

UTS PRAKTIKUM DASAR PEMROGRAMAN

NAMA: STEVAN ZAKY SETYANTO

NIM: 2341720101

KELAS/NO: TI 1A / 29

1. Buatlah program untuk menginputkan tiga buah bilangan yang menyatakan sudut-sudut sebuah segitiga. Bila **salah satu sudut** nilainya sama dengan **jumlah nilai dua sudut yang lain**, maka cetak **SIKU-SIKU**, bila tidak maka cetak **BUKAN SIKU-SIKU**.
Dengan ketentuan nilai input sebagai berikut:
Total ketiga sudut harus **sama dengan 180°**
Sudut yang dimasukkan tidak boleh bernilai **negative**
Setiap sudut yang dimasukkan nilainya harus **lebih besar** dari 0° dan **kurang dari 180°**
Tampilkan pesan kesalahan jika nilai input sudut tidak sesuai.

```
UtsDerajatSegitiga29.java > UtsDerajatSegitiga29 > main(String[])
1  import java.util.Scanner;
2
3  public class UtsDerajatSegitiga29 {
4      public static void main(String[] args) {
5          Scanner input = new Scanner(System.in);
6
7          System.out.println(x:"Masukkan tiga sudut segitiga:");
8          System.out.print(s:"Sudut 1: ");
9          int sudut1 = input.nextInt();
10         System.out.print(s:"Sudut 2: ");
11         int sudut2 = input.nextInt();
12         System.out.print(s:"Sudut 3: ");
13         int sudut3 = input.nextInt();
14
15
16         if (sudut1 + sudut2 + sudut3 == 180) {
17
18             if (sudut1 > 0 && sudut2 > 0 && sudut3 > 0) {
19
20                 if (sudut1 == 90 || sudut2 == 90 || sudut3 == 90) {
21                     System.out.println(x:"Segitiga ini SIKU-SIKU.");
22                 } else {
23                     System.out.println(x:"Segitiga ini BUKAN SIKU-SIKU.");
24                 }
25             } else {
26                 System.out.println(x:"Sudut harus lebih besar dari 0° dan kurang dari 180°.");
27             }
28         } else {
29             System.out.println(x:"Nilai input sudut tidak sesuai.");
30         }
31
32     }
33 }
34
35
```

```

AppData\Roaming\Code\User\workspaceStorage\07115d02cbeac7f842256174908d7da\redhat.java\jdt_ws\BELAJAR_MANDIRI_9cc65100\bin' 'UtsDerajatSegitiga29'
Masukkan tiga sudut segitiga:
Sudut 1: 0
Sudut 2: 0
Sudut 3: 0
Nilai input sudut tidak sesuai.
PS D:\BELAJAR_MANDIRI> d:; cd 'd:\BELAJAR_MANDIRI'; & 'C:\Program Files\Java\jdk-20\bin\java.exe' '-enable-preview' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\Lenovo\AppData\Roaming\Code\User\workspaceStorage\07115d02cbeac7f842256174908d7da\redhat.java\jdt_ws\BELAJAR_MANDIRI_9cc65100\bin' 'UtsDerajatSegitiga29'
Masukkan tiga sudut segitiga:
Sudut 1: 90
Sudut 2: 60
Sudut 3: 30
Segitiga ini SIKU-SIKU.
PS D:\BELAJAR_MANDIRI> d:; cd 'd:\BELAJAR_MANDIRI'; & 'C:\Program Files\Java\jdk-20\bin\java.exe' '-enable-preview' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\Lenovo\AppData\Roaming\Code\User\workspaceStorage\07115d02cbeac7f842256174908d7da\redhat.java\jdt_ws\BELAJAR_MANDIRI_9cc65100\bin' 'UtsDerajatSegitiga29'
Masukkan tiga sudut segitiga:
Sudut 1: 50
Sudut 2: 50
Sudut 3: 80
Segitiga ini BUKAN SIKU-SIKU.
PS D:\BELAJAR_MANDIRI>

```

1. Import Kelas Scanner
Program menggunakan kelas Scanner untuk menerima masukan dari pengguna.
2. Membuat Objek Scanner
Membuat objek Scanner yang digunakan untuk membaca dari pengguna.
3. Meminta Masukan Sudut Segitiga
Program meminta pengguna memasukkan tiga sudut segitiga
4. Memeriksa Total Sudut
Program memeriksa apakah total dari ketiga sudut adalah 180, sesuai dengan aturan bahwa total sudut dalam segitiga adalah 180
5. Memeriksa Sudut Positif
Selanjutnya, program memeriksa apakah setiap sudut adalah bilangan positif yang lebih besar dari 0, sesuai dengan aturan sudut dalam segitiga
6. Memeriksa Ketentuan Siku-Siku
Jika segitiga memenuhi ketentuan total sudut 180 dan semua sudut positif, program memeriksa apakah salah satu dari sudut tersebut adalah siku (90). Jika salah satunya adalah sudut siku, program mencetak "Segitiga ini SIKU-SIKU". Jika tidak, program mencetak "Segitiga ini BUKAN SIKU-SIKU".
7. Penanganan Kesalahan
Menangani kesalahan jika total sudut tidak sama dengan 180 atau jika salah satu sudut tidak atau kurang dari 0. Dalam kasus ini, program akan mencetak pesan kesalahan yang sesuai.

2. Buatlah sebuah program dengan menggunakan bahasa Java yang meminta masukan *user* sebuah bilangan bulat **N** ($N > 0$). Kemudian, program menampilkan penjumlahan **N** bilangan kuadrat pertama. Bilangan kuadrat adalah = {1, 4, 9, 16, 25, 36, ..., N^2 }.

Contoh:

- Jika *user* memasukkan $N = 2$, itu artinya : $1 + 4 = 5$
- Jika *user* memasukkan $N = 3$, itu artinya : $1 + 4 + 9 = 14$

```
J UtsNilai29.java > UtsNilai29 > main(String[])
1  import java.util.Scanner;
2
3  public class UtsNilai29 {
4      public static void main(String[] args) {
5          Scanner input = new Scanner(System.in);
6
7          System.out.print(s:"Masukkan bilangan bulat N (N > 0): ");
8          int N = input.nextInt();
9
10         if (N <= 0) {
11             System.out.println(x:"N harus lebih dari 0.");
12         } else {
13             int sum = 0;
14             for (int i = 1; i <= N; i++) {
15                 int kuadrat = i * i;
16                 sum += kuadrat;
17             }
18
19             System.out.println("Hasil penjumlahan N bilangan kuadrat pertama adalah: " + sum);
20
21         }
22     }
23 }
```

```
Masukkan bilangan bulat N (N > 0): 3
Hasil penjumlahan N bilangan kuadrat pertama adalah: 14
PS D:\BELAJAR MANDIRI>
```

1. Impor Kelas Scanner:

Program menggunakan kelas Scanner untuk menerima masukan dari pengguna.

2. Membuat Objek Scanner:

Ini membuat objek Scanner yang digunakan untuk membaca masukan dari pengguna.

3. Meminta Masukan Bilangan Bulat N:

Program meminta pengguna memasukkan bilangan bulat N.

4. Memeriksa Validitas N:

Program memeriksa apakah N lebih besar dari 0. Jika N kurang dari atau sama dengan 0, program mencetak pesan kesalahan.

5. Menghitung Penjumlahan Bilangan Kuadrat Pertama:

Jika N valid, program menggunakan loop for untuk menghitung penjumlahan N bilangan kuadrat pertama. Variabel sum awalnya diatur ke 0, dan dalam setiap iterasi, nilai $i * i$ dihitung dan ditambahkan ke sum.

6. Menampilkan Hasil:

Setelah semua bilangan kuadrat pertama dijumlahkan, program mencetak hasilnya.

3. Buatlah sebuah program dengan menggunakan bahasa Java yang meminta masukan *user* sebuah bilangan bulat **N** ($N > 0$). Program kemudian memeriksa setiap digit yang ada di angka tersebut, dan menampilkan berapa jumlah digit yang ganjil dari bilangan **N** tersebut.

Contoh:

Jika *user* memasukkan $N = 2345$, jumlah digit yang ganjil =

Jika *user* memasukkan $N = 993312$, jumlah digit yang ganjil = 5

```
J Ganjil29.java
1  import java.util.Scanner;
2
3  public class Ganjil29 {
4      Run | Debug
5      public static void main(String[] args) {
6          Scanner input = new Scanner(System.in);
7
8          System.out.print(s:"Masukkan bilangan bulat N (N > 0): ");
9          int N = input.nextInt();
10
11         if (N <= 0) {
12             System.out.println(x:"N harus lebih dari 0.");
13         } else {
14             int jumlahGanjil = 0;
15
16             while (N > 0) {
17                 int digit = N % 10;
18                 if (digit % 2 != 0) {
19                     jumlahGanjil++;
20                 }
21                 N = N / 10;
22             }
23
24             System.out.println("Jumlah digit ganjil dalam bilangan N = " + jumlahGanjil);
25         }
26     }
27 }
```

```
Masukkan bilangan bulat N (N > 0): 3
Jumlah digit ganjil dalam bilangan N = 1
PS D:\BELAJAR MANDIRI> d:; cd 'd:\BELAJAR MANDIRI\workspaceStorage\
Masukkan bilangan bulat N (N > 0): 13
Jumlah digit ganjil dalam bilangan N = 2
PS D:\BELAJAR MANDIRI> 
```

1.Import Kelas Scanner:

Program menggunakan kelas Scanner untuk menerima masukan dari pengguna.

2.Membuat Objek Scanner:

Ini membuat objek Scanner yang digunakan untuk membaca masukan dari pengguna.

3. Meminta Masukan Bilangan Bulat N:

Program meminta pengguna memasukkan bilangan bulat N.

4.Memeriksa Validitas N:

Program memeriksa apakah N lebih besar dari 0. Jika N kurang dari atau sama dengan 0, program mencetak pesan kesalahan.

5.Menghitung Jumlah Digit Ganjil:

Program menggunakan loop while untuk mengiterasi melalui digit-digit bilangan N. Pada setiap iterasi, digit paling kanan diambil dengan menghitung $N \bmod 10$. Kemudian, program memeriksa apakah digit tersebut ganjil ($\text{digit} \% 2 \neq 0$). Jika iya, jumlah digit ganjil ditingkatkan. Selanjutnya, digit paling kanan dihapus dengan membagi N dengan 10.

6. Menampilkan Hasil:

Setelah semua digit telah diperiksa, program mencetak jumlah digit ganjil yang ditemukan dalam bilangan N.