

Nama : Akbar Bagus Wicaksana

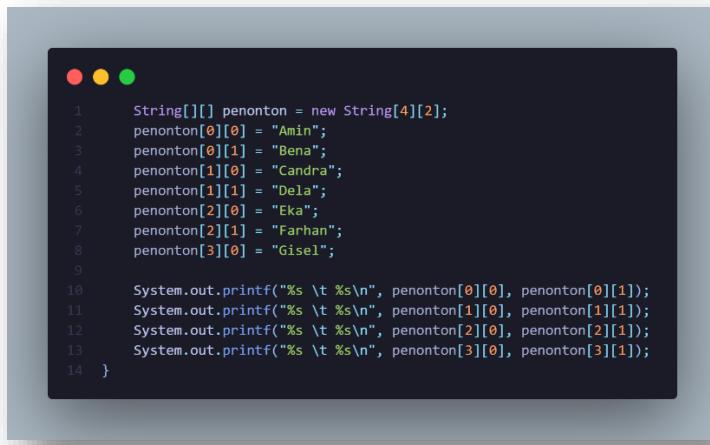
Nim : 254107020067

Kelas : TI – 1H

Matkul : Praktikum Daspro

Pertemuan ke 10

## Percobaan 1



```
String[][] penonton = new String[4][2];
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";
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]);
}
```

### Pertanyaan!

- Apakah pengisian elemen array harus dilakukan secara berurutan mulai dari indeks ke-0? Jelaskan!
  - Jawaban: Ya, karena array pada java indeks awal sebuah array dimulai dari 0
- Mengapa terdapat null pada daftar nama penonton?
  - Jawaban: Karena elemen array tersebut belum diinstansiasi (belum diberi nilai).
- Lengkapi daftar penonton pada langkah ke-4 sebagai berikut
  - Jawaban:



```
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";
```

- Tambahkan kode program sebagai berikut:  
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?
  - Jawaban: Fungsi penonton.length untuk mengembalikan baris dalam array 2d penonton, maka penonton.length adalah 4 karena terdapat 4 baris. Sedangkan

fungsi penonton[0].length untuk mengembalikan jumlah kolom pada baris 0, maka penonton[0].length adalah 2 karena setiap baris dibuat dengan ukuran 2. Ya, semua memiliki nilai yang sama yaitu 2, karena array didefinisikan dengan ukuran kolom yang tetap yaitu 2.

5. Modifikasi kode program pada pertanyaan 4 untuk menampilkan panjang setiap baris pada array menggunakan for loop. Compile, run, lalu amati hasilnya.

- Jawaban:

```
Amin    Bena
Eka     Farhan
Gisel   Hana
4
Panjang baris ke-1: 2
Panjang baris ke-2: 2
Panjang baris ke-3: 2
Panjang baris ke-4: 2
```

6. Modifikasi kode program pada pertanyaan 5 untuk menampilkan panjang setiap baris pada array menggunakan foreach loop. Compile, run, lalu amati hasilnya.

- Jawaban:

```
Amin    Bena
Candra  Dela
Eka     Farhan
Gisel   Hana
4
Panjang baris: 2
Panjang baris: 2
Panjang baris: 2
Panjang baris: 2
```

7. Tambahkan kode program untuk menampilkan nama penonton pada baris ke-3 menggunakan for loop. Compile, run, lalu amati hasilnya.

- Jawaban:

```
Amin    Bena
Candra  Dela
Eka     Farhan
Gisel   Hana
4
Panjang baris: 2
Panjang baris: 2
Panjang baris: 2
Panjang baris: 2
Penonton pada baris ke-3:
Eka
Farhan
```

8. Modifikasi kode program pada pertanyaan 7 menjadi perulangan dengan foreach loop. Compile, run, lalu lakukan amati hasilnya.

- Jawaban:

```
Amin    Bena
Candra  Dela
Eka     Farhan
Gisel   Hana
4
Panjang baris: 2
Panjang baris: 2
Panjang baris: 2
Panjang baris: 2
Penonton pada baris ke-3:
Eka
Farhan
```

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

- Jawaban:

```
Penonton pada baris ke-1: Amin, Bena
Penonton pada baris ke-2: Candra, Dela
Penonton pada baris ke-3: Eka, Farhan
Penonton pada baris ke-4: Gisel, Hana
4
Panjang baris: 2
Panjang baris: 2
Panjang baris: 2
Panjang baris: 2
Penonton pada baris ke-3:
Eka
Farhan
```

10. Menurut Anda, apa kekurangan dan kelebihan foreach loop dibandingkan dengan for loop?

- Jawaban: Kelebihannya lebih sederhana, lebih mudah dibaca, dan tidak berisiko salah indeks karena tidak menggunakan indeks sama sekali. Kekurangannya Tidak bisa mengakses indeks, tidak cocok untuk mengubah elemen berdasarkan posisi, dan tidak dapat digunakan untuk kontrol iterasi yang membutuhkan manipulasi indeks.

11. Berapa indeks baris maksimal untuk array penonton?

- Jawaban: Indeks baris maksimal untuk array penonton adalah 3.

12. Berapa indeks kolom maksimal untuk array penonton?

- Jawaban: Indeks kolom maksimal untuk array penonton adalah 1. Karena setiap baris dibuat dengan panjang 2.

13. Apa fungsi dari String.join()?

- Jawaban: Untuk menggabungkan beberapa elemen string menjadi satu string dengan pemisah yang ditentukan.

14. Commit dan push hasil modifikasi Anda ke Github dengan pesan “Modifikasi Percobaan 1”

The screenshot shows a GitHub repository interface. The top navigation bar includes links for Code, Issues, Pull requests, Actions, Projects, Wiki, Security, Insights, and Settings. The main area displays a file list under the 'Files' tab, showing a folder named 'jobsheet10' containing files like 'Bioskop2.java', 'jobsheet5', 'jobsheet6', 'jobsheet7', 'jobsheet8', and 'jobsheet9'. To the right, a detailed view of a commit for 'Bioskop2.java' is shown, with the message 'Modifikasi Percobaan 1' and a timestamp of 'now'. The commit history shows a single entry from 'e3122e9 · now'.

## Percobaan 2

```
1 Scanner sc = new Scanner(System.in);
2 int baris, kolom;
3 String nama, next;
4 String[][] penonton = new String[4][2];
5
6 while (true){
7     System.out.println("Masukkan nama: ");
8     nama = sc.nextLine();
9     System.out.println("Masukkan baris: ");
10    baris = sc.nextInt();
11    System.out.println("Masukkan kolom: ");
12    kolom = sc.nextInt();
13    sc.nextLine();
14
15    penonton[baris-1][kolom-1] = nama;
16
17    System.out.println("Input penonton lainnya? (y/n): ");
18    next = sc.nextLine();
19
20    if (next.equalsIgnoreCase("n")){
21        break;
22    }
23 }
```

## Pertanyaan!

1. Apakah pengisian elemen array dari scanner harus dilakukan secara berurutan mulai dari indeks ke-0? Jelaskan!
  - Jawaban: Tidak, Pengisian elemen array dari scanner tidak harus dilakukan secara berurutan mulai dari indeks ke-0. Array dapat diisi pada indeks mana pun selama indeks tersebut masih valid tidak di luar batas array.
2. Modifikasi kode program untuk memberikan opsi menu sebagai berikut:
  - Menu 1: Input data penonton
  - Menu 2: Tampilkan daftar penonton
  - Menu 3: Exit
  - Jawaban:

The screenshot shows a Java code editor with a dark theme. At the top, there are three colored circular icons (red, yellow, green). The code itself is a menu-driven program for managing movie theater ticket bookings. It uses a Scanner object to read input from the user. The menu options are:

- 1. Input data penonton (Input viewer data)
- 2. Tampilkan daftar penonton penonton (Display viewer list)
- 3. Exit (Exit)

The code handles each menu option by printing prompts to the console and reading user input. For option 1, it asks for a name and the row and column numbers, then updates a 4x2 dimensional String array named 'penonton'. For option 2, it prints the contents of the array. For option 3, it prints a message and breaks out of the loop. If an invalid menu choice is made, it prints an error message and loops back to the menu.

```
1 Scanner sc = new Scanner(System.in);
2 int baris, kolom, menu;
3 String nama;
4 String[][] penonton = new String[4][2];
5
6 while (true){
7     System.out.println("\n====MENU BIOSKOP====");
8     System.out.println("1. Input data penonton");
9     System.out.println("2. Tampilkan daftar penonton penonton");
10    System.out.println("3. Exit");
11    System.out.println("Pilih Menu: ");
12    menu = sc.nextInt();
13    sc.nextLine();
14
15    if (menu == 1){
16        System.out.println("Masukkan nama: ");
17        nama = sc.nextLine();
18        System.out.println("Masukkan baris (1-4): ");
19        baris = sc.nextInt();
20        System.out.println("Masukkan kolom (1-2): ");
21        kolom = sc.nextInt();
22        sc.nextLine();
23
24        penonton[baris-1][kolom-1] = nama;
25
26        System.out.println("Data penonton berhasil ditambahkan!");
27
28    } else if (menu == 2){
29        System.out.println("\n==== Daftar Penonton ====");
30        for (int i = 0; i < penonton.length; i++){
31            System.out.println("Baris "+(i+1)+" : ");
32            for (int j = 0; j < penonton[i].length; j++){
33                if (penonton[i][j] == null){
34                    System.out.println("KOSONG");
35                } else{
36                    System.out.println(penonton[i][j]);
37                }
38            }
39            System.out.println();
40        }
41    } else if (menu == 3){
42        System.out.println("Program selesai");
43        break;
44    } else{
45        System.out.println("Menu tidak valid. Silahkan pilih 1-3");
46    }
47 }
```

3. Modifikasi kode program untuk menghandle apabila nomor baris/kolom kursi yang tidak tersedia
- Jawaban:

```
1 if (menu == 1){
2     System.out.println("Masukkan nama: ");
3     nama = sc.nextLine();
4     while (true) {
5         System.out.println("Masukkan baris (1-4): ");
6         baris = sc.nextInt();
7         if (baris >= 1 && baris <= 4) break;
8         System.out.println("Error: Baris tidak tersedia!");
9     }
10    while (true) {
11        System.out.println("Masukkan kolom (1-2): ");
12        kolom = sc.nextInt();
13        if (kolom >= 1 && kolom <= 2) break;
14        System.out.println("Error: Kolom tidak tersedia!");
15    }
16    sc.nextLine();
17
18    penonton[baris-1][kolom-1] = nama;
19
20    System.out.println("Data penonton berhasil ditambahkan!");
21
22 }
```

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

- Jawaban:

```
1 switch (menu) {
2     case 1:
3         String nama;
4         int baris, kolom;
5
6         System.out.println("Masukkan nama: ");
7         nama = sc.nextLine();
8         while (true) {
9             System.out.println("Masukkan baris (1-4): ");
10            baris = sc.nextInt();
11            System.out.println("Masukkan kolom (1-2): ");
12            kolom = sc.nextInt();
13            sc.nextLine();
14
15            if (baris < 1 || baris > 4 || kolom < 1 || kolom > 2){
16                System.out.println("Kursi tidak tersedia! Silahkan masukkan ulang");
17                continue;
18            }
19
20            if (penonton[baris - 1][kolom - 1] != null){
21                System.out.println("Kursi tersebut sudah terisi!");
22                System.out.println("Silahkan pilih kursi lain\n");
23                continue;
24            }
25
26            penonton[baris-1][kolom-1] = nama;
27            System.out.println("Data penonton berhasil ditambahkan!");
28            break;
29        }
30    break;
```

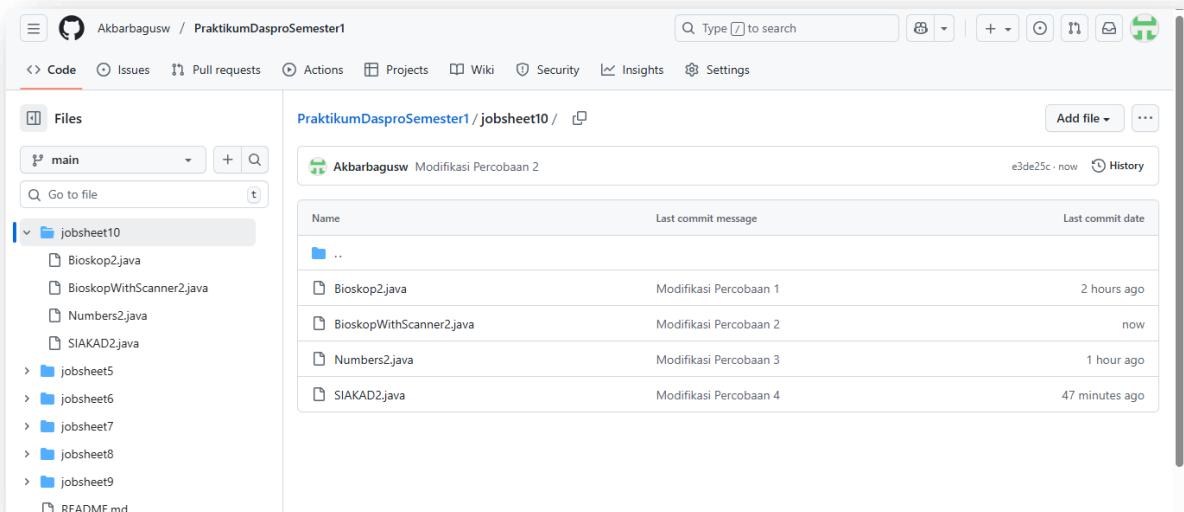
5. Pada menu 2, jika kursi kosong, ganti null dengan \*\*\*

- s Jawaban:



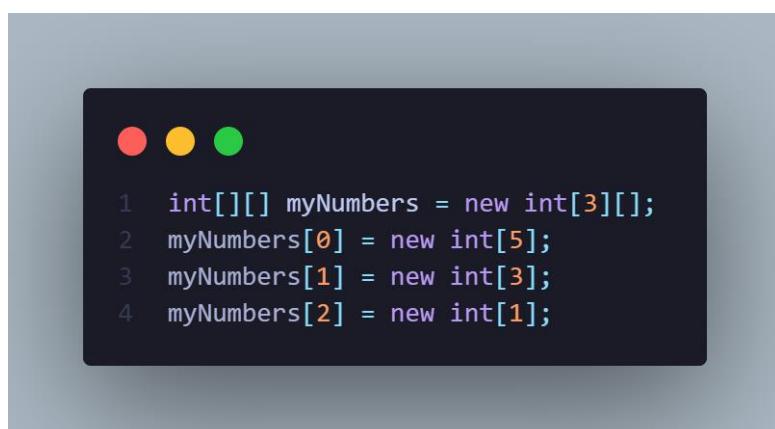
```
1 case 2:
2     System.out.println("\n==== Daftar Penonton ===");
3     for (int i = 0; i < penonton.length; i++){
4         System.out.print("Baris "+(i+1)+": ");
5         for (int j = 0; j < penonton[i].length; j++){
6             if (penonton[i][j] == null){
7                 System.out.print("***");
8             } else{
9                 System.out.print(penonton[i][j]);
10            }
11        }
12        System.out.println();
13    }
14 break;
```

6. Commit dan push hasil modifikasi Anda ke Github dengan pesan “Modifikasi Percobaan 2”



The screenshot shows a GitHub repository named 'PraktikumDasproSemester1'. The 'Code' tab is selected. On the left, there's a sidebar with a tree view of files: 'main', 'jobsheet10' (which contains 'Bioskop2.java', 'BioskopWithScanner2.java', 'Numbers2.java', and 'SIAKAD2.java'), and other branches like 'jobsheet5', 'jobsheet6', 'jobsheet7', 'jobsheet8', 'jobsheet9', and 'README.md'. On the right, under 'PraktikumDasproSemester1/jobsheet10', a commit titled 'Modifikasi Percobaan 2' is shown. The commit message is 'e3de25c · now ⏱ History'. The commit details show four files were modified: 'Bioskop2.java' (last commit message 'Modifikasi Percobaan 1', last commit date '2 hours ago'), 'BioskopWithScanner2.java' (last commit message 'Modifikasi Percobaan 2', last commit date 'now'), 'Numbers2.java' (last commit message 'Modifikasi Percobaan 3', last commit date '1 hour ago'), and 'SIAKAD2.java' (last commit message 'Modifikasi Percobaan 4', last commit date '47 minutes ago').

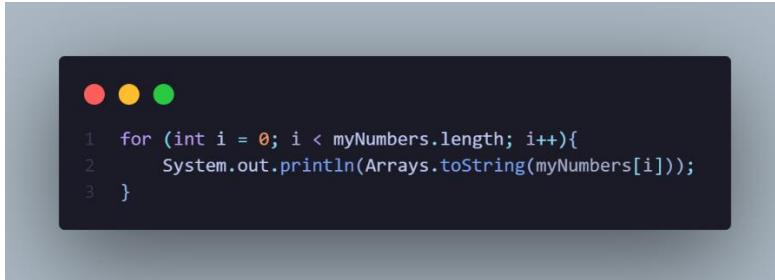
### Percobaan 3



```
1 int[][] myNumbers = new int[3][];
2 myNumbers[0] = new int[5];
3 myNumbers[1] = new int[3];
4 myNumbers[2] = new int[1];
```

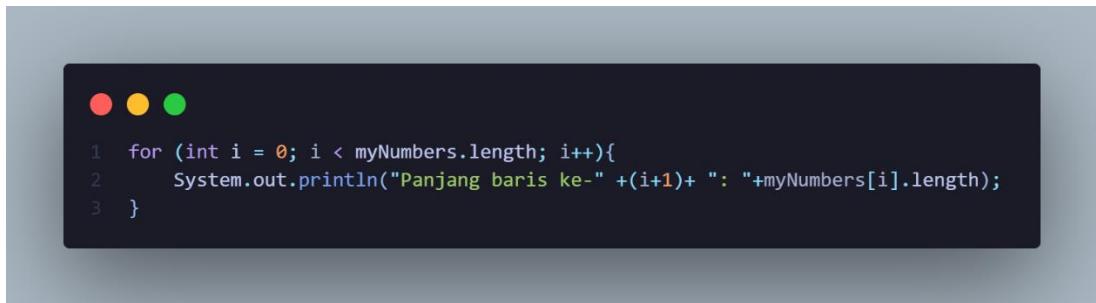
## Pertanyaan!

1. Tambahkan kode program sebagai berikut
  - Jawaban:



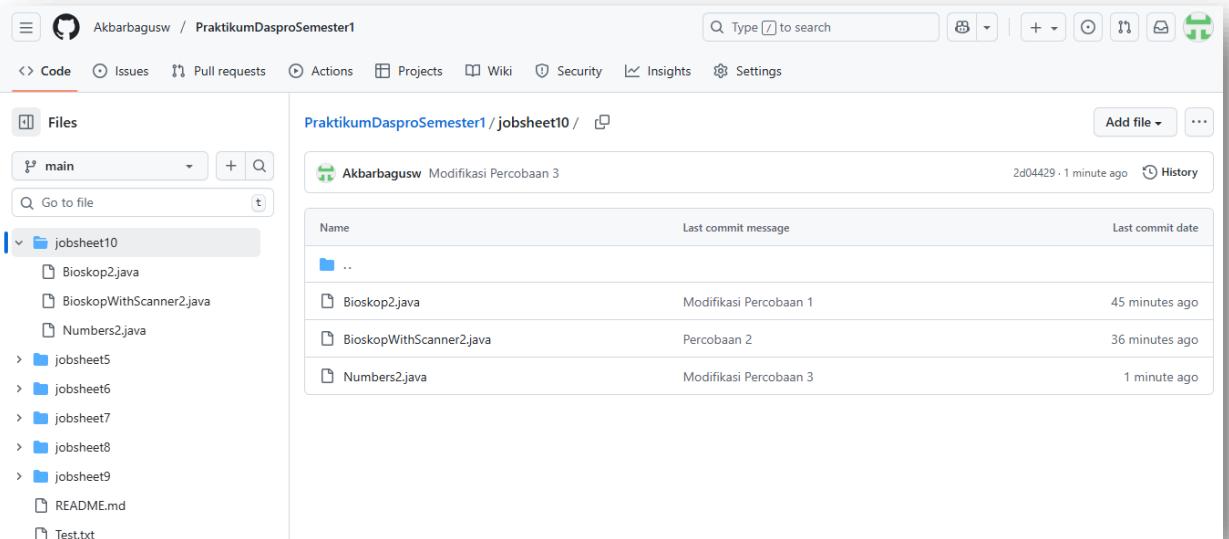
```
1 for (int i = 0; i < myNumbers.length; i++){
2     System.out.println(Arrays.toString(myNumbers[i]));
3 }
```

2. Apa fungsi dari Arrays.toString()?
  - Jawaban: Untuk mengubah array menjadi teks yang rapi agar bisa ditampilkan ke layar.
3. Apa nilai default untuk elemen pada array dengan tipe data int?
  - Jawaban: Nilai defaultnya adalah 0.
4. Tambahkan kode program berikut
  - Jawaban:



```
1 for (int i = 0; i < myNumbers.length; i++){
2     System.out.println("Panjang baris ke-" +(i+1)+ ":" +myNumbers[i].length);
3 }
```

5. Array myNumbers memiliki length berbeda untuk setiap barisnya. Apakah panjang array dapat dimodifikasi setelah diinstansiasi?
  - Jawaban: Tidak, panjang array tidak bisa diubah setelah array tersebut diinstansiasi.
6. Commit dan push hasil modifikasi Anda ke Github dengan pesan “Modifikasi Percobaan 3”



Akbarbagusw / PraktikumDasproSemester1

Type ⌂ to search

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

Add file ⌂ ...

Files

main

Go to file t

jobsheet1

- Bioskop2.java
- BioskopWithScanner2.java
- Numbers2.java

jobsheet5

jobsheet6

jobsheet7

jobsheet8

jobsheet9

README.md

Test.txt

PraktikumDasproSemester1 / jobsheet10 /

Akbarbagusw Modifikasi Percobaan 3 2d04429 · 1 minute ago History

Name	Last commit message	Last commit date
...		
Bioskop2.java	Modifikasi Percobaan 1	45 minutes ago
BioskopWithScanner2.java	Percobaan 2	36 minutes ago
Numbers2.java	Modifikasi Percobaan 3	1 minute ago

## Percobaan 4

```
1 Scanner sc = new Scanner(System.in);
2 int[][] nilai = new int[4][3];
3
4 for (int i = 0; i < nilai.length; i++){
5     System.out.println("Input nilai mahasiswa ke-"(i+1));
6     double totalPerSiswa = 0;
7     for (int j = 0; j < nilai[i].length; j++){
8         System.out.println("Nilai mata kuliah "(j+1)" : ");
9         nilai[i][j] = sc.nextInt();
10        totalPerSiswa += nilai[i][j];
11    }
12    System.out.println("Nilai rata-rata: "totalPerSiswa/3);
13 }
14
15 System.out.println("\n=====");
16 System.out.println("Rata-rata Nilai setiap Mata Kuliah:");
17
18 for (int j = 0; j < 3; j++){
19     double totalPerMatkul = 0;
20     for (int i = 0; i < 4; i++){
21         totalPerMatkul += nilai[i][j];
22     }
23     System.out.println("Mata kuliah "(j+1)" : "totalPerMatkul/4);
24 }
```

## Pertanyaan!

1. Bagaimana jika terdapat perubahan jumlah siswa dan jumlah mata kuliah? Modifikasi kode program SIAKAD untuk mengakomodasi jumlah siswa dan jumlah mata kuliah yang dinamis.
  - Jawaban:

```
1 Scanner sc = new Scanner(System.in);
2
3 System.out.print("Masukkan jumlah siswa: ");
4 int jumlahSiswa = sc.nextInt();
5 System.out.print("Masukkan jumlah mata kuliah: ");
6 int jumlahMatkul = sc.nextInt();
7 int[][] nilai = new int[jumlahSiswa][jumlahMatkul];
8
9 for (int i = 0; i < jumlahSiswa; i++){
10     System.out.println("Input nilai mahasiswa ke-"(i+1));
11     double totalPerSiswa = 0;
12     for (int j = 0; j < jumlahMatkul; j++){
13         System.out.println("Nilai mata kuliah "(j+1)" : ");
14         nilai[i][j] = sc.nextInt();
15         totalPerSiswa += nilai[i][j];
16     }
17     System.out.println("Nilai rata-rata: "totalPerSiswa/jumlahMatkul);
18 }
19
20 System.out.println("\n=====");
21 System.out.println("Rata-rata Nilai setiap Mata Kuliah:");
22
23 for (int j = 0; j < jumlahMatkul; j++){
24     double totalPerMatkul = 0;
25     for (int i = 0; i < jumlahSiswa; i++){
26         totalPerMatkul += nilai[i][j];
27     }
28     System.out.println("Mata kuliah "(j+1)" : "totalPerMatkul/jumlahSiswa);
29 }
```

## Tugas!

1. Implementasikan flowchart yang telah dibuat pada latihan Dasar Pemrograman (Teori) ke dalam kode program Java. Commit dan push program Anda ke Github dengan pesan “Tugas”

The screenshot shows a GitHub repository interface. On the left, the file structure is displayed under the 'Code' tab, showing a main folder and several sub-folders named 'jobsheet10' through 'jobsheet9'. Inside these folders are various Java source code files like 'Bioskop2.java', 'Numbers2.java', 'SIAKAD2.java', and 'Tugas.java'. A 'README.md' file is also present. On the right, the commit history for the 'jobsheet10' folder is shown, with the most recent commit being 'Tugas' at 'now'.

Name	Last commit message	Last commit date
..		
Bioskop2.java	Modifikasi Percobaan 1	2 hours ago
BioskopWithScanner2.java	Modifikasi Percobaan 2	31 minutes ago
Numbers2.java	Modifikasi Percobaan 3	1 hour ago
SIAKAD2.java	Modifikasi Percobaan 4	1 hour ago
Tugas.java	Tugas	now