

JOBSCHEET 8

Nama : Jonathan Abdiel Haryono
NIM : 254107020036
Absen : 14
Kelas : 1G

1. PERCOBAAN 1 : REVIEW PERULANGAN 1

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

```
1 import java.util.Scanner;
2
3 public class Star14 {
4     public static void main(String[] args) {
5         Scanner sc = new Scanner(System.in);
6
7         System.out.print("Masukkan nilai n : ");
8         int n = sc.nextInt();
9
10        for (int i = 1; i <= n; i++) {
11            System.out.print("*");
12        }
13    }
14 }
```

Below the editor, there is a terminal window showing the output of the program:

```
Masukkan nilai n : 5
*****
```

The screenshot shows a GitHub commit history interface. It displays a single commit titled "Percobaan 1" made by "jooo-nathan" 44 minutes ago. The commit hash is "ba7dd13".

Commits
History for Praktikum-Daspro / Jobsheet 8 on main
- Commits on Oct 29, 2025
Percobaan 1
jooo-nathan committed 44 minutes ago
ba7dd13
- End of commit history for this file

A. Pertanyaan

1. Jika pada perulangan for, inisialisasi `i=1` diubah menjadi `i=0`, apa akibatnya? Mengapa bisa demikian?
2. Jika pada perulangan for, kondisi `i<=n` diubah menjadi `i<n`, bagaimana bentuk outputnya jika input `n = 5`? Mengapa hasilnya berbeda?
3. Jika pada perulangan for, kondisi `i <= n` diubah menjadi `i > n`, apa akibatnya? Mengapa bisa demikian?
4. Jika pada perulangan for, kondisi step `i++` diubah menjadi `i--` apa akibatnya? Mengapa bisa demikian?
5. Jika pada perulangan for, step `i++` diubah menjadi `i += 2`, bagaimana pola outputnya jika input `n = 6`? Apa yang menyebabkan perubahan tersebut?

B. Jawaban

1. Perulangan akan diulang sebanyak $n+1$ kali karena nilai awal untuk iterasi adalah 0 sehingga program mengulang dari 0 hingga n (termasuk 0 dan n).
2. Program akan mengulang sebanyak $n-1$ kali, karena pada iterasi terakhir, yakni ketika nilai i sama dengan n , program tidak melakukan perulangan karena dibatasi oleh $i < n$. Misalnya, dengan input $n = 5$, program akan mengeluarkan output berupa bintang sebanyak 5-1, yakni 4 kali (****) karena pada perulangan ke-5, pengecekan apakah $5 < 5$ akan bernilai salah sehingga program hanya mengulang sebanyak 4 kali.
3. Jika $i \leq n$ diubah menjadi $i > n$, program tidak akan berjalan sebagaimana mestinya tergantung pada inputnya. Jika input untuk n nilainya kurang dari 1, maka perulangan akan berjalan terus menerus tanpa berhenti (infinite loop) karena nilai i jika ditambah dengan 1 secara terus menerus akan selalu lebih dari atau sama dengan n . Sedangkan, jika input untuk n nilainya lebih dari 0, program tidak akan mengeluarkan output apapun karena tidak memenuhi $i > n$, yang mana $i = 1$.
4. Apabila $i++$ diubah menjadi $i--$, program tidak akan berjalan sebagaimana mestinya tergantung pada inputnya. Jika input untuk n nilainya lebih dari 0, maka perulangan akan berjalan terus menerus tanpa berhenti (infinite loop) karena nilai i jika dikurang dengan 1 secara terus menerus akan selalu kurang dari atau sama dengan n . Sedangkan, jika input untuk n nilainya kurang dari 1, program tidak akan mengeluarkan output apapun karena sudah tidak memenuhi $i \leq n$.
5. Jika $i++$ diubah menjadi $i += 2$, setiap mengulang, program akan menambahkan i dengan 2 sehingga program hanya akan melakukan perulangan sebanyak $n/2$ kali. Apabila diberikan input $n = 6$, program akan memberikan output berupa 3 buah bintang (***) karena $6/2 = 3$.

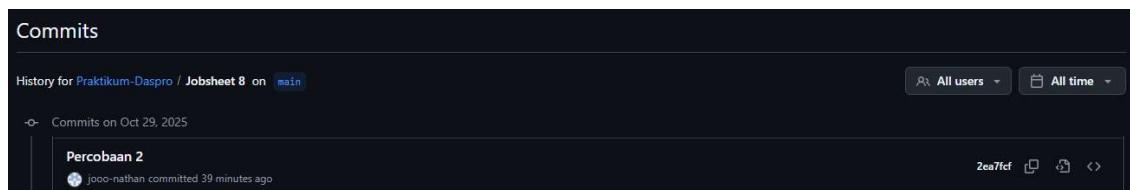
2. PERCOBAAN 2 : BINTANG PERSEGI

The screenshot shows a Java code editor with the following code:

```
J Square14.java > ...
1 import java.util.Scanner;
2
3 public class Square14 {
4     Run | Debug
5     public static void main(String[] args) {
6         Scanner sc = new Scanner(System.in);
7
8         System.out.print("Masukkan nilai n : ");
9         int n = sc.nextInt();
10
11         for (int iOuter = 1; iOuter <= n; iOuter++) {
12             for (int i = 1; i <= n; i++) {
13                 System.out.print("*");
14             }
15             System.out.println();
16         }
17     }
}
```

Below the code, there is a terminal window showing the output of the program. It prompts the user to enter a value for n, and then prints a square pattern of asterisks. The output is:

```
Masukkan nilai n : 5
*****
*****
*****
*****
*****
```



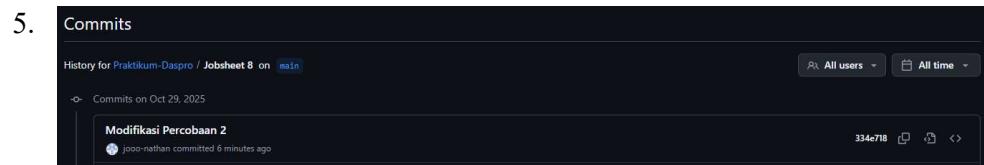
A. Pertanyaan

1. Perhatikan perulangan luar. Jika pada sintaks `for`, inisialisasi `iOuter=1` diubah menjadi `iOuter=0`, apa yang akibatnya? Mengapa bisa demikian?
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?
3. Apakah perbedaan kegunaan antara perulangan luar dengan perulangan yang berada di dalamnya?
4. Mengapa perlu ditambahkan sintaks `System.out.println();` di bawah perulangan dalam? Apa akibatnya jika sintaks tersebut dihilangkan?
5. **Commit dan push hasil modifikasi Anda ke Github dengan pesan "Modifikasi Percobaan 2"**

B. Jawaban

1. Jika `iOuter = 1` diubah menjadi `iOuter = 0`, jumlah baris yang dikeluarkan akan menjadi $n+1$, sedangkan jumlah kolom tetap sebanyak n . Hal tersebut terjadi karena perulangan outer (luar, yang mana merupakan baris) akan diulang sebanyak $n+1$ kali karena nilai awal untuk iterasi adalah 0 sehingga program mengulang dari 0 hingga n (termasuk 0 dan n).

2. Jika **i = 1** diubah menjadi **i = 0**, (sama halnya dengan yang sebelumnya, bedanya hanya pada baris dan kolom) jumlah kolom yang dikeluarkan akan menjadi $n+1$, sedangkan jumlah baris tetap sebanyak n . Hal tersebut terjadi karena perulangan inner (dalam, yang mana merupakan kolom) akan diulang sebanyak $n+1$ kali karena nilai awal untuk iterasi adalah 0 sehingga program mengulang dari 0 hingga n (termasuk 0 dan n).
3. Perulangan luar untuk mengulang baris (ingin ada berapa banyak baris?), sedangkan perulangan dalam untuk mengulang kolom (ingin ada berapa banyak kolom?).
4. Sintaks **System.out.println();** perlu ditambahkan sebagai enter dari tiap baris yang dibuat agar output tidak hanya horizontal melainkan juga vertikal. Jika sintaks tersebut dihilangkan, output akan dicetak secara horizontal sebanyak n^2 , misal untuk $n = 3$, karena $3^2 = 9$, program akan mengeluarkan output berupa 9 bintang sejajar (*****).



3. PERCOBAAN 3 : BINTANG SEGITIGA

```
J Triangle14.java > ...
1 import java.util.Scanner;
2
3 public class Triangle14 {
4     Run | Debug
5     public static void main(String[] args) {
6         Scanner sc = new Scanner(System.in);
7
8         System.out.print("Masukkan nilai n : ");
9         int n = sc.nextInt();
10
11         int i = 0;
12         while (i <= n) {
13             int j = 0;
14             while (j < i) {
15                 System.out.print("*");
16                 j++;
17             }
18             i++;
19         }
20     }
21 }
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

Masukkan nilai n : 5

A. Pertanyaan

1. Perhatikan, apakah output yang dihasilkan dengan nilai n = 5 sesuai dengan tampilan berikut?

•
• •
• * •
• * * *
• * * * *

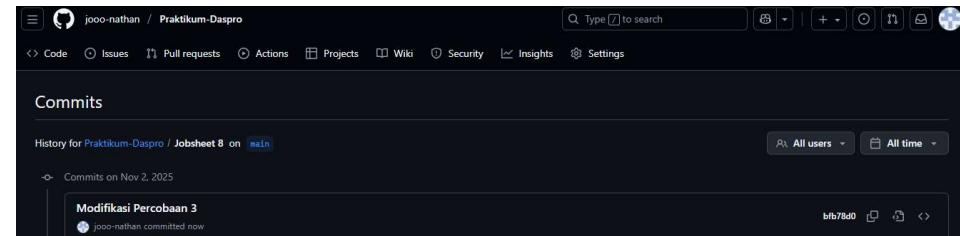
2. Jika tidak sesuai, bagian mana saja yang harus diperbaiki/ditambahkan? Jelaskan setiap bagian yang perlu diperbaiki/ditambahkan.
 3. Jelaskan peran masing-masing variabel i dan j dalam program ini. Mengapa j di-set ulang ke 0 di awal setiap iterasi outer loop? Apa yang akan terjadi jika j tidak di-reset?
 4. Commit dan push hasil modifikasi Anda ke Github dengan pesan "Modifikasi Percobaan 3"

B. Jawaban

1. Tidak, melainkan seperti ini

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS Run: Triangle14 + - x ... x
Masukkan nilai n : 5
*****
PS C:\Users\ACER\Documents\Politeknik Negeri Malang\TUGAS KAMPUS\Prakt. Dasar Pemrograman\Praktikum-Daspro\Jsheet 8> [redacted]
saydi                                     Ln 1, Col 1   Spaces:4   UFT-8   CRLF   () Java   Go Live
```

2. Menambahkan **System.out.println();** sebagai enter untuk setiap baris. Bagian yang perlu ditambahkan yakni **System.out.println();** tepat setelah inner loop agar setiap inner loop menyelesaikan perulangannya, akan dikeluarkan sebuah enter untuk memberi jarak antar baris sehingga hasilnya tidak horizontal, melainkan horizontal dan vertikal.
 3. Variabel **i** berperan sebagai iterator outer loop, sedangkan variabel **j** berperan sebagai iterator inner loop. Variabel **j** di-set ulang ke 0 di awal setiap iterasi outer loop agar inner loop selanjutnya bisa berjalan dengan normal dan tidak mengawali loop dengan nilai variabel **j** sebelumnya, melainkan dengan nilai 0.



4. PERCOBAAN 4 : STUDI KASUS NILAI TUGAS PROYEK KELOMPOK

```

nyoba.py J Tugas1.java 1.M J Triangle14.java 1 J NilaiKelompok14.java 1.u X J Tugas3
osheet 8 > J NilaiKelompok14.java > ...
1 import java.util.Scanner;
2
3 public class NilaiKelompok14 {
4     Run | Debug
5     public static void main(String[] args) {
6         Scanner sc = new Scanner(System.in);
7
8         int i = 1;
9         float totalNilai = 0;
10
11         while (i <= 6) {
12             System.out.println("Kelompok " + i);
13             totalNilai = 0;
14
15             for (int j = 1; j <= 5; j++) {
16                 System.out.print(" Nilai dari Kelompok Penilai " + j + ": ");
17                 int nilai = sc.nextInt();
18                 totalNilai += nilai;
19             }
20             float rataNilai = (float) (totalNilai / 5);
21             System.out.print("Kelompok " + i);
22             System.out.printf(format(": nilai rata-rata = %.1f\n", rataNilai));
23             i++;
24         }
25     }
}

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```

ShowCodeDetailsInExceptionMessages' -cp' 'C:\Users\ACER\AppData\Roaming\Code\User\workspaceStorage\6d12c6777480175e6db0d8d475757426\venchat\java\jdt_ws\osheet_8_a0962bf\bin' 'NilaiKelompok14'
Kelompok 1: nilai dari Kelompok Penilai 1: 88
Nilai dari Kelompok Penilai 2: 79
Nilai dari Kelompok Penilai 3: 90
Nilai dari Kelompok Penilai 4: 82
Nilai dari Kelompok Penilai 5: 85
Kelompok 1: nilai rata-rata = 84,8
Kelompok 2: nilai dari Kelompok Penilai 1: 88
Nilai dari Kelompok Penilai 2: 85
Nilai dari Kelompok Penilai 3: 90
Nilai dari Kelompok Penilai 4: 85
Nilai dari Kelompok Penilai 5: 82
Kelompok 2: nilai rata-rata = 86,2
Kelompok 3: nilai dari Kelompok Penilai 1: 90
Nilai dari Kelompok Penilai 2: 91
Nilai dari Kelompok Penilai 3: 90
Nilai dari Kelompok Penilai 4: 84
Nilai dari Kelompok Penilai 5: 90
Kelompok 3: nilai rata-rata = 88,2
Kelompok 4: nilai dari Kelompok Penilai 1: 77
Nilai dari Kelompok Penilai 2: 75
Nilai dari Kelompok Penilai 3: 83
Nilai dari Kelompok Penilai 4: 79
Nilai dari Kelompok Penilai 5: 76
Kelompok 4: nilai rata-rata = 77,4
Kelompok 5: nilai dari Kelompok Penilai 1: 88
Nilai dari Kelompok Penilai 2: 82
Nilai dari Kelompok Penilai 3: 83
Nilai dari Kelompok Penilai 4: 80
Nilai dari Kelompok Penilai 5: 76
Kelompok 5: nilai rata-rata = 80,6
Kelompok 6: nilai dari Kelompok Penilai 1: 91
Nilai dari Kelompok Penilai 2: 90
Nilai dari Kelompok Penilai 3: 85
Nilai dari Kelompok Penilai 4: 88
Nilai dari Kelompok Penilai 5: 90
Kelompok 6: nilai rata-rata = 88,8

```

A. Pertanyaan

- Jelaskan apa yang terjadi pada variabel **totalNilai** di setiap iterasi *outer loop* dan mengapa inisialisasinya (**total = 0**) berada di dalam outer loop, bukan di luar.
- Modifikasi program di atas, sehingga dapat mencari kelompok dengan rata-rata nilai tertinggi dan tampilkan nomor kelompok tersebut.
- Commit dan push hasil modifikasi Anda ke Github dengan pesan “Modifikasi Percobaan 4”**

B. Jawaban

- Setiap iterasi *outer loop*, variabel **totalNilai** akan di-reset menjadi 0 (dengan cara diinisialisasi menjadi 0) agar variabel ini kosong setelah selesai menjumlahkan nilai sehingga dia bisa menampung nilai yang lain dan mengoperasikannya tanpa tercampur dengan nilai sebelumnya.

```

nyoba.py J Tugas1.java 1.M J Triangle14.java 1 J NilaiKelompok14.java 1.u X J Tug...
osheet 8 > J NilaiKelompok14.java > ...
1 import java.util.Scanner;
2
3 public class NilaiKelompok14 {
4     Run | Debug
5     public static void main(String[] args) {
6         Scanner sc = new Scanner(System.in);
7         int i = 1, kelTertinggi = 0;
8         float totalNilai = 0, tertinggi = 0;
9
10        while (i <= 6) {
11            System.out.println("Kelompok " + i);
12            totalNilai = 0;
13
14            for (int j = 1; j <= 5; j++) {
15                System.out.print(" Nilai dari Kelompok Penilai " + j + ": ");
16                int nilai = sc.nextInt();
17                totalNilai += nilai;
18            }
19            float rataNilai = (float) (totalNilai / 5);
20            System.out.print("Kelompok " + i);
21            System.out.printf(format(": nilai rata-rata = %.1f\n", rataNilai));
22
23            if (rataNilai > tertinggi) {
24                tertinggi = rataNilai;
25                kelTertinggi = i;
26            }
27        }
28        System.out.println("Kelompok dengan nilai tertinggi adalah");
29        System.out.print(" kelompok " + kelTertinggi + " dengan nilai ");
30        System.out.printf(format("%.\n", tertinggi));
31    }
}

```

2.

3.

Commits

History for Praktikum-Daspro / Jobsheet 8 on main

Commits on Nov 2, 2025

Modifikasi Percobaan 4

jooo-nathan committed now

34beec2

TUGAS

Commits

History for Praktikum-Daspro / Jobsheet 8 on main

Commits on Nov 2, 2025

Tugas 3

jooo-nathan committed 1 minute ago

Tugas 2

jooo-nathan committed 3 minutes ago

Tugas 1

jooo-nathan committed 4 minutes ago

TUGAS 1

```

1 import java.util.Scanner;
2
3 public class Tugas1 {
4     public static void main(String[] args) {
5         Scanner sc = new Scanner(System.in);
6
7         System.out.print("Masukkan nilai n : ");
8         int n = sc.nextInt();
9         System.out.println();
10
11        for (int i = 1; i <= n; i++) {
12            System.out.printf("n = %d -> jumlah kuadrat = ", i);
13            int total = 0;
14
15            for (int j = 1; j <= i; j++) {
16                total += (j*j);
17
18                if (j > 1 && j != i+1) {
19                    System.out.print(" + ");
20                }
21                System.out.printf("%d", j*j);
22            }
23
24            if (i != 1) {
25                System.out.printf(" = %d", total);
26            }
27            System.out.println();
28        }
29    }
30}
31

```

TUGAS 2

```
1 import java.util.Scanner;
2
3 public class Tugas2 {
4     public static void main(String[] args) {
5         Scanner sc = new Scanner (System.in);
6
7         System.out.print("Masukkan nilai n : ");
8         int n = sc.nextInt();
9
10        float temp = n;
11        int jDigit = 1;
12        if (temp < 10) {
13            jDigit = 1;
14        }
15        else {
16            while (temp >= 10) {
17                temp /= 10;
18                jDigit++;
19            }
20        }
21        // System.out.println(jDigit);
22
23        for (int i = 1; i <= n; i++) {
24            if (i == 1 || i == n) {
25                for (int j = 1; j <= n; j++) {
26                    System.out.printf("%d", n);
27                }
28            }
29            else {
30                for (int j = 1; j <= n; j++) {
31                    if (j == 1 || j == n) {
32                        System.out.printf("%d", n);
33                    }
34                    else {
35                        for (int k = 1; k <= jDigit; k++) {
36                            System.out.print(" ");
37                        }
38                    }
39                }
40            }
41            System.out.println();
42        }
43    }
44 }
```

TUGAS 3

```
● ● ●

1 import java.util.Scanner;
2
3 public class Tugas3 {
4     public static void main(String[] args) {
5         Scanner sc = new Scanner(System.in);
6
7         System.out.println("MANAJEMEN KAFE 'Kopi Senja'\n");
8         System.out.print("Jumlah cabang kafe : ");
9         int jumKafe = sc.nextInt();
10        int totalItem = 0, totalCust = 0;
11
12        System.out.println("\n===== Input Penjualan Per Cabang =====");
13        for (int i = 1; i <= jumKafe; i++) {
14            System.out.printf("\n----- Cabang %d -----", i);
15            System.out.print("\nJumlah pelanggan : ");
16            int jumCust = sc.nextInt();
17            int totalItemPerCab = 0;
18
19            for (int j = 1; j <= jumCust; j++) {
20                System.out.printf(" - Pelanggan %d memesan berapa item? ", j);
21                int jumItem = sc.nextInt();
22                totalItemPerCab += jumItem;
23            }
24            if (i != jumKafe) {
25                System.out.print("\n-----");
26            }
27
28            System.out.printf("\nAnalisis Cabang %d\n", i);
29            System.out.printf(" -- Total Pelanggan\t: %d orang", jumCust);
30            System.out.printf("\n -- Total Item Terjual\t: %d item\n", totalItemPerCab);
31
32            totalItem = totalItem + totalItemPerCab;
33            totalCust = totalCust + jumCust;
34
35        }
36        System.out.println("\n=====");
37        System.out.println("Total seluruh cabang");
38        System.out.printf("Total Pelanggan\t: %d orang\n", totalCust);
39        System.out.printf("Item terjual\t: %d item\n\n", totalItem);
40    }
41 }
42
```