

**HANDS-ON LABS 2
PEMROGRAMAN
BERORIENTASI OBJEK
+ LAB**



Oleh:

Martogi Jekson C. Siagian 2105541054

PROGRAM STUDI TEKNIK ELEKTRO

FAKULTAS TEKNIK

UNIVERSITAS UDAYANA

2023

Hands-On Labs 2

Tugas

1. Buatlah sebuah program konversi nilai angka ke nilai huruf dengan range berikut :

Nilai ≥ 85 and nilai $\leq 100 = A$

Nilia ≥ 70 and nilai $\leq 84 = B$

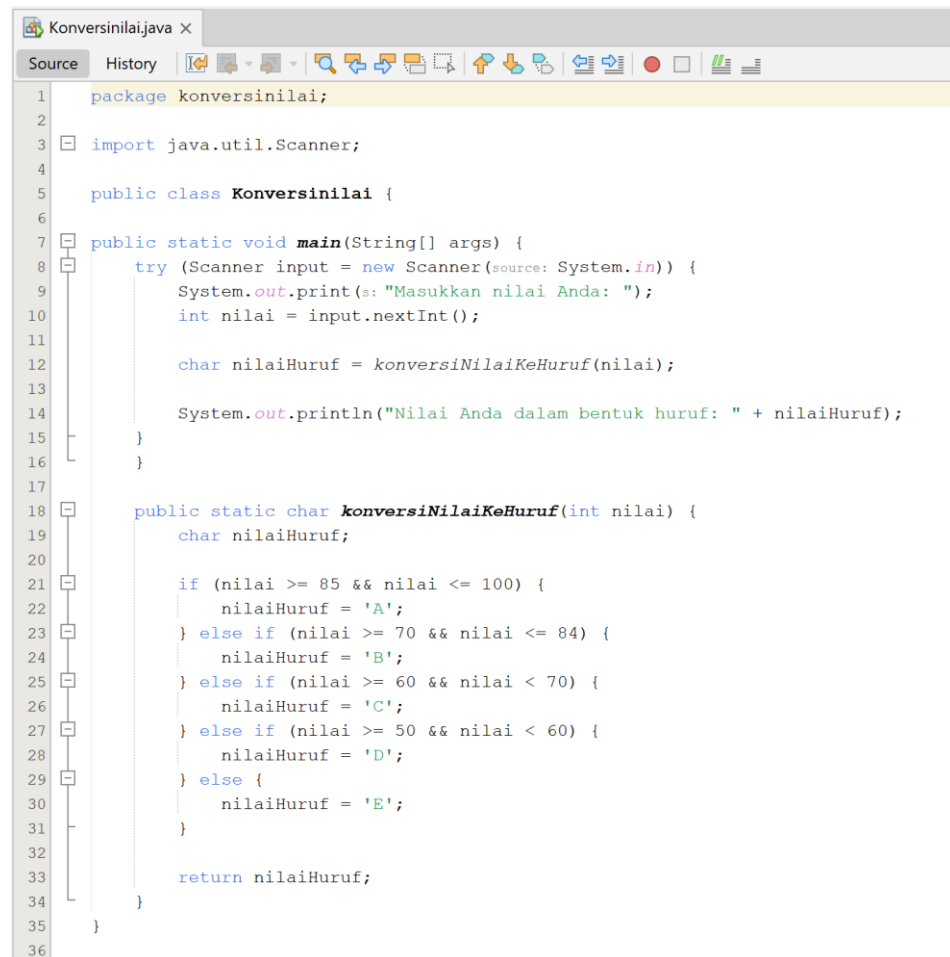
Nilai ≥ 60 and nilai $< 70 = C$

Nilai ≥ 50 and nilai $< 60 = D$

Nilai $< 50 = E$

Jawab :

Berikut Kode yang saya buat untuk program tersrbut



```
1 package konversinilai;
2
3 import java.util.Scanner;
4
5 public class Konversinilai {
6
7     public static void main(String[] args) {
8         try (Scanner input = new Scanner(System.in)) {
9             System.out.print("Masukkan nilai Anda: ");
10            int nilai = input.nextInt();
11
12            char nilaiHuruf = konversiNilaiKeHuruf(nilai);
13
14            System.out.println("Nilai Anda dalam bentuk huruf: " + nilaiHuruf);
15        }
16    }
17
18    public static char konversiNilaiKeHuruf(int nilai) {
19        char nilaiHuruf;
20
21        if (nilai >= 85 && nilai <= 100) {
22            nilaiHuruf = 'A';
23        } else if (nilai >= 70 && nilai <= 84) {
24            nilaiHuruf = 'B';
25        } else if (nilai >= 60 && nilai < 70) {
26            nilaiHuruf = 'C';
27        } else if (nilai >= 50 && nilai < 60) {
28            nilaiHuruf = 'D';
29        } else {
30            nilaiHuruf = 'E';
31        }
32
33        return nilaiHuruf;
34    }
35 }
36
```

Program yang telah saya buat adalah program Java sederhana yang melakukan konversi nilai angka menjadi nilai huruf berdasarkan kriteria yang diberikan di soal. Berikut penjelasan lebih detail tentang program tersebut:

- `import java.util.Scanner;;` Pernyataan ini mengimpor kelas `Scanner` dari pustaka Java, yang digunakan untuk membaca masukan dari pengguna.
- `public class KonversiNilai {:` Program dimulai dengan mendefinisikan kelas utama bernama `KonversiNilai`.
- `public static void main(String[] args) {:` Ini adalah metode utama yang akan dieksekusi ketika program dimulai. Ini adalah tempat di mana program akan berjalan.
- `Scanner input = new Scanner(System.in);;` Ini membuat objek `Scanner` yang akan digunakan untuk membaca masukan dari pengguna.
- `System.out.print("Masukkan nilai Anda: ");;` Ini mencetak pesan kepada pengguna untuk memasukkan nilai numerik.
- `int nilai = input.nextInt();;` Ini membaca masukan pengguna sebagai nilai integer dan menyimpannya dalam variabel `nilai`.
- `char nilaiHuruf = konversiNilaiKeHuruf(nilai);;` Ini memanggil metode `konversiNilaiKeHuruf` dengan `nilai` sebagai argumen dan menyimpan hasil konversi dalam variabel `nilaiHuruf`.
- `System.out.println("Nilai Anda dalam bentuk huruf: " + nilaiHuruf);;` Ini mencetak nilai huruf yang sudah dikonversi ke layar.
- `input.close();;` Ini menutup objek `Scanner` setelah selesai digunakan.
- `public static char konversiNilaiKeHuruf(int nilai) {:` Ini mendefinisikan metode `konversiNilaiKeHuruf` yang akan melakukan konversi dari nilai numerik ke nilai huruf.
- `char nilaiHuruf;` Ini mendeklarasikan variabel `nilaiHuruf` yang akan digunakan untuk menyimpan nilai huruf.
- Blok `if-else if` digunakan untuk mengecek kriteria-kriteria yang Anda sebutkan:
Jika nilai berada dalam rentang 85 hingga 100, maka `nilaiHuruf` akan diatur sebagai `'A'`.

Jika nilai berada dalam rentang 70 hingga 84, maka nilaiHuruf akan diatur sebagai 'B'.

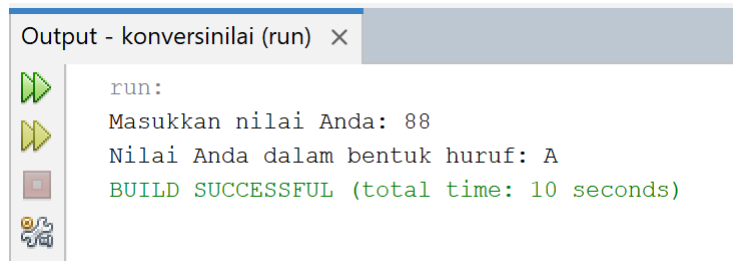
Jika nilai berada dalam rentang 60 hingga 69, maka nilaiHuruf akan diatur sebagai 'C'.

Jika nilai berada dalam rentang 50 hingga 59, maka nilaiHuruf akan diatur sebagai 'D'.

Jika nilai kurang dari 50, maka nilaiHuruf akan diatur sebagai 'E'.

- return nilaiHuruf;: Metode ini mengembalikan nilai huruf yang sesuai.

Dengan demikian, program ini memungkinkan pengguna untuk memasukkan nilai numerik, dan program akan mengkonversinya ke nilai huruf. Berikut Hasil Outputnya



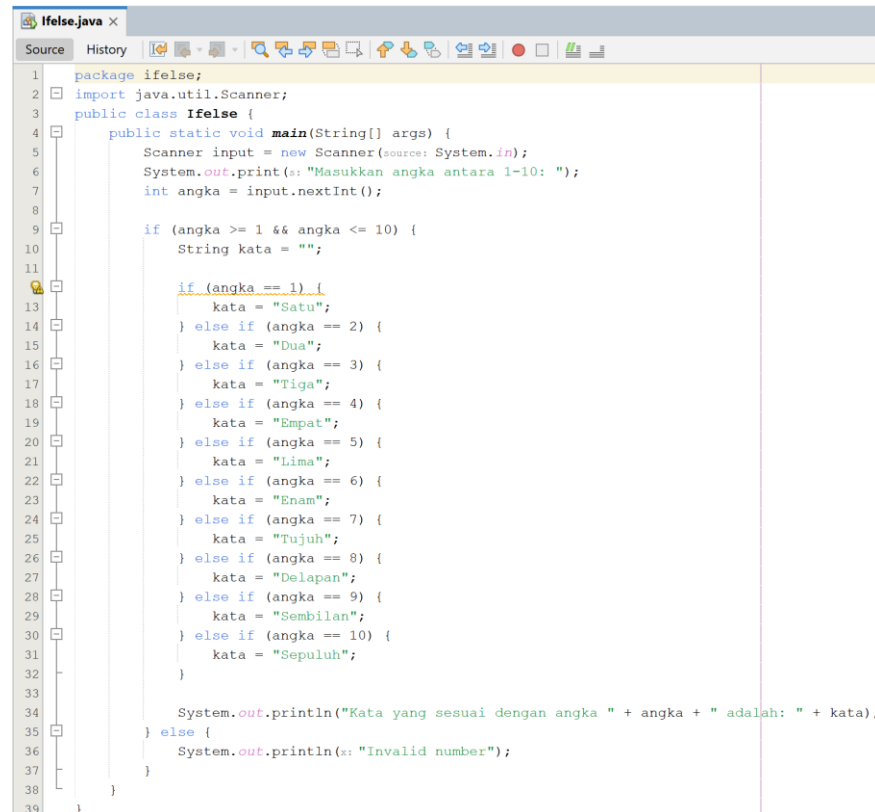
```
run:
Masukkan nilai Anda: 88
Nilai Anda dalam bentuk huruf: A
BUILD SUCCESSFUL (total time: 10 seconds)
```

2. Ambil sebuah angka sebagai input dari user, dan outputnya berupa kata yang sesuai dengan angka. Angka yang dimasukkan antara 1-10. Jika user memasukkan nilai yang tidak sesuai berikan output "Invalid number".
 - a. Gunakan statement if-else untuk menyelesaikan
 - b. Gunakan statement switch untuk menyelesaikan

Jawab:

A. If-else

Berikut kode yang saya buat untuk program tersebut



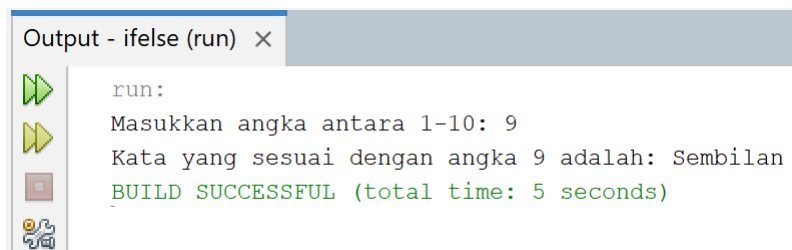
```
1 package ifelse;
2 import java.util.Scanner;
3 public class Ifelse {
4     public static void main(String[] args) {
5         Scanner input = new Scanner(System.in);
6         System.out.print("Masukkan angka antara 1-10: ");
7         int angka = input.nextInt();
8
9         if (angka >= 1 && angka <= 10) {
10             String kata = "";
11
12             if (angka == 1) {
13                 kata = "Satu";
14             } else if (angka == 2) {
15                 kata = "Dua";
16             } else if (angka == 3) {
17                 kata = "Tiga";
18             } else if (angka == 4) {
19                 kata = "Empat";
20             } else if (angka == 5) {
21                 kata = "Lima";
22             } else if (angka == 6) {
23                 kata = "Enam";
24             } else if (angka == 7) {
25                 kata = "Tujuh";
26             } else if (angka == 8) {
27                 kata = "Delapan";
28             } else if (angka == 9) {
29                 kata = "Sembilan";
30             } else if (angka == 10) {
31                 kata = "Sepuluh";
32             }
33
34             System.out.println("Kata yang sesuai dengan angka " + angka + " adalah: " + kata);
35         } else {
36             System.out.println("Invalid number");
37         }
38     }
39 }
```

Kode yang telah saya buat adalah sebuah program Java sederhana yang memungkinkan pengguna untuk memasukkan angka antara 1 hingga 10, dan kemudian program akan mencetak kata yang sesuai dengan angka tersebut. Berikut adalah penjelasan singkat tentang bagaimana kode ini berfungsi:

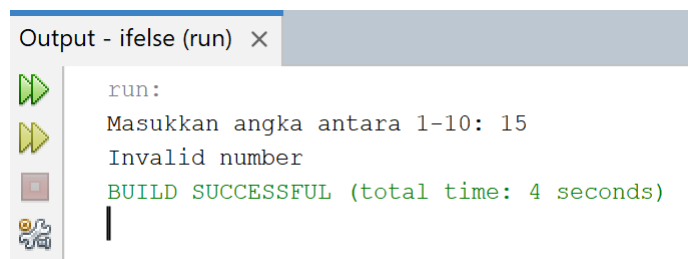
- **Impor Library:** Kode diawali dengan mengimpor pustaka `java.util.Scanner` untuk memungkinkan interaksi dengan pengguna melalui input dari keyboard.
- **Membuka Objek Scanner:** Program membuat objek `Scanner` yang disebut `input` untuk membaca input dari pengguna.
- **Input pengguna:** Program mencetak pesan untuk meminta pengguna memasukkan angka antara 1 hingga 10. Kemudian, input yang diberikan oleh pengguna disimpan dalam variabel `angka`.
- **Periksa apakah angka valid :** Program memeriksa apakah angka yang dimasukkan oleh pengguna berada dalam rentang antara 1 hingga 10.

- Menentukan kata yang sesuai : Jika angka yang dimasukkan oleh pengguna valid, program akan menggunakan pernyataan if-else untuk menentukan kata yang sesuai dengan angka tersebut dan menyimpannya dalam variabel kata.
- Mencetak hasil: Akhirnya, program mencetak kata yang sesuai dengan angka yang dimasukkan oleh pengguna atau mencetak "Invalid number" jika angka yang dimasukkan tidak sesuai dengan rentang yang diizinkan.

Berikut Hasil Outputnya



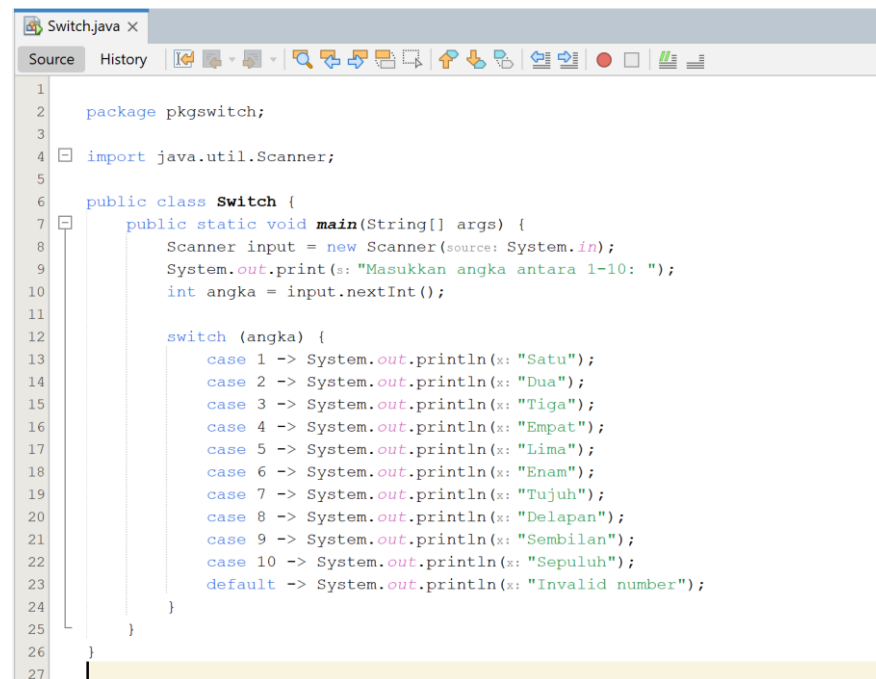
```
Output - ifelse (run) ×  
run:  
Masukkan angka antara 1-10: 9  
Kata yang sesuai dengan angka 9 adalah: Sembilan  
BUILD SUCCESSFUL (total time: 5 seconds)
```



```
Output - ifelse (run) ×  
run:  
Masukkan angka antara 1-10: 15  
Invalid number  
BUILD SUCCESSFUL (total time: 4 seconds)  
|
```

B. Switch

Berikut kode yang saya buat untuk membuat program tersebut

A screenshot of a Java IDE window titled 'Switch.java'. The window shows a Java program using a switch statement. The code is as follows:

```
1 package pkgswitch;
2
3
4 import java.util.Scanner;
5
6 public class Switch {
7     public static void main(String[] args) {
8         Scanner input = new Scanner(System.in);
9         System.out.print("Masukkan angka antara 1-10: ");
10        int angka = input.nextInt();
11
12        switch (angka) {
13            case 1 -> System.out.println("Satu");
14            case 2 -> System.out.println("Dua");
15            case 3 -> System.out.println("Tiga");
16            case 4 -> System.out.println("Empat");
17            case 5 -> System.out.println("Lima");
18            case 6 -> System.out.println("Enam");
19            case 7 -> System.out.println("Tujuh");
20            case 8 -> System.out.println("Delapan");
21            case 9 -> System.out.println("Sembilan");
22            case 10 -> System.out.println("Sepuluh");
23            default -> System.out.println("Invalid number");
24        }
25    }
26 }
27
```

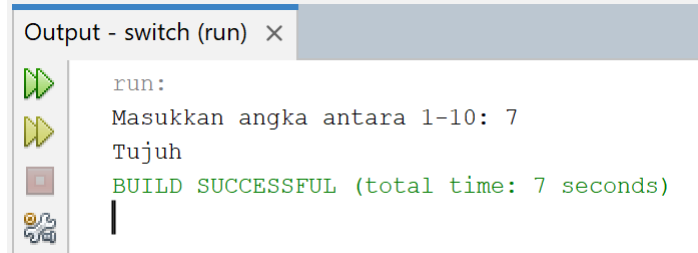
Kode yang menggunakan switch berfungsi dengan cara yang mirip, tetapi menggunakan struktur yang sedikit berbeda.

- switch digunakan untuk memilih satu dari banyak blok kode yang akan dieksekusi berdasarkan nilai ekspresi yang sama.
- Pada switch, hanya dapat membandingkan nilai yang sama dengan ekspresi yang diberikan (dalam hal ini, angka).
- Jika nilai ekspresi cocok dengan salah satu kasus (case), blok kode yang sesuai dengan kasus tersebut akan dieksekusi. Jika tidak ada yang cocok, blok kode dalam default (jika ada) akan dieksekusi.
- switch lebih sesuai untuk kasus-kasus di mana jika kita memiliki serangkaian nilai yang diskrit (seperti angka dari 1 hingga 10) dan ingin mengaitkan setiap nilai dengan blok kode yang berbeda.

Jadi, perbedaan utama antara keduanya adalah dalam cara pengujian kondisi dan pemilihan kode yang akan dieksekusi. Keduanya dapat digunakan untuk tugas yang sama, tetapi switch cenderung lebih bersih dan efisien. Ketika kita memiliki sejumlah besar kasus yang harus ditangani berdasarkan nilai

yang sama. Namun, dalam kasus seperti yang di tunjukkan (angka 1 hingga 10), baik if-else maupun switch dapat digunakan dengan baik.

Berikut Hasil output Ketika menggunakan switch



```
Output - switch (run) X
run:
Masukkan angka antara 1-10: 7
Tujuh
BUILD SUCCESSFUL (total time: 7 seconds)
```

3. Buatlah sebuah program untuk menampilkan sederatan angka genap dan ganjil beserta jumlahnya.

Contoh :

Input nilai = 10

1 3 5 7 9 = 25

2 4 6 8 10 = 30

Jawab:

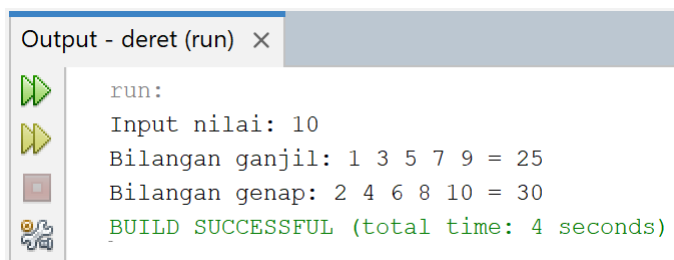
Berikut kode yang saya buat untuk program tersebut

```
1 package deret;
2
3 import java.util.Scanner;
4
5 public class Deret {
6     public static void main(String[] args) {
7         Scanner input = new Scanner(source: System.in);
8         System.out.print(s: "Input nilai: ");
9         int nilai = input.nextInt();
10
11         int totalGenap = 0;
12         int totalGanjil = 0;
13
14         System.out.print(s: "Bilangan ganjil: ");
15         for (int i = 1; i <= nilai; i++) {
16             if (i % 2 != 0) {
17                 System.out.print(i + " ");
18                 totalGanjil += i;
19             }
20         }
21
22         System.out.println(" = " + totalGanjil);
23
24         System.out.print(s: "Bilangan genap: ");
25         for (int i = 1; i <= nilai; i++) {
26             if (i % 2 == 0) {
27                 System.out.print(i + " ");
28                 totalGenap += i;
29             }
30         }
31
32         System.out.println(" = " + totalGenap);
33     }
34 }
35
```


Program di atas adalah program Java sederhana yang melakukan hal berikut:

- Mengimpor kelas Scanner untuk membaca input dari pengguna.
- Membuat objek Scanner dengan nama input untuk membaca input dari pengguna.
- Meminta pengguna untuk memasukkan nilai dengan pesan "Input nilai: " dan menyimpan nilai yang dimasukkan dalam variabel nilai.
- Membuat dua variabel totalGenap dan totalGanjil untuk menyimpan jumlah angka genap dan ganjil.
- Mencetak pesan "Bilangan ganjil: " dan melakukan perulangan dari 1 hingga nilai. Selama perulangan, program memeriksa apakah angka saat ini adalah ganjil (menggunakan operasi modulo % dengan 2). Jika angka adalah ganjil, maka angka tersebut dicetak dan ditambahkan ke totalGanjil.
- Setelah perulangan ganjil selesai, program mencetak total jumlah ganjil yang telah dihitung.
- Mencetak pesan "Bilangan genap: " dan melakukan perulangan dari 1 hingga nilai. Selama perulangan, program memeriksa apakah angka saat ini adalah genap (menggunakan operasi modulo % dengan 2). Jika angka adalah genap, maka angka tersebut dicetak dan ditambahkan ke totalGenap.
- Setelah perulangan genap selesai, program mencetak total jumlah genap yang telah dihitung.

Jadi, program ini meminta pengguna untuk memasukkan sebuah nilai, kemudian menghitung dan menampilkan sederetan angka ganjil dan genap beserta jumlahnya sesuai dengan nilai yang dimasukkan oleh pengguna. Berikut hasilnya

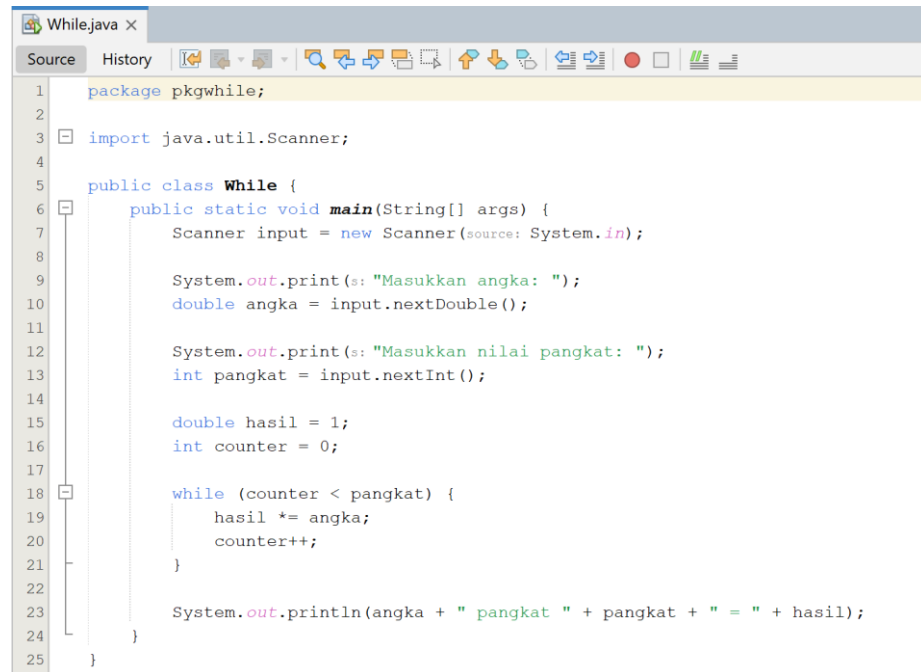


```
run:
Input nilai: 10
Bilangan ganjil: 1 3 5 7 9 = 25
Bilangan genap: 2 4 6 8 10 = 30
BUILD SUCCESSFUL (total time: 4 seconds)
```

4. Hitung pangkat sebuah nilai berdasarkan angka dan nilai pangkatnya. Angka dan nilai pangkat diinputkan. Buat tiga versi dari program ini menggunakan while loop, do-while dan for-loop.

Jawab :

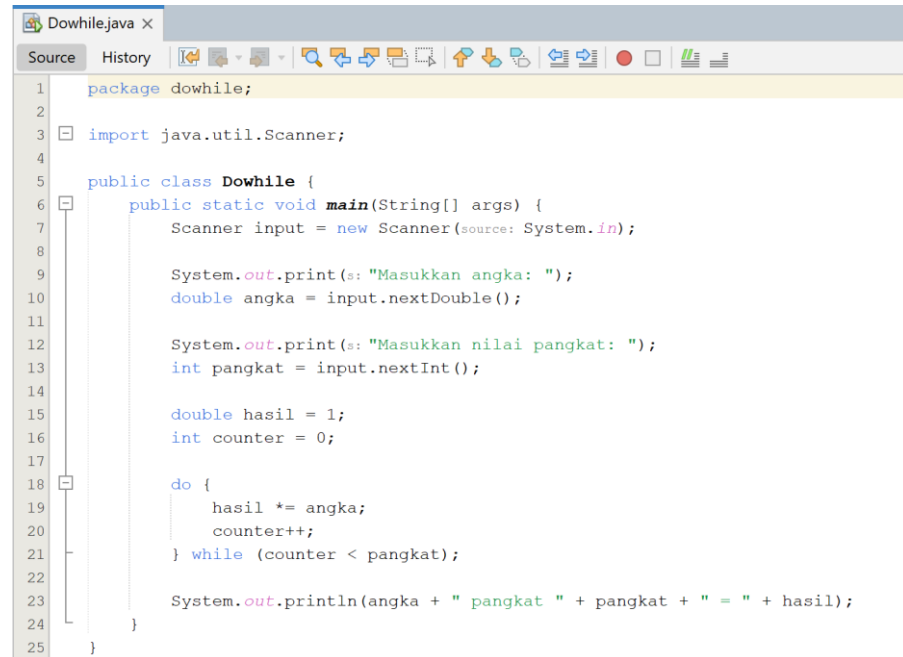
A. While Loop



```
1 package pkgwhile;
2
3 import java.util.Scanner;
4
5 public class While {
6     public static void main(String[] args) {
7         Scanner input = new Scanner(System.in);
8
9         System.out.print("Masukkan angka: ");
10        double angka = input.nextDouble();
11
12        System.out.print("Masukkan nilai pangkat: ");
13        int pangkat = input.nextInt();
14
15        double hasil = 1;
16        int counter = 0;
17
18        while (counter < pangkat) {
19            hasil *= angka;
20            counter++;
21        }
22
23        System.out.println(angka + " pangkat " + pangkat + " = " + hasil);
24    }
25 }
```

- Program ini menggunakan while loop untuk menghitung pangkat.
- Pengguna diminta untuk memasukkan angka dan nilai pangkat.
- Variabel hasil digunakan untuk menyimpan hasil pangkat.
- Loop akan terus berjalan selama counter kurang dari pangkat.
- Setiap iterasi, nilai hasil akan dikalikan dengan angka, dan counter akan ditingkatkan.
- Hasil pangkat akan dicetak setelah loop selesai.

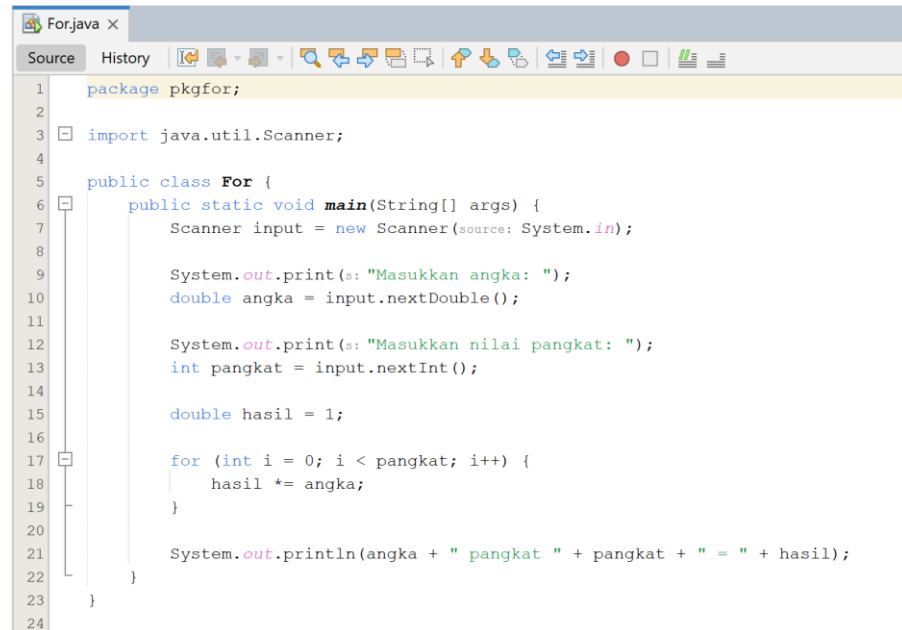
B. DoWhile Loop

A screenshot of a Java IDE window titled 'Dowhile.java'. The code is as follows:

```
1 package dowhile;
2
3 import java.util.Scanner;
4
5 public class Dowhile {
6     public static void main(String[] args) {
7         Scanner input = new Scanner(System.in);
8
9         System.out.print(s: "Masukkan angka: ");
10        double angka = input.nextDouble();
11
12        System.out.print(s: "Masukkan nilai pangkat: ");
13        int pangkat = input.nextInt();
14
15        double hasil = 1;
16        int counter = 0;
17
18        do {
19            hasil *= angka;
20            counter++;
21        } while (counter < pangkat);
22
23        System.out.println(angka + " pangkat " + pangkat + " = " + hasil);
24    }
25 }
```

- Program ini menggunakan do-while loop untuk menghitung pangkat.
- Seperti sebelumnya, pengguna diminta untuk memasukkan angka dan nilai pangkat.
- Variabel hasil digunakan untuk menyimpan hasil pangkat.
- Loop akan pertama kali dijalankan, dan kemudian kondisi akan diperiksa. Jika kondisi terpenuhi, loop akan terus berjalan.
- Seperti while loop, dalam setiap iterasi, nilai hasil akan dikalikan dengan angka.
- Hasil pangkat akan dicetak setelah loop selesai.

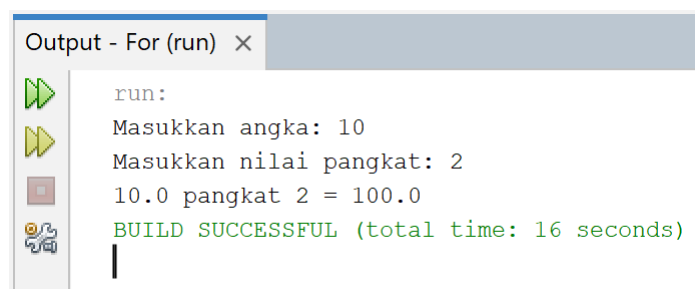
C. For Loop



```
1 package pkgfor;
2
3 import java.util.Scanner;
4
5 public class For {
6     public static void main(String[] args) {
7         Scanner input = new Scanner(System.in);
8
9         System.out.print(s: "Masukkan angka: ");
10        double angka = input.nextDouble();
11
12        System.out.print(s: "Masukkan nilai pangkat: ");
13        int pangkat = input.nextInt();
14
15        double hasil = 1;
16
17        for (int i = 0; i < pangkat; i++) {
18            hasil *= angka;
19        }
20
21        System.out.println(angka + " pangkat " + pangkat + " = " + hasil);
22    }
23 }
24
```

- Program ini menggunakan for loop untuk menghitung pangkat.
- Pengguna diminta untuk memasukkan angka dan nilai pangkat.
- Variabel hasil digunakan untuk menyimpan hasil pangkat.
- Loop for memiliki tiga bagian: inisialisasi (`int i = 0`), kondisi (`i < pangkat`), dan pembaruan (`i++`).
- Loop akan berjalan selama kondisi terpenuhi.
- Dalam setiap iterasi, nilai hasil akan dikalikan dengan angka.
- Hasil pangkat akan dicetak setelah loop selesai.

Perbedaan utama antara ketiga program ini adalah jenis loop yang digunakan untuk menghitung pangkat, tetapi hasil akhirnya akan sama. Program while dan do-while memeriksa kondisi sebelum loop dimulai, sementara program for mendefinisikan semua bagian loop dalam satu baris. Berikut hasil dari program tersebut



```
Output - For (run) X
run:
Masukkan angka: 10
Masukkan nilai pangkat: 2
10.0 pangkat 2 = 100.0
BUILD SUCCESSFUL (total time: 16 seconds)
```

5. Buatlah sebuah program untuk menginputkan beberapa buah nilai dan mencari total nilai serta rata-rata dari nilai yang dimasukkan menggunakan konsep array. User dapat menentukan jumlah nilai yang akan diinputkan. Selain nilai rata-rata, program juga dapat menentukan nilai yang paling besar dan nilai yang paling rendah

Jawab:

Berikut Kode yang saya buat untuk program tersebut

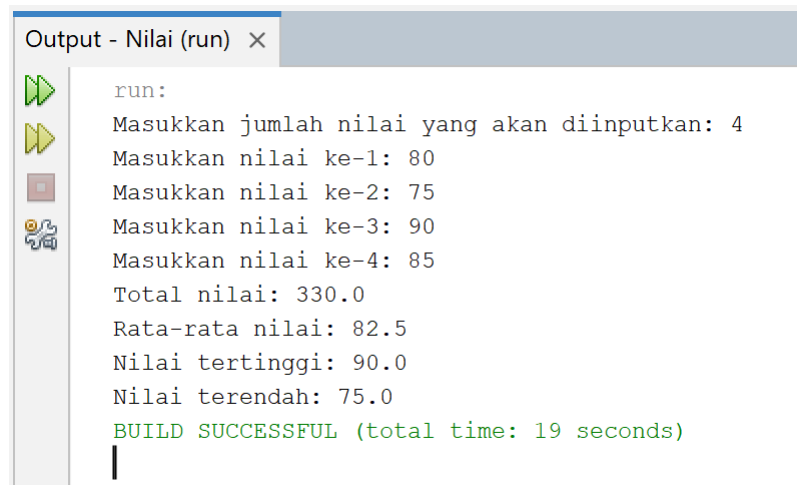


```
1 package nilai;
2
3 import java.util.Scanner;
4
5 public class Nilai {
6     public static void main(String[] args) {
7         try (Scanner input = new Scanner(System.in)) {
8             System.out.print("Masukkan jumlah nilai yang akan diinputkan: ");
9             int jumlahNilai = input.nextInt();
10
11             // Membuat array untuk menyimpan nilai-nilai
12             double[] nilai = new double[jumlahNilai];
13
14             // Memasukkan nilai-nilai ke dalam array
15             for (int i = 0; i < jumlahNilai; i++) {
16                 System.out.print("Masukkan nilai ke-" + (i + 1) + ": ");
17                 nilai[i] = input.nextDouble();
18             }
19
20             // Menghitung total nilai
21             double total = 0;
22             for (int i = 0; i < jumlahNilai; i++) {
23                 total += nilai[i];
24             }
25
26             // Menghitung rata-rata
27             double rataRata = total / jumlahNilai;
28
29             // Mencari nilai tertinggi dan terendah
30             double nilaiTertinggi = nilai[0];
31             double nilaiTerendah = nilai[0];
32             for (int i = 1; i < jumlahNilai; i++) {
33                 if (nilai[i] > nilaiTertinggi) {
34                     nilaiTertinggi = nilai[i];
35                 }
36                 if (nilai[i] < nilaiTerendah) {
37                     nilaiTerendah = nilai[i];
38                 }
39             }
40
41             // Menampilkan hasil
42             System.out.println("Total nilai: " + total);
43             System.out.println("Rata-rata nilai: " + rataRata);
44             System.out.println("Nilai tertinggi: " + nilaiTertinggi);
45             System.out.println("Nilai terendah: " + nilaiTerendah);
46         }
47     }
48 }
49
```

Program yang saya buat adalah program Java sederhana yang memungkinkan pengguna untuk memasukkan beberapa nilai, dan kemudian menghitung total nilai, rata-rata, nilai tertinggi, dan nilai terendah dari nilai-nilai yang dimasukkan. Berikut adalah penjelasan lebih rinci tentang bagian-bagian program tersebut:

- **Impor Pustaka Scanner:** Ini adalah bagian awal program yang mengimpor pustaka Scanner dari Java, yang akan digunakan untuk mendapatkan masukan dari pengguna.
- **Membaca Jumlah Nilai yang Akan Diinputkan:** Program akan meminta pengguna untuk memasukkan jumlah nilai yang akan diinputkan. Nilai ini akan digunakan untuk menentukan panjang array yang akan digunakan untuk menyimpan nilai-nilai tersebut.
- **Membuat dan Mengisi Array:** Setelah mendapatkan jumlah nilai yang akan diinputkan, program membuat sebuah array bernama "nilai" dengan tipe data double dan panjang sesuai dengan jumlah nilai yang dimasukkan oleh pengguna. Selanjutnya, program meminta pengguna untuk memasukkan nilai-nilai tersebut dalam loop for, dan nilai-nilai tersebut disimpan dalam array pada indeks yang sesuai.
- **Menghitung Total Nilai:** Program menghitung total nilai dengan menjumlahkan semua nilai dalam array "nilai".
- **Menghitung Rata-Rata Nilai:** Rata-rata nilai dihitung dengan membagi total nilai dengan jumlah nilai yang diinputkan.
- **Mencari Nilai Tertinggi dan Terendah:** Program mencari nilai tertinggi dan terendah dalam array "nilai" dengan menggunakan loop for. Pada awalnya, kita menginisialisasi nilai tertinggi dan terendah dengan nilai pertama dalam array, kemudian kita membandingkan setiap nilai dalam array dengan nilai tertinggi dan terendah saat ini, dan mengupdate nilai tertinggi dan terendah jika ditemukan nilai yang lebih besar atau lebih kecil.
- **Menampilkan Hasil:** Program akhirnya menampilkan total nilai, rata-rata nilai, nilai tertinggi, dan nilai terendah kepada pengguna.

Berikut Hasil dari program Ketika dijalankan :



The screenshot shows an IDE's output window titled "Output - Nilai (run) ×". On the left side of the window is a vertical toolbar with icons for running (green play button), stepping through (yellow play button), stopping (red square), and a bug icon. The output text is as follows:

```
run:
Masukkan jumlah nilai yang akan diinputkan: 4
Masukkan nilai ke-1: 80
Masukkan nilai ke-2: 75
Masukkan nilai ke-3: 90
Masukkan nilai ke-4: 85
Total nilai: 330.0
Rata-rata nilai: 82.5
Nilai tertinggi: 90.0
Nilai terendah: 75.0
BUILD SUCCESSFUL (total time: 19 seconds)
```

A cursor is visible at the end of the last line of output.