3f) + (kuis * 0.1f) +(tugas * 0.2f);

        String message;

        if (total > 80) {
            message = "A";
        } else if (total > 73) {
            message = "B+";
        } else if (total > 65) {
            message = "B";
        } else if (total > 60) {
            message = "C+";
        } else if (total > 50) {
            message = "C";
        } else if (total > 39) {
            message = "D";
        } else {
            message = "E";
        }
        System.out.println("Nilai akhir: " + total + " sehingga " + message + "");
    }
}
```

3. Setelah penambahan kode program pada pertanyaan nomor 1, berapakah jumlah kondisi yang ada serta jelaskan jenis operator yang digunakan!  
Ada 7 kondisi, operator yang digunakan adalah operator lebih dari

### Percobaan 3

1. Jelaskan fungsi dari *break* dan *default* pada percobaan 4 diatas!  
Break digunakan untuk menghentikan eksekusi di dalam switch case saat kondisi ditemukan dan akan dilanjutkan untuk menjalankan eksekusi diluar program switch. Sedangkan default digunakan Ketika kondisi dalam switch case tidak terpenuhi.
2. Modifikasi kode program diatas, hapus *break* pertama. Kemudian jalankan program.Tampilkan hasilnya dan jelaskan hasilnya!

```
Masukkan angka kedua:  
5  
Masukkan operator:  
+  
5.0 + 5.0 = 10.0  
5.0 - 5.0 = 0.0
```

Ketikan break pertama dihapus saat kita akan melakukan penjumlahan maka kondisi terpenuhi di case + namun eksekusi tidak dihentikan karena breaknya tidak ada, sehingga dilanjutkan ke case berikutnya.

3. Jelaskan fungsi perintah kode program dibawah ini pada percobaan 4!

```
operator = sc.next().charAt(0);
```

Operator diatas digunakan untuk mengambil karakter dari input yaitu karakter pertama, karena 0 adalah index pertama

### Tugas

Buatlah kode program berdasarkan flowchart yang telah dibuat pada Tugas pertemuan 5  
Matakuliah Dasar Pemrograman!  
Push dan commit hasil kode program anda ke repository project Anda!

**Catatan :** tugas hanya boleh menerapkan materi dari pertemuan 1 hingga pertemuan 5.

```
import java.util.Scanner;

public class Login {

    public static void main(String[] args) {
        String defUser = "admin", defPass = "admin";
        String user, pass;
        Scanner input = new Scanner(System.in);

        System.out.print("Masukkan username: ");
        user = input.nextLine();
        System.out.print("Masukkan password: ");
        pass = input.nextLine();

        if (user.equals(defUser) && pass.equals(defPass)){
            System.out.println("Login berhasil");
        }else{
            System.out.println("Username atau password salah");
        }

        input.close();
    }
}
```

```
import java.util.Scanner;

public class HitungJarakTembak {

    public static void main(String[] args) {

        Scanner input = new Scanner(System.in);
        int jarak;

        System.out.print("Masukkan jarak: ");
        jarak = input.nextInt();

        if (jarak <= 5){
            System.out.println("Meele Weapon");
        }else{
            System.out.println("Range Weapon");
        }

        input.close();
    }
}
```

```
import java.util.Scanner;

public class HitungAkarKuadrat {

    public static void main(String[] args) {
        int a, b, c;
        double D, x1, x2;

        Scanner input = new Scanner(System.in);
        System.out.print("Masukkan nilai a: ");
        a = input.nextInt();
        System.out.print("Masukkan nilai b: ");
        b = input.nextInt();
        System.out.print("Masukkan nilai c: ");
        c = input.nextInt();

        if (a<0){
            System.out.println("Bukan Persamaan Kuadrat");
        }else{
            D = Math.pow(b, 2) - (4*a*c);
            System.out.println("D = " + D);
            if (D>0) {
                x1 = (-b + Math.sqrt(D))/(2*a);
                x2 = (-b - Math.sqrt(D))/(2*a);

                System.out.println("x1 = " + x1);
                System.out.println("x2 = " + x2);
            }else{
                System.out.println("Akar-akar persamaan kuadrat tidak real");
            }
        }

        input.close();
    }
}
```

main ▾ 2 branches 0 tags

Go to file Add file ▾ <> Code ▾

bagusok add close scanner 12f48e8 42 minutes ago 4 commits

HitungAkarKuadrat.java	add close scanner	42 minutes ago
HitungJarakTembak.java	add close scanner	42 minutes ago
Login.java	add close scanner	42 minutes ago
PemilihanPercobaan225.java	add close scanner	42 minutes ago
PemilihanPercobaan25.java	add close scanner	42 minutes ago
PemilihanPercobaan325.java	add close scanner	42 minutes ago

Add a README with an overview of your project.

Add a README

Ti  
A  
Re  
No  
Pa  
No  
Pul

## Jobsheet 6

### Percobaan 1

1. Bagaimana outputnya ketika diberikan input tahun 2100 ? Jelaskan! Bagaimana agar output sesuai dengan ketentuan (Tahun 2100 bukan tahun kabisat)
2. Modifikasi program sesuai jawaban no 1 !

```
Pemilihan2Percobaan125.java > Pemilihan2Percobaan125 > main(String[])
4     public static void main(String[] args) {
5         Scanner sc = new Scanner(System.in);
6
7         System.out.print("Masukkan tahun: ");
8         int tahun = sc.nextInt();
9
10        if (tahun % 4 == 0){
11            if (tahun % 100 != 0){
12                System.out.println(tahun + " adalah tahun kabisat");
13            }else{
14                System.out.println(tahun + " adalah bukan tahun kabisat");
15            }
16        } else {
17            System.out.println(tahun + " adalah bukan tahun kabisat");
18        }
19    }
```

3. Push dan commit hasil modifikasi anda ke repository !
4. Tahun 2000 adalah kelipatan 4 dan kelipatan 100, tetapi tahun 2000 merupakan tahun kabisat, ketentuan tambahan (pengecualian) adalah ketika tahun kelipatan 100 **dan juga** kelipatan 400 maka tahun tersebut merupakan tahun kabisat. Modifikasi program untuk menyesuaikan ketentuan tersebut ! (selesaikan tanpa menggunakan operator logika)

```
public static void main(String[] args) {
    Scanner sc = new Scanner(System.in);

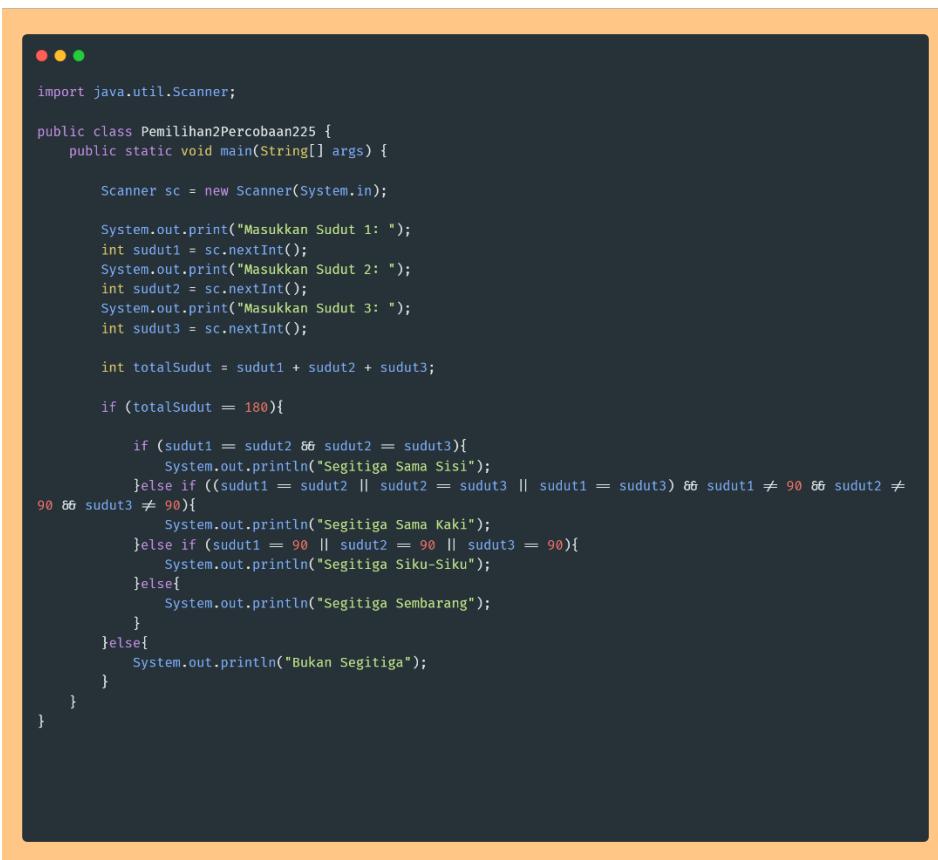
    System.out.print("Masukkan tahun: ");
    int tahun = sc.nextInt();

    if (tahun % 4 == 0){
        if (tahun % 100 != 0){
            System.out.println(tahun + " adalah tahun kabisat");
        }else{
            if (tahun % 400 == 0){
                System.out.println(tahun + " adalah tahun kabisat");
            }else{
                System.out.println(tahun + " adalah bukan tahun kabisat");
            }
        }
    } else {
        System.out.println(tahun + " adalah bukan tahun kabisat");
    }
}
```

5. Push dan commit hasil modifikasi anda ke repository !

## Percobaan 2

1. Jenis bangun datar segitiga selain segitiga siku-siku terdapat segitiga sama sisi, segitiga sama kaki dan segitiga sembarang. Modifikasi program tersebut untuk dapat mengeluarkan output jenis segitiga yang lain (selain segitiga siku-siku). Silahkan menggunakan operator logika untuk menghubungkan multi kondisi.



```
import java.util.Scanner;

public class Pemilihan2Percobaan225 {
    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        System.out.print("Masukkan Sudut 1: ");
        int sudut1 = sc.nextInt();
        System.out.print("Masukkan Sudut 2: ");
        int sudut2 = sc.nextInt();
        System.out.print("Masukkan Sudut 3: ");
        int sudut3 = sc.nextInt();

        int totalSudut = sudut1 + sudut2 + sudut3;

        if (totalSudut == 180){

            if (sudut1 == sudut2 && sudut2 == sudut3){
                System.out.println("Segitiga Sama Sisi");
            }else if ((sudut1 == sudut2 || sudut2 == sudut3 || sudut1 == sudut3) && sudut1 != 90 && sudut2 != 90 && sudut3 != 90){
                System.out.println("Segitiga Sama Kaki");
            }else if (sudut1 == 90 || sudut2 == 90 || sudut3 == 90){
                System.out.println("Segitiga Siku-Siku");
            }else{
                System.out.println("Segitiga Sembarang");
            }
        }else{
            System.out.println("Bukan Segitiga");
        }
    }
}
```

2. Push dan commit hasil modifikasi anda ke repository !

## Percobaan 3

1. Jelaskan fungsi dari **(int)** pada sintaks gajiBersih = (int) (penghasilan - (penghasilan \* pajak));

Digunakan untuk mengubah tipe data gaji bersih ke interger

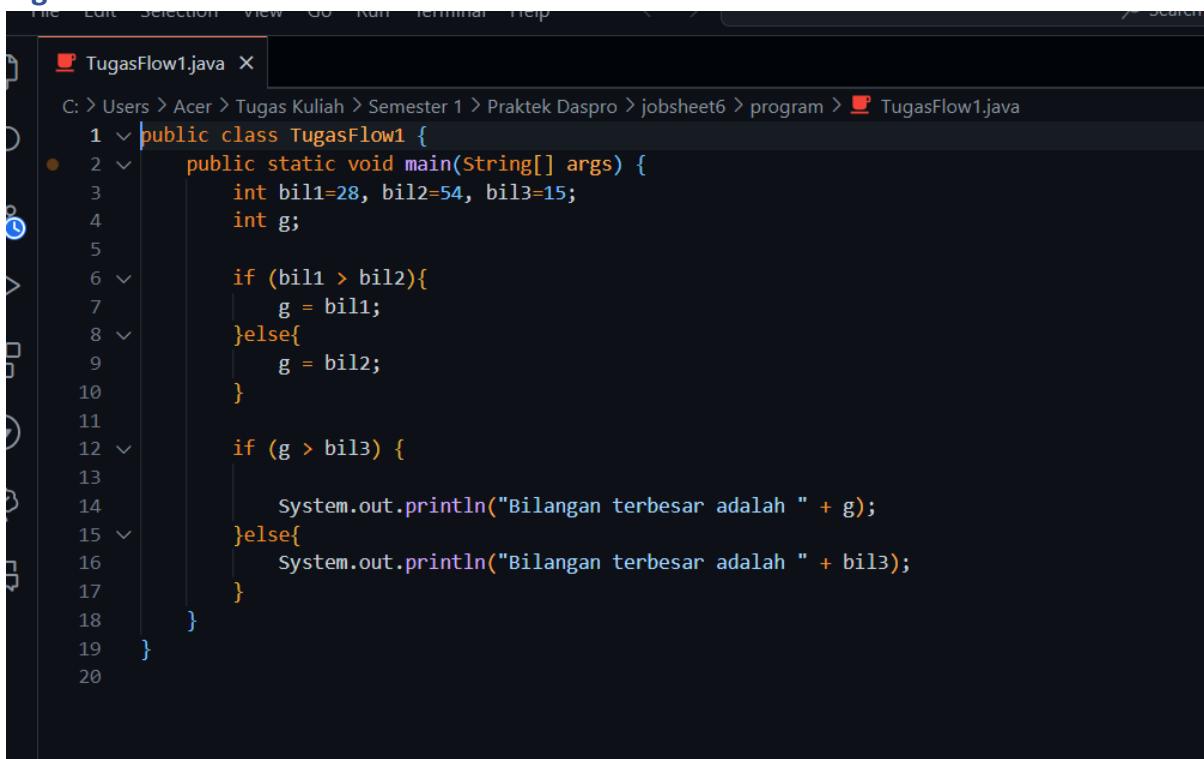
2. Jalankan program dengan memasukkan kategori = PEBISNIS dan penghasilan = 2000000. Amati apa yang terjadi! Apa kegunaan dari **equalsIgnoreCase?**

EqualsIgnore case akan mencocokkan apakah kata PEBISNIS cocok dengan pebinis, tapi tidak memperhatikan besar kecil hurufnya (tidak case sensitive)

- Ubah **equalsIgnoreCase** menjadi **equals**, kemudian jalankan program dengan memasukkan kategori = PEBISNIS dan penghasilan = 2000000. Amati apa yang terjadi! Mengapa hasilnya demikian? Apa kegunaan dari **equals**?

Karena equals itu memcocokkan dengan memperhatikan besar kecil nya huruf (case sensitive) sehingga PEBISNIS dan pebisnis itu tidak sama

## Tugas



```
File Edit Selection View Go Run Terminal Help / Search
TugasFlow1.java X
C: > Users > Acer > Tugas Kuliah > Semester 1 > Praktek Daspro > jobsheet6 > program > TugasFlow1.java
1 public class TugasFlow1 {
2     public static void main(String[] args) {
3         int bil1=28, bil2=54, bil3=15;
4         int g;
5
6         if (bil1 > bil2){
7             g = bil1;
8         }else{
9             g = bil2;
10        }
11
12        if (g > bil3) {
13
14            System.out.println("Bilangan terbesar adalah " + g);
15        }else{
16            System.out.println("Bilangan terbesar adalah " + bil3);
17        }
18    }
19 }
20
```

```
File Edit Selection View Go Run Terminal Help
TugasFlow2.java X
C: > Users > Acer > Tugas Kuliah > Semester 1 > Praktek Daspro > jobsheet6 > program > TugasFlow2.java
1 import java.util.Scanner;
2
3 public class TugasFlow2 {
4     public static void main(String[] args) {
5         double diskon = 0;
6
7         Scanner sc = new Scanner(System.in);
8
9         System.out.print("Masukkan jenis buku:");
10        String jenis = sc.nextLine();
11        System.out.print("Masukkan jumlah buku:");
12        int jumlah = sc.nextInt();
13
14        if (jenis != "kamus" && jenis != "novel") {
15            if (jumlah >= 3) {
16                diskon = 0.05;
17            }
18        }else if (jenis == "kamus") {
19            if (jumlah <= 2) {
20                diskon = 0.1;
21            }else{
22                diskon = 0.1 + 0.02;
23            }
24        }else if (jenis == "novel") {
25            if (jumlah <= 3) {
26                diskon = 0.07 + 0.01;
27            }else{
28                diskon = 0.07 + 0.02;
29            }
30        }
31
32        System.out.println("Jumlah Diskon: " + diskon);
33        sc.close();
34    }
35 }
36
```

## Jobsheet 7

### Percobaan 1

1. Terdapat tiga komponen perulangan pada sintaks FOR. Berdasarkan Percobaan 1 tersebut, sebutkan dan tunjukkan masing-masing komponen perulangan FOR pada kode program yang telah dibuat!

int i = 1 adalah inisialisasi  
i<= 50 adalah kondisi  
i++ update

2. Jelaskan alur kerja dari potongan kode program berikut!

```
for (int i = 1; i <= 50; i++) {  
    if (i % kelipatan == 0) {  
        jumlah += i;  
        counter++;  
    }  
}
```

Jadi program tersebut merupakan perulangan dimana perulangan di lakukan dari 1 sampai 50.

Didalam kode tersebut ada if dimana jika i di modulus dengan kelipatan dan sisanya adalah 0, maka jumlah akan di tambah dengan jumlah + i  
Dan counternya akan ditambah 1

3. Modifikasi kode program yang telah dibuat dengan menambahkan variabel baru untuk menghitung rata-rata dari seluruh bilangan kelipatan yang ditentukan! Push dan commit kode program ke github.



```
import java.util.Scanner;  
  
class ForKelipatan25{  
    public static void main(String[] args) {  
        Scanner scan = new Scanner(System.in);  
        int kelipatan, jumlah = 0, counter = 0;  
  
        System.out.print("Masukkan bilangan kelipatan (1-9): ");  
        kelipatan = scan.nextInt();  
  
        for (int i = 1; i <= 50; i++) {  
            if (i % kelipatan == 0) {  
                jumlah += i;  
                counter++;  
            }  
        }  
  
        double avg = (double) jumlah / counter;  
  
        System.out.printf("Banyaknya bilangan %d dari 1 sampai 50 adalah %d\n", kelipatan, counter );  
        System.out.printf("Jumlah bilangan kelipatan %d adalah %d\n", kelipatan, jumlah );  
        System.out.print("Rata-rata bilangan kelipatan " + kelipatan + " adalah " + avg);  
    }  
}
```

- Buatlah file baru dengan nama WhileKelipatanNoAbsen.java. Buatlah kode program dengan tujuan serupa tetapi menggunakan WHILE. Push dan commit kode program ke github.



```

import java.util.Scanner;

class WhileKelipatan25{
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        int kelipatan, jumlah = 0, counter = 0;
        System.out.print("Masukkan bilangan kelipatan (1-9): ");
        kelipatan = scan.nextInt();
        int i = 1;
        while (i <= 50) {
            if (i % kelipatan == 0) {
                jumlah += i;
                counter++;
            }
            i++;
        }
        double avg = (double) jumlah / counter;
        System.out.printf("Banyaknya bilangan %d dari 1 sampai 50 adalah %d\n", kelipatan, counter );
        System.out.printf("Jumlah bilangan kelipatan %d adalah %d\n", kelipatan, jumlah );
        System.out.print("Rata-rata bilangan kelipatan " + kelipatan + " adalah " + avg);
    }
}

```

## Percobaan 2

- Tunjukkan bagian kode program yang digunakan sebagai syarat untuk menghentikan perulangan WHILE! Berapa kali perulangan dilakukan?

```
while (i < jumlahKaryawan) {
```

- Pada potongan kode berikut, Apa yang sebenarnya terjadi jika variabel jabatan berisi nilai “DIREKTUR”? Apa peran CONTINUE yang dituliskan di dalam sintaks perulangan?

```
if (jabatan.equalsIgnoreCase("direktur")) {
|   continue;
```

Jika berisii direktur maka dia akan menjalankan fungsi continue, Fungsi continue berfungsi untuk melanjutkan perulangan atau men-skip perulangan, sehingga dilanjutkan ke index berikutnya. Jadi setelah di skip dia tidak mengeksekusi kode dibawahnya didalam blok while.

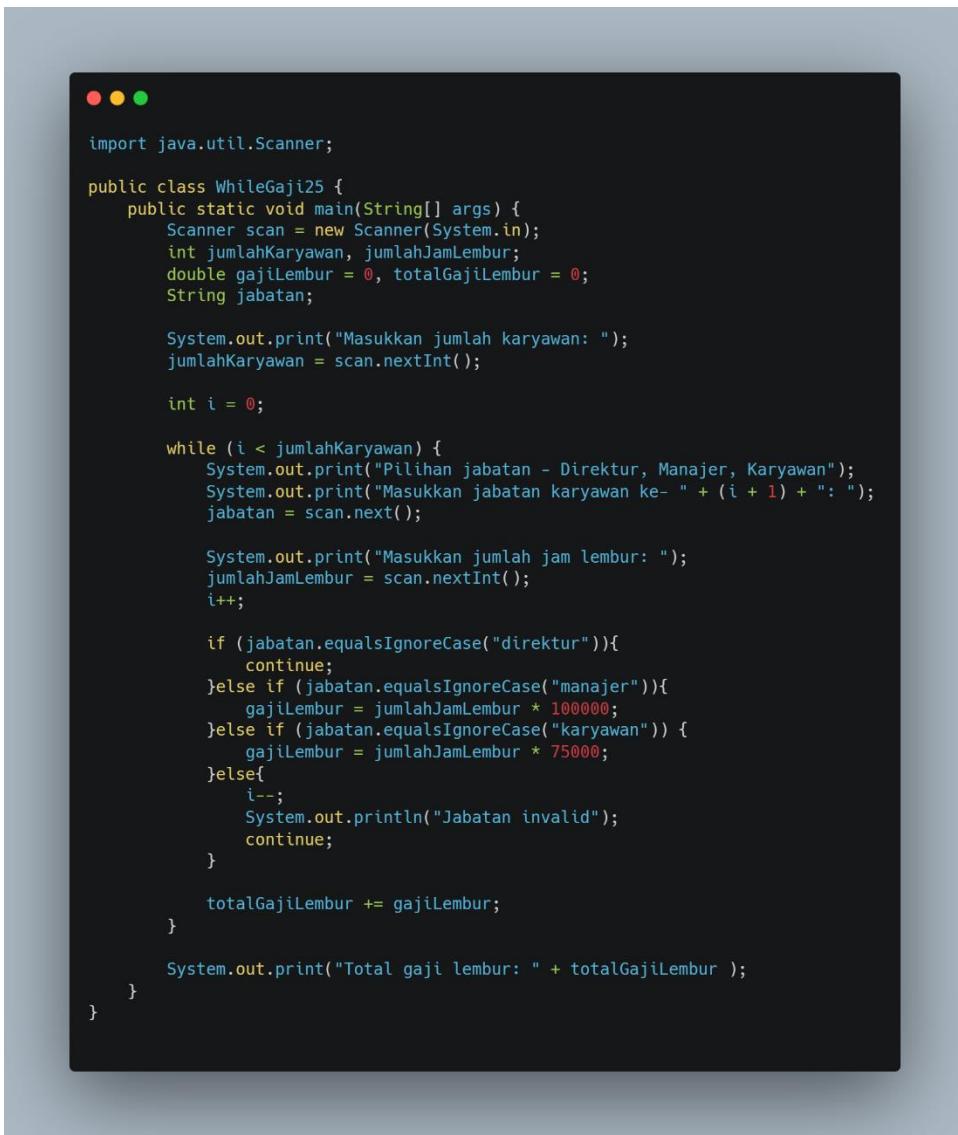
- Mengapa komponen update `i++` diletakkan di posisi tengah, tidak di bagian akhir statement? Pindahkan `i++` di bagian akhir, lalu jalankan kembali program dengan

memasukkan “direktur” sebagai jabatan karyawan pertama. Apa yang terjadi? Jelaskan!

Jika diletakkan dibawah, apabila jabatan = direktur maka perulangan akan lanjut ke index berikutnya tanpa mengeksekusi i++; alhasil jumlah perulangan akan semakin banyak.

4. Modifikasi kode program untuk menghandle jabatan yang invalid seperti contoh berikut:

```
Masukkan jumlah karyawan: 3
Pilihan jabatan - Direktur, Manajer, Karyawan
Masukkan jabatan karyawan ke-1: manajer
Masukkan jumlah jam lembur: 10
Pilihan jabatan - Direktur, Manajer, Karyawan
Masukkan jabatan karyawan ke-2: direktur
Masukkan jumlah jam lembur: 5
Pilihan jabatan - Direktur, Manajer, Karyawan
Masukkan jabatan karyawan ke-3: pegawai
Masukkan jumlah jam lembur: 4
Jabatan invalid
Pilihan jabatan - Direktur, Manajer, Karyawan
Masukkan jabatan karyawan ke-3: karyawan
Masukkan jumlah jam lembur: 4
Total gaji lembur: 1300000.0
```



```
import java.util.Scanner;

public class WhileGaji25 {
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        int jumlahKaryawan, jumlahJamLembur;
        double gajiLembur = 0, totalGajiLembur = 0;
        String jabatan;

        System.out.print("Masukkan jumlah karyawan: ");
        jumlahKaryawan = scan.nextInt();

        int i = 0;

        while (i < jumlahKaryawan) {
            System.out.print("Pilihan jabatan - Direktur, Manajer, Karyawan");
            System.out.print("Masukkan jabatan karyawan ke- " + (i + 1) + ": ");
            jabatan = scan.next();

            System.out.print("Masukkan jumlah jam lembur: ");
            jumlahJamLembur = scan.nextInt();
            i++;

            if (jabatan.equalsIgnoreCase("direktur")){
                continue;
            }else if (jabatan.equalsIgnoreCase("manajer")){
                gajiLembur = jumlahJamLembur * 100000;
            }else if (jabatan.equalsIgnoreCase("karyawan")) {
                gajiLembur = jumlahJamLembur * 75000;
            }else{
                i--;
                System.out.println("Jabatan invalid");
                continue;
            }

            totalGajiLembur += gajiLembur;
        }

        System.out.print("Total gaji lembur: " + totalGajiLembur );
    }
}
```

### Percobaan 3

1. Apa kegunaan sintaks BREAK di dalam sintaks perulangan?  
Break digunakan untuk menghentikan perulangan
2. Modifikasi kode program sehingga jika jumlah hari cuti yang ingin diambil lebih besar daripada jatah yang tersisa, program tidak berhenti sehingga pengguna masih memiliki kesempatan untuk mengisikan jumlah hari sesuai jatah cuti.

```
import java.util.Scanner;

public class DowhileCuti25 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        int jatahCuti, jumlahHari;
        String konfirmasi;

        System.out.print("Masukkan jatah cuti: ");
        jatahCuti = sc.nextInt();

        do{
            System.out.print("Apakah anda ingin mengambil cuti (y/t) ? ");
            konfirmasi = sc.nextLine();

            if (konfirmasi.equalsIgnoreCase("y")){
                System.out.print("Jumlah hari: ");
                jumlahHari = sc.nextInt();

                if (jumlahHari <= jatahCuti) {
                    jatahCuti -= jumlahHari;
                    System.out.println("Sisa jatah cuti anda adalah " + jatahCuti);

                    if (jatahCuti == 0) {
                        System.out.println("Jatah cuti anda sudah habis");
                        break;
                    }
                } else{
                    System.out.println("Sisa jatah cuti anda tidak mencukupi, tinggal sisa: " +
jatahCuti + " hari");
                }
            }
        }while(jatahCuti > 0);
    }
}
```

3. Push dan commit kode program ke github
4. Pada saat input konfirmasi, ketikkan “t”, apa yang terjadi? Mengapa demikian?  
Program akan terus berjalan, karena tidak ada fungsi break
5. Modifikasi kode program sehingga saat pengguna mengetikkan “t” sebagai input konfirmasi, maka program akan berhenti

```
● ● ●

import java.util.Scanner;

public class DoWhileCuti25 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        int jatahCuti, jumlahHari;
        String konfirmasi;

        System.out.print("Masukkan jatah cuti: ");
        jatahCuti = sc.nextInt();

        do{
            System.out.print("Apakah anda ingin mengambil cuti (y/t) ? ");
            konfirmasi = sc.nextLine();

            if (konfirmasi.equalsIgnoreCase("y")){
                System.out.print("Jumlah hari: ");
                jumlahHari = sc.nextInt();

                if (jumlahHari <= jatahCuti) {
                    jatahCuti -= jumlahHari;
                    System.out.println("Sisa jatah cuti anda adalah " + jatahCuti);

                    if (jatahCuti == 0) {
                        System.out.println("Jatah cuti anda sudah habis");
                        break;
                    }
                }else{
                    System.out.println("Sisa jatah cuti anda tidak mencukupi, tinggal sisa: " +
jatahCuti + " hari");
                }
            }else{
                break;
            }
        }while(jatahCuti > 0);
    }
}
```

## 6. Push dan commit kode program ke github

### Tugas

```
● ● ●

public class DeretBil25 {
    public static void main(String[] args) {
        int i = 25;
        int total = 0;

        while (i >= 1) {
            total += i;
            i--;
        }

        System.out.print("Total deret bilangan 25: " + total);
    }
}
```

```
public class Deret1sampai50 {
    public static void main(String[] args)
    {
        for (int i = 1; i <= 50; i++) {
            if (i % 3 != 0) {
                System.out.print(i + " ");
            }
        }
    }
}
```

```
public class Deret1sampai50 {
    public static void main(String[] args)
    {
        for (int i = 1; i <= 50; i++) {
            if (i % 3 != 0) {
                System.out.print(i + " ");
            }
        }
    }
}
```

## Jobsheet 9

### Percobaan 1

1. Isi masing-masing elemen array bil diubah dengan angka 5.0, 12867, 7.5, 2000000. Apa Jika yang terjadi? Mengapa bisa demikian?

Akan terjadi error “Type Missmatch” karena variable bil kita set int dimana seluruh element didalam array harus bertipe integer.

2. Modifikasi kode program di atas dengan melakukan inisialisasi elemen array sekaligus pada saat deklarasi array.

```
InsertArr.java > InsertArr > main(String[])
1 class InsertArr{
2
3     Run | Debug
4     public static void main(String[] args) {
5         int[] bil = {5, 13, -7, 17};
6
7         System.out.println(bil[0]);
8         System.out.println(bil[1]);
9         System.out.println(bil[2]);
10        System.out.println(bil[3]);
11    }
12 }
```

3. Ubah statement pada langkah No 4 menjadi seperti berikut Apa keluaran dari program? Jelaskan maksud dari statement tersebut.

```
InsertArr.java > InsertArr > main(String[])
1 class InsertArr{
2
3     Run | Debug
4     public static void main(String[] args) {
5         int[] bil = {5, 13, -7, 17};
6
7         for(int i=0; i<4; i++){
8             System.out.println(bil[i]);
9         }
10    }
11 }
```

Statement for digunakan untuk melakukan looping terhadap array. For akan melakukan iterasi setiap index array dan melakukan print disetiap index arraynya.

4. Jika kondisi pada statement for-loop di atas diubah menjadi:  $i \leq 4$ , apa keluaran dari program? Mengapa demikian?

Akan ada Error “Index 4 out of bounds”

Karena jumlah index arraynya Cuma sampai 3, sedangkan saat kita looping indexnya dari 0 sampai 4, sedangkan index 4 tidak ada.

5. Push dan commit kode program ke github.

## Percobaan 2

1. Ubah statement pada langkah nomor 5 menjadi seperti berikut ini

```
for (int i = 0; i < nilaiAkhir.length; i++) {
    System.out.print("Masukkan nilai akhir ke-" + i + " : ");
    nilaiAkhir[i] = sc.nextInt();
}
```

Jalankan program. Apakah terjadi perubahan? Mengapa demikian?

Sama saja, karena array.length digunakan untuk menghitung jumlah element yang ada di dalam array, yang dimana array nilaiAkhir sudah di set 10

2. Apa yang dimaksud dengan kondisi:  $i < nilaiAkhir.length$  ?

Jika nilai I kurang dari jumlah element yang ada di array nilaiAkhir

3. Ubah statement pada langkah nomor 6 menjadi seperti berikut ini,

```
for (int i = 0; i < nilaiAkhir.length; i++) {
    if (nilaiAkhir[i] > 70) {
        System.out.println("Mahasiswa ke-" + i + " lulus!");
    }
}
```

sehingga program hanya menampilkan nilai Mahasiswa yang lulus saja (yaitu mahasiswa yang memiliki nilai  $> 70$ ): Jalankan program dan jelaskan alur program!

```
ArrayNilai25.java > ArrayNilai25 > main(String[])
1  import java.util.Scanner;
2
3  public class ArrayNilai25 {
4      Run | Debug
5      public static void main(String[] args) {
6          Scanner sc = new Scanner(System.in);
7
8          int[] nilaiAkhir = new int[10];
9
10         for(int i=0; i<nilaiAkhir.length; i++) {
11             System.out.print("Masukkan nilai ke-" + (i+1) + ": ");
12             nilaiAkhir[i] = sc.nextInt();
13         }
14
15         for(int i=0; i<nilaiAkhir.length; i++) {
16             if (nilaiAkhir[i] > 70) {
17                 System.out.println("Nilai ke-" + (i+1) + ": " + nilaiAkhir[i] + " (lulus)");
18             }
19         }
20     }
21 }
22 }
```

Alur program diatas adalah, perulangan akan melakukan looping sesuai jumlah element yang ada di dalam array nilaiAkhir, lalu dia akan mengecek apakah nilaiAkhir setiap index nya lebih dari 70, jika iya maka akan print lulus, jika tidak maka tidak print apa-apa.

4. Modifikasi program agar menampilkan status kelulusan semua mahasiswa berdasarkan nilai, yaitu dengan menampilkan status mana mahasiswa yang lulus dan tidak lulus, seperti ilustrasi output berikut:

```
Masukkan nilai akhir ke-0 : 87
Masukkan nilai akhir ke-1 : 65
Masukkan nilai akhir ke-2 : 78
Masukkan nilai akhir ke-3 : 95
Masukkan nilai akhir ke-4 : 92
Masukkan nilai akhir ke-5 : 58
Masukkan nilai akhir ke-6 : 89
Masukkan nilai akhir ke-7 : 67
Masukkan nilai akhir ke-8 : 85
Masukkan nilai akhir ke-9 : 78
Mahasiswa ke-0 lulus!
Mahasiswa ke-1 tidak lulus!
Mahasiswa ke-2 lulus!
Mahasiswa ke-3 lulus!
Mahasiswa ke-4 lulus!
Mahasiswa ke-5 tidak lulus!
Mahasiswa ke-6 lulus!
Mahasiswa ke-7 tidak lulus!
Mahasiswa ke-8 lulus!
Mahasiswa ke-9 lulus!
```

```
File: ArrayNilai25.java > Run | Debug > main(String[])
1 import java.util.Scanner;
2
3 public class ArrayNilai25 {
4     public static void main(String[] args) {
5         Scanner sc = new Scanner(System.in);
6
7         int[] nilaiAkhir = new int[10];
8
9         for(int i=0; i<nilaiAkhir.length; i++) {
10            System.out.print("Masukkan nilai ke-" + (i+1) + ": ");
11            nilaiAkhir[i] = sc.nextInt();
12        }
13
14         for(int i=0; i<nilaiAkhir.length; i++) {
15             if (nilaiAkhir[i] > 70) {
16                 System.out.println("Nilai ke-" + (i) + " Lulus!");
17             } else{
18                 System.out.println("Nilai ke-" + (i) + " Tidak Lulus!");
19             }
20
21         }
22     }
23 }
24
```

5. Push dan commit kode program ke github.

### Percobaan 3

1. Modifikasi kode program pada praktikum percobaan 3 di atas (ArrayRataNilaiXX.java) agar program dapat menampilkan banyaknya mahasiswa yang lulus, yaitu mahasiswa yang memiliki lebih besar dari 70 (>70).

```

ArrayRata25.java > ArrayRata25 > main(String[])
1 import java.util.Scanner;
2
3 public class ArrayRata25 {
4     Run | Debug
5     public static void main(String[] args) {
6         Scanner sc = new Scanner(System.in);
7
8         int[] nilaiMhs = new int[10];
9         double total = 0;
10        double rata2;
11        int lulus = 0;
12
13        for(int i=0; i<nilaiMhs.length; i++) {
14            System.out.print("Masukkan nilai ke-" + (i+1) + ": ");
15            nilaiMhs[i] = sc.nextInt();
16            if (nilaiMhs[i] > 70) {
17                lulus++;
18            }
19        }
20        for(int i=0; i<nilaiMhs.length; i++) {
21            total += nilaiMhs[i];
22        }
23
24        rata2 = total / nilaiMhs.length;
25        System.out.println("Rata-rata nilai mahasiswa adalah: " + rata2);
26        System.out.println("Banyak Mahasiswa yang lulus: " + lulus);
27
28    }
29 }
30

```

```

s\program_286c8074\bin' 'ArrayRata25'
Masukkan nilai ke-1: 70
Masukkan nilai ke-2: 71
Masukkan nilai ke-3: 77
Masukkan nilai ke-4: 1
Masukkan nilai ke-5: 1
Masukkan nilai ke-6: 1
Masukkan nilai ke-7: 1
Masukkan nilai ke-8: 1
Masukkan nilai ke-9: 1
Masukkan nilai ke-10: 1
Rata-rata nilai mahasiswa adalah: 22.5
Banyak Mahasiswa yang lulus: 2
PS C:\Users\Acer\Tugas Kuliah\Semester 1\Pra

```

2. Modifikasi program pada praktikum percobaan 3 di atas (ArrayRataNilaiXX.java) sehingga program menerima jumlah elemen berdasarkan input dari pengguna dan mengeluarkan output seperti berikut ini:

```
Masukkan jumlah mahasiswa : 5
Masukkan nilai mahasiswa ke-1 : 80
Masukkan nilai mahasiswa ke-2 : 60
Masukkan nilai mahasiswa ke-3 : 90
Masukkan nilai mahasiswa ke-4 : 85
Masukkan nilai mahasiswa ke-5 : 65
Rata-rata nilai lulus = 85.0
Rata-rata nilai tidak lulus = 62.5
```



The screenshot shows a Java code editor with a dark theme. The code is a program named `ArrayRata25` that reads student scores from the console and calculates the average for those who passed (above 70) and failed (below or equal to 70). The code uses `Scanner` to read integers and `System.out.println` to output results.

```
1 import java.util.Scanner;
2
3 public class ArrayRata25 {
4     public static void main(String[] args) {
5         Scanner sc = new Scanner(System.in);
6
7         System.out.print("Masukkan jumlah siswa: ");
8         int jml = sc.nextInt();
9
10        int[] nilaiMhs = new int[jml];
11        double total = 0;
12        int lulus = 0;
13        double sumLulus = 0;
14        double sumTidakLulus = 0;
15
16        for(int i=0; i<nilaiMhs.length; i++) {
17            System.out.print("Masukkan nilai ke-" + (i+1) + ": ");
18            nilaiMhs[i] = sc.nextInt();
19            if (nilaiMhs[i] > 70) {
20                lulus++;
21                sumLulus += nilaiMhs[i];
22            } else{
23                sumTidakLulus += nilaiMhs[i];
24            }
25        }
26
27        for(int i=0; i<nilaiMhs.length; i++) {
28            total += nilaiMhs[i];
29        }
30
31        double rataLulus = sumLulus / lulus;
32        double rataTidakLulus = sumTidakLulus / (nilaiMhs.length - lulus);
33
34        System.out.println("Rata-rata nilai lulus = " + rataLulus);
35        System.out.println("Rata-rata nilai tidak lulus = " + rataTidakLulus);
36
37
38    }
39 }
40 }
```

3. Push dan commit kode program ke github.

## Percobaan 4 (Searching)

1. Jelaskan maksud dari statement **break**; pada baris ke-11 kode program percobaan 4 di atas.

Break digunakan untuk menghentikan perulangan, pada kode diatas perulangan akan berhenti saat array indeks ke l valuenya sama dengan key.

2. Modifikasi kode program pada percobaan 4 di atas sehingga program dapat menerima input berupa banyaknya elemen array, isi array, dan key yang ingin dicari. Lalu cetak ke layar indeks posisi elemen dari key yang dicari. Contoh hasil program:

```
Masukkan jumlah elemen array: 8
Masukkan elemen array ke-0 : 12
Masukkan elemen array ke-1 : 18
Masukkan elemen array ke-2 : -6
Masukkan elemen array ke-3 : 10
Masukkan elemen array ke-4 : 6
Masukkan elemen array ke-5 : 15
Masukkan elemen array ke-6 : 11
Masukkan elemen array ke-7 : 9
Masukkan key yang ingin dicari: 10
Key ada di posisi indeks ke-3
```

```
1 import java.util.Scanner;
2
3 public class LinearSearch25 {
4     public static void main(String[] args) {
5
6         Scanner sc = new Scanner(System.in);
7         int[] arrayInt;
8
9         System.out.print("Masukkan jumlah element: ");
10        int count = sc.nextInt();
11
12        arrayInt = new int[count];
13
14        for (int i = 0; i<arrayInt.length; i++) {
15            System.out.print("Masukkan bilangan: ");
16            int bil = sc.nextInt();
17
18            arrayInt[i] = bil;
19        }
20
21        System.out.print("Masukkan key yang ingin dicari: ");
22        int key = sc.nextInt();
23
24        for(int i=0; i<arrayInt.length; i++) {
25            if (arrayInt[i] == key) {
26                System.out.println("Key ada dalam posisi indeks ke- " + i);
27                break;
28            }
29        }
30    }
31 }
32 }
```

3. Modifikasi program pada percobaan 4 di atas, sehingga program akan memberikan pesan “key tidak ditemukan” jika key tidak ada di dalam array. Contoh tampilan program sebagai berikut:

```
Masukkan jumlah elemen array: 6
Masukkan elemen array ke-0 : 19
Masukkan elemen array ke-1 : 23
Masukkan elemen array ke-2 : 29
Masukkan elemen array ke-3 : 31
Masukkan elemen array ke-4 : 37
Masukkan elemen array ke-5 : 43
Masukkan key yang ingin dicari: 11
Key tidak ditemukan
```

```
1 import java.util.Scanner;
2
3 public class LinearSearch25 {
4     public static void main(String[] args) {
5
6         Scanner sc = new Scanner(System.in);
7         int[] arrayInt;
8
9         System.out.print("Masukkan jumlah element: ");
10        int count = sc.nextInt();
11
12        arrayInt = new int[count];
13
14        for (int i = 0; i<arrayInt.length; i++) {
15            System.out.print("Masukkan bilangan: ");
16            int bil = sc.nextInt();
17
18            arrayInt[i] = bil;
19        }
20
21        System.out.print("Masukkan key yang ingin dicari: ");
22        int key = sc.nextInt();
23
24        for(int i=0; i<arrayInt.length; i++) {
25            if (arrayInt[i] == key) {
26                System.out.println("Key ada dalam posisi indeks ke- " + i);
27                break;
28            }else{
29                System.out.println("Key tidak ditemukan");
30                break;
31            }
32        }
33    }
34 }
35
```

#### 4. Push dan commit kode program ke github.

### Tugas

1. Buat program untuk menghasilkan nilai tertinggi, nilai terendah, dan rata-rata dari suatu array berisi bilangan bertipe integer.

Ketentuan:

- Input: Banyaknya elemen, nilai tiap elemen

– Output: Nilai tertinggi, nilai terendah, nilai rata-rata

```
1 import java.util.Arrays;
2 import java.util.Scanner;
3
4 public class Tugas25 {
5     public static void main(String[] args) {
6         Scanner sc = new Scanner(System.in);
7         int total = 0;
8
9         System.out.print("Masukkan Jumlah Element: ");
10        int count = sc.nextInt();
11
12        int arr[] = new int[count];
13        int temp = 0;
14
15        for (int i = 0; i<arr.length; i++) {
16            System.out.print("Masukkan Bilangan: ");
17            int bil = sc.nextInt();
18
19            arr[i] = bil;
20
21            total += bil;
22        }
23
24
25        for(int i=0; i<arr.length; i++) {
26            for(int j=1; j<(arr.length-i); j++) {
27                if(arr[j-1] > arr[j]) {
28                    temp = arr[j-1];
29                    arr[j-1] = arr[j];
30                    arr[j] = temp;
31
32                }
33            }
34        }
35
36        System.out.println("Nilai terendah: " + arr[0]);
37        System.out.println("Nilai rata-rata: " + (double) total / arr.length);
38        System.out.println("Nilai tertinggi: " + arr[arr.length - 1]);
39
40        sc.close();
41    }
42 }
43
44 }
```

```
Masukkan Jumlah Element: 5
Masukkan Bilangan: 5
Masukkan Bilangan: 4
Masukkan Bilangan: 3
Masukkan Bilangan: 2
Masukkan Bilangan: 1
Nilai terendah: 1
Nilai rata-rata: 3.0
Nilai tertinggi: 5
PS C:\Users\Acer\Tugas Kuliah\Semester
```

## Jobsheet 10

### Percobaan 1

1. Apakah pengisian elemen array harus dilakukan secara berurutan mulai dari indeks ke-0? Jelaskan!

Tidak, pengisian elemen array tidak harus dilakukan secara berurutan mulai dari indeks '0'. Pengisian elemen array bebas tergantung kebutuhan logika kita.

2. Mengapa terdapat `null` pada daftar nama penonton?  
Karena elemen-elemen array yang tidak diinisialisasi akan memiliki nilai 'null' untuk tipe data referensi seperti String.
3. Lengkapi daftar penonton pada langkah ke-4 sebagai berikut

```
penonton[0][0] = "Amin";
penonton[0][1] = "Bena";
penonton[1][0] = "Candra";
penonton[1][1] = "Dela";
penonton[2][0] = "Eka";
penonton[2][1] = "Farhan";
penonton[3][0] = "Gisel";
penonton[3][1] = "Hana";
```

```
Bioskop25.java
1  public class Bioskop25 {
2      Run | Debug
3      public static void main(String[] args) {
4          String[][] penonton = new String[4][2];
5          penonton[0][0] = "Ammin";
6          penonton[0][1] = "Bena";
7          penonton[1][0] = "Candra";
8          penonton[1][1] = "Dela";
9          penonton[2][0] = "Eka";
10         penonton[2][1] = "Farhan";
11         penonton[3][0] = "Gisel";
12         penonton[3][1] = "Hana";
13         System.out.printf("%s \t %s\n", penonton[0][0], penonton[0][1]);
14         System.out.printf("%s \t %s\n", penonton[1][0], penonton[1][1]);
15         System.out.printf("%s \t %s\n", penonton[2][0], penonton[2][1]);
16         System.out.printf("%s \t %s\n", penonton[3][0], penonton[3][1]);
17     }
18 }
19
20
21
22
```

4. Tambahkan kode program sebagai berikut:

```
System.out.println(penonton.length);
System.out.println(penonton[0].length);
System.out.println(penonton[1].length);
System.out.println(penonton[2].length);
System.out.println(penonton[3].length);
```

Jelaskan fungsi dari `penonton.length` dan `penonton[0].length`!

Apakah `penonton[0].length`, `penonton[1].length`, `penonton[2].length`, dan `penonton[3].length` memiliki nilai yang sama? Mengapa?

The screenshot shows a Java code editor with a dark theme. The code in the editor is as follows:

```
1  public class Bioskop25 {
2      public static void main(String[] args) {
3          String[][] penonton = new String[4][2];
4
5          penonton[0][0] = "Ammin";
6          penonton[0][1] = "Bena";
7          penonton[1][0] = "Candra";
8          penonton[1][1] = "Dela";
9          penonton[2][0] = "Eka";
10         penonton[2][1] = "Farhan";
11         penonton[3][0] = "Gisel";
12         penonton[3][1] = "Hana";
13
14         System.out.printf("%s \t %s\n", penonton[0][0], penonton[0][1]);
15         System.out.printf("%s \t %s\n", penonton[1][0], penonton[1][1]);
16         System.out.printf("%s \t %s\n", penonton[2][0], penonton[2][1]);
17         System.out.printf("%s \t %s\n", penonton[3][0], penonton[3][1]);
18
19         System.out.println(penonton.length);
20         System.out.println(penonton[0].length);
21         System.out.println(penonton[1].length);
22         System.out.println(penonton[2].length);
23         System.out.println(penonton[3].length);
24
25     }
26
27 }
```

The code defines a 2D string array named `penonton` with 4 rows and 2 columns. It then prints each row and its length. The code editor interface includes a sidebar with icons for file operations, a search bar, and a terminal tab at the bottom.

- `Penonton.length` : memberikan Panjang dari dimensi pertama array `penonton`, adalah 4. Ini mengindikasikan jumlah baris dalam array.
- `Penonton[0].length` : memberikan Panjang dari dimensi kedua array `penonton`, pada indeks ke-0 adalah 3. Ini mengindikasikan jumlah kolom pada baris pertama.
- `Penonton[1].length` : memberikan Panjang dari dimensi kedua array `penonton`, pada indeks ke-1 adalah 3. Ini mengindikasikan jumlah kolom pada baris kedua.
- `Penonton[2].length` : memberikan Panjang dari dimensi kedua array `penonton`, pada indeks ke-2 adalah 3. Ini mengindikasikan jumlah kolom pada baris ketiga.
- `Penonton[3].length` : memberikan Panjang dari dimensi kedua array `penonton`, pada indeks ke-3 adalah 3. Ini mengindikasikan jumlah kolom pada baris keempat.

5. Modifikasi kode program pada langkah 4 untuk menampilkan panjang setiap baris pada array menggunakan for loop. Compile, run, lalu lakukan commit.

```
System.out.println(penonton.length);

for (int i = 0; i < penonton.length; i++) {
    System.out.println("Panjang baris ke-" + (i+1) + " : " + penonton[i].length);
}

Bioskop25.java x
Bioskop25.java > Bioskop25 > main(String[])
7     penonton[0][1] = "Bena";
8     penonton[1][0] = "Candra";
9     penonton[1][1] = "Dela";
10    penonton[2][0] = "Eka";
11    penonton[2][1] = "Farhan";
12    penonton[3][0] = "Gisel";
13    penonton[3][1] = "Hana";
14
15    System.out.printf("%s \t %s\n", penonton[0][0], penonton[0][1]);
16    System.out.printf("%s \t %s\n", penonton[1][0], penonton[1][1]);
17    System.out.printf("%s \t %s\n", penonton[2][0], penonton[2][1]);
18    System.out.printf("%s \t %s\n", penonton[3][0], penonton[3][1]);
19
20    // System.out.println(penonton.length);
21    // System.out.println(penonton[0].length);
22    // System.out.println(penonton[1].length);
23    // System.out.println(penonton[2].length);
24    // System.out.println(penonton[3].length);
25
26    for (int i = 0; i < penonton.length; i++) {
27        System.out.println("Panjang baris ke-" + (i+1) + " : " + penonton[i].length);
28    }
29
```

```
25'
Ammin      Bena
Candra     Dela
Eka        Farhan
Gisel      Hana
Panjang baris ke-1 : 2
Panjang baris ke-2 : 2
Panjang baris ke-3 : 2
Panjang baris ke-4 : 2
PS C:\Users\Acer\Tugas Kuliah\S
```

6. Modifikasi kode program pada langkah 5 untuk menampilkan panjang setiap baris pada array menggunakan foreach loop. Compile, run, lalu lakukan commit.

```
Bioskop25.java x
Bioskop25 > main(String[])
penonton[0][0] - nida ,  
14  
15     System.out.printf("%s \t %s\n", penonton[0][0], penonton[0][1]);  
16     System.out.printf("%s \t %s\n", penonton[1][0], penonton[1][1]);  
17     System.out.printf("%s \t %s\n", penonton[2][0], penonton[2][1]);  
18     System.out.printf("%s \t %s\n", penonton[3][0], penonton[3][1]);  
19  
20     // System.out.println(penonton.length);  
21     // System.out.println(penonton[0].length);  
22     // System.out.println(penonton[1].length);  
23     // System.out.println(penonton[2].length);  
24     // System.out.println(penonton[3].length);  
25  
26     for (String[] a : penonton) {  
27         System.out.println("Panjang baris" + " : " + a.length);  
28     }  
29  
30 }  
31 }  
32 }  
33 }
```

```
am' ; & 'C:\Program Files\Zulu\zulu-11\bin\java.exe
ata\Roaming\Code\User\workspaceStorage\d107de9a9bt.java\jdt_ws\program_12ae1d8e\bin' 'Bioskop25'
Ammin    Bena
Candra   Dela
Eka       Farhan
Gisel     Hana
Panjang baris : 2
Panjang baris : 2
Panjang baris : 2
Panjang baris : 2
PS C:\Users\Acer\Tugas Kuliah\Semester 1\Praktek
```

---

```
System.out.println(penonton.length);

for (String[] barisPenonton : penonton) {
    System.out.println("Panjang baris: " + barisPenonton.length);
}
```

7. Menurut Anda, apa kekurangan dan kelebihan foreach loop dibandingkan dengan forloop?

Kelebihan foreach loop:

- Lebih mudah dibaca dan ditulis.
- Lebih efisien dalam beberapa kasus.

Kekurangan foreach loop:

- Tidak memberikan kontrol yang sama seperti for loop.
- Tidak dapat digunakan untuk melakukan operasi pada kumpulan secara keseluruhan.
- Tidak bisa mengakses index

8. Berapa indeks baris maksimal untuk array penonton?

4 => 0, 1, 2, 3

9. Berapa indeks kolom maksimal untuk array penonton?

3 => 0, 1, 2

10. Tambahkan kode program untuk menampilkan nama penonton pada baris ke-3 menggunakan for loop. Compile, run, lalu lakukan commit.

```
System.out.println("Penonton pada baris ke-3: ");

for (int i = 0; i < penonton[2].length; i++) {
    System.out.println(penonton[2][i]);
}
```

```
Bioskop25.java
Bioskop25.java > Bioskop25 > main(String[])
14
15     System.out.printf("%s \t %s\n", penonton[0][0], penonton[0][1]);
16     System.out.printf("%s \t %s\n", penonton[1][0], penonton[1][1]);
17     System.out.printf("%s \t %s\n", penonton[2][0], penonton[2][1]);
18     System.out.printf("%s \t %s\n", penonton[3][0], penonton[3][1]);
19
20     // System.out.println(penonton.length);
21     // System.out.println(penonton[0].length);
22     // System.out.println(penonton[1].length);
23     // System.out.println(penonton[2].length);
24     // System.out.println(penonton[3].length);
25
26     System.out.print("Panjang baris ke-3");
27     for (int i = 0; i < penonton[2].length; i++) {
28         System.out.print(" : " + penonton[2][i]);
29     }
30
31     for (String[] a : penonton) {
32         System.out.println("Panjang baris" + " : " + a.length);
33     }
34
35
36
```

PROBLEMS 1 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\Acer\Tugas Kuliah\Semester 1\Praktek Daspro\jobsheet10\program>
c:; cd 'c:\Users\Acer\Tugas Kuliah\Semester 1\Praktek Daspro\jobsheet10\program'; & 'C:\Program Files\Zulu\zulu-11\bin\java.exe' '-cp' 'C:\Users\Acer\AppData\Roaming\Code\User\workspaceStorage\d107de9a9b1830da50d676d20b214a9a\redhat.java\jdt_ws\program_12ae1d8e\bin' 'Bioskop25'
Ammin  Béna
Candra  Dela
Eka      Farhan
Gisel   Hana
Panjang baris ke-3 : Eka : FarhanPanjang baris : 2
Panjang baris : 2
Panjang baris : 2
Panjang baris : 2
PS C:\Users\Acer\Tugas Kuliah\Semester 1\Praktek Daspro\jobsheet10\program>
```

11. Modifikasi kode pada langkah 10 menjadi perulangan dengan for each loop.

Compile, run, lalu lakukan commit.

```
System.out.println("Penonton pada baris ke-3: ");

for (String i : penonton[2]) {
    System.out.println(i);
}
```

The screenshot shows a Java code editor with a file named 'Bioskop25.java'. The code prints a 2D array of names. A yellow warning icon is on line 27. The output terminal shows the names Ammin, Bena, Candra, Dela, Eka, Farhan, Gisel, and Hana, followed by their respective array lengths (2, 2, 2, 2). A powershell terminal window is also visible.

```
Bioskop25.java X
Bioskop25.java > Bioskop25 > main(String[])
13     penonton[3][1] = null;
14
15     System.out.printf(format:"%s \t %s\n", penonton[0][0], penonton[0][1]);
16     System.out.printf(format:"%s \t %s\n", penonton[1][0], penonton[1][1]);
17     System.out.printf(format:"%s \t %s\n", penonton[2][0], penonton[2][1]);
18     System.out.printf(format:"%s \t %s\n", penonton[3][0], penonton[3][1]);
19
20     // System.out.println(penonton.length);
21     // System.out.println(penonton[0].length);
22     // System.out.println(penonton[1].length);
23     // System.out.println(penonton[2].length);
24     // System.out.println(penonton[3].length);
25
26     System.out.print(s."Panjang baris ke-3");
27     for (String i : penonton[2]) {
28         System.out.print(i);
29     }
30     for (String[] a : penonton) {
31         System.out.println("Panjang baris" + " : " + a.length);
32     }
33
34
35 }
```

PROBLEMS 1 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\Acer\Tugas Kuliah\Semester 1\Praktek Daspro\jobsheet10\program>
c;; cd 'c:\Users\Acer\Tugas Kuliah\Semester 1\Praktek Daspro\jobsheet10\program'; & 'C:\Program Files\Zulu\zulu-11\bin\java.exe' '-cp' 'C:\Users\Acer\AppData\Roaming\Code\User\workspaceStorage\d107de9a9b1830da50d676d20b214a9a\redhat.java\jdt_ws\program_12ae1d8e\bin' 'Bioskop25'
Ammin Bena
Candra Dela
Eka Farhan
Gisel Hana
Panjang baris ke-3EkaFarhanPanjang baris : 2
Panjang baris : 2
Panjang baris : 2
Panjang baris : 2
PS C:\Users\Acer\Tugas Kuliah\Semester 1\Praktek Daspro\jobsheet10\program>
```

12. Modifikasi kembali kode program pada langkah 11 untuk menampilkan nama penonton untuk setiap baris. Compile dan run program kemudian amati hasilnya. Lakukan commit.

```
for (int i = 0; i < penonton.length; i++) {
    System.out.println("Penonton pada baris ke-" + (i+1) + ": " + String.join(", ", penonton[i]));
}
```

```

Bioskop25.java
Bioskop25.java > Bioskop25 > main(String[])
penonton[0][0] = "Ammin"
penonton[0][1] = "Bena"
penonton[1][0] = "Candra"
penonton[1][1] = "Dela"
penonton[2][0] = "Eka"
penonton[2][1] = "Farhan"
penonton[3][0] = "Gisel"
penonton[3][1] = "Hana"

System.out.printf("%s \t %s\n", penonton[0][0], penonton[0][1]);
System.out.printf("%s \t %s\n", penonton[1][0], penonton[1][1]);
System.out.printf("%s \t %s\n", penonton[2][0], penonton[2][1]);
System.out.printf("%s \t %s\n", penonton[3][0], penonton[3][1]);

// System.out.println(penonton.length);
// System.out.println(penonton[0].length);
// System.out.println(penonton[1].length);
// System.out.println(penonton[2].length);
// System.out.println(penonton[3].length);

for (int i = 0; i < penonton.length; i++) {
    System.out.println("Penonton pada baris ke-" + i + " : " + String.join(delimiter:", ", penonton[i]));
}

System.out.print(s:"Panjang baris ke-3");
for (String i : penonton[2]) {
    System.out.print(i);
}
for (String[] a : penonton) {
    System.out.println("Panjang baris" + " : " + a.length);
}

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```

t.java\jdt ws\program_12ae1d8e\bin' 'Bioskop25'
Ammin Bena
Candra Dela
Eka Farhan
Gisel Hana
Penonton pada baris ke-0 : Ammin, Bena
Penonton pada baris ke-1 : Candra, Dela
Penonton pada baris ke-2 : Eka, Farhan
Penonton pada baris ke-3 : Gisel, Hana
Panjang baris ke-3EkaFarhanPanjang baris : 2
Panjang baris : 2
Panjang baris : 2
Panjang baris : 2
PS C:\Users\Acer\tugas_kuliah\Semester_1\Praktek Dasar\jobsheet10\program>

```

### 13. Apa fungsi dari `String.join()`?

Fungsi dari `String.join()` adalah menggabungkan elemen-elemen suatu array menjadi

satu string, dengan menggunakan delimiter tertentu di antara setiap pasangan array.

### 14. Commit dan push ke github

## Percobaan 2

- Apakah pengisian elemen array dari scanner harus dilakukan secara berurutan mulai dari indeks ke-0? Jelaskan!

Tidak, Pengisian bisa dilakukan di index mana saja sesuai selera.

- Modifikasi kode program untuk memberikan opsi menu sebagai berikut:

- Menu 1: Input data penonton
- Menu 2: Tampilkan daftar penonton
- Menu 3: Exit

The screenshot shows a Java code editor with a dark theme. The code is contained within a class named `BioskopWithScanner25`. The program starts by importing the `java.util.Scanner` class. It then defines a `main` method that initializes a `Scanner` object to read input from the console. A `String` array `penonton` is created with dimensions `4x2`. The program enters a `while` loop that continues until the user selects option 3 (Exit). Inside the loop, it prints a menu with three options: 1. Input data penonton, 2. Tampilkan daftar penonton, and 3. Exit. It then prompts the user to choose an option. If the choice is 1, it asks for the name and seat number (baris/kolom) and stores them in the `penonton` array. If the choice is 2, it prints the entire `penonton` array. If the choice is 3, it prints a message and exits the loop. Any other choice is treated as invalid.

```
1 import java.util.Scanner;
2
3 public class BioskopWithScanner25 {
4     public static void main(String[] args) {
5         Scanner input = new Scanner(System.in);
6         String[][] penonton = new String[4][2];
7
8         while (true) {
9             System.out.println("Menu:");
10            System.out.println("1. Input data penonton");
11            System.out.println("2. Tampilkan daftar penonton");
12            System.out.println("3. Exit");
13            System.out.print("Pilih menu (1-3): ");
14
15            int menuChoice = input.nextInt();
16            input.nextLine(); // Membersihkan newline
17
18            if (menuChoice == 1) {
19                System.out.println("Masukkan nama penonton: ");
20                String nama = input.nextLine();
21                System.out.println("Masukkan baris: ");
22                int baris = input.nextInt();
23                System.out.println("Masukkan kolom: ");
24                int kolom = input.nextInt();
25
26                // Membersihkan newline
27                input.nextLine();
28
29                penonton[baris - 1][kolom - 1] = nama;
30
31                System.out.println("Data penonton berhasil dimasukkan.");
32            } else if (menuChoice == 2) {
33                System.out.println("Daftar penonton:");
34                for (int i = 0; i < penonton.length; i++) {
35                    for (int j = 0; j < penonton[i].length; j++) {
36                        System.out.print(penonton[i][j] + "\t");
37                    }
38                    System.out.println();
39                }
40            } else if (menuChoice == 3) {
41                System.out.println("Terima kasih. Program selesai.");
42                break;
43            } else {
44                System.out.println("Menu tidak valid. Silakan pilih lagi.");
45            }
46        }
47    }
48 }
49 }
```

3. Modifikasi kode program untuk menghandle apabila nomor baris/kolom kursi tidak tersedia

```
Bioskop25.java | BioskopWithScanner25.java 1 | main(String[])
BioskopWithScanner25.java > BioskopWithScanner25 > main(String[])
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
String nama = input.nextLine();
System.out.println("Masukkan baris (1-4): ");
int baris = input.nextInt();
if (baris < 1 || baris > penonton.length) {
    System.out.println("Nomor baris tidak valid. Silakan pilih lagi.");
    continue;
}

System.out.println("Masukkan kolom (1-2): ");
int kolom = input.nextInt();
if (kolom < 1 || kolom > penonton[baris - 1].length) {
    System.out.println("Nomor kolom tidak valid. Silakan pilih lagi.");
    continue;
}

// Membersihkan newline
input.nextLine();

if (penonton[baris - 1][kolom - 1] != null) {
    System.out.println("Maaf, kursi tersebut sudah terisi. Silakan pilih kursi lain.");
} else {
    penonton[baris - 1][kolom - 1] = nama;
    System.out.println("Data penonton berhasil dimasukkan.");
}
} else if (menuChoice == 2) {
    System.out.println("Daftar penonton:");
}
}

Masukkan kolom (1-2):
4
Nomor kolom tidak valid. Silakan pilih lagi.
Menu:
```

4. Pada menu 1, modifikasi kode program untuk memberikan warning apabila kursi yang dipilih sudah terisi oleh penonton lainnya lalu munculkan perintah untuk memasukkan baris dan kolom kembali

```

1 terminal Help ← → ⌂ program
2 Bioskop25.java BioskopWithScanner25.java 1 ✎
3 BioskopWithScanner25.java > ↗ BioskopWithScanner25 > main(String[])
4
5     String nama = input.nextLine();
6
7     System.out.println("Masukkan baris (1-4): ");
8     int baris = input.nextInt();
9     if (baris < 1 || baris > penonton.length) {
10         System.out.println("Nomor baris tidak valid. Silakan pilih lagi.");
11         continue;
12     }
13
14     System.out.println("Masukkan kolom (1-2): ");
15     int kolom = input.nextInt();
16     if (kolom < 1 || kolom > penonton[baris - 1].length) {
17         System.out.println("Nomor kolom tidak valid. Silakan pilih lagi.");
18         continue;
19     }
20
21     // Membersihkan newline
22     input.nextLine();
23
24     if (penonton[baris - 1][kolom - 1] != null) {
25         System.out.println("Maaf, kursi tersebut sudah terisi. Silakan pilih kursi lain.");
26     } else {
27         penonton[baris - 1][kolom - 1] = nama;
28         System.out.println("Data penonton berhasil dimasukkan.");
29     }
30
31 } else if (menuChoice == 2) {
32     System.out.println("Daftar penonton:");
33
34     for (int i = 0; i < penonton.length; i++) {
35         for (int j = 0; j < penonton[i].length; j++) {
36             if (penonton[i][j] != null) {
37                 System.out.print(penonton[i][j] + "\t");
38             } else {
39                 System.out.print("****\t");
40             }
41         }
42         System.out.println();
43     }
44
45 } else if (menuChoice == 3) {
46     System.out.println("Terima kasih. Program selesai.");
47     break;
48 } else {
49     System.out.println("Menu tidak valid. Silakan pilih lagi.");
50 }
51
52
53
54
55
56
57
58
59
60
61

```

PROBLEMS 2 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```

1
1 Masukkan kolom (1-2):
1
1 Maaf, kursi tersebut sudah terisi. Silakan pilih kursi lain.
Menu:
1. Input data penonton

```

##### 5. Pada menu 2, jika kursi kosong, ganti `null` dengan `***`

```

1 terminal Help ← → ⌂ program
2 Bioskop25.java BioskopWithScanner25.java 1 ✎
3 BioskopWithScanner25.java > ↗ BioskopWithScanner25
4
5     String nama = input.nextLine();
6
7     if (penonton[baris - 1][kolom - 1] != null) {
8         System.out.println("Maaf, kursi tersebut sudah terisi. Silakan pilih kursi lain.");
9     } else {
10        penonton[baris - 1][kolom - 1] = nama;
11        System.out.println("Data penonton berhasil dimasukkan.");
12    }
13
14    } else if (menuChoice == 2) {
15        System.out.println("Daftar penonton:");
16        for (int i = 0; i < penonton.length; i++) {
17            for (int j = 0; j < penonton[i].length; j++) {
18                if (penonton[i][j] != null) {
19                    System.out.print(penonton[i][j] + "\t");
20                } else {
21                    System.out.print("****\t");
22                }
23            }
24            System.out.println();
25        }
26
27    } else if (menuChoice == 3) {
28        System.out.println("Terima kasih. Program selesai.");
29        break;
30    } else {
31        System.out.println("Menu tidak valid. Silakan pilih lagi.");
32    }
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
59
60
61

```

PROBLEMS 2 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```

Ahmed ****
**** ****
**** ****
**** ****
Menu:
1. Input data penonton
2. Tampilkan daftar penonton
3. Exit
Pilih menu (1-3): 1

```

##### 6. Commit dan push kode program ke github.

## Percobaan 3

1. Tambahkan kode program sebagai berikut

```
for (int i = 0; i < myNumbers.length; i++) {  
    System.out.println(Arrays.toString(myNumbers[i]));  
}
```

2. Apa fungsi dari `Arrays.toString()`?

untuk mengonversi sebuah array menjadi representasi string sehingga bisa dicetak.

3. Apa nilai default untuk elemen pada array dengan tipe data int?

Nilai defaultnya adalah 0.

4. Tambahkan kode program berikut

```
for (int i = 0; i < myNumbers.length; i++) {  
    System.out.println("Panjang baris ke-" + (i+1) + ": " + myNumbers[i].length);  
}
```

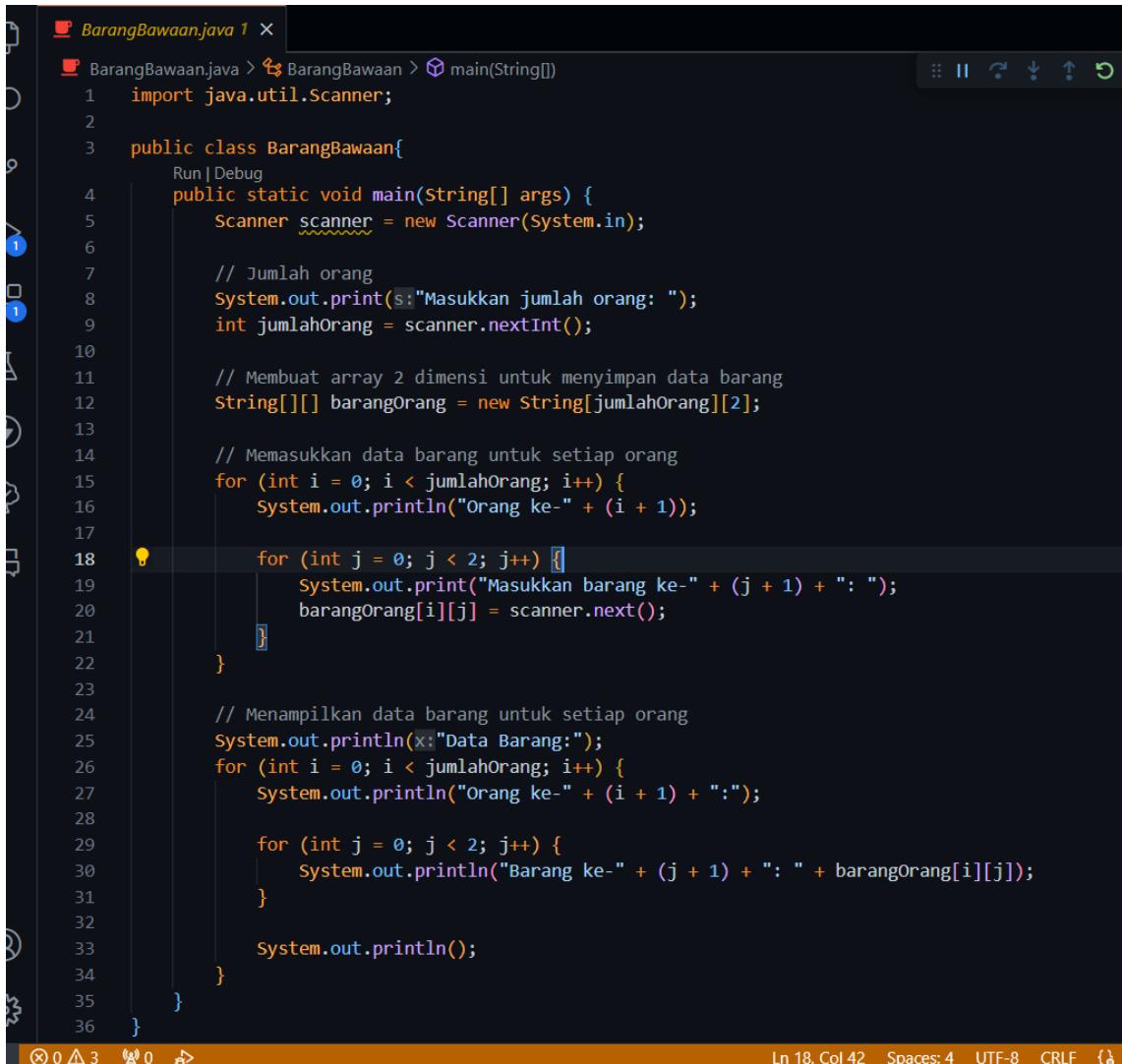
The screenshot shows a Java development environment with several tabs open. The active tab is `Numbers25.java`. The code defines a class `Numbers25` with a `main` method. Inside the `main` method, an array `myNumbers` is declared and initialized with three rows of different lengths: 5, 3, and 1 respectively. A `for` loop then iterates through each row, printing its length. The output window below shows the results: `[0, 0, 0]`, followed by the printed output: `Panjang baris ke-2: 1`, `[0]`, and `PS C:\Users\Acer\Kuliah\Semester 1\Praktek Daspro\jobsheet10\program>`.

5. Array `myNumbers` memiliki `length` berbeda untuk setiap barisnya. Bagaimana cara agar `length` untuk setiap baris sama? Apakah panjang array dapat dimodifikasi?

Harus menentukan diawal agar pannjangnya sama, kalau sudah ditentukan tidak dapat dimodifikasi

## Tugas Kelompok

- Implementasikan flowchart yang telah dibuat pada tugas pertemuan 10 mata kuliah Dasar Pemrograman terkait project kelompok ke dalam kode program Java.
- Push dan commit kode program Anda ke repository github untuk final project
- Catatan: tugas hanya boleh menerapkan materi dari pertemuan 1 hingga pertemuan 10



```
BarangBawaan.java 1 X
BarangBawaan.java > BarangBawaan > main(String[])
1 import java.util.Scanner;
2
3 public class BarangBawaan{
4     Run | Debug
5     public static void main(String[] args) {
6         Scanner scanner = new Scanner(System.in);
7
8         // Jumlah orang
9         System.out.print("Masukkan jumlah orang: ");
10        int jumlahOrang = scanner.nextInt();
11
12        // Membuat array 2 dimensi untuk menyimpan data barang
13        String[][] barangOrang = new String[jumlahOrang][2];
14
15        // Memasukkan data barang untuk setiap orang
16        for (int i = 0; i < jumlahOrang; i++) {
17            System.out.println("Orang ke-" + (i + 1));
18
19            for (int j = 0; j < 2; j++) {
20                System.out.print("Masukkan barang ke-" + (j + 1) + ": ");
21                barangOrang[i][j] = scanner.next();
22            }
23
24            // Menampilkan data barang untuk setiap orang
25            System.out.println("Data Barang:");
26            for (int i = 0; i < jumlahOrang; i++) {
27                System.out.println("Orang ke-" + (i + 1) + ":");

28                for (int j = 0; j < 2; j++) {
29                    System.out.println("Barang ke-" + (j + 1) + ": " + barangOrang[i][j]);
30                }
31
32            System.out.println();
33        }
34    }
35}
36
```

Ln 18, Col 42 Spaces: 4 UTF-8 CRLF { }

```
\Code\Users\workspaceStorage\d107de9a9b1830da50d676d20b214a9a\redhat.java\jdt_ws\program_12ae1d8e\bin' 'BarangBawaan'
Masukkan jumlah orang: 2
Orang ke-1
Masukkan barang ke-1: Apel
Masukkan barang ke-2: Tisu
Orang ke-2
Masukkan barang ke-1: Mangga
Masukkan barang ke-2: Pisau
Data Barang:
Orang ke-1:
Barang ke-1: Apel
Barang ke-2: Tisu

Orang ke-2:
Barang ke-1: Mangga
Barang ke-2: Pisau

PS C:\Users\Acer\Tugas Kuliah\Semester 1\Praktek Daspro\jobsheet10\program> █
```

## Jobsheet 11

### Percobaan 1

1. Jika pada perulangan for, inisialisasi **i=1** diubah menjadi **i=0**, apa yang akibatnya? Mengapa bisa demikian?  
Jumlah bintangnya jadi 6, karena indexnya dimulai dari 0-6
2. Jika pada perulangan for, kondisi **i <= N** diubah menjadi **i > N**, apa akibatnya? Mengapa bisa demikian?  
Tidak terjadi perulangan karena nilai i tidak lebih dari n
3. Jika pada perulangan for, kondisi step **i++** diubah menjadi **i--** apa akibatnya? Mengapa bisa demikian? Akan terjadi infinity loop karena nilai i akan selalu kurang dari N

### Percobaan 2

1. Perhatikan perulangan luar. Jika pada sintaks **for**, inisialisasi **iOuter=1** diubah menjadi **iOuter=0**, apa yang akibatnya? Mengapa bisa demikian?  
**Akan ada 6 baris bintang karena index perulangan iOuter dimulai dari 0-5**
2. Kembalikan program semula dimana inisialisasi **iOuter=1**.  
Kemudian perhatikan perulangan dalam, Jika pada sintaks **for**, inisialisasi **i=1** diubah menjadi **i=0**, apa yang akibatnya? Mengapa bisa demikian?  
**Akan ada 6 bintang disetiap barisnya, karena perulangannya dimulai dari 0- 6**
3. Jadi, apakah perbedaan kegunaan antara perulangan luar dengan perulangan yang berada didalamnya?

Perulangan luarr digunakan untuk looping baris, sedangkan perulangan dalam digunakan untuk kolom

4. Mengapa perlu ditambahkan sintaks **System.out.println();** di bawah perulangan dalam? Apa akibatnya jika sintaks tersebut dihilangkan?  
Agar jika perulangan dalam selesai akan pindah baris ke bawah
5. Silakan commit dan push ke repository Anda.

### Percobaan 3

berikut?

```
*  
**  
***
```

1. Perhatikan, apakah output yang dihasilkan dengan nilai N = 5 sesuai dengan tampilan  
tidak
2. Jika tidak sesuai, bagian mana saja yang harus diperbaiki/ditambahkan? Jelaskan setiap bagian yang perlu diperbaiki/ditambahkan.

```
public class Triangle {  
  
    Run | Debug  
    public static void main(String[] args) {  
  
        Scanner sc = new Scanner(System.in);  
        System.out.print("Masukkan nilai N = ");  
        int N = sc.nextInt();  
        int i = 0;  
  
        while(i<N) {  
            int j = 0;  
            while (j<i) {  
                System.out.print("*");  
                j++;  
            }  
            System.out.println();  
            i++;  
        }  
    }  
}
```

Tambahkan **System.out.println()** di sebelum **i++** agar setelah perulangan while dalam selesai langsung pindah baris ke bawah

### Percobaan 4

1. Jelaskan alur program di atas!

Program ini menghasilkan angka acak antara 1 dan 10, lalu meminta pengguna menebak angka tersebut. Pengguna ditanya apakah mereka ingin bermain lagi setelah setiap putaran. Program akan berlanjut selama pengguna ingin bermain lagi dan berakhir ketika pengguna memilih untuk tidak bermain lagi.

2. Apa yang harus dilakukan untuk tidak melanjutkan (tidak mengulangi) permainan tersebut?

Menginputkan huruf selain y/Y

3. Modifikasi program di atas, sehingga bisa menampilkan informasi mengenai : input nilai tebakan yang dimasukan oleh user apakah lebih kecil atau lebih besar dari jawaban/number yang di random!

Menambahkan if else

```
boolean success = false;
do {
    System.out.print("Tebak angka (1-10): ");
    int answer = input.nextInt();
    input.nextLine();
    success = (answer == number);
    if (answer < number) {
        System.out.println("Angka terlalu kecil");
    } else if (answer > number) {
        System.out.println("Angka terlalu besar");
    } else {
        System.out.println("Tebakan Anda benar!");
    }
} while(!success);
System.out.print("Apakah Anda ingin mengulang permainan (Y/n)?");
menu = input.nextLine().charAt(0);

} while(menu=='y' || menu=='Y');
```

4. Silakan commit dan push ke repository Anda.

## Percobaan 5

```
1 import java.util.Scanner;
2
3 public class NestedLoop25 {
4     public static void main(String[] args) {
5         Scanner scanner = new Scanner(System.in);
6         double temps[][] = new double[4][7];
7
8         for (int i = 0; i < temps.length; i++) {
9             System.out.println("Kota ke-" + i);
10            for (int j = 0; j < temps[0].length; j++) {
11                System.out.print("Hari ke-" + (j + 1) + ": ");
12                temps[i][j] = scanner.nextDouble();
13            }
14            System.out.println();
15        }
16
17        int k = 1;
18        for(double[] data: temps){
19            System.out.print("Kota ke-"+k+" : ");
20            for(double item: data){
21                System.out.print(item + " ");
22            }
23            System.out.println();
24            k++;
25        }
26
27
28    }
29 }
30
```

1. Modifikasi program di atas sehingga bisa menampilkan nilai rata-rata masing-masing kota!

```
● ● ●

1 import java.util.Scanner;
2
3 public class NestedLoop25 {
4     public static void main(String[] args) {
5         Scanner scanner = new Scanner(System.in);
6         double temps[][] = new double[4][7];
7
8         for (int i = 0; i < temps.length; i++) {
9             System.out.println("Kota ke-" + i);
10            for (int j = 0; j < temps[0].length; j++) {
11                System.out.print("Hari ke-" + (j + 1) + ": ");
12                temps[i][j] = scanner.nextDouble();
13            }
14            System.out.println();
15        }
16
17        int k = 1;
18        int dummy = 0;
19        for(double[] data: temps){
20            System.out.print("Kota ke-"+ k +" : ");
21            for(double item: data){
22                System.out.print(item + " ");
23                dummy += item;
24            }
25            System.out.println("Rata-rata :" + dummy/7 + "" );
26            dummy = 0;
27            System.out.println();
28            k++;
29        }
30
31
32    }
33 }
34
```

2. Silakan commit dan push ke repository Anda.

### Tugas Individu dan Kelompok

1. Buatlah program untuk mencetak tampilan segitiga angka seperti dibawah ini berdasarkan input N (nilai N minimal 3). Contoh N = 5

```
1  
12  
123  
1234
```

```
1 import java.util.Scanner;  
2  
3 public class TugasIndividu1 {  
4     public static void main(String[] args) {  
5         Scanner input = new Scanner(System.in);  
6  
7         System.out.print("Masukkan jumlah: ");  
8         int n = input.nextInt();  
9  
10        for (int i = 1; i<=n; i++) {  
11  
12            for (int j = n; j>i; j--) {  
13                System.out.print(" ");  
14            }  
15  
16            for (int j = 1; j<=i; j++) {  
17                System.out.print(j);  
18            }  
19            System.out.println();  
20        }  
21  
22        input.close();  
23    }  
24}  
25
```

2. Buatlah program untuk mencetak tampilan segitiga bintang seperti dibawah ini berdasarkan input N (nilai N minimal 5). Contoh N = 7

```
*****
*****
*****
****
***
***
```

```
1 import java.util.Scanner;
2
3 public class TugasIndividu2 {
4     public static void main(String[] args) {
5         Scanner input = new Scanner(System.in);
6
7         System.out.print("Masukkan jumlah: ");
8         int n = input.nextInt();
9
10        for (int i = 1; i<=n; i++) {
11
12            for (int j = n; j>=i; j--) {
13                System.out.print("*");
14            }
15
16            System.out.println();
17        }
18
19        input.close();
20    }
21 }
22 }
```

3. Buatlah program untuk mencetak tampilan persegi angka seperti di bawah ini berdasarkan input N (nilai N minimal 3). Contoh N = 3. dan

	5	5	5	5	5
	3	3	3	5	5
	3	3	5	5	5
	3	3	5	5	5



```
1 import java.util.Scanner;
2
3 public class TugasIndividu3 {
4     public static void main(String[] args) {
5         Scanner input = new Scanner(System.in);
6
7         System.out.print("Masukkan jumlah: ");
8         int n = input.nextInt();
9
10        for (int i = 1; i<=n; i++) {
11
12            for(int j = 1; j<=n; j++){
13                if (i == 1 || i == n) {
14                    System.out.print(n + " ");
15
16                }else{
17
18                    if (j == 1 || j == n){
19                        System.out.print(n + " ");
20                    }else{
21                        System.out.print(" ");
22                    }
23                }
24            }
25
26
27            System.out.println();
28        }
29
30        input.close();
31    }
32 }
33
```

4. Implementasikan flowchart dari fitur-fitur yang telah Anda buat pada tugas teori sebelumnya tentang nested loop!
5. Jangan lupa, semoga kode program harus di-push ke repository Anda.

## Jobsheet 13

### Percobaan 1: Fungsi tanpa parameter

- Apakah fungsi tanpa parameter selalu digunakan untuk fungsi “void” saja?  
Tidak
- Apakah bisa kalimat “Thank you for....dst” dituliskan tanpa fungsi UcapanTerimaKasih? modifikasi kode program tersebut untuk dapat menampilkan suatu kalimat **tanpa dan dengan** menggunakan fungsi!

```
TerimaKasih25.java > TerimaKasih25 > main(String[])
1 public class TerimaKasih25 {
2     Run | Debug
3     public static void main(String[] args) {
4         System.out.println("Thank you for being best teacher in the woorld. \n"+
5             "You inspired in me a love for learning and made me feel like I could ask you anything. \n");
6         ucapanTerimaKasih();
7     }
8     public static void ucapanTerimaKasih() {
9         System.out.println("Thank you for being best teacher in the woorld. \n"+
10            "You inspired in me a love for learning and made me feel like I could ask you anything. \n");
11    }
12 }
```

- Apakah keuntungan menggunakan fungsi di dalam program?

Fungsi adalah cara untuk memecah program besar menjadi bagian-bagian kecil yang lebih mudah dipahami. Fungsi dapat digunakan kembali di dalam program yang sama, maupun di program yang lain.

### Percobaan 2: Fungsi dengan Parameter

- Apakah kegunaan suatu parameter di dalam fungsi?  
digunakan untuk menampung nilai yang akan dikirimkan ke dalam fungsi
- Apakah parameter sama dengan variabel? jelaskan!  
Beda. Parameter tidak sama dengan variabel. Parameter adalah variabel yang digunakan untuk menampung nilai yang akan dikirimkan ke dalam fungsi. Variabel adalah simbol yang digunakan untuk mewakili suatu nilai yang dapat berubah-ubah.
- Apakah parameter di dalam Bahasa pemrograman Java hanya digunakan untuk melewatkkan data input saja? Bagaimana dengan data output?  
Tidak, parameter juga dapat digunakan untuk melewatkkan data output.

Pada dasarnya, parameter adalah variabel yang dideklarasikan di dalam fungsi. Variabel ini dapat digunakan untuk menampung nilai yang dilewatkan dari luar fungsi (data input) atau nilai yang dihasilkan oleh fungsi (data output).

Untuk melewakan data output dari fungsi, kita dapat menggunakan kata kunci return. Kata kunci ini akan mengembalikan nilai dari fungsi ke tempat pemanggilan fungsi.

### Percobaan 3: Membuat fungsi kembalian

1. Jelaskan mengenai, kapan suatu fungsi membutuhkan nilai kembalian (return)!  
Suatu fungsi membutuhkan nilai kembalian (return) jika fungsi tersebut menghasilkan suatu nilai yang ingin digunakan di luar fungsi.
2. Apakah bisa ditambahkan statement System.out.println di dalam fungsi dengan nilai kembalian? Apa pengaruh nya?  
Bisa, tidak berpengaruh apa apa, dia juga akan menjalankan fungsi print tersebut.
3. Apakah bisa fungsi tanpa nilai kembalian di dalam fungsi main dipanggil tanpa dilewatkan ke variabel? seperti pada percobaan 1? Jelaskan!  
Bisa, langsung saja parsing data ke parameternya.

### Percobaan 4: Fungsi yang mengcalling fungsi lain

1. Berdasarkan uji coba pada percobaan 4, manakah fungsi yang akan eksekusi pertama kali? Jelaskan!  
Ucapan terima kasih, krena ucapan terimaTerimaKasih yang dimasukkan di fungsi main.
2. Manakah tata cara penulisan fungsi benar di dalam class? Di atas fungsi main ataukah dibawah fungsi main? Jelaskan!  
Dibawah fungsi main, agar fungsi main ter eksekusi duluan
3. Modifikasi program di atas dengan menambahkan fungsi **UcapanTambahan()** dengan input parameter String. Fungsi **UcapanTambahan()** berisi ucapan tambahan yang ingin disampaikan kepada penerima ucapan.

```
JcapanTerimaKasih25.java > UcapanTerimaKasih25 > main(String[])
    import java.util.Scanner;

    public class UcapanTerimaKasih25 {
        Run | Debug
        public static void main(String[] args) {
            ucapanTerimaKasih();
            ucapanTambahan(ucapan: "Ini Ucapan Tambahan");
        }

        >     public static String penerimaUcapan() { ... }

        >     public static void ucapanTerimaKasih() { ... }

        public static void ucapanTambahan(String ucapan) {
            System.out.println(ucapan);
        }
    }
```

### Percobaan 5: Fugsi varargs

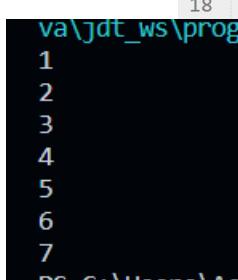
1. Jelaskan mengapa penulisan parameter di praktikum 5 di tulis dengan `int... a!`  
 Penulisan parameter dengan `int... a` di praktikum 5 menggunakan **varargs**, yaitu variabel yang dapat menampung sejumlah nilai dari berbagai tipe data. Variabel varargs diawali dengan tanda `...` dan diikuti oleh tipe data variabel tersebut.
  
2. Sebutkan kegunaan varargs dalam implementasi kode program untuk menyelesaikan permasalah dalam dunia nyata! (minimal 3)
  - Untuk menerima sejumlah nilai dari pengguna. Misalnya, fungsi `tampil()` dalam praktikum 5 dapat digunakan untuk menampilkan sejumlah nilai integer dari pengguna.
  - Untuk membuat fungsi lebih fleksibel. Misalnya, fungsi `sum()` dapat digunakan untuk menjumlahkan sejumlah nilai, tanpa perlu menentukan jumlah nilai yang akan dijumlahkan.
  - Untuk menghemat kode. Misalnya, fungsi `tambah()` dapat digunakan untuk menambahkan dua bilangan, tanpa perlu menulis fungsi baru untuk menambahkan tiga bilangan, empat bilangan, dan seterusnya.
  
3. Bisakah kita menggunakan dua tipe data varargs dalam satu fungsi? Berikan contohnya!

Tidak bisa karena vararg hanya bisa 1 disetiap fungsi, dan harus terletak di akhir

## Percobaan 6: Membuat program dengan fungsi vs tanpa fungsi

1. Sebutkan tahapan dan urutan ekskusi pada percobaan 6 di atas!
  - Deklarasi variable
  - Menerima input panjang
  - Menerima input lebar
  - Menerima input tinggi
  - Menghitung luas dan cetak luas
  - Menghitung volume dan mencetak volume
  -
2. Apakah output dari program dibawah ini kemudian jelaskan jalurnya program tersebut!

```
1 public class programKu {  
2     public static void TampilHinggaKei(int i) {  
3         for (int j = 1; j <= i; j++) {  
4             System.out.print(j);  
5         }  
6     }  
7  
8     public static int Jumlah (int bil1, int bil2) {  
9         return (bil1 + bil2);  
10    }  
11  
12    public static void TampilJumlah (int bil1, int bil2) {  
13        TampilHinggaKei(Jumlah(bil1, bil2));  
14    }  
15  
16    public static void main (String[] args) {  
17        int temp = Jumlah(1, 1);  
18        TampilJumlah(temp, 5);  
    }
```



Pertama di fungsi main menjalankan fungsi Jumlah dengan parameter bil1 dan 2. lalu fungsi jumlah mereturn hasil penjumlahan dari bil1 dan 2 dan ditangkap oleh variable temp.

Lalu fungsi main akan menjalankan fungsi tampil jumlah dengan parameter 1 temp dan 5.

Lalu fungsi tampilJumlah akan menjalankan fungsi tampilHinggaKei dengan parameter 1 adalah hsil returt dari fungsi jumlah dengan parameter temp dan 5.

3. Pada saat apakah fungsi yang kita buat harus menggunakan parameter atau tidak? Pada saat apakah fungsi yang kita buat harus memiliki nilai kembalian atau tidak? Jelaskan!

Fungsi yang kita buat harus menggunakan parameter jika fungsi tersebut membutuhkan informasi dari luar untuk melakukan prosesnya.

Fungsi harus memiliki return jika kita membutuhkan hasil proses dari fungsi tersebut.

## Tugas

- Buatlah sebuah *class KubusNoAbsen* yang di dalamnya terdapat fungsi untuk menghitung volume kubus dan luas permukaan kubus!

```
Kubus25.java  Kubus25  main(String[])
1 import java.util.Scanner;
2
3 public class Kubus25 {
4     Run | Debug
5     public static void main(String[] args) {
6         Scanner sc = new Scanner(System.in);
7         System.out.print("Masukkan sisi kubus: ");
8         int s = sc.nextInt();
9
10        hitungVolume(s);
11        luasPermukaan(s);
12    }
13
14    static void hitungVolume(int s) {
15        int vol = s * s * s;
16        System.out.println("Volume kubus adalah " + vol);
17    }
18
19    static void luasPermukaan(int s) {
20        int L = 6 * s * s;
21        System.out.println("Luas permukaan kubus adalah " + L);
22    }
23}
```

- Buatlah program untuk mengisi array 2 dimensi nilai tugas lima mahasiswa sesuai dengan studi kasus pada tugas pada materi teori sebagai berikut :

	Minggu ke 1	Minggu ke 2	Minggu ke 3	Minggu ke 4	Minggu ke 5	Minggu ke 6	Minggu ke 7
Sari	20	19	25	20	10	0	10
Rina	30	30	40	10	15	20	25

<b>Yani</b>	5	0	20	25	10	5	45
<b>Dwi</b>	50	0	7	8	0	30	60
<b>Lusi</b>	15	10	16	15	10	10	5

Tambahkan fungsi untuk mengambil informasi dari data diatas dengan rincian sebagai berikut :

- Fungsi untuk meninputkan data nilai mahasiswa
- Fungsi untuk menampilkan seluruh nilai mahasiswa mulai dari minggu pertama sampai ketujuh
- Fungsi untuk mencari pada hari keberapakah terdapat nilai tertinggi dibanding hari lain dari keseluruhan mahasiswa
- Fungsi untuk menampilkan mahasiswa yang memiliki nilai tertinggi (tampilkan pula keterangan nilai dari minggu ke-)

```

● ● ●
1 import java.util.Scanner;
2
3 class NilaiMatkul{
4     static String[] nama = new String[5];
5     static int[][] nilai = new int[5][7];
6
7     static Scanner sc = new Scanner(System.in);
8
9     public static void main(String[] args) {
10         inputNilai();
11         System.out.println("*****");
12         tampilNilai();
13         int[] index = cariNilaiTertinggi();
14         mhsTertinggi(index[0], index[1]);
15     }
16
17     static void inputNilai() {
18         for (int i = 0; i<5; i++) {
19             System.out.print("Mahasiswa ke-" + (i+1));
20             System.out.print("Masukkan nama Mahasiswa: ");
21             nama[i] = sc.nextLine();
22             for (int j = 0; j<7; j++) {
23                 System.out.print("Masukkan nilai ke-" + (j+1) + ": ");
24                 nilai[i][j] = sc.nextInt();
25             }
26         }
27         System.out.println();
28     }
29
30 }
31
32 static void tampilNilai(){
33     for (int i = 0; i<5; i++) {
34         System.out.print("Mahasiswa ke-" + (i+1));
35         System.out.print("Nama : " + nama[i]);
36         for (int j = 0; j<7; j++) {
37             System.out.print("Nilai minggu ke-" + (j+1) + ": " + nilai[i][j]);
38         }
39         System.out.println();
40     }
41 }
42
43
44 static int[] cariNilaiTertinggi() {
45
46     int indexNamaMinggu[] = {0, 0};
47
48     int i = 0;
49     for (int nilai[]: nilai) {
50         int j = 0;
51         for (int nilai: nilai) {
52             if (nilai > nilai[indexNamaMinggu[0]][indexNamaMinggu[1]]) {
53                 indexNamaMinggu[0] = i;
54                 indexNamaMinggu[1] = j;
55             }
56         }
57         j++;
58     }
59     i++;
60 }
61
62     System.out.println("Nilai tertinggi pada minggu ke " + (indexNamaMinggu[1]+1) + " adalah " + nilai[indexNamaMinggu[0]][indexNamaMinggu[1]]);
63
64     return indexNamaMinggu;
65 }
66
67 static void mhsTertinggi(int indexNama, int indexMinggu) {
68     System.out.println("Nilai tertinggi " + nama[indexNama] + " pada minggu ke-" + (indexMinggu+1) + " adalah " + nilai[indexNama][indexMinggu] );
69 }
70 }

```

3. Modifikasi program tugas no 2 dengan memastikan terdapat input dari user untuk menentukan jumlah mahasiswa dan juga jumlah tugas!

```
1 import java.util.Scanner;
2
3 class NilaiMatkul{
4     static String[] nama;
5     static int[][] nilai;
6
7     static Scanner sc = new Scanner(System.in);
8
9     public static void main(String[] args) {
10
11         System.out.print("Masukkan jumlah mahasiswa: ");
12         int jumlahMhs = sc.nextInt();
13         System.out.print("Masukkan jumlah tugas: ");
14         int jumlahTugas = sc.nextInt();
15
16         nama = new String[jumlahMhs];
17         nilai = new int[jumlahMhs][jumlahTugas];
18
19
20         inputNilai();
21         System.out.println("=====");
22         tampilNilai();
23         int[] index = cariNilaiTertinggi();
24         mhsTertinggi(index[0], index[1]);
25     }
26
27     static void inputNilai() {
28         for (int i = 0; i<nama.length; i++) {
29             System.out.println("Mahasiswa ke-" + (i+1));
30             System.out.print("Masukkan nama Mahasiswa: ");
31             nama[i] = sc.next();
32             for (int j = 0; j<nilai[i].length; j++) {
33                 System.out.print("Masukkan nilai ke-" + (j+1) + ": ");
34                 nilai[i][j] = sc.nextInt();
35             }
36
37             System.out.println();
38         }
39     }
40 }
41
42 static void tampilNilai(){
43     for (int i = 0; i<nama.length; i++) {
44         System.out.println("Mahasiswa ke-" + (i+1));
45         System.out.println("Nama: " + nama[i]);
46         for (int j = 0; j<nilai[i].length; j++) {
47             System.out.println("Nilai minggu ke-" + (j+1) + ": " + nilai[i][j]);
48         }
49
50         System.out.println();
51     }
52 }
53
54 static int[] cariNilaiTertinggi() {
55
56     int indexNamaMinggu[] = {0, 0};
57
58     int i = 0;
59     for (int nilai[]: nilai) {
60         int j = 0;
61         for (int nil: nilai) {
62             if (nil > nilai[indexNamaMinggu[0]][indexNamaMinggu[1]]) {
63                 indexNamaMinggu[0] = i;
64                 indexNamaMinggu[1] = j;
65             }
66             j++;
67         }
68         i++;
69     }
70
71     System.out.println("Nilai tertinggi pada minggu ke " + (indexNamaMinggu[1]+1) + " adalah " + nilai[indexNamaMinggu[0]][indexNamaMinggu[1]]);
72
73     return indexNamaMinggu;
74 }
75
76 static void mhsTertinggi(int indexNama, int indexMinggu) {
77     System.out.println("Nilai tertinggi " + nama[indexNama] + " pada minggu ke-" + (indexMinggu+1) + " adalah " + nilai[indexNama][indexMinggu] );
78 }
79
80 }
```

## Jobsheet 14

### Percobaan 1

1. Apa yang dimaksud dengan fungsi rekursif?  
Fungsi yang memanggil dirinya sendiri
2. Bagaimana contoh kasus penggunaan fungsi rekursif ?  
Bisa digunakan untuk menghitung factorial
3. Pada **Percobaan1**, apakah hasil yang diberikan fungsi **faktorialRekursif()** dan fungsi **faktorialIteratif()** sama? Jelaskan perbedaan alur jalannya program pada penggunaan fungsi rekursif dan fungsi iteratif!  
Sama,  
Fungsi rekursif menggunakan proses pemanggilan diri sendiri secara berulang. Sedangkan, Fungsi iteratif menggunakan proses perulangan.

### Percobaan 2

1. Pada **Percobaan2**, terdapat pemanggilan fungsi rekursif **hitungPangkat(bilangan, pangkat)** pada fungsi main, kemudian dilakukan pemanggilan fungsi **hitungPangkat()** secara berulangkali. Jelaskan sampai kapan proses pemanggilan fungsi tersebut akan dijalankan!  
Pada Percobaan2, proses pemanggilan fungsi hitungPangkat() akan dijalankan sampai kondisi dasar terpenuhi. Kondisi dasar tersebut adalah ketika nilai pangkat sama dengan 0. Pada saat itu, fungsi hitungPangkat() akan mengembalikan nilai 1, yang merupakan hasil dari bilangan pokok yang dipangkatkan dengan 0.  
Jika nilai pangkat tidak sama dengan 0, maka fungsi hitungPangkat() akan melakukan pemanggilan dirinya sendiri dengan nilai pangkat yang berkurang 1. Hal ini akan terus dilakukan sampai nilai pangkat menjadi 0.
2. Tambahkan kode program untuk mencetak deret perhitungan pangkatnya. Contoh : **hitungPangkat(2,5)** dicetak  $2 \times 2 \times 2 \times 2 \times 2 = 32$

```
1 import java.util.Scanner;
2
3 public class Percobaan2 {
4
5
6     static String ctk = "";
7
8     public static void main(String[] args) {
9         int bilangan, pangkat;
10        Scanner sc = new Scanner(System.in);
11        System.out.print("Bilangan yang dihitung: ");
12        bilangan = sc.nextInt();
13        System.out.print("Pangkat: ");
14        pangkat = sc.nextInt();
15
16        System.out.println(hitungPangkat(bilangan, pangkat));
17        System.out.println(ctk + "1 = " + hitungPangkat(bilangan, pangkat));
18
19        sc.close();
20    }
21
22    static int hitungPangkat(int x, int y) {
23
24        if (y == 0) {
25            return 1;
26        } else {
27            ctk += x + " x ";
28            return x * hitungPangkat(x, y-1);
29        }
30    }
31
32
33
34
35
36 }
37
```

### Percobaan 3

1. Pada **Percobaan3**, sebutkan blok kode program manakah yang merupakan “base case” dan “recursion call”!

2. **System.out.print(hitungLaba(saldo, tahun))**

Ini adalah pemanggilan fungsi rekursif pertama kali.

3. Jabarkan trace fase ekspansi dan fase substitusi algoritma perhitungan laba di atas jika diberikan nilai **hitungLaba(100000,3)**

Pada fase ekspansi, fungsi hitungLaba() akan terus melakukan pemanggilan diri sendiri sampai kondisi dasar terpenuhi. Pada kasus hitungLaba(100000,3), proses ekspansi akan berlangsung sebagai berikut:

Tahun | Saldo

----- | -----

1 | 100000

2 | 111000

3 | 122110

Pada tahun pertama, fungsi hitungLaba() akan dipanggil dengan nilai saldo awal sebesar 100.000 dan jumlah tahun sebesar 2. Fungsi ini akan menghitung bunga untuk tahun pertama, yaitu  $100.000 * 0,111 = 11.100$ . Kemudian, fungsi akan mengembalikan nilai saldo akhir, yaitu  $100.000 + 11.100 = 111.000$ .

Pada tahun kedua, fungsi hitungLaba() akan dipanggil dengan nilai saldo awal sebesar 111.000 dan jumlah tahun sebesar 1. Fungsi ini akan menghitung bunga untuk tahun kedua, yaitu  $111.000 * 0,111 = 12.211$ . Kemudian, fungsi akan mengembalikan nilai saldo akhir, yaitu  $111.000 + 12.211 = 123.211$ .

Pada tahun ketiga, fungsi hitungLaba() akan dipanggil dengan nilai saldo awal sebesar 123.211 dan jumlah tahun sebesar 0. Fungsi ini akan mencapai kondisi dasar, karena jumlah tahun telah menjadi 0. Oleh karena itu, fungsi akan mengembalikan nilai saldo, yaitu 123.211.

### Tugas

1. Buatlah program untuk menampilkan bilangan n sampai 0 dengan menggunakan fungsi rekursif dan fungsi iteratif. (**DeretDescendingRekursif**).

```
1  /**
2   * TugasIndividu1
3   */
4  public class TugasIndividu1 {
5
6      public static void main(String[] args) {
7          System.out.println(deretDescRekursif(5));
8          System.out.print(deretDescIteratif(5));
9      }
10
11     static int deretDescRekursif(int n) {
12         if (n == 0) {
13             return 0;
14         } else {
15             System.out.print(n + " ");
16             return deretDescRekursif(n-1);
17         }
18     }
19
20     static int deretDescIteratif(int n) {
21         for (int i = n; i >= 1; i--) {
22             System.out.print(i + " ");
23         }
24
25         return 0;
26
27     }
28 }
29
30
31 }
```

```
Acer\AppData\Roaming
5 4 3 2 1 0
5 4 3 2 1 0
PS C:\Users\Acer\T
```

2. Buatlah program yang di dalamnya terdapat fungsi rekursif untuk menghitung penjumlahan bilangan. Misalnya  $f = 8$ , maka akan dihasilkan  $1+2+3+4+5+6+7+8 = 36$  (**PenjumlahanRekursif**).

```
1 public class TugasIndividu2 {
2
3     private static String ctk = "";
4     private static int dummy = 0;
5
6     public static void main(String[] args) {
7         System.out.print(hitungPenjumlahan(8));
8     }
9
10    static int hitungPenjumlahan(int n) {
11        dummy++;
12        if (n == 0) {
13            System.out.print(ctk);
14            return 0;
15        } else {
16
17            if (n == 1) {
18                ctk += dummy + " = ";
19            } else {
20                ctk += dummy+ " + ";
21            }
22            return n + hitungPenjumlahan(n-1);
23        }
24    }
25}
26
```

```
PS C:\Users\Acer\Tugas Kuliah\Semester 1\Praktek Daspro\jo> cd C:\Users\Acer\Tugas Kuliah\Semester 1\Praktek Daspro\jo> java TugasIndividu2
1 + 2 + 3 + 4 + 5 + 6 + 7 + 8 = 36
PS C:\Users\Acer\Tugas Kuliah\Semester 1\Praktek Daspro\jo>
```

3. Buat program yang di dalamnya terdapat fungsi rekursif untuk mengecek apakah suatu bilangan  $n$  merupakan bilangan prima atau bukan.  $n$  dikatakan bukan bilangan prima jika ia habis dibagi dengan bilangan kurang dari  $n$ . (**CekPrimaRekursif**).

```
 1 public class TugasIndividu3 {
 2     public static void main(String[] args) {
 3         System.out.println(cekPrimaRekursive(13));
 4     }
 5
 6
 7     static boolean cekPrimaRekursive(int n) {
 8         if (n <= 1) {
 9             return false;
10         } else if (n == 2) {
11             return true;
12         } else {
13             for (int i = 2; i * i <= n; i++) {
14                 if (n % i == 0) {
15                     return false;
16                 }
17             }
18             return true;
19         }
20     }
21 }
22 }
```

```
s\program_63bfd992\bin' 'TugasIndividu3'
true
PS C:\Users\Acer\Tugas Kuliah\Semester 1\Praktek Dasn
```

4. Sepasang marmut yang baru lahir (jantan dan betina) ditempatkan pada suatu pembiakan. Setelah dua bulan pasangan marmut tersebut melahirkan sepasang marmut kembar (jantan dan betina). Setiap pasangan marmut yang lahir juga akan melahirkan sepasang marmut juga setiap 2 bulan. Berapa pasangan marmut yang ada pada akhir bulan ke-12? Buatlah programnya menggunakan fungsi rekursif! (Fibonacci).

Berikut ini adalah ilustrasinya dalam bentuk tabel.

Bulan ke-	Jumlah Pasangan		Total Pasangan
	Produktif	Belum Produktif	
1	0	1	1
2	0	1	1
3	1	1	2
4	1	2	3
5	2	3	5
6	3	5	8

7	5	8		13
8	8	13		21
9	13	21		34
10	21	34		55
11	34	55		89
12	55	89		144



```

1  public class TugasIndividu4 {
2      public static void main(String[] args) {
3
4          for (int bulan = 1; bulan <=12; bulan++) {
5              if (bulan == 1 || bulan == 2 ) {
6                  System.out.println("Bulan ke-" + bulan + " : " + 1);
7                  continue;
8              }
9              int total = fibonanci(bulan-1);
10             System.out.println("Bulan ke-" + bulan + " : " + total);
11         }
12
13     }
14
15     static int fibonanci(int n) {
16         if (n == 0 || n == 1) {
17             return 1;
18         } else {
19             return fibonanci(n - 1) + fibonanci(n - 2);
20         }
21     }
22 }
23

```

```
Acer\ UserData\ Roaming\ Code\ user\ workspaces\ Data\ Bulan ke-1 : 1  
Bulan ke-2 : 1  
Bulan ke-3 : 2  
Bulan ke-4 : 3  
Bulan ke-5 : 5  
Bulan ke-6 : 8  
Bulan ke-7 : 13  
Bulan ke-8 : 21  
Bulan ke-9 : 34  
Bulan ke-10 : 55  
Bulan ke-11 : 89  
Bulan ke-12 : 144  
PS C:\Users\Acer\Tugas Kuliah\Semester 1\Pr
```