



## MODUL PERKULIAHAN #5 **STRUKTUR PROGRAM PENGULANGAN**

|                      |   |   |
|----------------------|---|---|
| Capaian Pembelajaran | : | <b>Mahasiswa Mengerti dan Mampu:</b><br>Menuliskan penggunaan struktur program pengulangan dalam penulisan kode program dengan bahasa Java  |
| Sub Pokok Bahasan    | : | <ol style="list-style-type: none"><li>1. Struktur <b>for</b></li><li>2. Struktur <b>while</b></li><li>3. Struktur <b>do..while</b></li><li>4. Perintah <b>break</b> dan <b>continue</b></li></ol> |

|                |   |  |
|----------------|---|--|
| Daftar Pustaka | : | <ol style="list-style-type: none"> <li>1. Farrell. Joyce, An Object-Oriented Approach to Programming Logic and Design 3rd Edition, Course Technology, 2009</li> <li>2. Thomas Wu. C, An Introduction to Object-Oriented Programming with Java™ 4th Edition, McGraw Hill, 2006</li> <li>3. Deitel, Java How To Program, Deitel, Prentice Hall, 2008</li> <li>4. Anif, M, Seri Aplikasi : Aplikasi Penjualan (Pemesanan Barang) dengan Java, Mitra Wacana Media, 2008</li> </ol> |
|----------------|---|--|

## PRAKTIKUM 5

### STRUKTUR PROGRAM PENGULANGAN

#### 5.1 Teori Singkat

Pengulangan (looping) adalah suatu bagian yang bertugas melakukan kegiatan mengulang suatu proses sesuai dengan yang diinginkan. Banyak dari aplikasi perangkat lunak yang melakukan pekerjaan berulang sampai sebuah kondisi yang diinginkan, oleh karena itu pengulangan merupakan bagian yang penting dalam pemrograman karena dengan adanya pengulangan pembuat program tidak perlu menulis kode program sebanyak pengulangan yang diinginkan.

Pengulangan mempunyai beberapa bagian yang harus dipenuhi yaitu:

1. Inisialisasi adalah tahap persiapan membuat kondisi awal sel melakukan pengulangan, misalnya mengisi variabel dengan nilai awal. Tahap ini dilakukan sebelum memasuki bagian pengulangan.
2. Proses terjadi di dalam bagian pengulangan dimana berisi semua proses yang perlu dilakukan secara berulang-ulang.
3. Iterasi terjadi di dalam pengulangan di mana merupakan kondisi pertambahan agar pengulangan dapat terus berjalan.
4. Terminasi adalah kondisi berhenti dari pengulangan, kondisi berhenti sangat penting dalam pengulangan agar pengulangan dapat berhenti, tidak menjadi pengulangan yang tanpa henti. Kondisi pengulangan adalah kondisi yang dipenuhi oleh kondisi jalannya algoritma untuk masuk ke dalam blok pengulangan.

Pengulangan merupakan salah satu inti dari analisis kasus pada pembuatan algoritma, sebuah kasus harus dipikirkan penyelesaiannya dengan pemikiran ada proses atau aksi yang harus dikerjakan secara berulang agar sebuah kasus terselesaikan. Struktur kontrol perulangan adalah berupa pernyataan dari Java yang mengijinkan untuk mengeksekusi blok kode secara berulang-ulang dengan jumlah tertentu sesuai yang kita inginkan.

Ada tiga macam jenis dari struktur kontrol perulangan yaitu **for**, **while** dan **do-while**.

### 5.1.1 Perulangan: for....

Struktur Perulangan **for** digunakan saat mengetahui berapa banyak perulangan yang akan dilakukan dan atau melakukan Pengulangan eksekusi code beberapa kali. Pada perulangan for pada umumnya digunakan untuk melakukan perulangan yang banyaknya sudah pasti atau sudah diketahui sebelumnya. Dalam perulangan for kita harus menentukan nilai awal perulangan dan nilai akhir perulangannya.

Perulangan **for** tidak membutuhkan counter untuk menaikkan variabel karena sudah disebutkan pada salah satu parameter perulangan tersebut.

Proses perulangan akan terus dilakukan selama kondisi loop bernilai true. Dengan kata lain proses perulangan hanya akan dihentikan apabila kondisinya telah bernilai false atau sudah tidak terpenuhi lagi. Perulangan for biasanya menggunakan suatu variabel untuk mengendalikan berapa kali tubuh loop akan dieksekusi dan menentukan kapan loop akan berhenti. Variabel ini disebut juga dengan variabel kontrol.

**Bentuk** dari Pengulangan **for**.

```
for (InitExpression; LoopCondition; StepExpression){  
    statement1;  
    statement2;  
    ...  
}
```

#### **Keterangan:**

1. **InitExpression / Inisialisasi:** Inisialisasi dari variabel loop.

Instruksi pemberian suatu nilai yang mempengaruhi nilai kondisi. Pada proses yang normal, pemberian nilai awal ini akan menyebabkan kondisi bernilai TRUE. Instruksi ini hanya pernah satu kali dilaksanakan, yaitu hanya pada saat awal struktur FOR dijalankan dan merupakan variabel kontrol.

2. **LoopCondition / KondisiLoop:** membandingkan variabel lop pada nilai batas. Suatu kondisi yang bernilai TRUE atau FALSE, dan akan membatasi proses perulangan. Blok perintah pada struktur perulangan akan dijalankan selama

kondisi masih bernilai TRUE.

3. **StepExpression / PerubahanNilai:** melakukan update pada variabel loop.

Suatu Instruksi yang dapat mempengaruhi nilai kondisi. Pada proses yang normal, perubahan nilai disini suatu saat akan membuat kondisi bernilai FALSE dan berfungsi menaikkan (increment) nilai variabel kontrol dan kondisi loop mengevaluasi apakah kondisi perulangan bernilai true atau false.

### 5.1.2 Perulangan: while ....

While Loop adalah statement atau blok statement yang diulang-ulang mencapai kondisi yang cocok.

Perulangan WHILE digunakan untuk mengulang suatu proses perulangan yang belum diketahui jumlahnya. Pada perulangan WHILE pengecekan kondisi akan dilakukan terlebih dahulu, jika kondisi bernilai TRUE, maka perulangan akan terus berlanjut dan sebaliknya jika bernilai FALSE maka perulangan akan dihentikan.

**Bentuk dari while,**

```
while( boolean_expression ){  
    statement1;  
    statement2;  
    ...  
}
```

### 5.1.3 Perulangan: do ... while

**Do-while loop** mirip dengan while-loop. Statement di dalam do-while loop akan dieksekusi beberapa kali selama kondisi bernilai true. Perbedaan antara while dan do-while loop adalah statemen di dalam do-while loop dieksekusi sedikit satu kali.

**Bentuk** dari **do-while**.


```
do{
    statement1;
    statement2;
    ...
}while( boolean_expression );
```

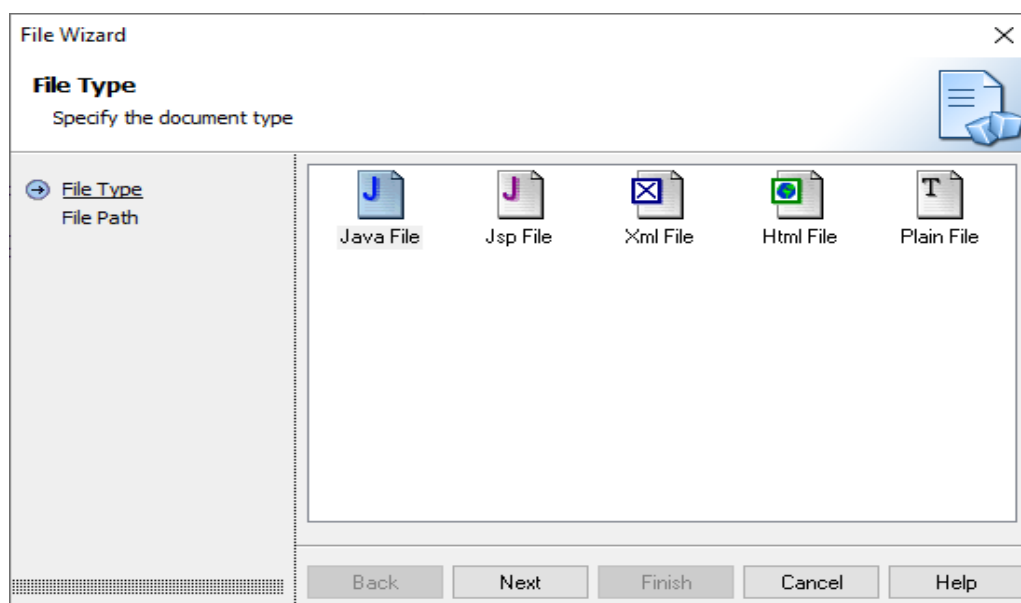
#### 5.1.4 Perintah Break dan Continue

- Break** : Perintah yang digunakan untuk menghentikan kegiatan pengulangan
- Break label** : Dengan menambahkan label pada awal for terluar dapat menjadikan Perintah **break...label** menghentikan semua pengulangan
- continue** : Perintah yang digunakan untuk melewati kegiatan yang ada dibawah Perintah tersebut dan melanjutkan pengulangan.

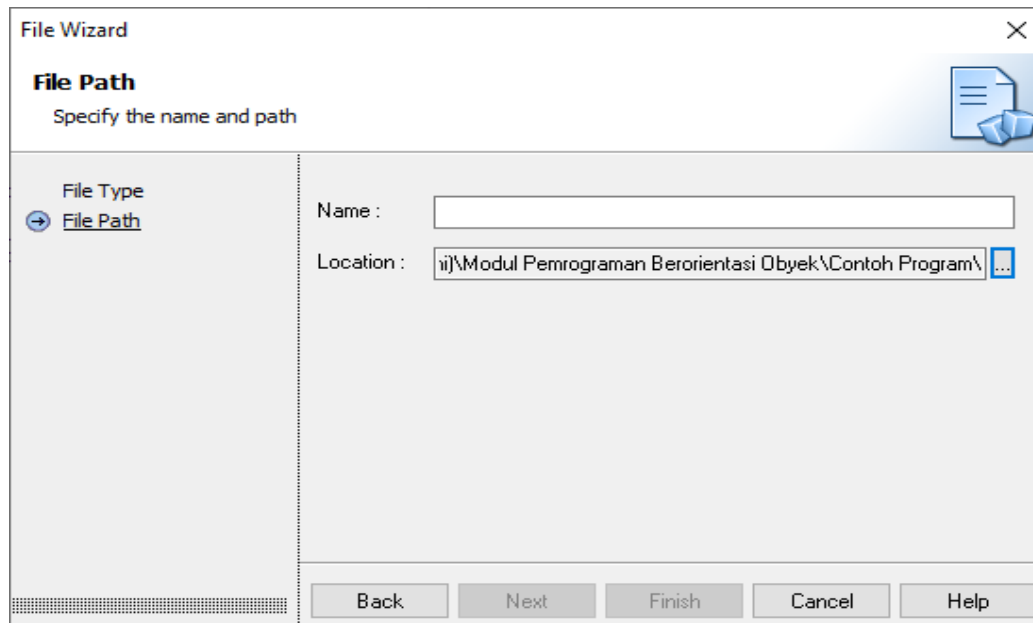
#### 5.1.5 Praktikum

##### Langkah-langkah Praktikum

13. Buka Editor JCreator
14. Buatlah file baru dengan membuka menu File > New > File atau dengan Shortcut Ctrl + N atau dengan klik icon  pada ribbon menu, kemudian pilih > Java File dan Klik tombol Next



15. Buat Nama File sesuai dengan nama file Java masing-masing Contoh program, isikan pada bagian > Name, pilih lokasi tempat penyimpanan file pada pojok kanan Bagian Location kemudian Klik tombol Finish.



16. Lanjutkan dengan menuliskan program pada layar editor JCreator

### Program 5.1: UlangDenganFor.java

Tuliskan program 5.1 berikut pada editor JCreator

```
PengulanganFor.java
1  /*
2   * implementasi struktur pengulangan for....
3   */
4
5  public class PengulanganFor{
6      public static void main(String[] args){
7
8          int H;
9
10         // pengulangan proses dengan struktur pengulangan for
11         for(H=0;H<=5;H++){
12             System.out.println("perulangan ke-: "+H);
13             System.out.println("Saya belajar java");
14             System.out.println("di Labkom UBL");
15             System.out.println(" ");
16         }
17     }
18 }
19
```

Lakukan Kompilasi dan Jalankan program 5.1 diatas dengan membuka menu Build

>Compile File  dan > Execute File 

### Program 5.2: PengulanganFor1.java

Tuliskan program 5.2 berikut pada editor JCreator

```
PengulanganFor1.java |
1  /*
2   * implementasi struktur pengulangan for....
3   */
4
5  public class PengulanganFor1 {
6      public static void main(String args[]){
7          int x;
8          for(x=1;x<10;x++){
9              System.out.println("Nilai X "+x);
10             }
11         }
12     }
13 }
```

Lakukan Kompilasi dan Jalankan program 5.2 diatas dengan membuka menu Build

>Compile File  dan > Execute File 

### Program 5.3: CetakBilanganGanjil.java

Tuliskan program 5.3 berikut pada editor JCreator

```
CetakBilanganGanjil.java * |
1  /*
2   * implementasi struktur pengulangan for.....
3   */
4
5  public class CetakBilanganGanjil{
6      public static void main(String[] argumen){
7          for(int i = 1; i <= 20; i += 2){
8              System.out.print( i + " ");
9          }
10         }
11     }
12 }
13 }
```

Lakukan Kompilasi dan Jalankan program 5.3 diatas dengan membuka menu Build

>Compile File  dan > Execute File 



### Program 5.4: PengulanganWhile.java

Tuliskan program 5.4 berikut pada editor JCreator

```
PengulanganWhile.java
1  /*
2   * implementasi struktur pengulangan while....
3   */
4
5  public class PengulanganWhile{
6      public static void main(String[] args){
7          int H=1;
8
9          // pengulangan proses dengan struktur pengulangan while
10         // minimal proses di jalankan nol kali,
11         // jika variabel dimulai dengan nilai 6
12
13         while (H<=5){
14             System.out.println("Perulangan pakai WHILE ke-:"+H);
15             System.out.println("Saya suka sekali belajar java");
16             System.out.println("di Labkom UBL");
17             System.out.println(" ");
18             H++;
19         }
20     }
21 }
22
23
```

Lakukan Kompilasi dan Jalankan program 5.4 diatas dengan membuka menu Build

>Compile File  dan > Execute File 

### Program 5.5: PengulanganWhile1.java

Tuliskan program 5.5 berikut pada editor JCreator

```
PengulanganWhile1.java *
1  /*
2   * implementasi struktur pengulangan while....
3   */
4
5  public class PengulanganWhile1 {
6      public static void main(String[] args) {
7          int batas = 0;
8          while (batas<10) {
9              System.out.println(batas);
10             batas++;
11         }
12     }
13 }
14
```

Lakukan Kompilasi dan Jalankan program 5.5 diatas dengan membuka menu Build

>Compile File  dan > Execute File 

### Program 5.6: PengulanganDo.java

Tuliskan program 5.6 berikut pada editor JCreator

```
PengulanganDo.java
1  /*
2   * implementasi struktur pengulangan do.....
3   */
4
5  public class PengulanganDo{
6      public static void main(String[] args){
7          int R=1;
8
9          // pengulangan proses dengan struktur pengulangan do ... while
10         // minimal proses di jalankan sekali
11         do {
12             System.out.println("perulangan DO..While ke-:"+R);
13             System.out.println("Saya suka sekali belajar java");
14             System.out.println("di Labkom UBL");
15             System.out.println(" ");
16             R++;
17         } while (R<=5);
18     }
19 }
20 }
```

Lakukan Kompilasi dan Jalankan program 5.6 diatas dengan membuka menu Build

>Compile File  dan > Execute File 

### Program 5.7: PengulanganDo1.java

Tuliskan program 5.7 berikut pada editor JCreator

```
PengulanganDo1.java *
1  /*
2   * implementasi struktur pengulangan do.....
3   */
4
5  public class PengulanganDo1 {
6      public static void main(String[] args) {
7          int batas = 0;
8          do {
9              System.out.println(batas);
10             batas++;
11         }while (batas<10);
12     }
13 }
14 }
```

Lakukan Kompilasi dan Jalankan program 5.7 diatas dengan membuka menu Build

>Compile File  dan > Execute File 

### Program 5.8: PengulanganBreak.java

Tuliskan program 5.8 berikut pada editor JCreator

```
PengulanganBreak.java
1  /*
2   * implementasi struktur pengulangan break.....
3   */
4
5  public class PengulanganBreak{
6      public static void main(String[] args){
7          int S;
8          for (S=0;S<=10;S++){
9              System.out.println("Nilai S:"+ S);
10             if (S>=5){
11                 break;
12             }
13         }
14     }
15 }
16 }
```

Lakukan Kompilasi dan Jalankan program 5.8 diatas dengan membuka menu Build

>Compile File  dan > Execute File 

### Program 5.9: PengulanganBreak1.java

Tuliskan program 5.9 berikut pada editor JCreator

```
PengulanganBreak1.java
1  /*
2   * implementasi struktur pengulangan break.....
3   */
4
5  public class PengulanganBreak1 {
6      public static void main(String[] args){
7
8          for (int i=0;i<10;i++){
9              if (i==4){
10                 break;
11             }
12             System.out.println(i);
13         }
14     }
15 }
16 }
```

Lakukan Kompilasi dan Jalankan program 5.9 diatas dengan membuka menu Build

>Compile File  dan > Execute File 

### Program 5.10: PengulanganBreak2.java

Tuliskan program 5.10 berikut pada editor JCreator

```
PengulanganBreak2.java
1  /*
2   * implementasi struktur pengulangan break.....
3   */
4
5  public class PengulanganBreak2 {
6      public static void main(String[] args){
7
8          outer: for (int i=0;i>10;i++){
9              inner: for (int j=10;i>0;i--){
10                 if (i !=j){
11                     System.out.println(i);
12                     break outer;
13                 }else {
14                     System.out.println("-->" +i);
15                     continue inner;
16                 }
17             }
18         }
19     }
20 }
21
```



Lakukan Kompilasi dan Jalankan program 5.10 diatas dengan membuka menu Build

>Compile File  dan > Execute File 

### Program 5.11: PengulanganContinue.java

Tuliskan program 5.11 berikut pada editor JCreator



```
PengulanganContinue.java
1  /*
2   * implementasi struktur pengulangan continue.....
3   */
4
5  public class PengulanganContinue{
6      public static void main(String[] args){
7          int S;
8          System.out.println("Cetak Bilangan Ganjil: ");
9          for (S=1;S<=10;S++){
10              if (S%2==0){
11                  continue;
12              }
13              System.out.println(S);
14          }
15      }
16  }
17
```

Lakukan Kompilasi dan Jalankan program 5.11 diatas dengan membuka menu Build  
>Compile File  dan > Execute File 

### Program 5.12: PengulanganContinue.java

Tuliskan program 5.12 berikut pada editor JCreator

```
PengulanganContinue1.java |
1  /*
2   * implementasi struktur pengulangan continue.....
3   */
4
5  public class PengulanganContinue1 {
6      public static void main(String[] args){
7
8          for (int i=0;i<10;i++){
9              if (i==4){
10                 continue;
11             }
12             System.out.println(i);
13         }
14     }
15 }
16
```

Lakukan Kompilasi dan Jalankan program 5.12 diatas dengan membuka menu Build  
>Compile File  dan > Execute File 

## 5.1.6 Latihan Mandiri

### ForLoop.java

```
1. class ForLoop {
2.     public static void main(String[] args){
3.         for(int i=1; i<11; i++){
4.             System.out.println("Count is: " + i);
5.         }
6.     }
7. }
```

#### WhileLoop.java

```
1. class WhileLoop {
2.     public static void main(String[] args){
3.         int count = 1;
4.         while (count < 11) {
5.             System.out.println("Count is: " + count); count++;
6.         }
7.     }
8. }
```

#### EnhancedForLoop.java

```
1. class EnhancedForLoop {
2.     public static void main(String[] args){
3.         int[] numbers = {1,2,3,4,5,6,7,8,9,10};
4.         for (int item : numbers) {
5.             System.out.println("Count is: " + item);
6.         }
7.     }
8. }
```

#### DoWhileLoop.java

```
1. class DoWhileLoop {
2.     public static void main(String[] args){
3.         int count = 1;
4.         do {
5.             System.out.println("Count is: " + count); count++;
6.         } while (count < 11);
7.     }
8. }
```

### 5.1.7 Kesimpulan

Pada Pertemuan ini dapat disimpulkan bahwa Mahasiswa mampu dan dapat:

1. Menjelaskan dan menggunakan struktur program Pengulangan dalam Penulisan kode program Pengulangan dalam Penulisan kode program dengan Bahasa Java
2. Menjelaskan dan menggunakan Perintah break dan continue dalam Penulisan kode program dengan Bahasa Java