**BAB IV**

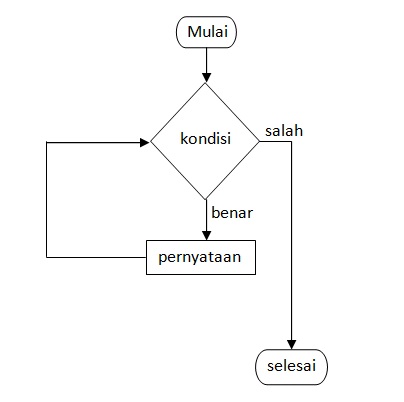
**PERULANGAN**

### **Tujuan**

1. Mengenal dan mengetahui perbedaan penggunaan beberapa *IDE* atau aplikasi pemrograman dan beberapa jenis bahasa pemrograman.
2. Memahami pengertian dan konsep Perulangan.
3. Mengetahui penggunaan Perulangan dengan bahasa pemrograman Phyton, C#, Java, dan PHP.
4. Mengetahui perbedaan *While, Do While, dan For* dalam bahasa pemrograman Phython, C#, Java, dan PHP.
5. Mampu mengimplementasikan Perulangan dalam bahasa pemrograman Phython, C#, Java, dan PHP.

### **Dasar Teori**

* + 1. **While**

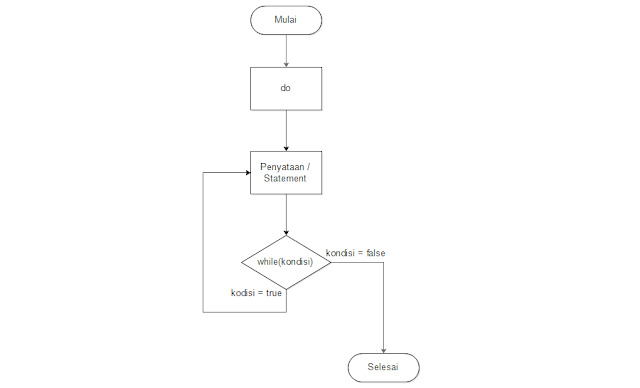
Perulangan *While* dilakukan dengan jumlah yang belum diketahui pasti. Pernyataan pada *While* akan dikerjakan setelah pengecekan kondisi pada *While* yang bernilai benar. Perulangan *While* merupakan perulangan yang termasuk dalam perulangan *uncounted loop*. Tetapi bisa menjadi perulangan *counted loop* dengan memberikan *counter* didalamnya.

Gambar 4. 1. Flowchart perulangan While

(*Sumber: dasarpemrogramangolang.novalagung.com, petanikode.com*)

* + 1. **Do While**

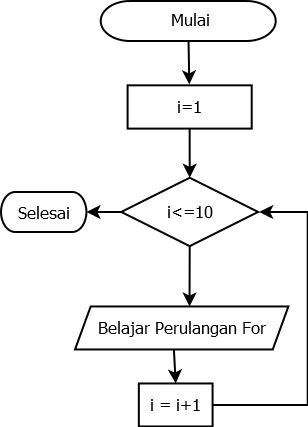
Perulangan ini hampir sama dengan *While* dan dipakai saat melakukan perulangan dengan jumlah yang belum diketahui pasti. Bedanya pernyataan pada *Do* akan dijalankan terlebih dahulu, setelah itu dilakukan pengecekan kondisinya pada *While*.



Gambar 4. 2. Flowchart perulangan Do While

*(Sumber: dasarpemrogramangolang.novalagung.com)*

* + 1. **For**

Perulangan ini dipakai pada saat melakukan perulangan dengan jumlah yang sudah diketahui pasti. *For* merupakan perulangan yang termasuk kedalam *couted loop*, karena sudah berapa kali perulangannya dilakukan.

Gambar 4. 3. Flowchart perulangan For

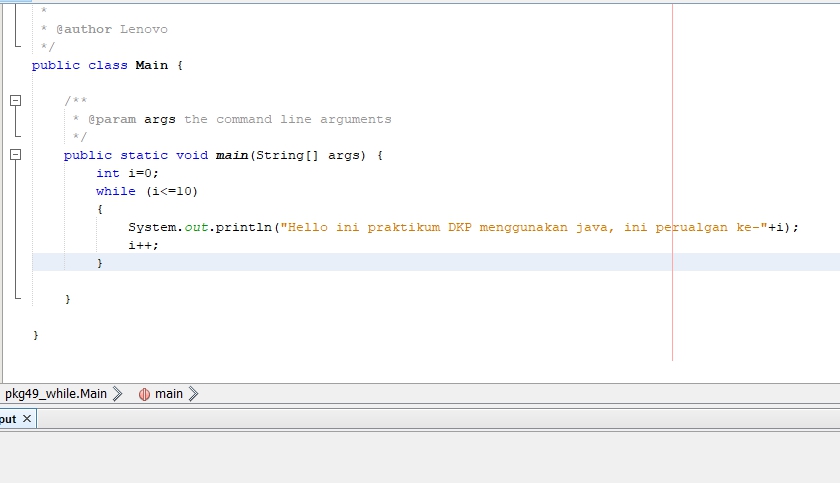
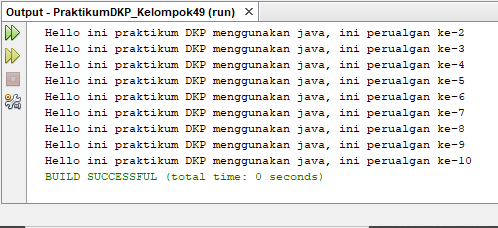
*(Sumber: petanikode.com)*

### **Hasil Percobaan dan Analisis**

* + 1. **Perulangan dalam Java**

1. **While**

|  |
| --- |
| int i=0;  while (i<=10)  {  System.out.println("Hello ini praktikum DKP menggunakan java, ini perualgan ke-"+i);  i++;  } |



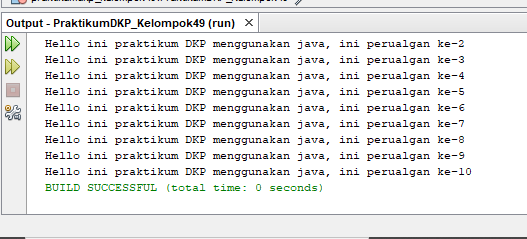
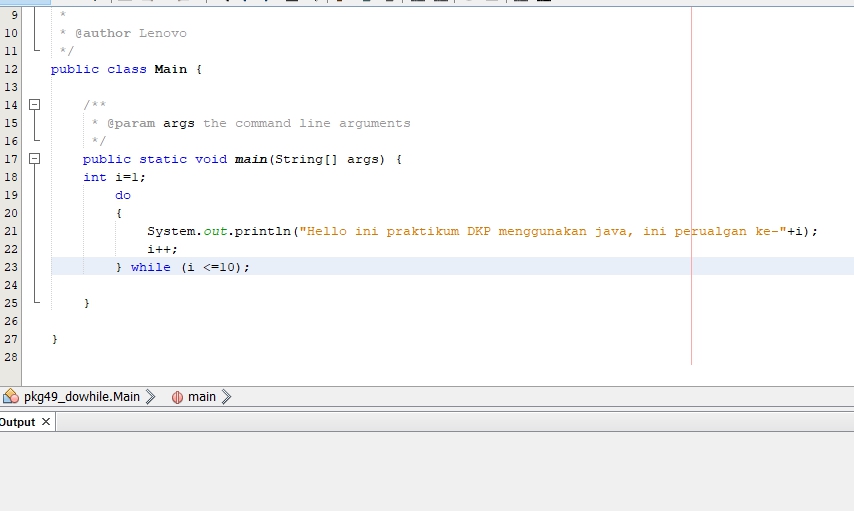
Gambar 4. 5. Output perulangan While dalam Java

Gambar 4. 4. Source Code perulangan While dalam Java

Pertama, deklarasi i terlebih dahulu, setelah itu masukkan *command while* dan beri syarat. Karena i memenuhi syarat di *while* maka *command* didalam *while* akan terus dijalankan, *Command* di dalam *while* akan terus berulang sampai i tidak memenuhi syarat.

1. **Do While**

|  |
| --- |
| int i=1;  do  {  System.out.println("Hello ini praktikum DKP menggunakan java, ini perualgan ke-"+i);  i++;  } while (i <=10); |

****

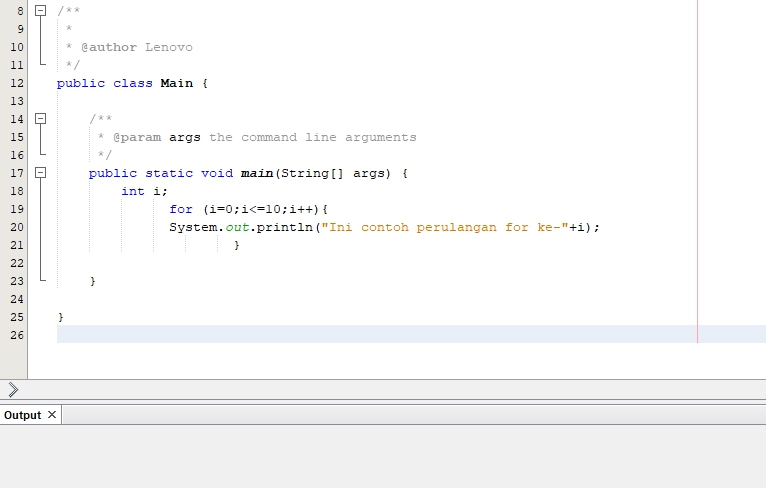
Gambar 4. 7. Output perulangan Do While dalam Java

Gambar 4. 6. Source Code perulangan Do While dalam Java

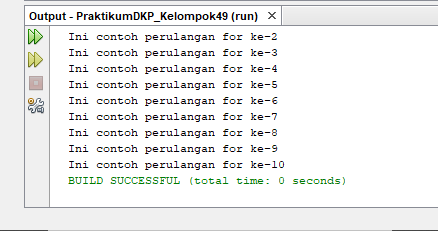
Pertama, deklarasi i terlebih dahulu, setelah itu masukkan *command Do while* dan beri syarat. *Command* di dalam *do while* akan dijalankan dan selanjutnya i akan di cek, karena i memenuhi syarat maka *command* didalam *do while* akan terus dijalankan. *Command* di dalam *do while* akan terus berulang sampai i tidak memenuhi syarat.

1. **For**

|  |
| --- |
| int i;  for (i=0;i<=10;i++){  System.out.println("Ini contoh perulangan for ke-"+i);  } |



Gambar 4. 8. Source Code perulangan For dalam Java

****

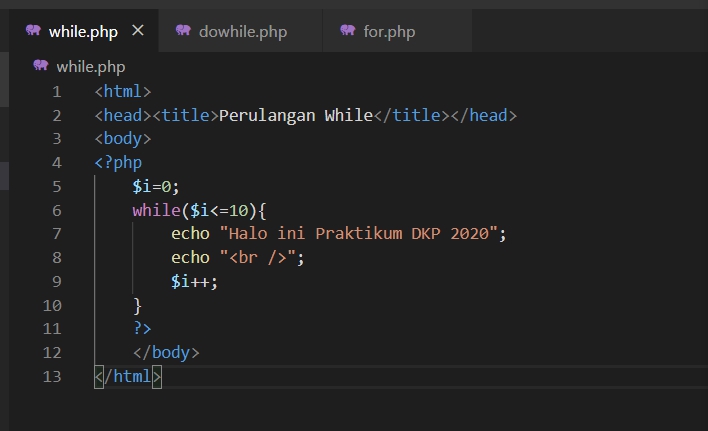
Gambar 4. 9. Output perulangan For dalam Java

Pertama, deklarasi bahwa I adalah *interger* terlebih dahulu, setelah itu masukkan *command* *for* dan beri syarat. *Command* didalam *for* akan terus berulang sampai i telah melewati batas maksimal, dengan perubahan jumlah angka yang konstan tiap perulangannya.

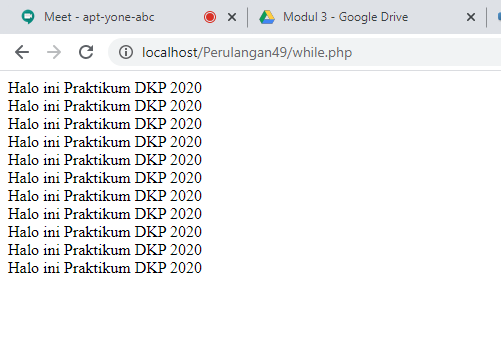
* + 1. **Perulangan dalam PHP**

1. **While**

|  |
| --- |
| <html>  <head><title>Perulangan While</title></head>  <body>  <?php  $i=0;  while($i<=10){  echo "Halo ini Praktikum DKP 2020";  echo "<br />";  $i++;  }  ?>  </body>  </html> |



Gambar 4. 10. Source Code perulangan While dalam PHP

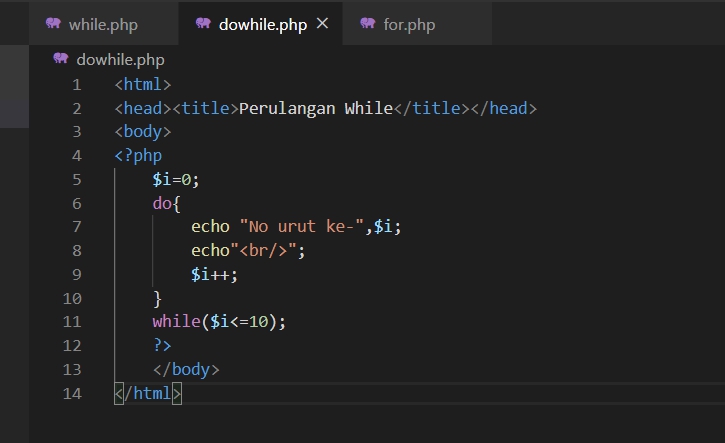
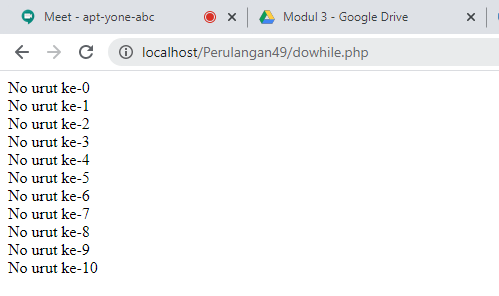
****

Gambar 4. 11. Output perulangan While dalam PHP

Pertama, deklarasi i terlebih dahulu, setelah itu masukkan *command while* dan beri syarat. Karena i memenuhi syarat di *while* maka *command* didalam *while* akan terus dijalankan,*Command* di dalam *while* akan terus berulang sampai i tidak memenuhi syarat.

1. **Do While**

|  |
| --- |
| <html>  <head><title>Perulangan While</title></head>  <body>  <?php  $i=0;  do{  echo "No urut ke-",$i;  echo"<br/>";  $i++;  }  while($i<=10);  ?>  </body>  </html> |

****

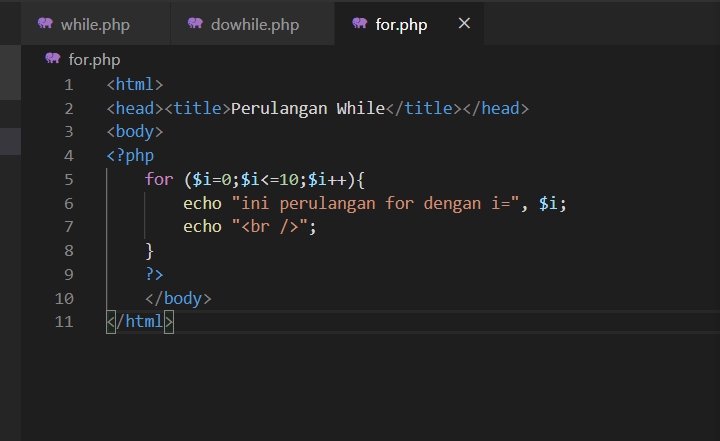
Gambar 4. 13. Output perulangan Do While dalam PHP

Gambar 4. 12. Source Code perulangan Do While dalam PHP

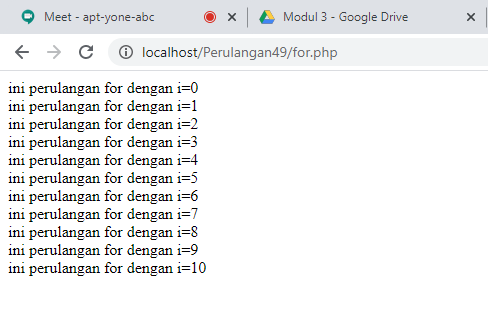
Pertama, deklarasi i terlebih dahulu, setelah itu masukkan *command Do while* dan beri syarat.*Command* di dalam *do while* akan dijalankan dan selanjutnya i akan di cek ,karena i memenuhi syarat maka *command* didalam *do while* akan terus dijalankan.*Command* di dalam *do while* akan terus berulang sampai i tidak memenuhi syarat.

1. **For**

|  |
| --- |
| <html>  <head><title>Perulangan While</title></head>  <body>  <?php  for ($i=0;$i<=10;$i++){  echo "ini perulangan for dengan i=", $i;  echo "<br />";  }  ?>  </body>  </html> |



Gambar 4. 14. Source Code perulangan For dalam PHP

****

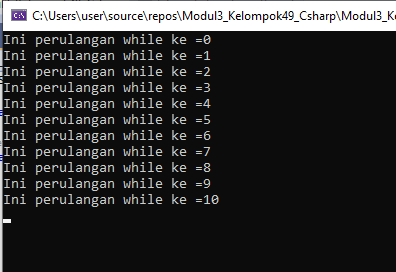
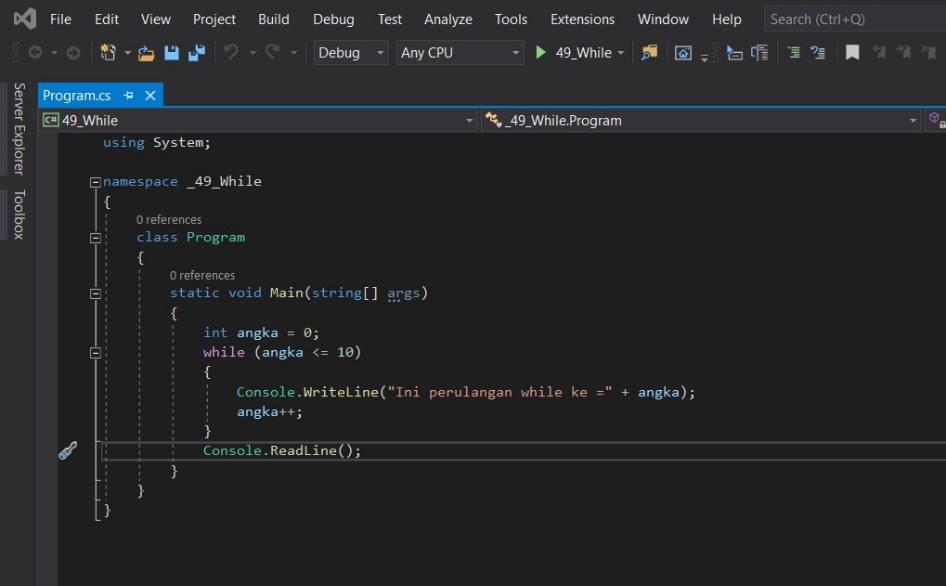
Gambar 4. 15. Output perulangan For dalam PHP

Pertama, deklarasi bahwa I adalah *interger* terlebih dahulu, setelah itu masukkan *command* *for* dan beri syarat. *Command* didalam *for* akan terus berulang sampai i telah melewati batas maksimal, dengan perubahan jumlah angka yang konstan tiap perulangannya.

* + 1. **Perulangan dalam C#**

1. **While**

|  |
| --- |
| int angka = 0;  while (angka <= 10)  {  Console.WriteLine("Ini perulangan while ke =" + angka);  angka++;  }  Console.ReadLine(); |

****

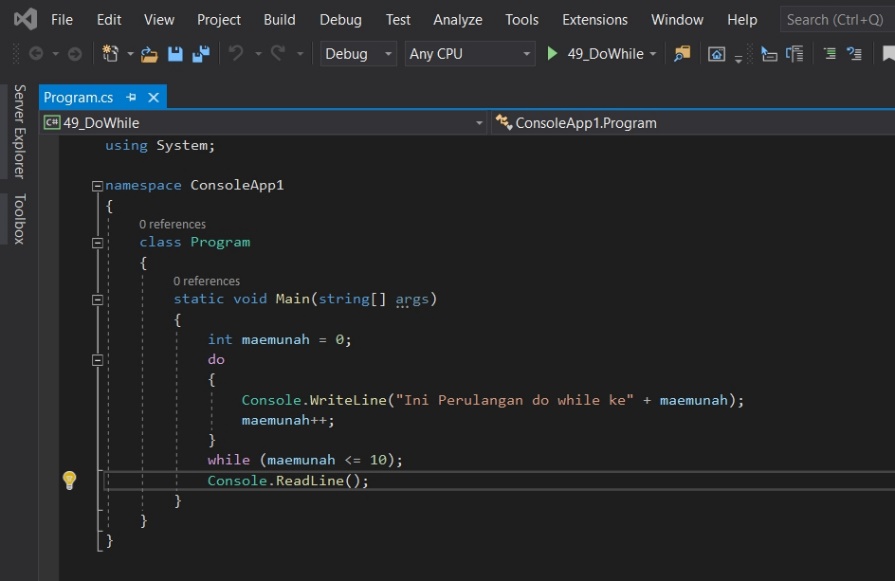
Gambar 4. 17. Output perulangan While dalam C#

Gambar 4. 16. Source Code perulangan While dalam C#

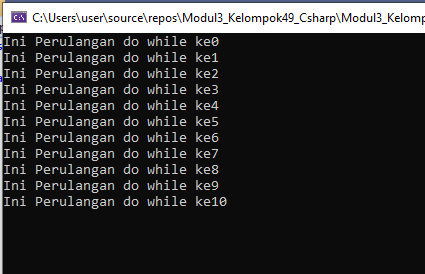
Pertama, deklarasi i terlebih dahulu, setelah itu masukkan *command while* dan beri syarat. Karena i memenuhi syarat di *while* maka *command* didalam *while* akan terus dijalankan,*Command* di dalam *while* akan terus berulang sampai i tidak memenuhi syarat.

1. **Do While**

|  |
| --- |
| int maemunah = 0;  do  {  Console.WriteLine("Ini Perulangan do while ke" + maemunah);  maemunah++;  }  while (maemunah <= 10);  Console.ReadLine(); |

****

Gambar 4. 18. Source Code perulangan Do While dalam C#

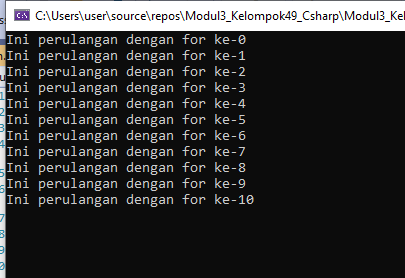
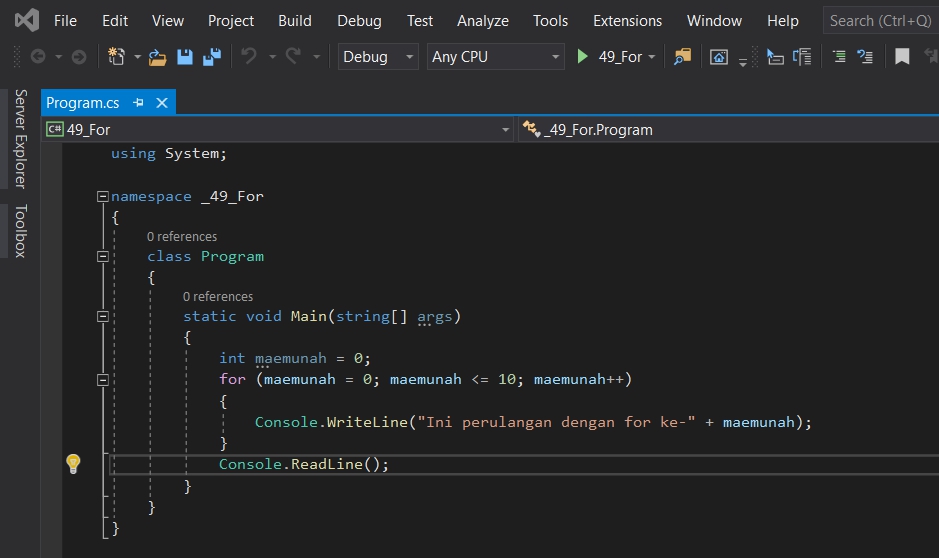
****

Gambar 4. 19. Output perulangan Do While dalam C#

Pertama, deklarasi i terlebih dahulu, setelah itu masukkan *command Do while* dan beri syarat.*Command* di dalam *do while* akan dijalankan dan selanjutnya i akan di cek ,karena i memenuhi syarat maka *command* didalam *do while* akan terus dijalankan.*Command* di dalam *do while* akan terus berulang sampai i tidak memenuhi syarat.

1. **For**

|  |
| --- |
| int maemunah = 0;  for (maemunah = 0; maemunah <= 10; maemunah++)  {  Console.WriteLine("Ini perulangan dengan for ke-" + maemunah);  }  Console.ReadLine(); |

****

Gambar 4. 21. Output perulangan For dalam C#

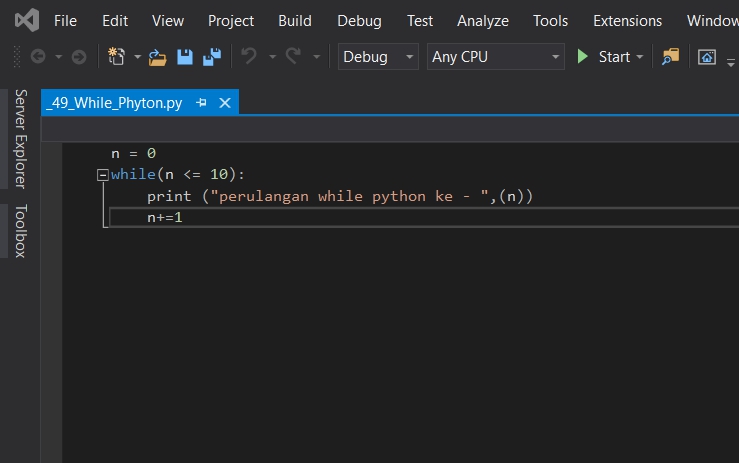
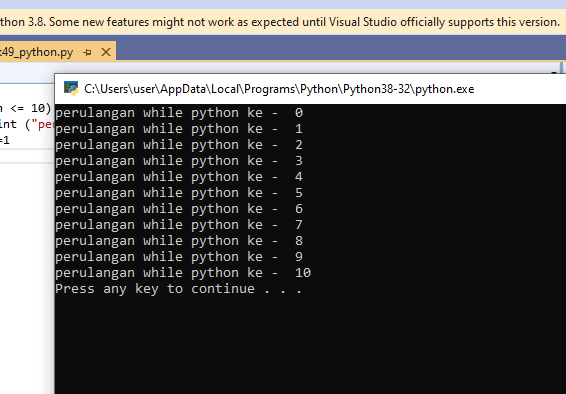
Gambar 4. 20. Source Code perulangan For dalam C#

Pertama, deklarasi bahwa I adalah *interger* terlebih dahulu, setelah itu masukkan *command* *for* dan beri syarat. *Command* didalam *for* akan terus berulang sampai i telah melewati batas maksimal, dengan perubahan jumlah angka yang konstan tiap perulangannya.

* + 1. **Perulangan dalam Phyton**

1. **While**

|  |
| --- |
| n = 0  while(n <= 10):  print ("perulangan while python ke - ",(n))  n+=1 |



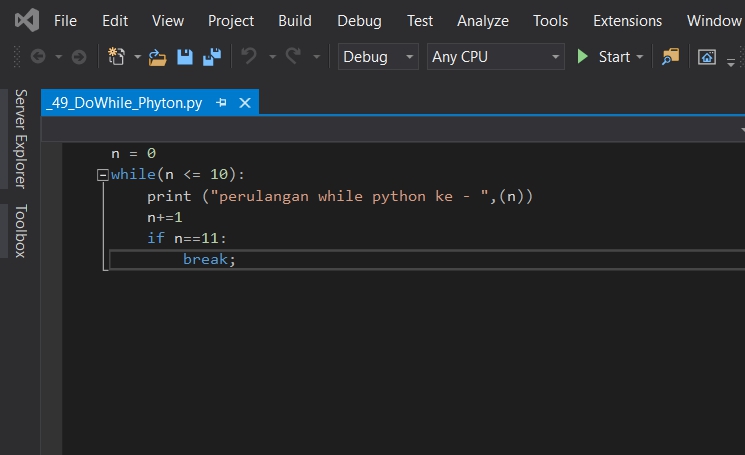
Gambar 4. 23. Output perulangan While dalam Phyton

Gambar 4. 22. Source Code perulangan While dalam Phyton

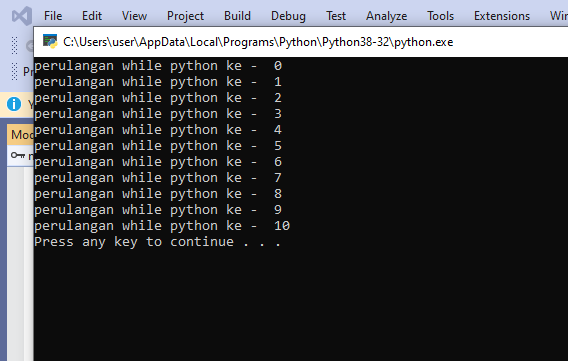
Pertama, deklarasi i terlebih dahulu, setelah itu masukkan *command while* dan beri syarat. Karena i memenuhi syarat di *while* maka *command* didalam *while* akan terus dijalankan,*Command* di dalam *while* akan terus berulang sampai i tidak memenuhi syarat.

1. **Do While**

|  |
| --- |
| n = 0  while(n <= 10):  print ("perulangan while python ke - ",(n))  n+=1  if n==11:  break; |



Gambar 4. 24. Source Code perulangan Do While dalam Phyton

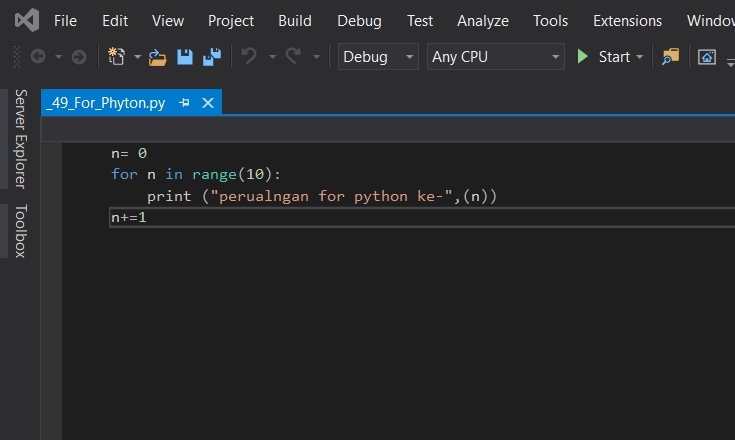


Gambar 4. 25. Output perulangan Do While dalam Phyton

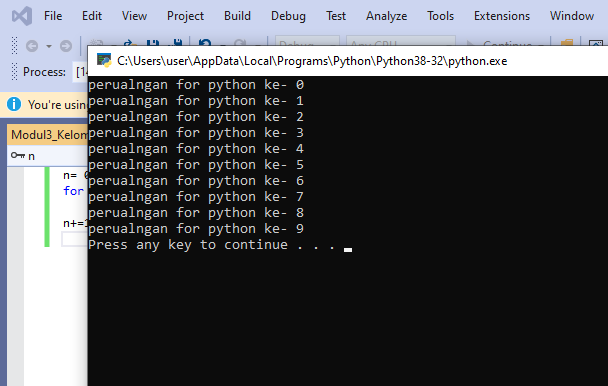
Sebenarnya di bahasa pemrograman python tidak ada yang namanya do while. Maka dari itu yang digunakan adalah *command while* diawal dan *if* diakhir.Pertama, deklarasi i terlebih dahulu, setelah itu masukkan *command while* dan *if* dan beri syaratnya . Karena i memenuhi syarat di *while* maka *command* didalam *while* akan terus dijalankan,dan selanjutnya i di cek di *if* dan masih memenuhi maka *command* didalam *while* akan diulang kembali. *Command* di dalam *while* akan terus berulang sampai i tidak memenuhi syarat *while* dan *if*.

1. **For**

|  |
| --- |
| int maemunah = 0;  for (maemunah = 0; maemunah <= 10; maemunah++)  {  Console.WriteLine("Ini perulangan dengan for ke-" + maemunah);  }  Console.ReadLine(); |



Gambar 4. 26. Source Code perulangan For dalam Phyton



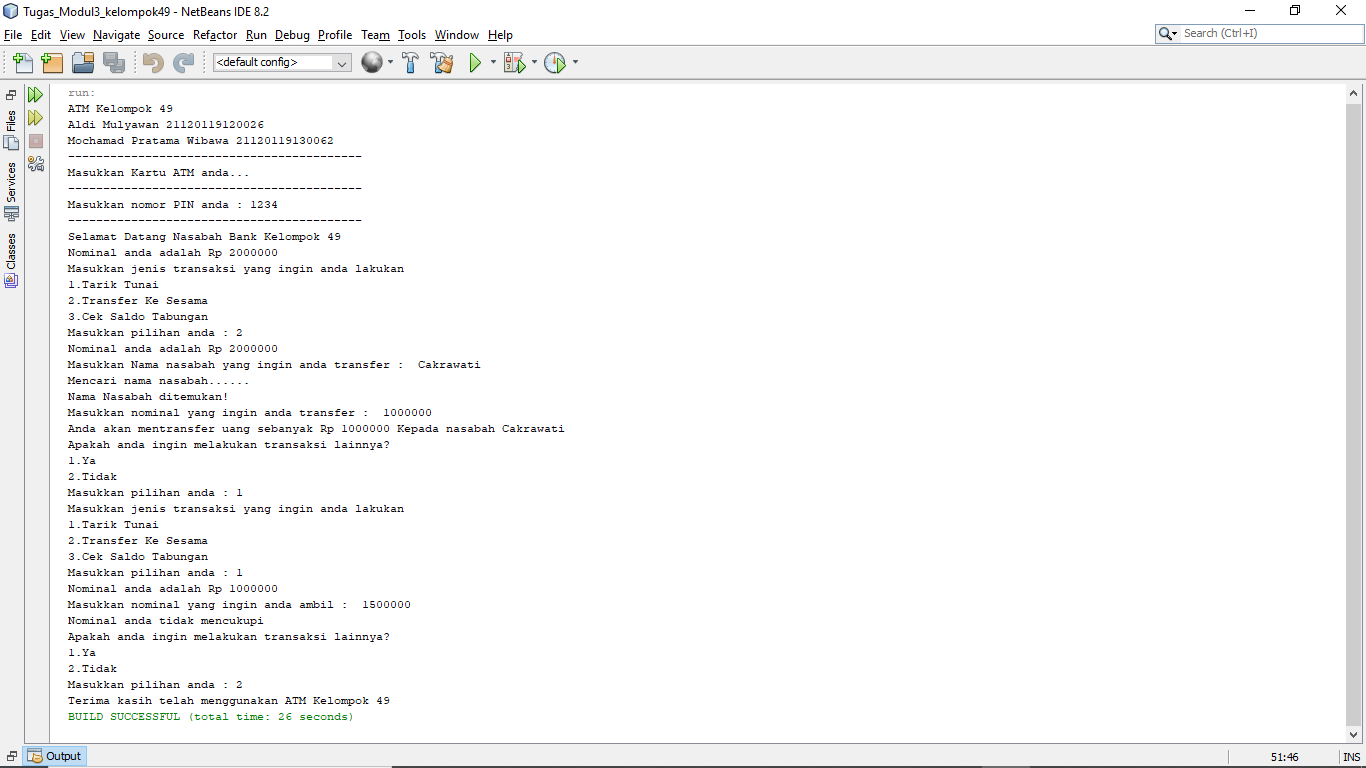
Gambar 4. 27. Output perulangan For dalam Phyton

Pertama, deklarasi bahwa I adalah *interger* terlebih dahulu, setelah itu masukkan *command* *for* dan beri syarat. *Command* didalam *for* akan terus berulang sampai i telah melewati batas maksimal, dengan perubahan jumlah angka yang konstan tiap perulangannya.

### **Tugas**

|  |
| --- |
| package tugas\_modul4\_kelompok49;  import java.util.Scanner;  import java.lang.Math;  public class Tugas\_Modul3\_Kelompok49 {  public static void main(String[] args) {  Scanner input = new Scanner( System.in );  Scanner in = new Scanner (System.in);  String[] pilihan\_fitur = {"1.Tarik Tunai", "2.Transfer Ke Sesama","3.Cek Saldo Tabungan"};  System.out.println("ATM Kelompok 49");  System.out.println("Aldi Mulyawan 21120119120026");  System.out.println("Mochamad Pratama Wibawa 21120119130062");  System.out.println("------------------------------------------");  System.out.println("Masukkan Kartu ATM anda...");  System.out.println("------------------------------------------");  System.out.print("Masukkan nomor PIN anda : ");  int zzzz = input.nextInt();  System.out.println("------------------------------------------");  int a = 2000000;  int d = 0;  System.out.println("Selamat Datang Nasabah Bank Kelompok 49");  System.out.println("Nominal anda adalah Rp " + a);  do  {  System.out.println("Masukkan jenis transaksi yang ingin anda lakukan");  for (String x : pilihan\_fitur){  System.out.println(x);  }  System.out.print("Masukkan pilihan anda : ");  int b = input.nextInt();  switch (b) {  case 1:  System.out.println("Nominal anda adalah Rp " + a);  System.out.print("Masukkan nominal yang ingin anda ambil : ");  int c = input.nextInt();  if (a>=c){  a=a-c;  System.out.println("Anda akan menarik uang sebanyak " + c);  break;  }  else  System.out.println("Nominal anda tidak mencukupi");  break;  case 2:  System.out.println("Nominal anda adalah Rp " + a);  System.out.print("Masukkan Nama nasabah yang ingin anda transfer : ");  String nama = in.nextLine();  System.out.println("Mencari nama nasabah......");  System.out.println("Nama Nasabah ditemukan!");  System.out.print("Masukkan nominal yang ingin anda transfer : ");  int f = input.nextInt();  if (a>=f){  a=a-f;  System.out.println("Anda akan mentransfer uang sebanyak Rp " + f + " Kepada nasabah " + nama);  break;  }  else  System.out.println("Nominal anda tidak mencukupi");  break;  case 3:  System.out.println("Nominal anda adalah Rp " + a);  break;  default:  System.out.println("Perintah tidak ditemukan");  break;  }  System.out.println("Apakah anda ingin melakukan transaksi lainnya?");  System.out.println("1.Ya");  System.out.println("2.Tidak");  System.out.print("Masukkan pilihan anda : ");  int e = input.nextInt();  d=d-d+e;    }while (d<2);  System.out.println("Terima kasih telah menggunakan ATM Kelompok 49");  }  } |

*Output* Hasil:



Gambar 4. 28. Output Tugas Modul 3 dalam Java

Penjelasan *Output* :

Pertama, kita masukkan *command* supaya kita dapat melakukan inputan. Selanjutnya, kita buat array tentang daftar fitur yang ada yaitu tarik tunai, transfer dan cek saldo. Selanjutnya, kita print perkenalan dan perintah untuk memasukkan PIN serta kita masukkan *command* supaya pengguna dapat memasukkan PIN-nya. Selanjutnya, kita definisikan jumlah uang yang dimiliki nasabah, dan kita keluarkan tulisan selamat datang serta keterangan jumlah tabungan nasabah. Selanjutnya, kita gunakan *Do While* untuk perintah selanjutnya supaya nasabah melakukan transaksi terlebih dahulu dan nasabah memilih untuk melakukan transaksi lagi atau tidak berada diakhir dan keputusan akan diulang atau tidak berada diakhir sehingga kita menggunakan *Do While*. Selanjutnya, keluarkan daftar fitur yang ada dengan memanggil array yang telah kita buat sebelumnya, dan beri inputan supaya nasabah dapat memilih fitur yang kita inginkan. Tiap fitur kita pisahkan dengan *switch*. Pada *switch* pertama kita melakukan transaksi tarik tunai, kita menunjukkan jumlah tabungan nasabah lalu memberikan inputan berapa uang yang diinginkan nasabah dan mengurangi uang yang dimiliki nasabah sekarang serta memberikan pemberitahuan bahwa nasabah akan menarik uang sejumlah uang yang diinputkan sebelumnya. Pada *switch* yang kedua kita sediakan fitur transfer uang, nasabah menginputkan nama dan nominal yang ingin di transfer dan selanjutnya tabungan nasabah akan berkurang sesuai dengan jumlah uang yang ia transfer serta kita keluarkan pemberitahuan bahwa nasabah akan mentransfer uang kepada tujuan serta nominal yang telah diinputkan sebelumnya. Pada *switch* yang ketiga kita sediakan fitur cek saldo, pada fitur ini kita hanya memanggil variabel “a” dan mengeluarkannya untuk memberitahu jumlah uang yang dimiliki oleh nasabah. Jika salah satu *switch* telah selesai maka akan ditanyakan apakah nasabah ingin melakukan transaksi lagi, jika nasabah menjawab iya maka *Do while* akan diulang dari pemilihan fitur yang diinginkan. Jika tidak maka akan keluar tulisan terima kasih.

Link Video: https://drive.google.com/open?id=1dZKwM\_DZrHmY5ZsAqsDoMFaepMXolju

### **Kesimpulan**

1. Perulangan atau *Looping* adalah cara atau struktur pemrograman yang memproses atau mengulang-ulang eksekusi blok kode tanpa henti, hingga kondisi yang dijadikan acuan terpenuhi.
2. Perulangan atau *Looping* dibagi menjadi beberapa jenis antara lain *While, Do While, dan For.*
3. Terdapat beberapa perbedaan antara perulangan satu dan lainnya, tergantung tujuan penggunaannya.
4. Bahasa pemrograman yang keluar dalam waktu dekat, memiliki kemampuan yang lebih cepat dan *simple* dari pendahulunya. Seperti phyton lebih *simple* dari C#.
5. Pemograman mempunyai tiga inti dasar yaitu input, proses, dan output.
6. Antara bahasa pemrograman satu dengan yang lainnya kadang terdapat perbedaan-perbendaan tertentu, karena memiliki aturannya masing-masing.
7. Bahasa pemrogaman tidak dapat dibanding-bandingkan, setiap bahasa pemograman memiliki kelebihan dan bisa dipakai sesuai kebutuhannya masing-masing.
8. Diperlukan ketelitian dalam mebuat sebuah program, karena kesalahan sedikit akan membuat program tidak berjalan dengan baik.