**BAB IV**

**PERULANGAN**

1. **Tujuan**
   1. Praktikan dapat memahami konsep dasar perulangan.
   2. Praktikan dapat memahami pernyataan perulangan WHILE.
   3. Praktikan dapat memahami pernyataan perulangan DO...WHILE.
   4. Praktikan dapat memahami pernyataan perulangan FOR.
2. **Dasar Teori**
   1. **Konsep dasar perulangan.**

Perulangan adalah salah satu bentuk pemrograman untuk menangani satu langkah yang berulang. Contoh penggunaan perulangan adalah, kita bisa membuat tulisan “Hello World” hingga ratusan bahkan ribuan kali. Kita bisa saja melakukan *Copy-Paste*, namun hal tersebut sangat tidak praktis.

Pernyataan dari sebuah perulangan akan terus dilakukan selama kondisi nilainya benar. Bila kondisi bernilai salah maka perulangan akan dihentikan.

Terdapat dua jenis perulangan yaitu *counted loop* dan *uncounted loop*. *Counted loop* merupakan perulangan yang jumlah pengulangannya terhitung atau tentu seperti perulangan FOR. *Uncounted loop* merupakan perulangan yang jumlah pengulangannya tidak terhitung atau tidak tentu seperti perulangan WHILE dan DO WHILE.

Ada beberapa model perulangan dalam pemrograman, di antaranya ada WHILE, DO...WHILE, dan FOR. Ketiga model perulangan tersebut sangat populer dalam mempelajari pemrograman tingkat dasar. Dalam praktikum modul IV ini, kita akan mempelajari ketiga model perulangan tersebut.

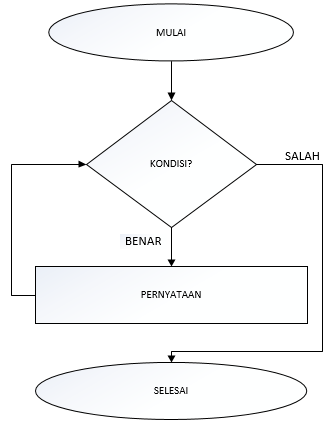
* 1. **Perulangan dengan WHILE.**

Perulangan While merupakan sebuah perulangan yang kondisi berhenti sudah di ketahui tetapi banyaknya proses perulangan belum di ketahui. Bentuk sederhana dari perulangan WHILE adalah sebagai berikut

|  |
| --- |
| while (kondisi)  pernyataan |

Pada pernyataan while diatas, kondisi diuji terlebih dahulu. Jika kondisinya bernilai benar, maka pernyataan akan dieksekusi dan kemudian kondisi diuji kembali hingga kondisi bernilai salah .

Pernyataan bisa juga berupa blok yang berisi kumpulan perintah-perintah di antara { dan }. perintah ini disebut juga dengan inti perulangan. Inti perulangan akan terus dieksekusi selama suatu\_kondisi bernilai true. suatu\_kondisi ini disebut juga penguji perulangan.

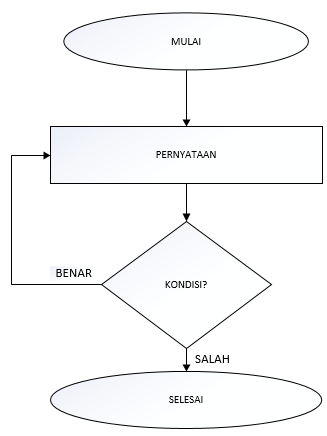


Gambar 4. 1 *Flowchart* perulangan WHILE

* 1. **Perulangan DO...WHILE.**

Kadang-kadang akan lebih mudah menulis perulangan jika penguji perulangan dilakukan di akhir badan perulangan. Dalam hal ini badan perulangan akan dieksekusi terlebih dahulu tanpa memperdulikan apakah suatu kondisi bernilai true atau false. Pengujian dilakukan di akhir setelah suatu kondisi didapat dalam eksekusi perulangan pertama kali. Pernyataan dengan DO...WHILE pada dasarnya merupakan pernyataan while terbalik, dengan bentuk.

|  |
| --- |
| do  pernyataan  while (kondisi) |



Gambar 4. 2 *Flowchart* perulangan DO... WHILE

* 1. **Perulangan FOR**

Perulangan FOR merupakan perulangan yang dilakukan ketika kondisi kapan perulangan dihentikan dan banyaknya proses perulangan sudah di ketahui. Dalam penggunaan perulangan dengan FOR, ditentukan oleh tiga bagian yang masing-masing dipisahkan oleh titik koma, yaitu :

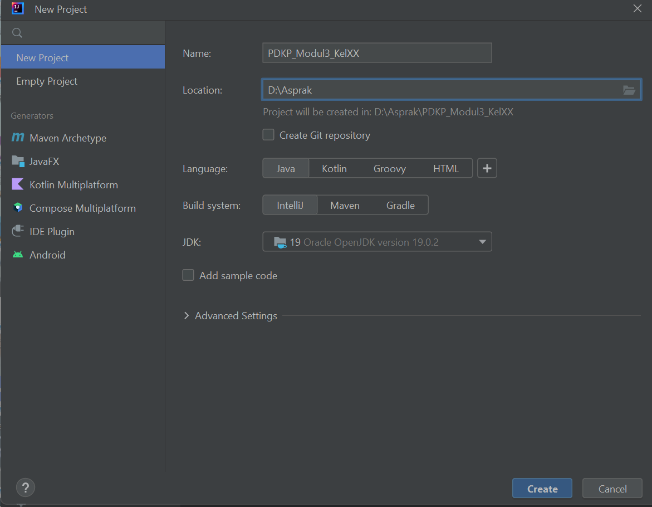
* + - 1. Inisialiasi
      2. Penghenti
      3. *Modifier(Increament/Decreament)*

Bentuk sederhana perulangan for adalah sebagai berikut :

|  |
| --- |
| for (inisialisasi; penghenti; modifier)  pernyataan |

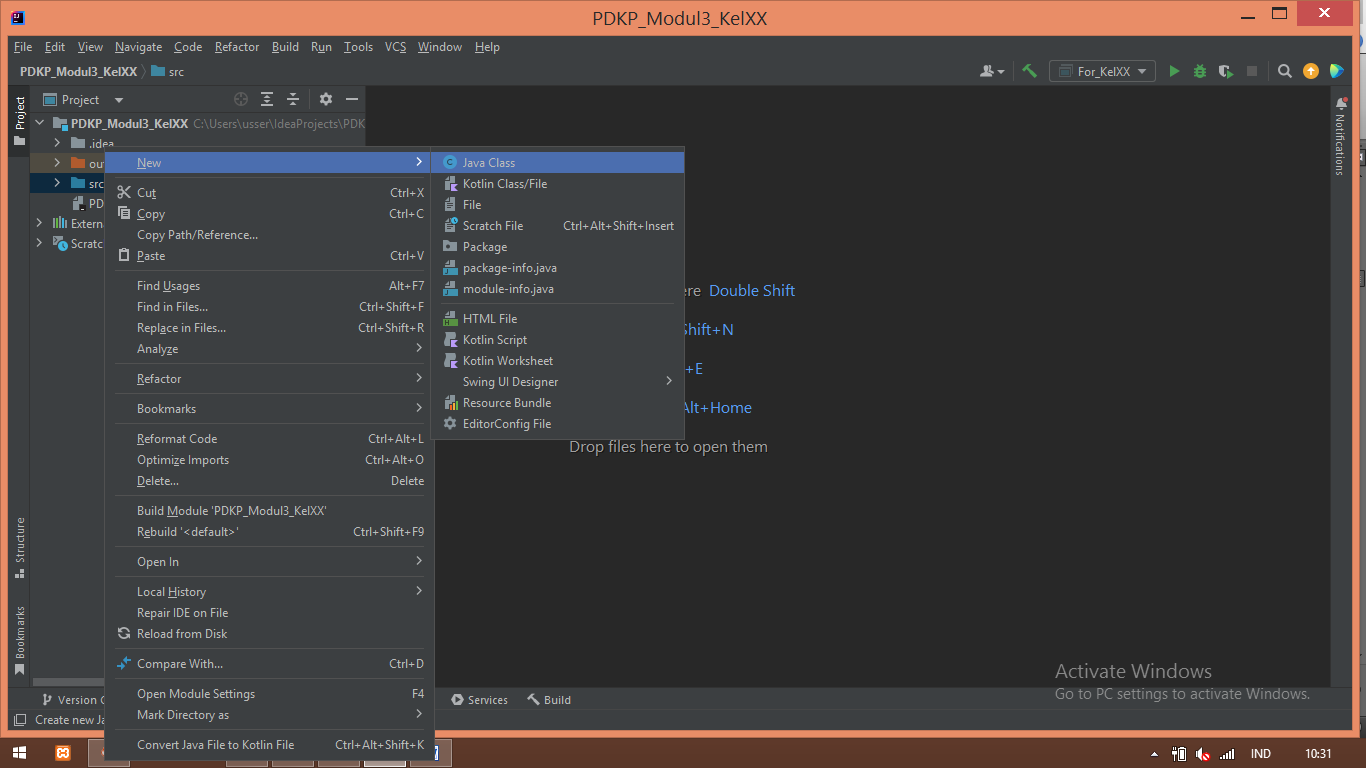
Bagian inisialisasi dijalankan sekali sebelum pernyataan dijalankan. Penghenti selalu dievaluasi terlebih dahulu sebelum pernyataan dieksekusi. Bila penghenti bernilai salah, maka eksekusi berhenti. *Modifier* dijalankan otomatis setelah pernyataan dieksekusi.

1. **Percobaan**
   * 1. **Perulangan pada JAVA**
2. Membuat *project* baru
3. Buka Intellij IDEA Community
4. Kemudian pilih *new project* (*file* -> *new project*)
5. Pilih java dan Intellij kemudian beri nama *project* dan klik *create*.



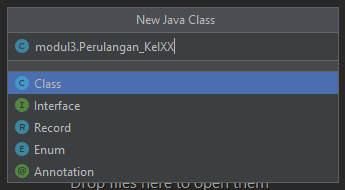
Gambar 4. 3 Memilih jenis *project*

1. Setelah itu, pada bagian *project* (sebelah kiri) klik kanan pada folder src, kemudian *hover* pilihan *new*, kemudian pilih Java Class

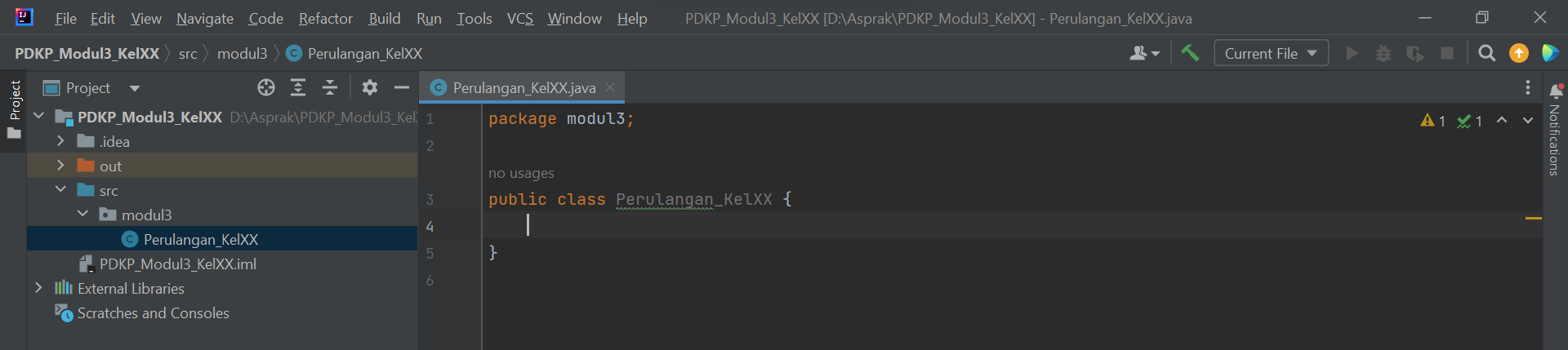


Gambar 4. 5 Membuat Java *Class* baru

1. Berikan nama modul3.Perulangan\_KelXX kemudian *enter*

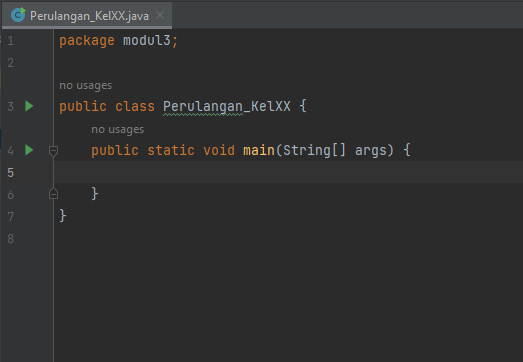


Gambar 4. 6 Membuat *package* berisi *Class* Perulangan\_KelXX



Gambar 4. 7 Tampilan ketika *Class* berhasil dibuat

1. Dalam public class, ketik main atau psvm kemudian pilih dropdown yang muncul. (Jika tidak muncul drop down, ketik saja public static void main(String[] args){})



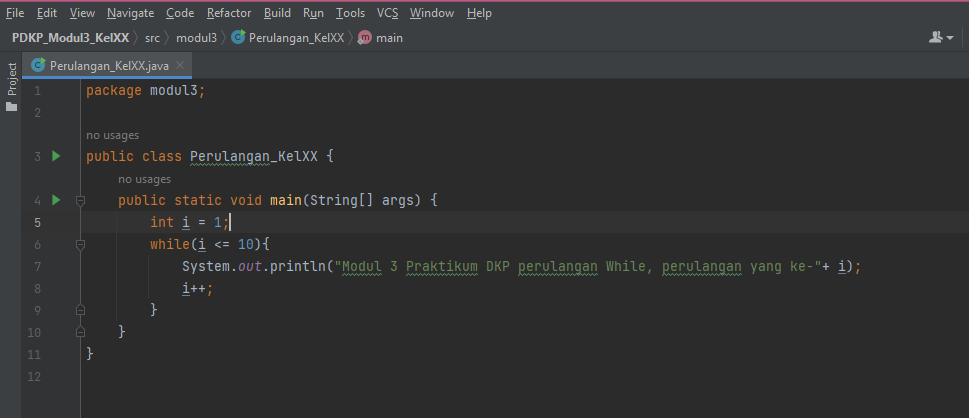
Gambar 4. 9 *Class* yang siap diprogram

1. Program sudah siap dibuat!
2. While

Berikut ini adalah beberapa contoh perulangan dengan WHILE dalam bahasa pemrograman JAVA :

Masukan *sourcecode* yang berada di dalam Main berikut ini pada *Class* yang telah dibuat :

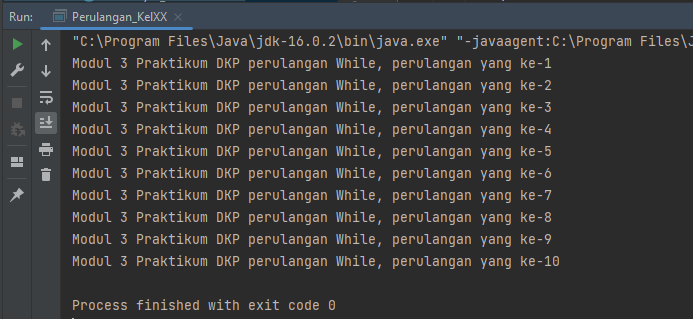
|  |
| --- |
| package modul3;  public class Perulangan\_KelXX {  public static void main(String[] args) {  int i = 1;  while(i <= 10){  System.out.println("Modul 3 Praktikum DKP perulangan While, perulangan yang ke-"+ i);  i++;  }  }  } |

Gambar 4. 10 Program WHILE pada Java

Setelah itu, pilih menu *run* kemudian pilih *run* (bentuk segitiga), tunggu sampai keluar hasilnya dan amati.



Gambar 4. 11 menjalankan program



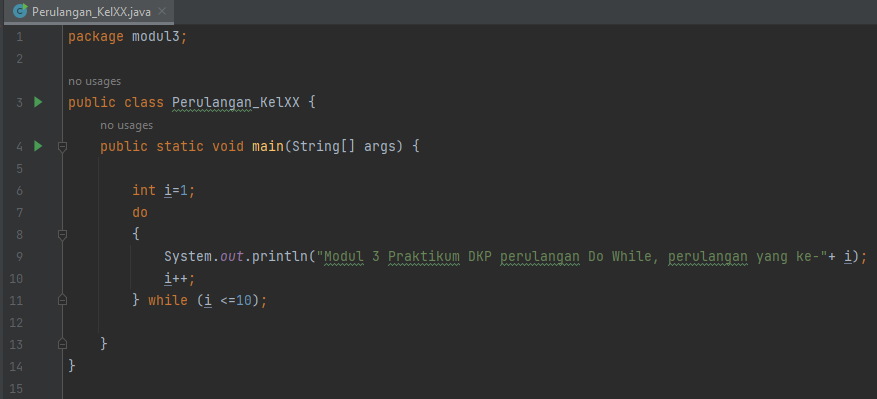
Gambar 4. 12 Hasil keluaran program

**INGAT** : Java adalah *Case Sensitive*. Artinya, kode dengan huruf besar dan huruf kecil dianggap berbeda.

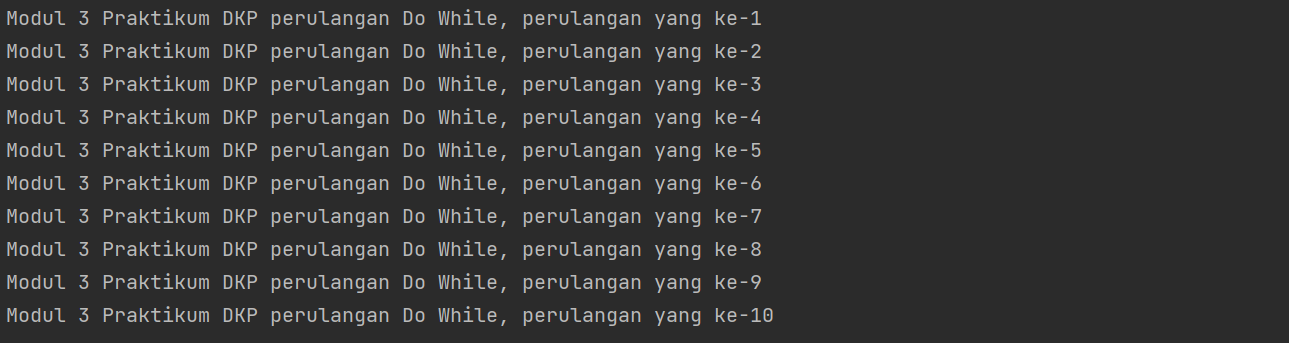
1. Do while

Ubah kode pada Main menjadi seperti berikut:

|  |
| --- |
| package modul3;  public class Perulangan\_KelXX {  public static void main(String[] args) {  int i=1;  do  {  System.out.println("Modul 3 Praktikum DKP perulangan Do While, perulangan yang ke-"+ i);  i++;  } while (i <=10);  }  } |

Gambar 4. 13 Program DO... WHILE... pada Java

*Run* program seperti sebelumnya kemudian amati hasilnya.



Gambar 4. 14 Hasil Keluaran Program

1. For

Sekali lagi, ubah program pada Main menjadi seperti berikut:

package modul3;

public class Perulangan\_KelXX {

public static void main(String[] args) {

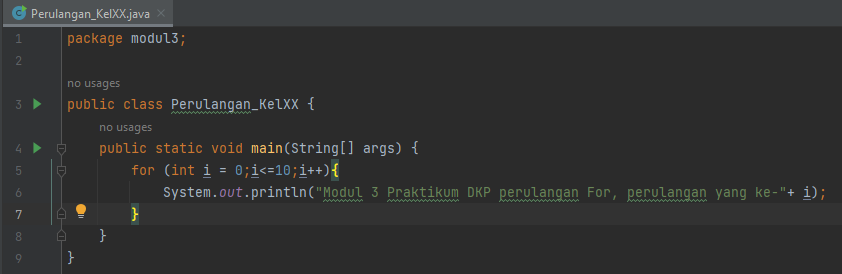
for (int i = 0;i<=10;i++){

System.out.println("Modul 3 Praktikum DKP perulangan For, perulangan yang ke-"+ i);

}

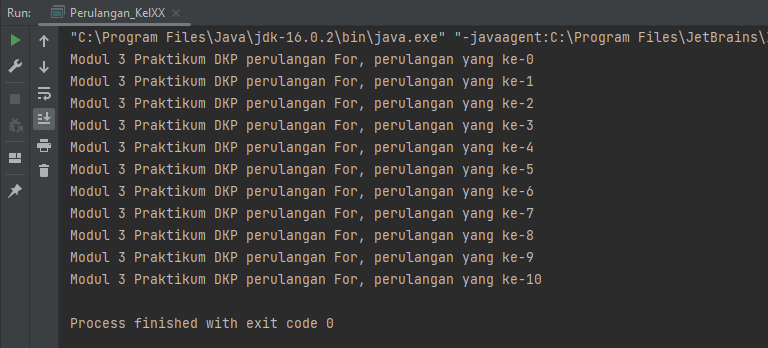
}

}



Gambar 4. 15 Program FOR pada Java

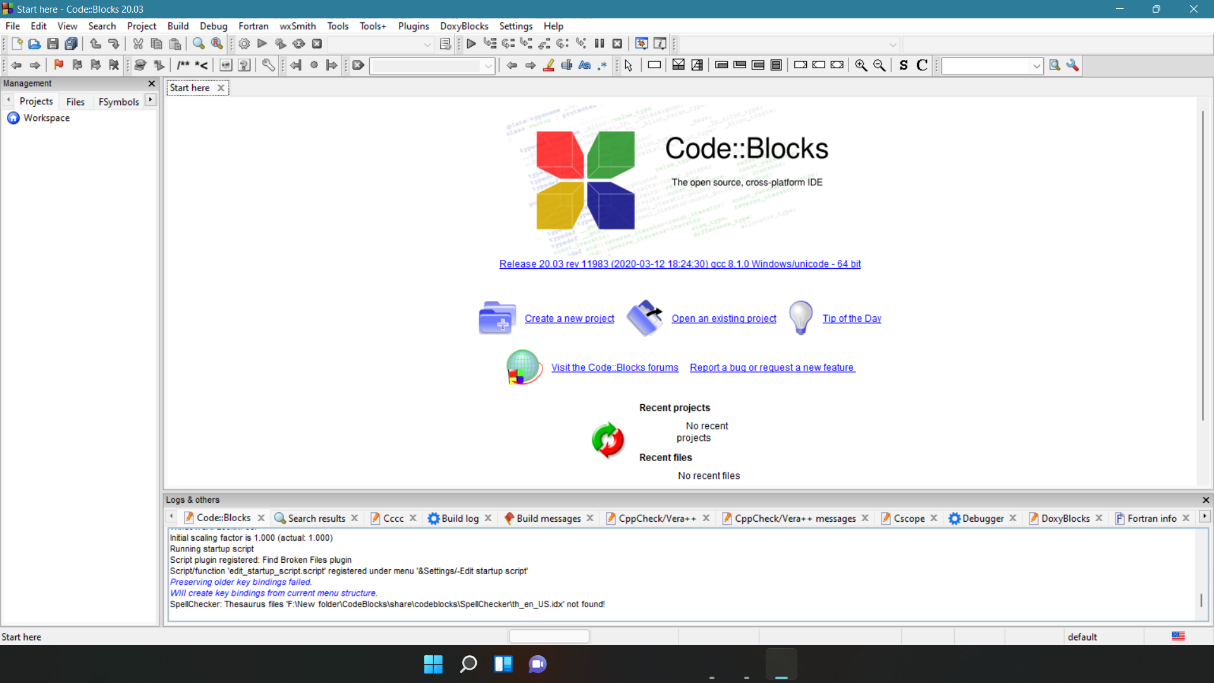
Setelah mengubah *Source Code*, *run* program dan amati hasilnya.

php

Gambar 4. 16 Keluaran hasil program

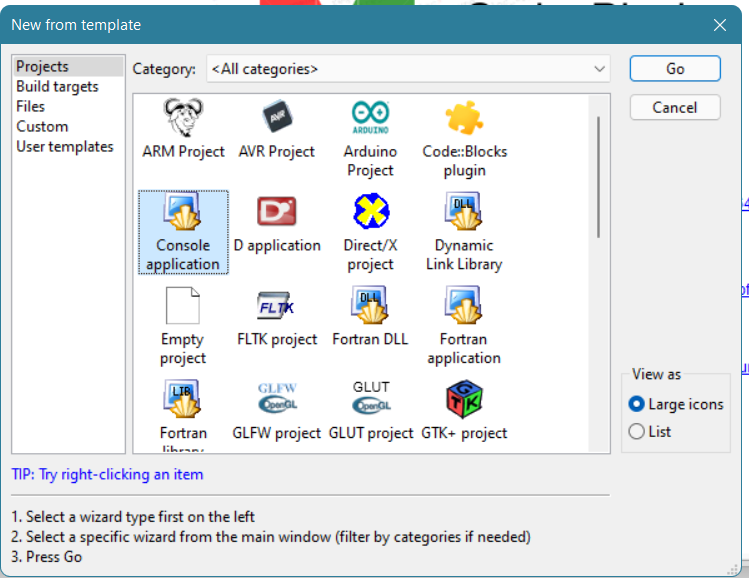
* + 1. **Perulangan pada C++**

1. Membuat *Project* baru
2. Buka CodeBlocks kalian
3. Pilih *create a new project*



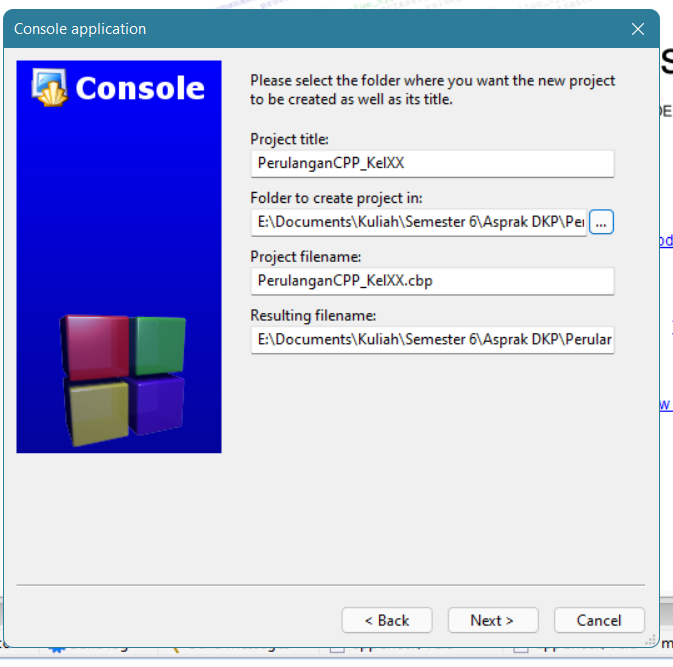
Gambar 4. 17 Memilih *Create a new Project*

1. Pilih *Console* AppC++ kemudian klik *Go, Next* dan pilih Bahasa C++



Gambar 4. 18 Memilih *Console* App C++

1. Beri nama project kemudian klik *Create*

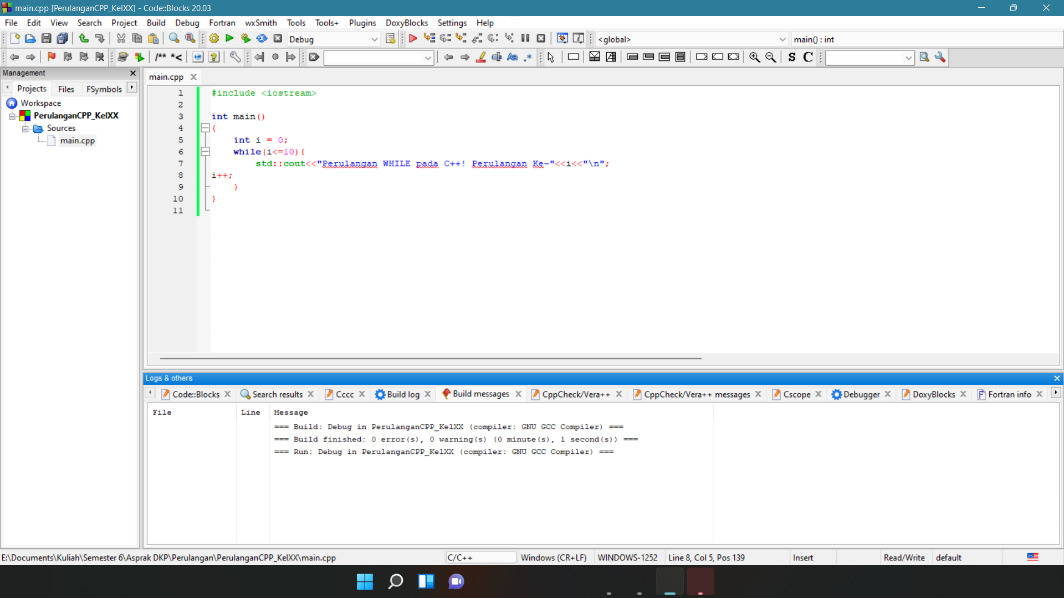


Gambar 4. 19 Memberi nama kemudian membuat *project*

1. Praktikum siap dimulai!
2. While

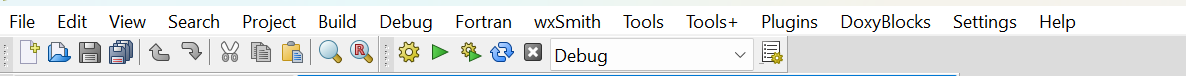
Masukkan *source code* berikut pada program:

|  |
| --- |
| #include <iostream>  int main()  {  int i = 0;  while(i<=10){  std::cout<<”Perulangan WHILE pada C++! Perulangan Ke-”<<i<<”\n”;  i++;  }  } |

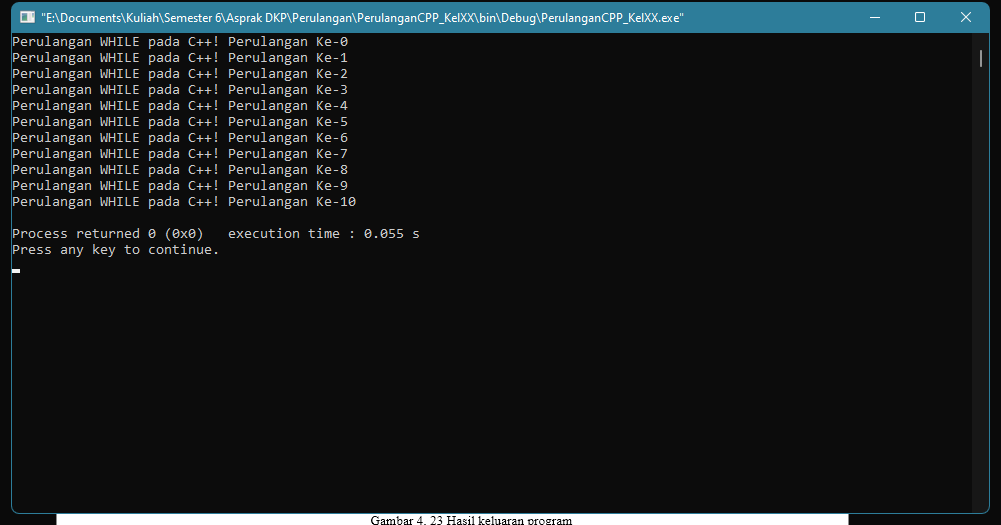


Gambar 4. 20 Perulangan WHILE pada C++

Setelah *source code* dimasukkan, lakukan *run and build* (logo segitiga hijau gear kuning)pada program untuk melihat hasilnya.



Gambar 4. 21 me-*run* program (*local windows debugger*)

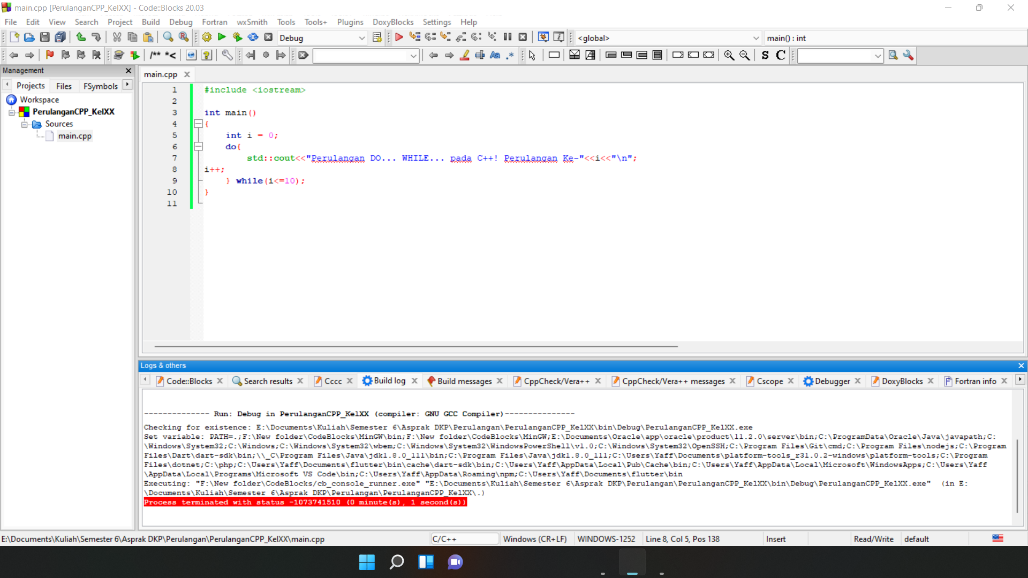


Gambar 4. 22 Hasil keluaran program

1. Do.. while..

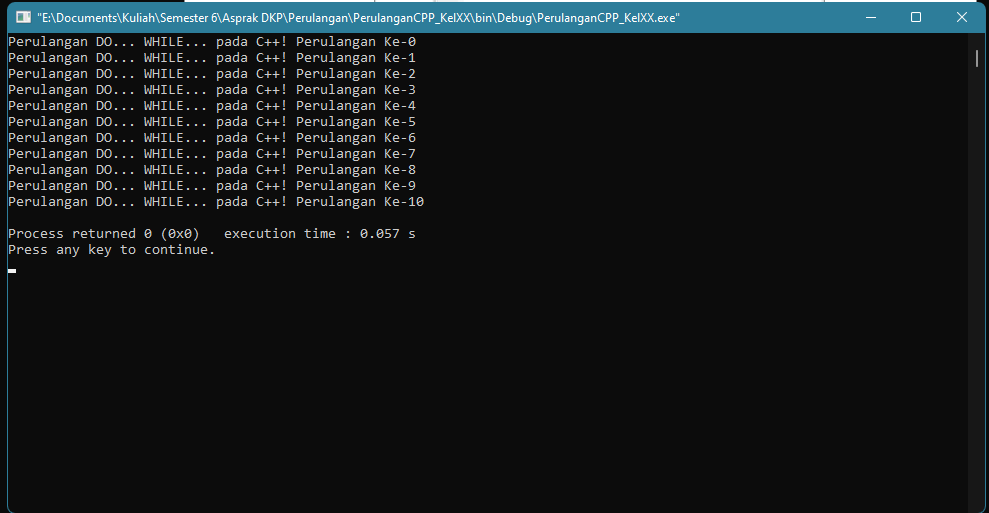
Masukkan *source code* berikut pada program:

|  |
| --- |
| #include <iostream>  int main()  {  int i = 0;  do{  std::cout<<”Perulangan DO... WHILE... pada C++! Perulangan Ke-”<<i<<”\n”;  i++;  } while(i<=10);  } |



Gambar 4. 23 Perulangan DO... WHILE... pada C++

Jika *source code* sudah dimasukkan, *run* program dan amati hasilnya



Gambar 4. 24 Hasil keluaran program

1. For

Masukkan *source code* berikut pada program:

#include <iostream>

int main()

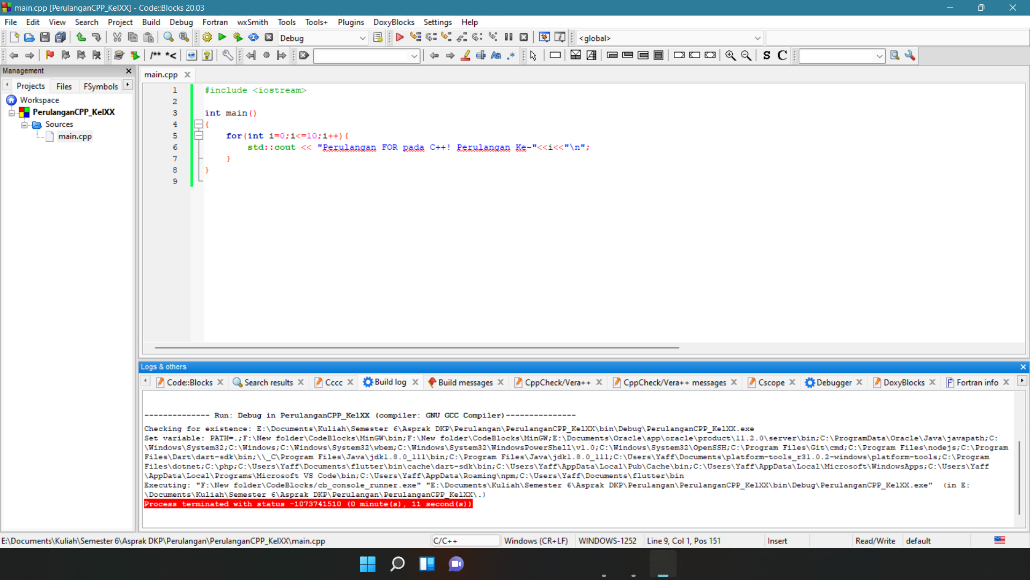
{

for(int i=0;i<=10;i++){

std::cout << "Perulangan FOR pada C++! Perulangan Ke-"<<i<<"\n";

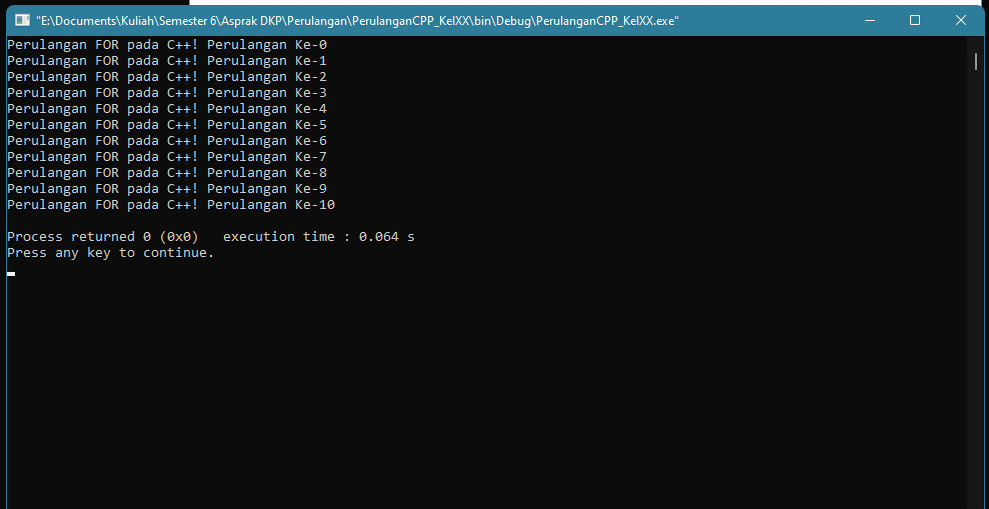
}

}



Gambar 4. 25 Perulangan FOR pada C++

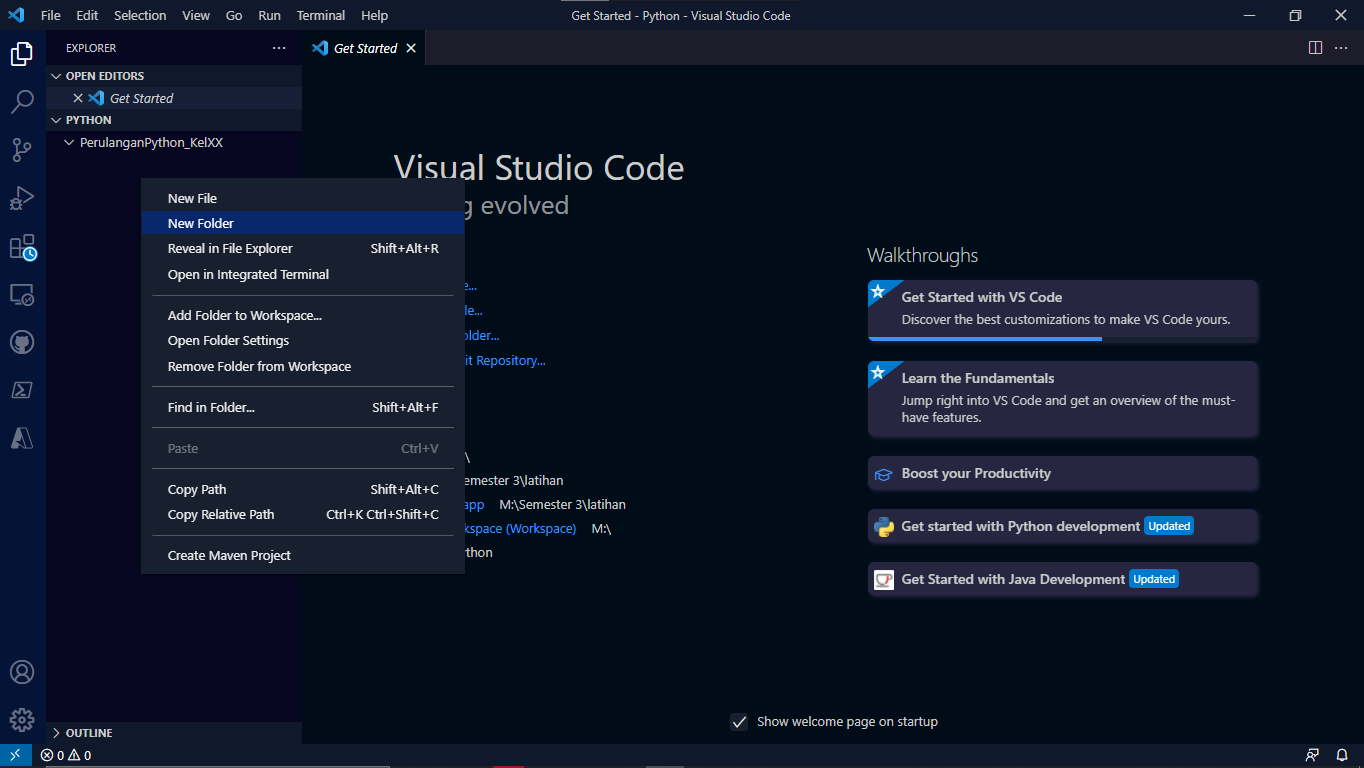
Jika *source code* sudah dimasukkan, *run* program dan amati hasilnya



Gambar 4. 26 Hasil keluaran program

* + 1. **Perulangan pada Python**

1. Membuat *project* baru
2. Buka Visual Studio Code
3. Buat folder baru dengan klik kanan pada workspace kemudian pilih *New Folder* dan beri nama folder PerulanganPython\_KelXX untuk tempat project baru



Gambar 4. 27 Membuat folder baru

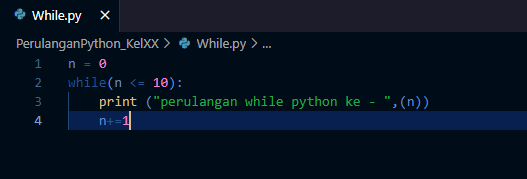
1. Pilih folder yang sudah dibuat tadi, kemudian klik kanan dan pilih *New File* untuk membuat file *Python*. Beri nama file *Python* dengan *Main.py*

Gambar 4. 28 Membuat file *Python* *Main.py*

1. Praktikum siap
2. While

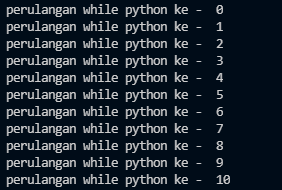
Masukkan *source code* berikut pada program:

|  |
| --- |
|  |



Gambar 4. 29 Perulangan WHILE pada Python

Apabila sudah, *run* program dan amati hasilnya



Gambar 4. 30 Hasil keluaran program

1. Do while

Sebenernya python itu tidak punya memiliki do.. while. Namun, kita dapat membuat program yang berfungsi secara mirip. Masukkan *source code* berikut pada program:

n = 0

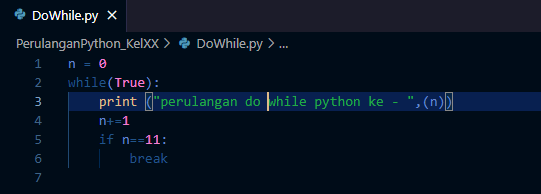
while(True):

print ("perulangan do while python ke - ",(n))

n+=1

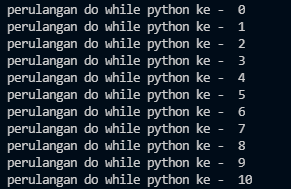
if n==11:

break



Gambar 4. 31 Perulangan DO... WHILE... pada Python

Apabila sudah, *run* program dan amati hasilnya



Gambar 4. 33 Hasil keluaran program

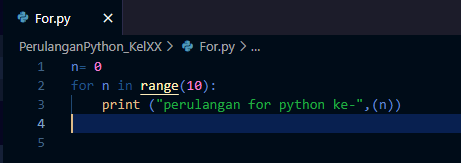
1. For

Masukkan *source code* berikut pada program:

n= 0

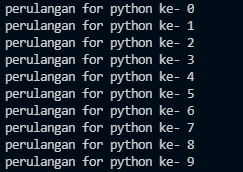
for n in range(10):

print ("perulangan for python ke-",(n))



Gambar 4. 34 Perulangan FOR pada Python

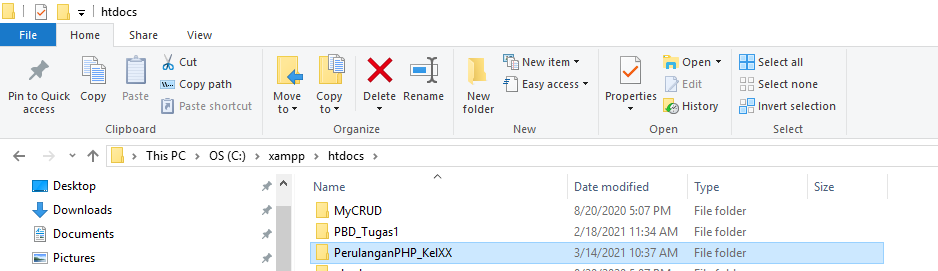
Apabila sudah, *run* program dan amati hasilnya



Gambar 4. 35 Hasil keluaran pada program

* + 1. **Perulangan pada PHP**

1. Membuat *file* baru
2. Buat sebuah *folder* baru bernama Perulangan\_PHP\_KelXX pada direktori C:/xampp/htdocs



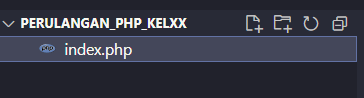
Gambar 4. 36 Membuat *folder* baru

1. Buka Visual Studio Code kalian
2. Pilih menu *file* kemudian *new* -> *new file*



Gambar 4. 37 Membuat *file* baru

1. Save *file* tersebut pada folder yang baru saja kalian buat dengan nama index.php (path: C:/xampp/htdocs/PerulanganPHP\_KelXX)



Gambar 4. 38 *Save file* di folder PerulanganPHP\_KelXX

1. Praktikum siap dimulai!
2. While

Buka Xampp Control Panel, jalankan Apache. Masukkan *source code* berikut pada *file* yang telah dibuat:

<html>

<head><title>Modul 3 | Perulangan</title></head>

<body>

<?php

$i=0;

while($i<=10){

echo "Halo ini Praktikum DKP 2024, perulangan ke-" . $i;

echo "<br />";

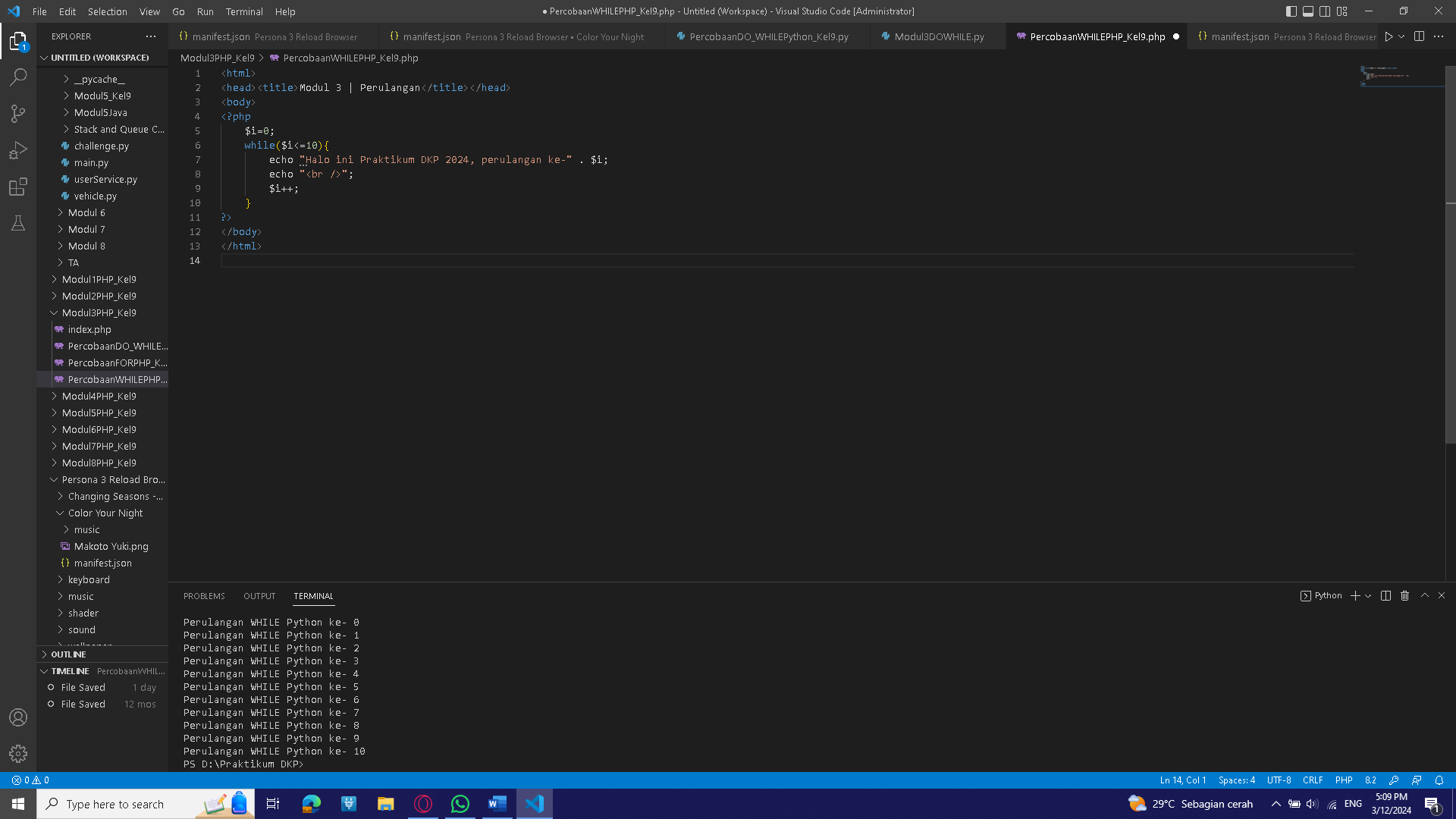
$i++;

}

?>

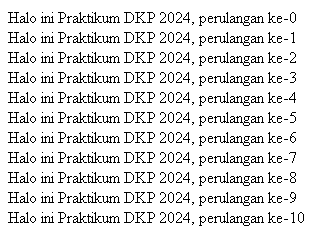
</body>

</html>



Gambar 4. 39 Perulangan WHILE pada PHP

Apabila sudah, *save file* dengan ctrl+s kemudian buka *browser* web kalian dan masukkan address localhost/PerulanganPHP\_KelXX untuk melihat hasilnya.

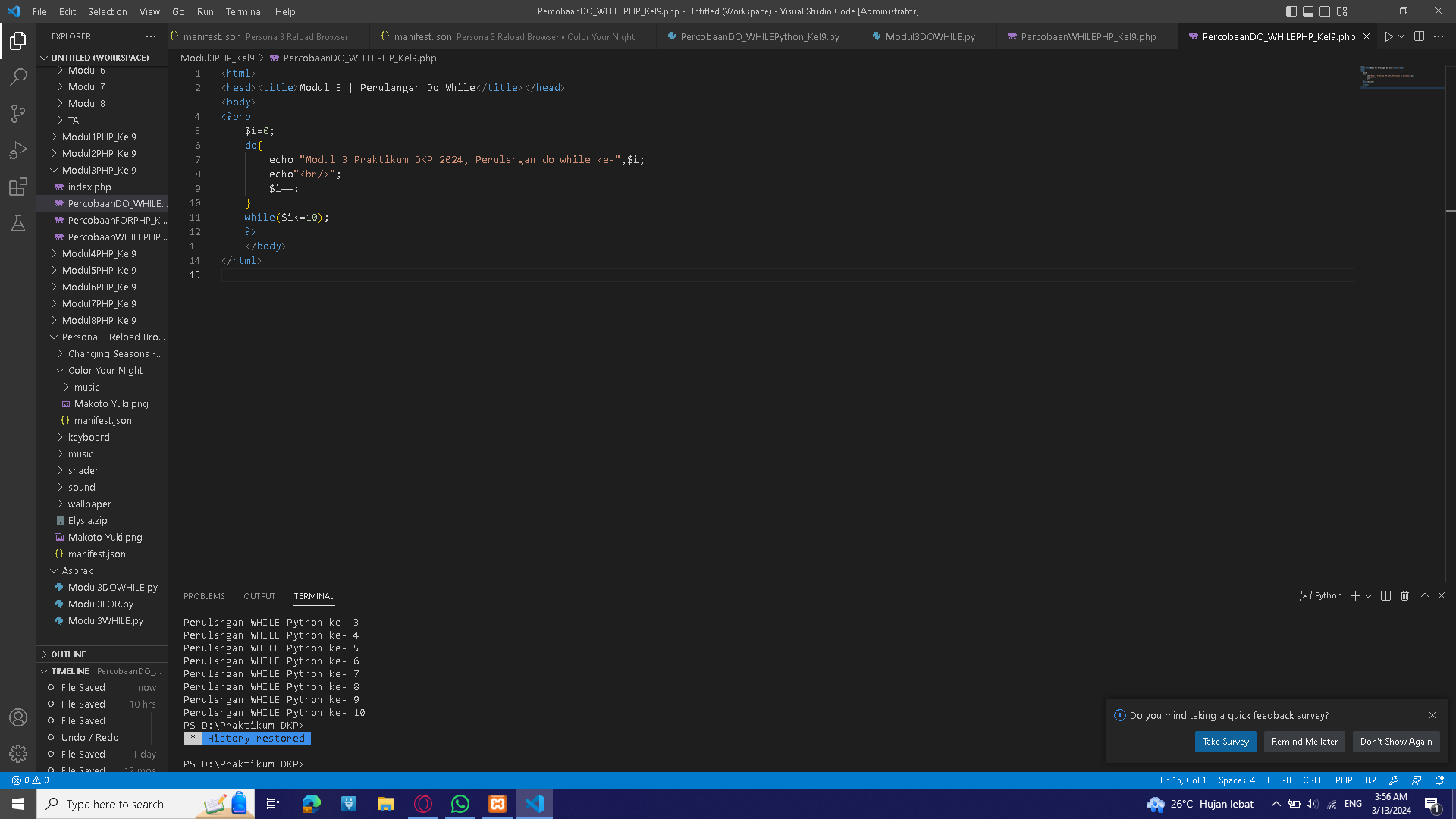


Gambar 4. 40 Hasil keluaran program

1. Do.. While..

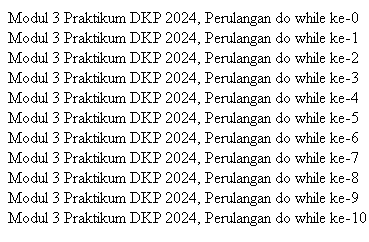
Masukkan *source code* berikut pada *file*:

|  |
| --- |
| <html>  <head><title> Modul 3 | Perulangan Do While </title></head>  <?php  $i=0;  do{ echo "Modul 3 Praktikum DKP 2024, perulangan do while ke-",$i;  echo "<br/>";  $i++;  }  while ($i<=10);  ?>  </body>  </html> |



Gambar 4. 41 Perulangan DO... WHILE... pada PHP

Apabila sudah, *save file* dengan ctrl+s kemudian buka *browser* web kalian dan masukkan address localhost/PerulanganPHP\_KelXX untuk melihat hasilnya.



Gambar 4. 42 Hasil keluaran program

1. For

Masukkan *source code* berikut pada *file*:

<html>

<head><title>Modul 3 Prak DKP | Perulangan For</title></head>

<body>

<?php

for ($i=0;$i<=10;$i++){

echo "ini perulangan for dengan i=", $i;

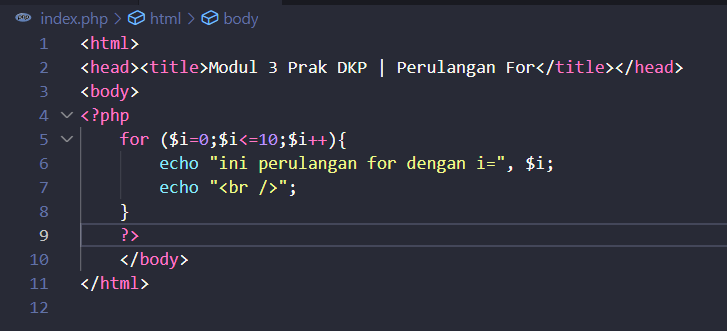
echo "<br />";

}

?>

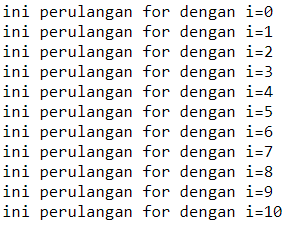
</body>

</html>



Gambar 4. 43 Perulangan FOR pada PHP

Apabila sudah, *save file* dengan ctrl+s kemudian buka *browser* web kalian dan masukkan address localhost/PerulanganPHP\_KelXX untuk melihat hasilnya.



Gambar 4. 44 Hasil keluaran program