

## **JOBSHEET 3**

### **LOOPING**



19

Muhammad Alif Febriansyah

2341720025

<https://github.com/alfbrynn>

Teknologi Informasi \_ Teknik Informatika

Polinema

2023/2024

## Percobaan 1

```
1 import java.util.Scanner;
2 public class ForKelipatan19 {
3     public static void main(String[] args) {
4         Scanner scan = new Scanner(System.in);
5         int kelipatan, jumlah, counter;
6         jumlah = 0;
7         counter = 0;
8         System.out.println("Masukkan bilangan kelipatan (1-9) : ");
9         kelipatan = scan.nextInt();
10
11         for (int i = 1; i <= 50; i++) {
12             if (i % kelipatan == 0) {
13                 jumlah += i;
14                 counter++;
15             }
16         }
17
18         System.out.printf("Banyaknya bilangan %d dari 1 sampai 50 adalah %d\n", kelipatan, counter);
19         System.out.printf("Total bilangan kelipatan %d dari 1 sampai 50 adalah %d\n", kelipatan, jumlah);
20     }
21 }
```

## Output

```
PS C:\Users\Zazil> java -Xmx2G -XX:+ShowCodeDetailsInExceptionMessages -cp 'C:\Users\Zazil\AppData\Local\Temp\vs
codesws_03c68\jdt_ws\jdt.ls-java-project\bin' 'ForKelipatan19'
Masukkan bilangan kelipatan (1-9) :
5
Banyaknya bilangan 5 dari 1 sampai 50 adalah 10
Total bilangan kelipatan 5 dari 1 sampai 50 adalah 275
PS C:\Users\Zazil>
```

## Pertanyaan

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!

Komponen perulangan pada sintaks FOR adalah:

- Inisialisasi: `int i = 1;` (inisialisasi variabel i dengan nilai 1)
  - Kondisi: `i <= 50;` (kondisi untuk menjalankan perulangan selama i kurang dari atau sama dengan 50)
  - Perubahan: `i++` (increment i setiap kali iterasi)
2. Jelaskan alur kerja dari potongan kode program berikut!  
Untuk menjelaskan alur kerja dari potongan kode program tersebut:
    1. Inisialisasi variabel: Variabel `i` diinisialisasi dengan nilai 1, yang akan digunakan sebagai penghitung untuk iterasi perulangan.
    2. Perulangan for: Perulangan akan berjalan sebanyak 50 kali, dimulai dari 1 hingga 50.

3. Pengecekan kelipatan: Pada setiap iterasi, dilakukan pengecekan apakah `i` merupakan kelipatan dari bilangan yang dimasukkan pengguna (`kelipatan`). Ini dilakukan dengan menggunakan operator modulo (`%`). Jika `i` adalah kelipatan, maka akan memasuki blok if.
4. Penjumlahan dan pencatatan: Jika `i` adalah kelipatan, maka nilai `i` akan ditambahkan ke variabel `jumlah`, yang digunakan untuk menghitung total dari bilangan-bilangan kelipatan. Selain itu, variabel `counter` akan diincrement untuk menghitung banyaknya bilangan kelipatan.
5. Perubahan nilai `i`: Setelah setiap iterasi, nilai `i` akan diincrement menggunakan pernyataan `i++`.
6. Pengecekan batas perulangan: Perulangan akan terus berlanjut selama nilai `i` masih kurang dari atau sama dengan 50.

Jadi, potongan kode program ini melakukan iterasi dari 1 hingga 50, memeriksa apakah setiap bilangan adalah kelipatan dari bilangan yang dimasukkan pengguna, dan mengakumulasi total serta mencatat banyaknya bilangan kelipatan.

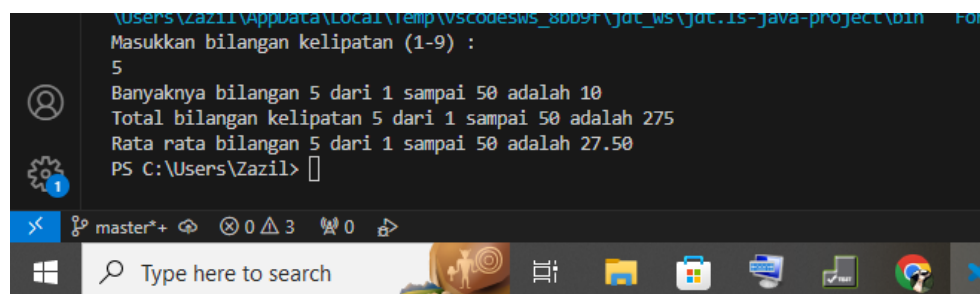
3. Modifikasi kode program yang telah dibuat dengan menambahkan variabel baru untuk menghitung rata-rata dari seluruh bilangan kelipatan yang ditentukan! Push dan commit kode program ke github.

```

1 import java.util.Scanner;
2 public class ForKelipatan19 {
3     public static void main(String[] args) {
4         Scanner scan = new Scanner(System.in);
5         int kelipatan, jumlah, counter;
6         double rataRata;
7         jumlah = 0;
8         counter = 0;
9         System.out.println("Masukkan bilangan kelipatan (1-9) : ");
10        kelipatan = scan.nextInt();
11
12        for (int i = 1; i <= 50; i++) {
13            if (i % kelipatan == 0) {
14                jumlah += i;
15                counter++;
16            }
17        }
18
19        rataRata = (double) jumlah / counter;
20
21        System.out.printf("Banyaknya bilangan %d dari 1 sampai 50 adalah %d\n", kelipatan, counter);
22        System.out.printf("Total bilangan kelipatan %d dari 1 sampai 50 adalah %d\n", kelipatan, jumlah);
23        System.out.printf("Rata rata bilangan %d dari 1 sampai 50 adalah %.2f\n", kelipatan, rataRata);
24    }
25 }

```

Output



```

C:\Users\Zazil\AppData\Local\Temp\vscodesws_8bb9f\Jot_ws\Jot-15-java-project\bin For
Masukkan bilangan kelipatan (1-9) :
5
Banyaknya bilangan 5 dari 1 sampai 50 adalah 10
Total bilangan kelipatan 5 dari 1 sampai 50 adalah 275
Rata rata bilangan 5 dari 1 sampai 50 adalah 27.50
PS C:\Users\Zazil>

```

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

```

1  import java.util.Scanner;
2
3  public class WhileKelipatan19 {
4      public static void main(String[] args) {
5          Scanner scan = new Scanner(System.in);
6          int kelipatan, jumlah, counter;
7          double rataRata;
8          jumlah = 0;
9          counter = 0;
10         System.out.println("Masukkan bilangan kelipatan (1-9) : ");
11         kelipatan = scan.nextInt();
12         int i = 1;
13
14         while (i <= 50) {
15             if (i % kelipatan == 0) {
16                 jumlah += i;
17                 counter++;
18             }
19             i++;
20         }
21
22         rataRata = (double) jumlah / counter;
23
24         System.out.printf("Banyaknya bilangan %d dari 1 sampai 50 adalah %d\n", kelipatan, counter);
25         System.out.printf("Total bilangan kelipatan %d dari 1 sampai 50 adalah %d\n", kelipatan, jumlah);
26         System.out.printf("Rata-rata bilangan kelipatan %d dari 1 sampai 50 adalah %.2f\n", kelipatan, rataRata);
27     }
28 }

```

Output

```

\Users\Zazil\AppData\Local\Temp\vscodesws_8bb9f\jdt_ws\jdt.ls-java-p
roject\bin' 'WhileKelipatan19'
Masukkan bilangan kelipatan (1-9) :
5
Banyaknya bilangan 5 dari 1 sampai 50 adalah 10
Total bilangan kelipatan 5 dari 1 sampai 50 adalah 275
Rata-rata bilangan kelipatan 5 dari 1 sampai 50 adalah 27.50
PS C:\Users\Zazil>

```

percobaan 2

```

import java.util.Scanner;

public class WhileGaji19 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int jumlahKaryawan, jumlahJamLembur;
        double gajiLembur, totalGajiLembur;
        gajiLembur = 0;
        totalGajiLembur = 0;
        String jabatan;

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

        int i = 0;
    }
}

```

```

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

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

    if (jabatan.equalsIgnoreCase("Direktur")) {
        continue;
    } else if (jabatan.equalsIgnoreCase("Manager")) {
        gajiLembur = jumlahJamLembur * 100000;
    } else if (jabatan.equalsIgnoreCase("Karyawan")) {
        gajiLembur = jumlahJamLembur * 75000;
    }

    totalGajiLembur += gajiLembur;
    System.out.println("Total gaji lembur: " + totalGajiLembur);
}
}
}

```

Output

```

PS C:\Users\Zazil> java -project\bin\WhileGaji19
Masukkan jumlah karyawan: 2
Pilihan jabatan - Direktur, Manager, Karyawan
Masukkan jabatan karyawan ke-1: direktur
Masukkan jumlah jam lembur: 3
Pilihan jabatan - Direktur, Manager, Karyawan
Total gaji lembur: 400000.0
Pilihan jabatan - Direktur, Manager, Karyawan
Masukkan jabatan karyawan ke-2: karyawan
Masukkan jumlah jam lembur: 5
Total gaji lembur: 775000.0
PS C:\Users\Zazil> & 'C:\Program Files\Java\jdk-20\bin\java.exe' '--er
sers\Zazil\AppData\Local\Temp\vscodesws_bff00\jdt_ws\jdt.ls-java-projec

```

Pertanyaan 1

1. Tunjukkan bagian kode program yang digunakan sebagai syarat untuk menghentikan perulangan WHILE! Berapa kali perulangan dilakukan?  
Perulangan WHILE dalam program ini akan berhenti saat variable 'i' mencapai nilai yang sama dengan 'JumlahKaryawan', sesuai dengan kondisi while 'i < jumlahKaryawan'.
2. Pada potongan kode berikut, Apa yang sebenarnya terjadi jika variabel jabatan berisi nilai "DIREKTUR"? Apa peran CONTINUE yang dituliskan di dalam sintaks perulangan?  
Jika variabel `jabatan` berisi nilai "DIREKTUR", maka kondisi `if (jabatan.equalsIgnoreCase("Direktur"))` akan dievaluasi sebagai true, dan pernyataan

`continue` akan memulai iterasi perulangan baru tanpa mengeksekusi kode di bawahnya dalam lingkup while. Dalam konteks ini, jika jabatan adalah "DIREKTUR", perulangan akan melanjutkan ke iterasi berikutnya tanpa menghitung gaji lembur dan mencetak total gaji lembur.

3. 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!

Pembaruan `i++` diletakkan di posisi tengah karena keperluan perulangan dalam kode ini adalah untuk mendapatkan input dan menghitung gaji lembur untuk setiap karyawan. Memindahkan `i++` ke bagian akhir akan mengakibatkan peningkatan `i` terjadi setelah semua perhitungan dan pencetakan gaji lembur dilakukan, sehingga perulangan akan keliru menghitung gaji lembur untuk karyawan selanjutnya. Saat "direktur" dimasukkan sebagai jabatan karyawan pertama, karena perulangan terus berlanjut tanpa melakukan perhitungan gaji lembur, program akan mengabaikan input "direktur" dan melanjutkan ke karyawan berikutnya tanpa memperhitungkan gaji lembur untuk direktur.

4. Modifikasi program untuk handle jabatan yang invalid

```
import java.util.Scanner;

public class WhileGaji19 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int jumlahKaryawan, jumlahJamLembur;
        double gajiLembur, totalGajiLembur;
        gajiLembur = 0;
        totalGajiLembur = 0;
        String jabatan;

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

        int i = 0;

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

            if (!(jabatan.equalsIgnoreCase("Direktur") || jabatan.equalsIgnoreCase("Manager") ||
                jabatan.equalsIgnoreCase("Karyawan"))) {
                System.out.println("Jabatan tidak valid. Silakan masukkan jabatan yang valid.3 ");
                continue;
            }

            System.out.print("Masukkan jumlah jam lembur: ");
```

```

    jumlahJamLembur = sc.nextInt();
    i++;

    if (jabatan.equalsIgnoreCase("Direktur")) {
        continue;
    } else if (jabatan.equalsIgnoreCase("Manager")) {
        gajiLembur = jumlahJamLembur * 100000;
    } else if (jabatan.equalsIgnoreCase("Karyawan")) {
        gajiLembur = jumlahJamLembur * 75000;
    }

    totalGajiLembur += gajiLembur;
    System.out.println("Total gaji lembur: " + totalGajiLembur);
}
}
}

```

output

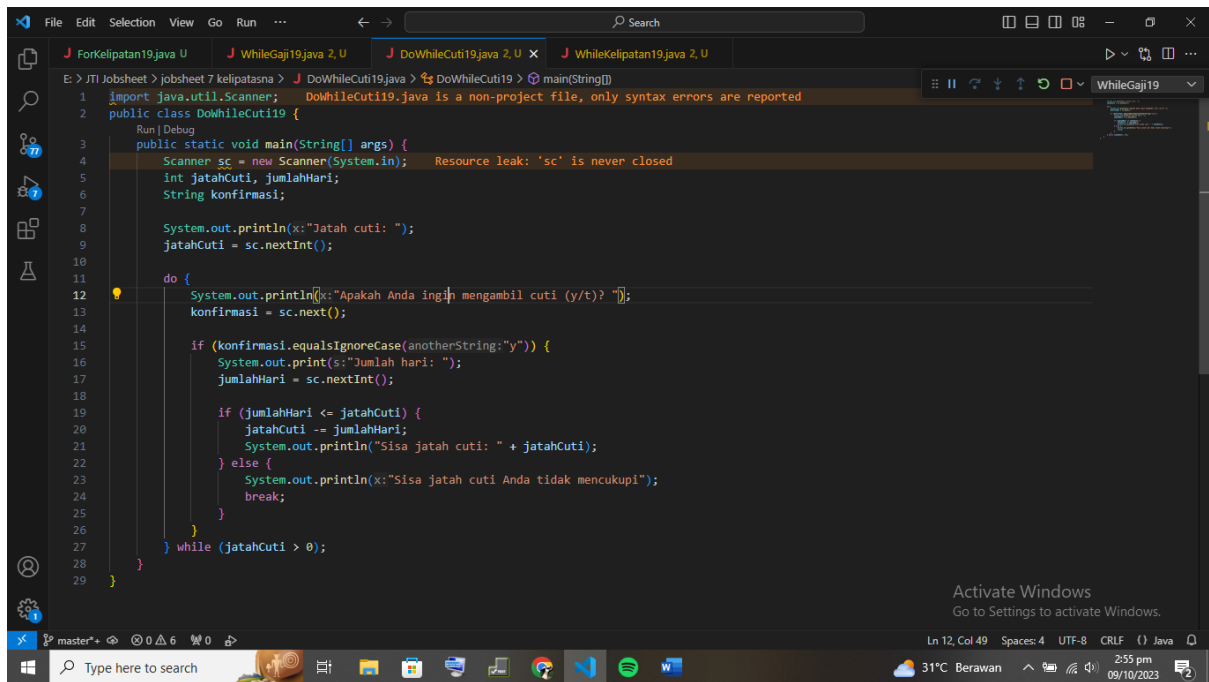
```

java-project\bin' 'WhileGaji19'
Masukkan jumlah karyawan: 3
Pilihan jabatan - Direktur, Manager, Karyawan
Masukkan jabatan karyawan ke-1: manager

Masukkan jumlah jam lembur: 5
Pilihan jabatan - Direktur, Manager, Karyawan
Masukkan jabatan karyawan ke-3: pegawai
Jabatan tidak valid. Silakan masukkan jabatan jabatan yang valid.3
Pilihan jabatan - Direktur, Manager, Karyawan
Masukkan jabatan karyawan ke-3: karyawan
Masukkan jumlah jam lembur: 6
Total gaji lembur: 1050000.0
PS C:\Users\Zazil>

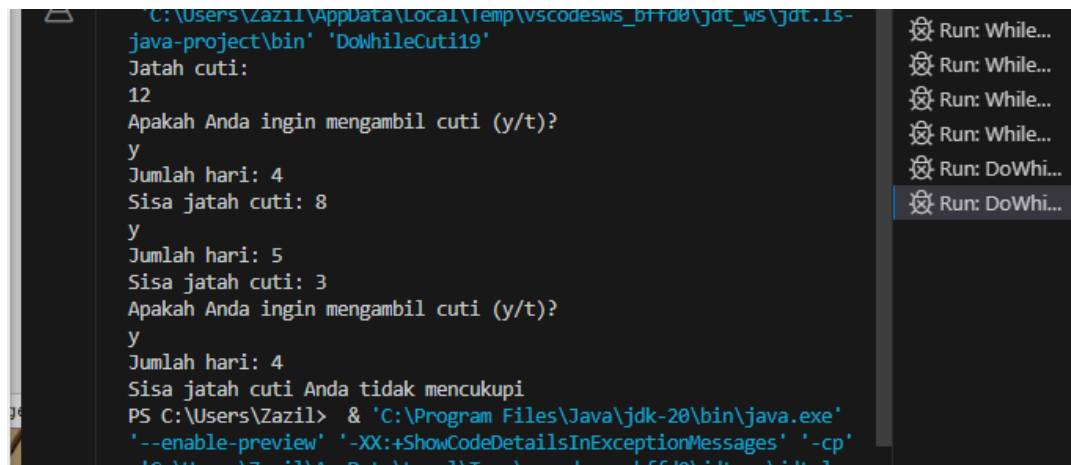
```

Percobaan 3



```
1 import java.util.Scanner;
2 public class DoWhileCuti19 {
3     public static void main(String[] args) {
4         Scanner sc = new Scanner(System.in);
5         int jatahCuti, jumlahHari;
6         String konfirmasi;
7
8         System.out.println(x:"Jatah cuti: ");
9         jatahCuti = sc.nextInt();
10
11         do {
12             System.out.println(x:"Apakah Anda ingin mengambil cuti (y/t)? ");
13             konfirmasi = sc.next();
14
15             if (konfirmasi.equalsIgnoreCase("y")) {
16                 System.out.print(s:"Jumlah hari: ");
17                 jumlahHari = sc.nextInt();
18
19                 if (jumlahHari <= jatahCuti) {
20                     jatahCuti -= jumlahHari;
21                     System.out.println("Sisa jatah cuti: " + jatahCuti);
22                 } else {
23                     System.out.println(x:"Sisa jatah cuti Anda tidak mencukupi");
24                     break;
25                 }
26             }
27         } while (jatahCuti > 0);
28     }
29 }
```

## Output



```
C:\Users\Zazil\AppData\Local\Temp\vscodesws_bffd0\jdt_ws\jdt.ls-
java-project\bin\DoWhileCuti19
Jatah cuti:
12
Apakah Anda ingin mengambil cuti (y/t)?
y
Jumlah hari: 4
Sisa jatah cuti: 8
y
Jumlah hari: 5
Sisa jatah cuti: 3
Apakah Anda ingin mengambil cuti (y/t)?
y
Jumlah hari: 4
Sisa jatah cuti Anda tidak mencukupi
PS C:\Users\Zazil> & 'C:\Program Files\Java\jdk-20\bin\java.exe'
'-enable-preview' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp'
'C:\Users\Zazil\AppData\Local\Temp\vscodesws_bffd0\jdt_ws\jdt.ls'
```

## Pertanyaan

1. Apa kegunaan sintaks BREAK di dalam sintaks perulangan?  
Untuk menghentikan eksekusi dari sebuah loop
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 mengisi jumlah hari sesuai jatah cuti.

```
public class DoWhileCuti119 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int jatahCuti, jumlahHari;
        String konfirmasi;

        System.out.print("Jatah cuti: ");
        jatahCuti = sc.nextInt();
```



```

outerLoop: do {
    System.out.print("Apakah Anda ingin mengambil cuti (y/t)? ");
    konfirmasi = sc.next();

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

        if (jumlahHari <= jatahCuti) {
            jatahCuti -= jumlahHari;
            System.out.println("Sisa jatah cuti: " + jatahCuti);
        } else {
            System.out.println("Sisa jatah cuti Anda tidak mencukupi");
            // Tambahkan inner loop untuk memungkinkan pengisian ulang jumlah hari
            while (true) {
                System.out.print("Apakah Anda ingin mengisi jumlah hari kembali (y/t)? ");
                konfirmasi = sc.next();
                if (konfirmasi.equalsIgnoreCase("y")) {
                    System.out.print("Jumlah hari: ");
                    jumlahHari = sc.nextInt();
                    if (jumlahHari <= jatahCuti) {
                        jatahCuti -= jumlahHari;
                        System.out.println("Sisa jatah cuti: " + jatahCuti);
                        break; // Keluar dari inner loop
                    } else {
                        System.out.println("Sisa jatah cuti Anda tidak mencukupi");
                    }
                } else {
                    break outerLoop; // Keluar dari outer loop jika pengguna tidak ingin isi
                    jumlah hari lagi
                }
            }
        }
    } while (jatahCuti > 0);

    System.out.println("Terima kasih!");
}
}

```

Output

```

mp\vscodesws_1139a\jdt_ws\jdt.ls-java-project\bin' 'DowhileCuti119'
Jatah cuti: 12
Apakah Anda ingin mengambil cuti (y/t)? y
Jumlah hari: 5
Sisa jatah cuti: 7
Apakah Anda ingin mengambil cuti (y/t)? y
Jumlah hari: 5
Sisa jatah cuti: 2
Apakah Anda ingin mengambil cuti (y/t)? y
Jumlah hari: 4
Sisa jatah cuti Anda tidak mencukupi
Apakah Anda ingin mengisi jumlah hari kembali (y/t)? y
Jumlah hari: 5
Sisa jatah cuti Anda tidak mencukupi
Apakah Anda ingin mengisi jumlah hari kembali (y/t)? y
Jumlah hari: 2
Sisa jatah cuti: 0
Terima kasih!
PS C:\Users\Zazil>

```

3. Push dan commit kode program ke github
4. Pada saat input konfirmasi, ketikkan “t”, apa yang terjadi? Mengapa demikian?

```

Apakah Anda ingin mengambil cuti (y/t)? t
Apakah Anda ingin mengambil cuti (y/t)? t
Apakah Anda ingin mengambil cuti (y/t)? t
Apakah Anda ingin mengambil cuti (y/t)? t
Apakah Anda ingin mengambil cuti (y/t)? t
Apakah Anda ingin mengambil cuti (y/t)? t

```

Jika user input t maka program akan melanjutkan ke Langkah selanjutnya sesuai dengan alur looping. Ini karena t tidak sama dengan y yang diharapkan dalam percabangan if

5. Modifikasi kode program sehingga saat pengguna mengetikkan “t” sebagai input konfirmasi, maka program akan berhenti

```

import java.util.Scanner;

public class DoWhileCuti219 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int jatahCuti, jumlahHari;
        String konfirmasi;

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

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

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

                if (jumlahHari <= jatahCuti) {
                    jatahCuti -= jumlahHari;
                    System.out.println("Sisa jatah cuti: " + jatahCuti);
                }
            }
        } while (konfirmasi.equalsIgnoreCase("y"));
    }
}

```

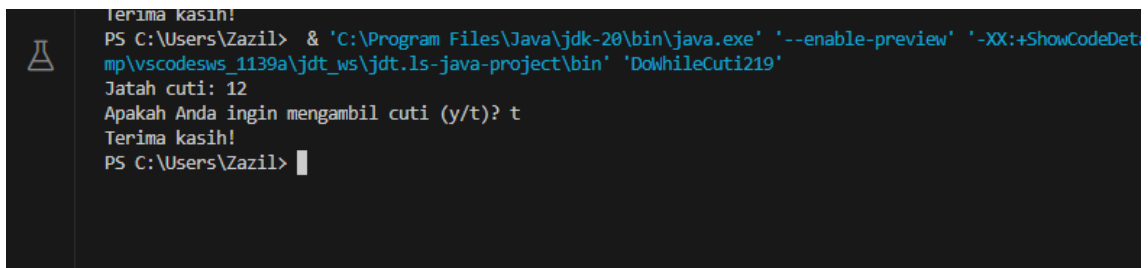
```

        } else {
            System.out.println("Sisa jatah cuti Anda tidak mencukupi");
            continue; // Lanjutkan ke iterasi berikutnya jika sisa jatah cuti tidak mencukupi
        }
    } else if (konfirmasi.equalsIgnoreCase("t")) {
        break; // Keluar dari loop jika pengguna memilih untuk tidak mengambil cuti
    } else {
        System.out.println("Input tidak valid. Silakan masukkan y atau t.");
    }
} while (jatahCuti > 0);

System.out.println("Terima kasih!");
}
}

```

Output



```

Terima kasih!
PS C:\Users\Zazil> & 'C:\Program Files\Java\jdk-20\bin\java.exe' '--enable-preview' '-XX:+ShowCodeDetails' '-Dmp\vscode\ws_1139a\jdt_ws\jdt.ls-java-project\bin' 'DowhileCuti219'
Jatah cuti: 12
Apakah Anda ingin mengambil cuti (y/t)? t
Terima kasih!
PS C:\Users\Zazil>

```

6. Push dan commit kode program ke github