**BAB 2**

**Variabel, Tipe Data, dan Array**

## TUJUAN

1. Praktikan dapat memahami pengertian dan konsep Variabel, Tipe Data, dan

Array

1. Praktikan dapat mengetahui penggunaan variabel, tipe data, dan array

dengan bahasa pemrograman Python, C++, Java, dan PHP

1. Praktikan mampu mengimplementasikan variabel, tipe data dan array dalam

bahasa pemrograman Python, C++, Java, dan PHP.

## DASAR TEORI

### **Variabel dan Tipe Data**

Bahasa pemrograman pada umumnya, mengenal adanya variabel yang digunakan untuk menyimpfcvan nilai atau data. Nilai dari variabel dapat di isi dengan informasi yang diinginkan dan dapat diubah nilainya pada saat kode program sedang berjalan. Sebuah variabel memiliki nama yang digunakan untuk mengakses nilai dari variabel itu. Pada sebagian besar bahasa pemrograman, variabel harus dideklarasikan lebih dulu untuk mempermudah *compiler* bekerja. Apabila variabel tidak dideklarasikan maka setiap kali *compiler* bertemu dengan variabel baru pada kode program akan terjadi waktu tunda karena *compiler* harus membuat variabel baru. Hal ini memperlambat proses kerja *compiler*. Bahkan pada beberapa bahasa pemrograman, *compiler* akan menolak untuk melanjutkan proses kompilasi.

Pemberian nama variabel harus mengikuti aturan yang ditetapkan oleh bahasa pemrograman yang kita gunakan. Namun secara umum ada aturan yang berlaku untuk hampir semua bahasa pemrograman. Aturan-aturan tersebut yaitu:

* Nama variabel harus diawali dengan huruf.
* Tidak boleh menggunakan spasi pada satu nama variabel. Spasi bisa diganti

dengan karakter *underscore* (\_).

* Nama variabel tidak boleh mengandung karakter-karakter khusus, seperti :

.,+, -, \*, /, <, >, &, (, ) dan lain-lain.

* Nama variabel tidak boleh menggunakan kata-kata kunci di bahasa

pemrograman

Tipe data pada dasarnya merupakan nama dari jenis data. Dalam bahasa pemrograman, tidak semua variabel memiliki karakteristik yang sama. Informasi yang dapat disimpan oleh variabel tersebut ditentukan berdasarkan jenis atau tipe data. Terdapat berbagai jenis tipe data dalam setiap bahasa pemrograman. Berikut ini adalah beberapa jenis tipe data yang umumnya ada pada hampir seluruh bahasa pemrograman:

1. **Tipe data Karakter (*character*/char)**

Jenis tipe data ini hanya memungkinkan sebuah variabel menyimpan informasi sebuah *single* karakter, seperti huruf “P”,, “c”, “7”, atau “@”.

1. **Tipe data *String***

Tipe data *string* ini memungkinkan variabel menyimpan informasi untaian karakter, seperti kata atau kalimat. Contohnya adalah nama yang digunakan untuk input pada *form*, informasi tersebut akan disimpan dalam variabel bertipe data *string*.

1. **Tipe data *Integer* (int)**

Jenis tipe data *integer* digunakan jika ingin menyimpan informasi angka (bilangan bulat) dalam variabel, tetapi bukan bilangan pecahan. Hal ini berarti nilai “7” akan menjadi nilai *integer* yang valid.

1. **Tipe data *Floating Point Number* (float)**

Tipe data ini digunakan untuk menyimpan informasi angka dengan format pecahan atau angka desimal. Angka “65,00” secara teknis merupakan nilai bertipe *floating* *point* meskipun nilai tersebut juga dapat direpresentasikan sebagai nilai “65”. Jenis tipe data ini membutuhkan ukuran memori yang lebih besar dibandingkan jenis tipe data angka *integer*.

1. **Tipe data *Boolean* (bool)**

Jenis tipe data ini biasanya digunakan hanya untuk merepresentasikan dua kondisi, yakni nilai *TRUE* dan *FALSE* atau data bernilai 1 dan 0 atau kondisi *ON* dan *OFF*. Jenis tipe data ini merupakan jenis yang paling sederhana, tetapi cukup sering digunakan dalam setiap pemrograman.

1. **Tipe data *Array***

Jenis tipe data *array* sering disebut juga sebagai tipe data larik atau disebut *list* pada Python. Tipe data ini pada dasarnya merupakan kumpulan sejumlah variabel bertipe data sama dengan ukuran tertentu, yang tersusun secara runtun.

1. **Variabel dan Tipe data pada Python**

Deklarasi variabel pada python tidak menyertakan tipe data secara eksplisit berbeda dengan C++ dan Java yang harus disertakan pada saat deklarasi variabelnya.

Tabel 1. 1 Tabel Tipe data Python

|  |  |  |
| --- | --- | --- |
| Tipe Data | Contoh | Keterangan |
| *Boolean* | *True* atau *False* | Menyatakan benar(*True*) yang bernilai 1, atau salah(*False*) yang bernilai 0 |
| *String* | "Praktikum DKP Modul 1" | Menyatakan karakter/kalimat bisa berupa huruf angka, dll (diapit tanda “atau”) |
| *Integer* | 1 atau 2112 | Menyatakan bilangan bulat |
| *Float* | 3.14 atau 0.6 | Menyatakan bilangan pecahan |
| *List/Array* | ['abc', 123, 3.23] | Data untaian yang menyimpan berbagai tipe data dan isinya bisa diubah-ubah |

Berikut contoh dari pengkodean deklarasi variabel pada python.

integer = 21120118120014

float = 3.1

string = 'Namaku Lina'

boolean = True

1. **Variabel dan Tipe data pada C++**

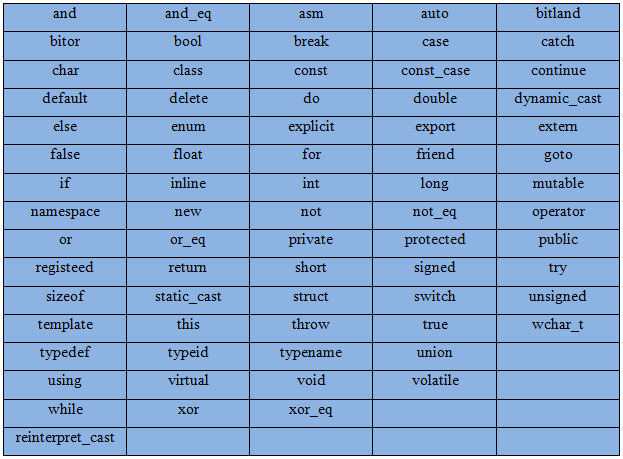
Deklarasi nama variabel pada pemrograman C++ memiliki aturan-aturan penulisan supaya tidak terjadi kesalahan pada program. Pemrograman C++ memiliki beberapa aturan yang harus dipahami sehingga memudahkan kita dalam memberi nama variabel. Berikut aturan yang biasa digunakan.

* + Bersifat Case-Sensitive

*Case-sensitive* artinya adalah membedakan penggunaan antara huruf kecil dan huruf besar. Variabel nama , Nama , NAMA akan dianggap nama variabel yang berbeda. Misalkan pada saat deklarasi variabel menggunakan variabel nama sedangkan ketika dipanggil menggunakan variabel Nama, maka terdapat kesalahan ketika program dijalankan.

* + Tidak boleh diambil dari kata *keyword*

Di dalam pemrograman c++ terdapat *keyword* yang tidak boleh digunakan untuk nama variabel. Berikut adalah daftar *keyword* pada pemrograman C++.



Gambar 1. 1 keyword table c++

* + Terdiri dari huruf dan angka

Penulisan nama variabel boleh menggunakan huruf, angka dan *underscore* " \_ ". Setiap akhir nama variabel ditutup dengan titik koma " ; " Contohnya:

|  |
| --- |
| int tinggi;  int tinggi2;  int tinggi\_badan; |

* + Tidak boleh menggunakan angka di depan

Syarat yang ini berhubungan dengan syarat no 3 ketika menggunakan angka dan huruf, angka tidak boleh berada di depan.

1. **Variabel dan Tipe data pada Java**

Java memiliki beberapa jenis variabel yang dapat dikelompokkan sebagai berikut :

* Instance Variables (tidak statis)

Dalam bahasa pemrograman berorientasi objek, objek menyimpan variabel yang tidak dideklarasikan dengan kata kunci *static* dalam kategori non-statis, atau dapat berubah-ubah. Suatu kelas dapat dijelmakan ke dalam beberapa objek. Nilai yang terkandung dalam variabel tak-statis ini berbeda untuk setiap objeknya.

* Class Variables (statis)

Variabel ini merupakan bagian integral dari suatu kelas, dan tidak ada satu objek pun yang dapat menyatakan kepemilikan atas variabel ini. Variabel yang dideklarasikan sebagai statis digunakan bersama oleh semua objek. Variabel ini lebih bersifat global yang nilainya sama untuk setiap objek pada kelas yang bersangkutan.

|  |
| --- |
| public class DemoInstance {    // Deklarasi instance variabel  **private String name;**  **private int dayaTahan;**      public void setName(String newName){      name = newName;    }      public void setDaya(int newDayaTahan){      dayaTahan = newDayaTahan;    }      public String getName(){      return name;    }      public int getDaya(){      return dayaTahan;    } |

* Local Variables

Variabel ini didefinisikan di dalam suatu metoda (*method*) atau dalam suatu prosedur. Variabel ini bersifat lokal karena hanya dapat diakses oleh metoda atau prosedur tersebut.

|  |
| --- |
| public class Lokal{  public void hitung(){  **int jumlah = 25 \* 5;**    System.out.println("\n");  System.out.println("Jumlah Perkalian dari Angka 25 dan 5 Adalah "+ jumlah);  System.out.println("\n");  } |

* Parameter

Paramater atau argumen adalah variabel yang digunakan pada saat suatu metoda atau prosedur dipanggil. Parameter berguna untuk memberikan nilai awal untuk diteruskan (*pass*) ke dalam suatu prosedur atau metoda.

**Literal**

Pada bagian ini akan dijelaskan tentang literal, yaitu rangkaian kata atau huruf yang menyatakan suatu nilai. Misalnya,

**int** angka = 10;

Pada pernyataan di atas, yang dinamakan literal adalah 10, karena 10 berarti bilangan bulat atau integer. Pada bahasa pemrograman java, terdapat beberapa jenis literal yang melambangkan bilangan bulat, riil, kalimat, atau boolean.

**Literal Bilangan Bulat**

Bilangan bulat dapat dilambangkan dalam beberapa bentuk. Bilangan bulat biasa dilambangkan dengan deretan angka yang dimulai dengan angka yang bukan nol.

int angka = -10;

**Literal Karakter**

Kita dapat melambangkan suatu karakter dengan tanda petik tunggal misalnya ‗a‘ atau ‗3′ atau ‗=‘. Suatu karakter dapat juga dilambangkan dengan kode ASCII nya. Caranya dengan memulainya dengan \u00 (garis miring terbalik) kemudian diikuti dengan kode ASCII nya dalam bentuk heksadesimal.

*// huruf 'A' dalam ASCII*

**char** huruf = '**\u**0041';

**Literal Boolean**

Nilai *true* dan *false* pada java merupakan *literal boolean*. Suatu variabel bertipe *boolean* hanya dapat memiliki nilai *true* atau *false*.

**boolean** ok = **true**;

Ingat bahwa *boolean true* atau *false* tidak menggunakan tanda petik tunggal seperti ekspresi pada karakter.

**Literal Bilangan Riil**

Bilangan riil, misalnya -0.00127 atau 415.842, pada java dapat disimpan baik sebagai *float* atau *double*. Bilangan real dapat direpresentasikan dalam bentuk desimal biasa, pecahan, atau eksponen (dilambangkan dengan e atau E).

Imbuhan akhir d atau D dan f atau F dapat pula ditambahkan untuk menentukan tipenya secara eksplisit. Berikut beberapa contohnya.

double d = 3.27E+32;

float f = 4f;

float pi = 3.14159;

**Literal *String***

*String* merupakan untaian huruf dan angka yang tersusun menjadi satu kalimat. *String* pada java tidak disimpan dalam bentuk *array* seperti pada C.. *String* ditulis di antara dua tanda petik ganda seperti contoh berikut.

String salam = "Selamat Datang";

*String* juga dapat mengandung karakter spesial seperti dibahas pada literal karakter. Misalnya

String hallo = "Selamat Datang";

System.out.println("Hallo\nSelamat datang praktikan DKP

2021");

1. **Variabel dan Tipe data pada PHP**

Dalam bahasa pemrograman PHP, variabel dikenali dengan adanya tanda dollar ($) yang kemudian bisa diikuti dengan angka, huruf, dan underscore, akan tetapi variabel tidak dapat mengandung spasi. Untuk dapat mendefinisikan variabel, kita hanya perlu menuliskan kemudian penerjemah akan mengenali secara otomatis.

|  |
| --- |
| $nama  $no\_telp  $\_pekerjaan |

misalnya:

|  |
| --- |
| $nilai = 98765656;  $kata\_acak = “praktikum DKP”; |

Berbeda dengan bahasa pemrograman lainnya, variabel di PHP lebih fleksibel. Kita tidak perlu mendefinisikan jenisnya ketika mendefinisikan untuk pertama kali. Ada 6 variabel dasar yang dapat diakomodasi di PHP yaitu :

Tabel 1. 2 Tabel Tipe data PHP

|  |  |  |
| --- | --- | --- |
| Tipe | Contoh | Penjelasan |
| *Integer* | 177982 | Semua angka bukan pecahan |
| *Double* | 5234.123 | Nilai Pecahan |
| *String* | “latihan” | Kumpulan karakter |
| *Boolean* | *True* | TRUE atau FALSE |
| *Object* |  | Sebuah *instance* dari *object* |
| *Array* |  | Larik |

### **Array**

Jenis tipe data *array* sering disebut juga sebagai tipe data larik. Tipe data ini pada dasarnya merupakan kumpulan sejumlah variabel bertipe data sama dengan ukuran tertentu, yang tersusun secara *run*tun. Ada beragam variasi dari tipe data array, bergantung pada bahasa pemrograman yang digunakan. Sebagai contoh, nilai “1”, “2”, “3”, “4”, dan “5” dapat Anda simpan ke dalam variabel bertipe data *array* dengan ukuran array “5” (5 buah nilai, mulai dari 1 sampai dengan 5) yang memiliki kumpulan nilai bertipe integer atau secara sederhana dapat dikatakan *array of* *integer values*.

Setiap variabel yang terdapat dalam array dapat diakses dengan menggunakan komponen indeks dari tiap-tiap variabel tersebut. Namun, variabel pertama yang ada dalam array bukan memiliki indeks 1, melainkan indeks 0. Menyimpan nilai-nilai tersebut ke dalam variabel bertipe data array akan memudahkan kita untuk melakukan manipulasi terhadap setiap data yang ada dalam array tersebut. Jenis tipe data string sebenarnya merupakan kumpulan variabel bertipe karakter atau bisa dikatakan array of *characters*.

Ada dua jenis tipe data array:

1. Array 1 dimensi

Array 1 dimensi yaitu sekelompok data yang mempunyai nama variabel dan tipe data yang sama dan dapat diakses memakai 1 buah index saja.

|  |
| --- |
| var nama\_variabel: array[range\_index] of tipe\_data |

Contoh:

Buatlah deklarasi array dengan nama variabel deret yang mempunyai kapasitas 10 dengan tipe data integer!

Jawab:

|  |
| --- |
| var deret:array[10] of integer |

1. Array 2 dimensi (multi dimensi)

Merupakan sebuah kumpulan data yang mempunyai nama dan tipe data yang sama demana elemennya dapat diakses memakai 2 index, dan dapat disebut sebagai index baris dan index kolom.

|  |
| --- |
| var nama\_variable:array[index\_baris][index\_kolom] of tipe\_data |

Contoh:

Deklarasikan variabel array yang mempunyai index baris 10 dan index kolom 5 dengan nama data yang mempunyai tipe data integer atau bilangan bulat!

Jawab:

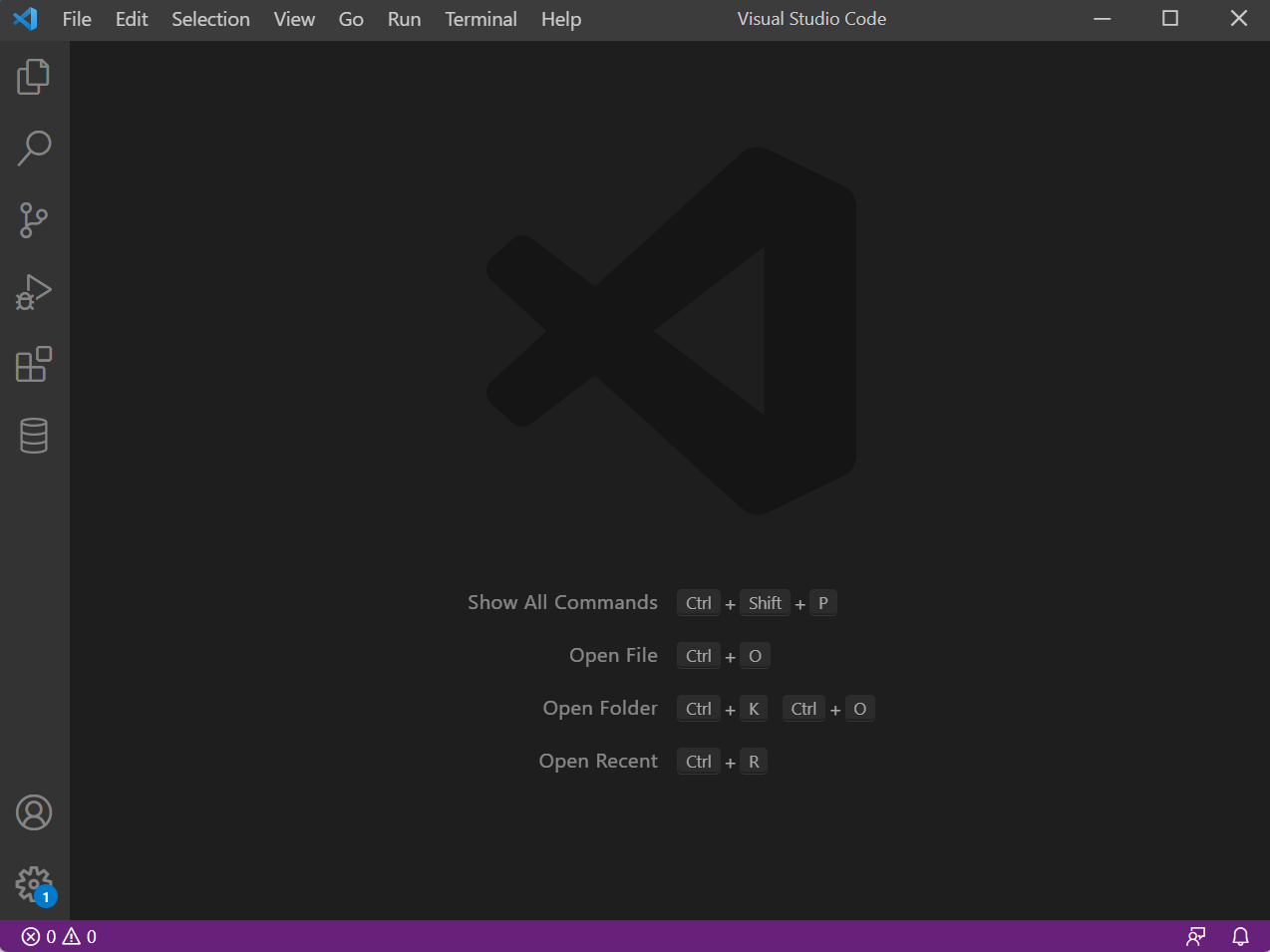
|  |
| --- |
| var data:array[10][5] of integer |

## PERCOBAAN

### **Percobaan 1 (Deklarasi variabel dan tipe data)**

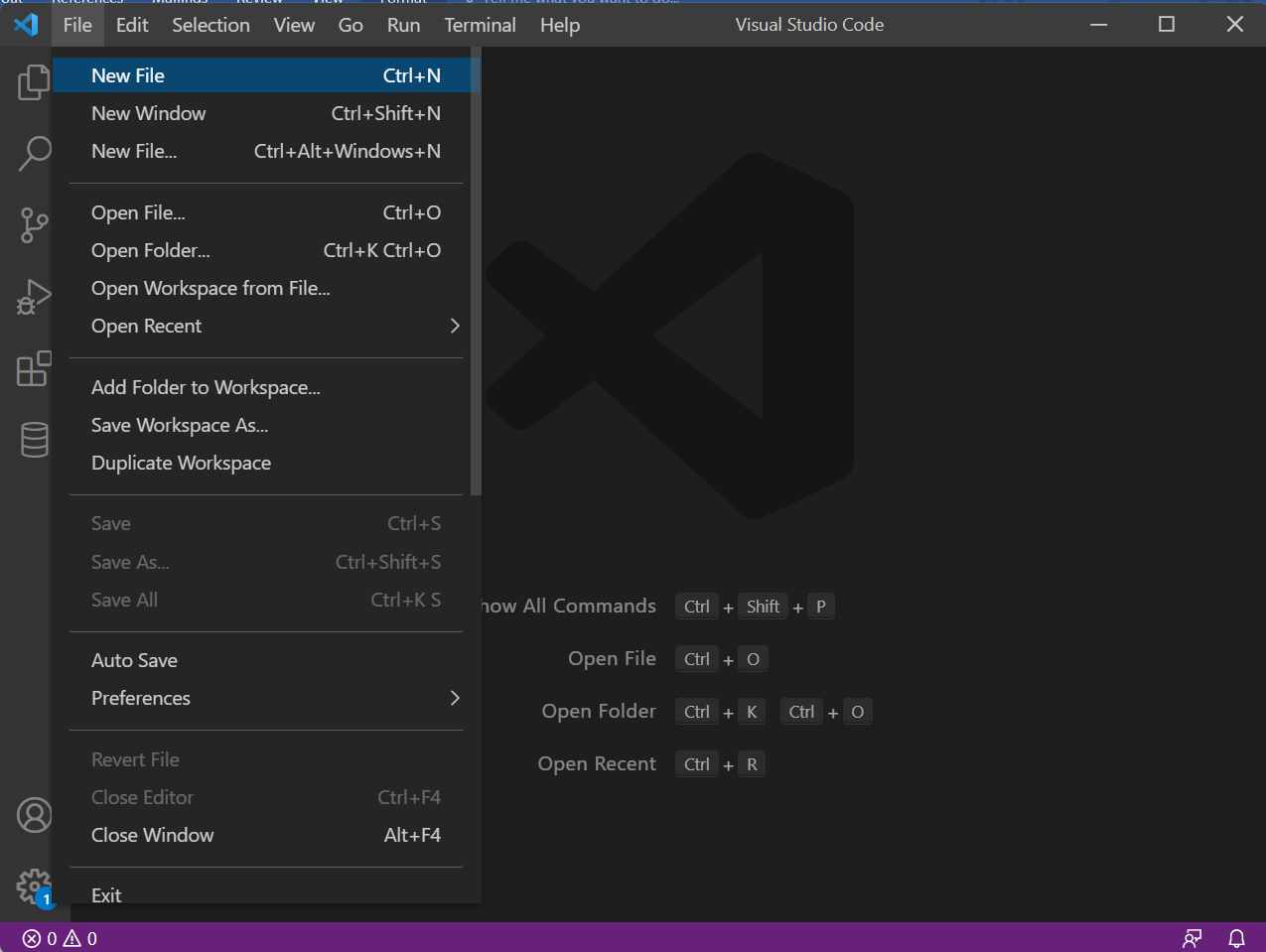
### **Deklarasi variabel dan tipe data pada Python**

1. Buka Visual Studio Code
2. Pilih *File*



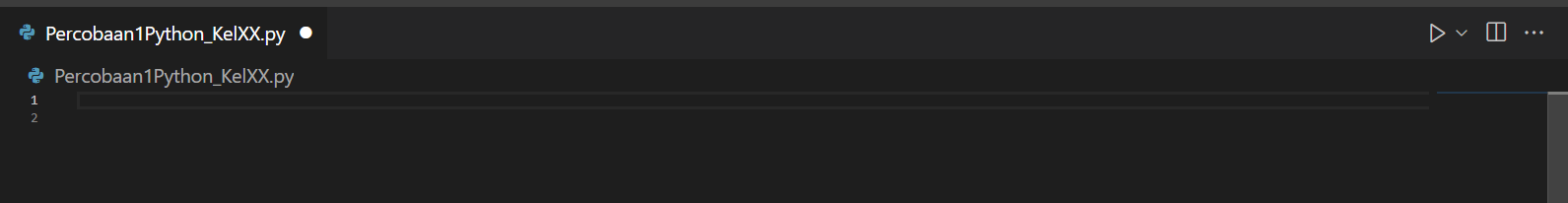
Gambar 2. 1 Memilih *File*

1. Pilih *New File*



Gambar 2. 2 Memilih *New File*

1. Beri nama *file* Percobaan1Python\_KelXX.py

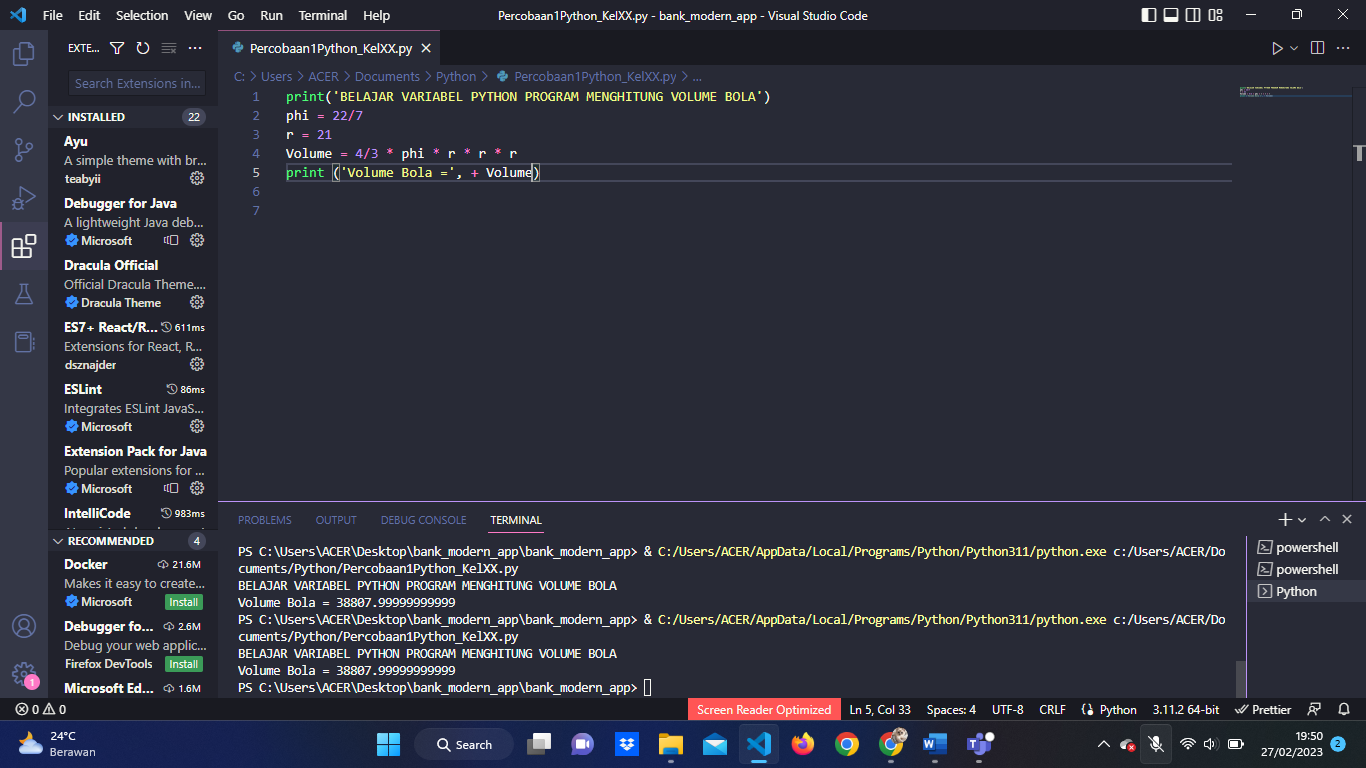


Gambar 2. 3 *File name*

1. Masukkan *source code* berikut pada program:

|  |
| --- |
| print('BELAJAR VARIABEL PYTHON PROGRAM MENGHITUNG VOLUME BOLA')  phi = 22/7  r = 7  V = 4/3 \* phi \* r^3  print ('Volume Bola =', + V) |

1. Klik *start* dan amati hasilnya

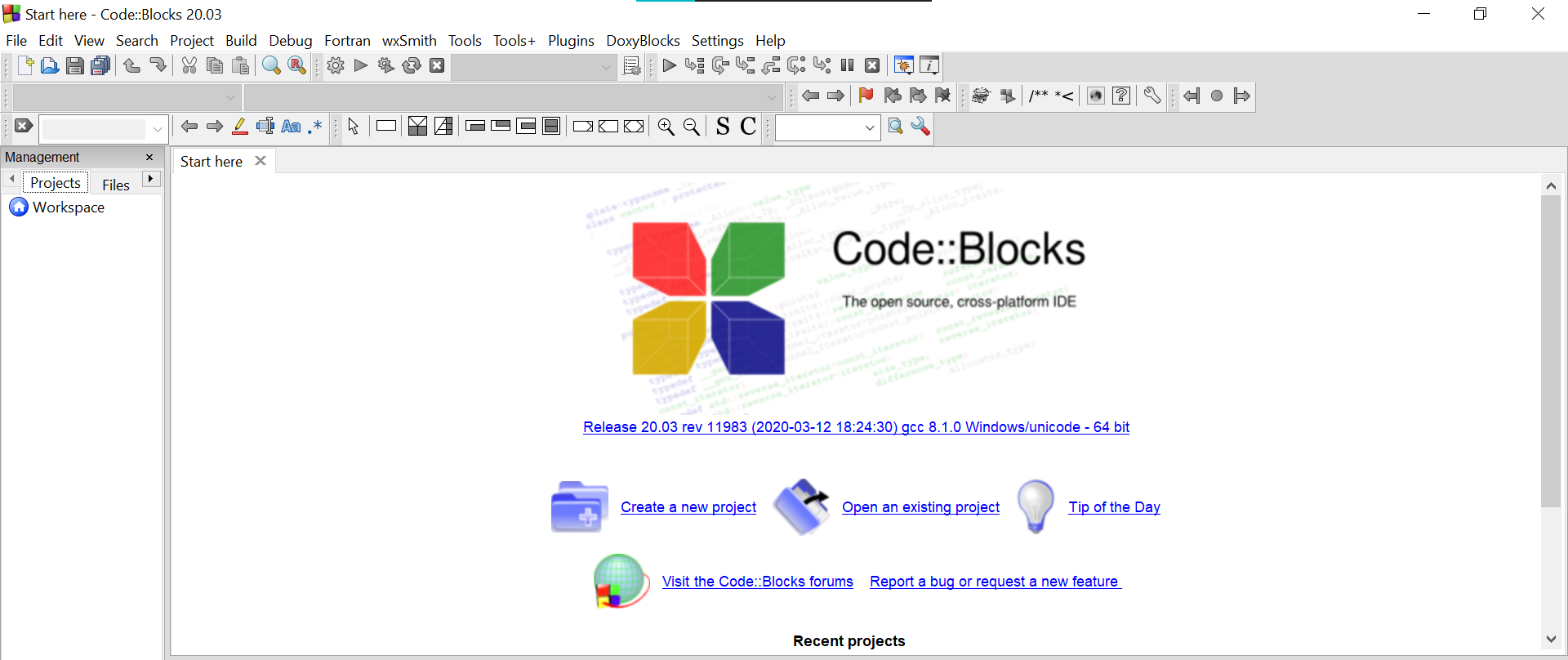


Gambar 2. 4 *Output* Percobaan1 Python

### **Deklarasi variabel dan tipe data pada C++**

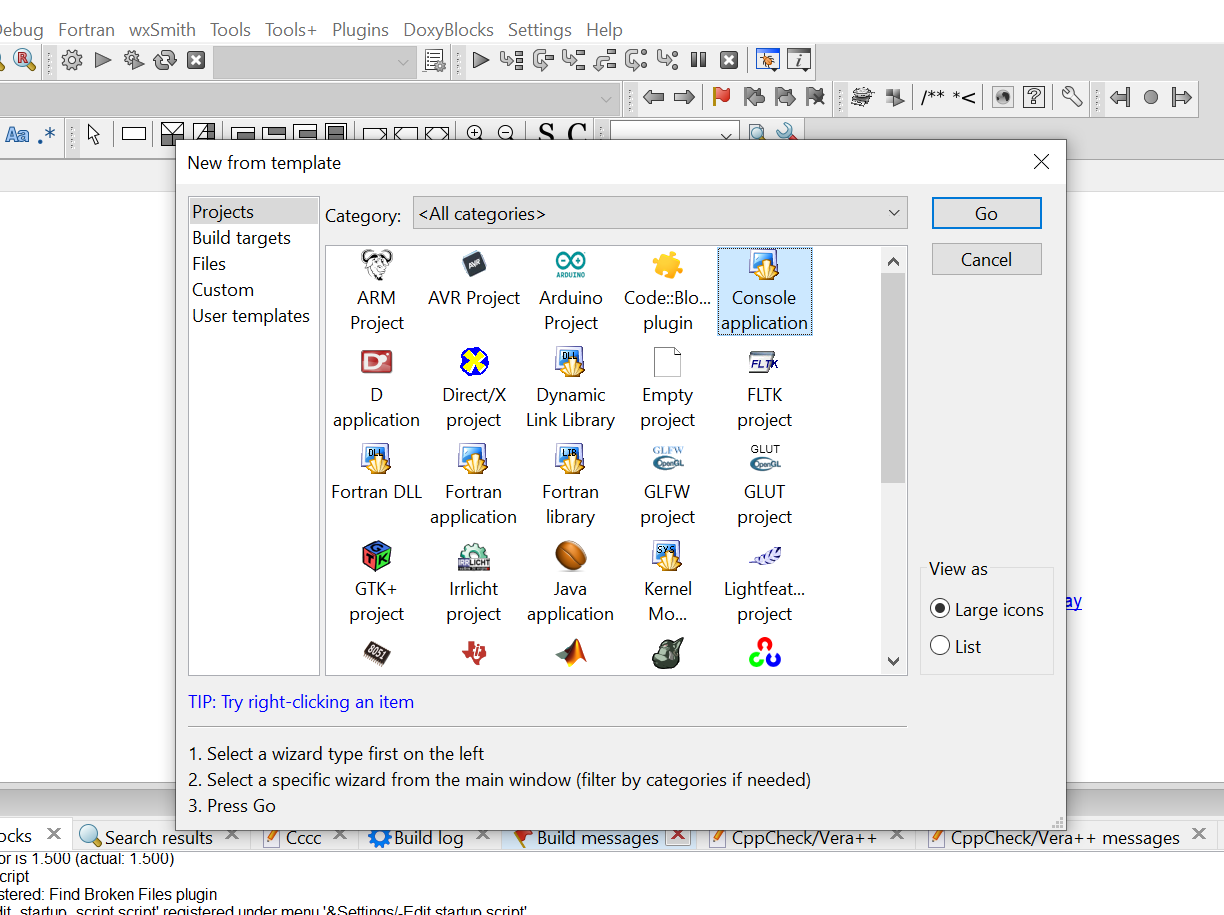
1. Buka program CodeBlocks yang telah diinstal.

2. Kemudian buat projek baru dengan cara klik *create a new project*.

****

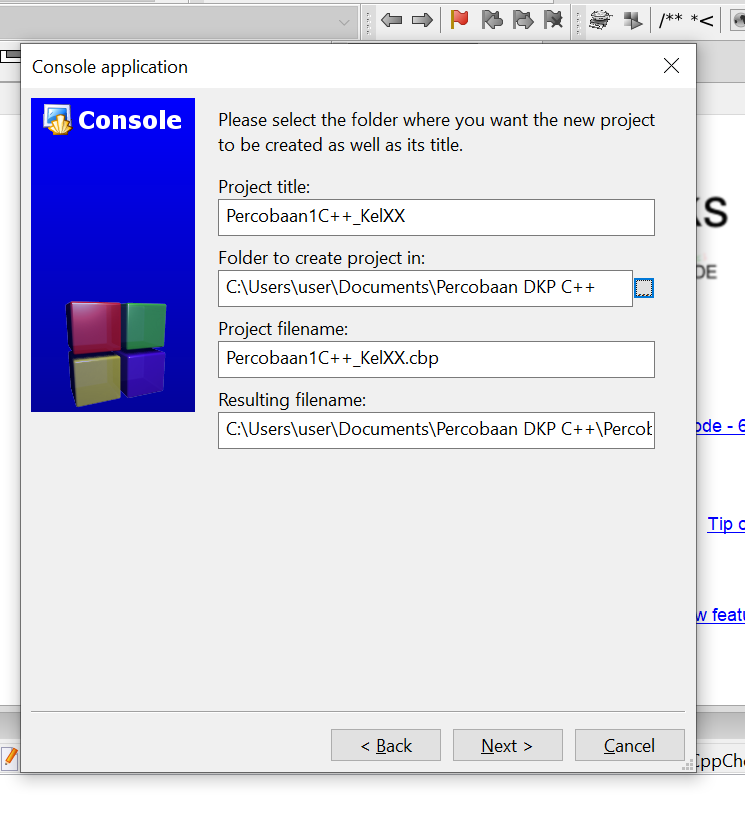
Gambar 2.5 Tampilan Pertama CodeBlock

3. Lalu pilih *console application.*



Gambar 2.6 *Build project* dengan *console application*

1. Selanjutnya klik *next* jika muncul *pop up* kemudian pilih menu C++ dan klik *next.*
2. Selanjutnya beri nama Percobaan1C++\_KelXX dan klik *next* selanjutnya akan muncul *pop up* pastikan *compiler* memakai GNU GCC Compiler lalu klik *finish.*

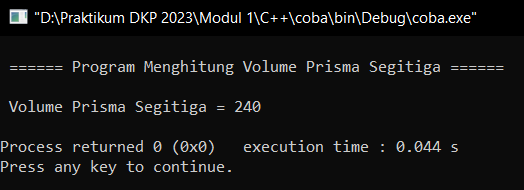


Gambar 2.7 Penamaan *Project*

1. Setelah itu masukkan kode berikut ini dan klik tombol *build and run* yang berada pada bagian atas sebelah *icon* *play*.

|  |
| --- |
| #include <iostream>  using namespace std;  int main() {  //Volume Prisma Segitiga  //V = Luas Alas \* Tinggi  int volume, alas, tinggisegitiga, tinggiprisma;  alas = 6;  tinggisegitiga = 8;  tinggiprisma = 10;  volume = 0.5 \* alas \* tinggisegitiga \* tinggiprisma;  cout << "\n ====== Program Menghitung Volume Prisma Segitiga ====== \n\n";  cout << " Volume Prisma Segitiga = " << volume << endl;  return 0;} |

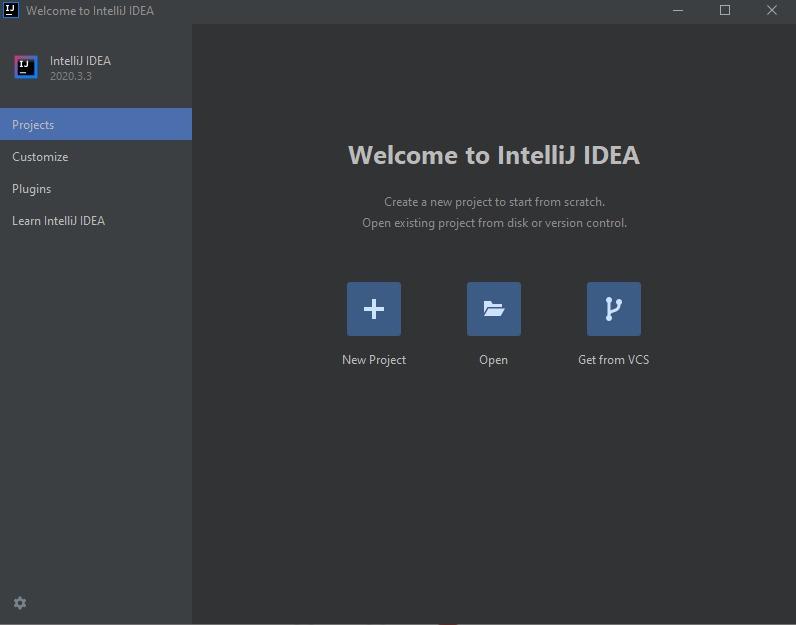
1. Maka hasilnya akan seperti ini.



Gambar 2.8 *Output Percobaan C++*

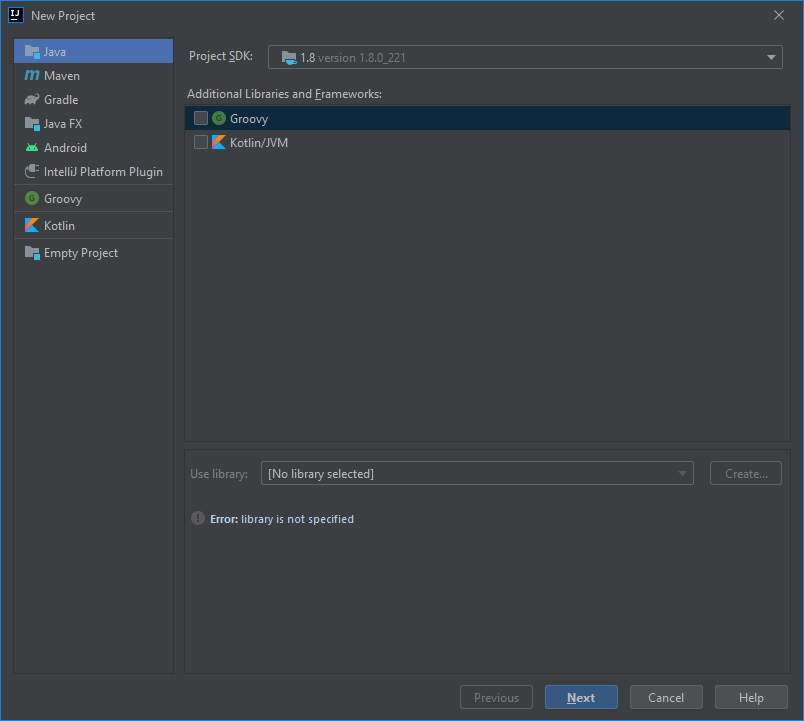
### **C. Deklarasi variabel dan tipe data pada Java**

1. Buat *project* baru dengan aplikasi Intellij IDEA Community
2. Pilih *new file* *project* (file -> *new project*)

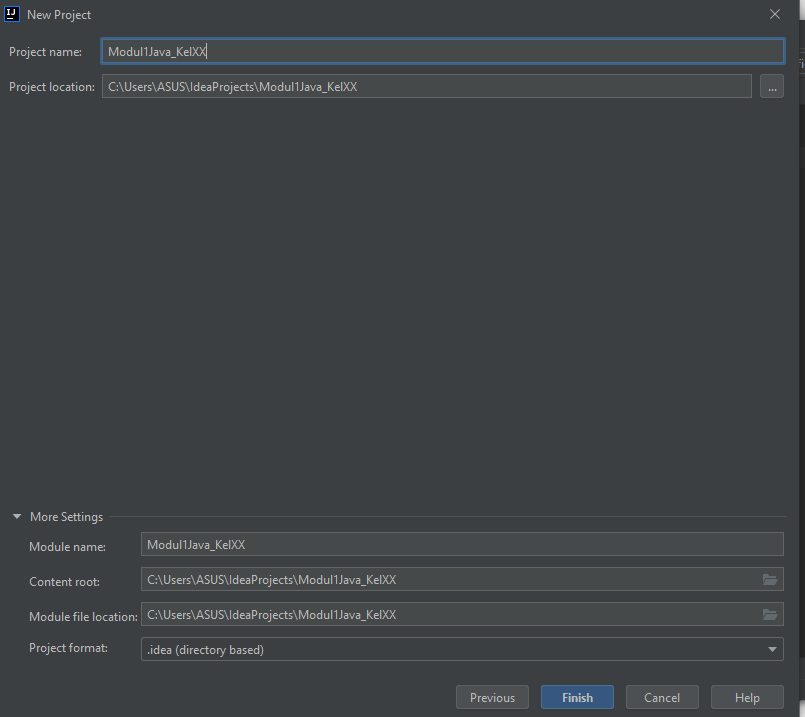


Gambar 2. 9 *New Project* Percobaan 1

1. Pilih java kemudian *next*, *next*, beri nama *project* Modul1Java\_KelXX kemudian *finish*

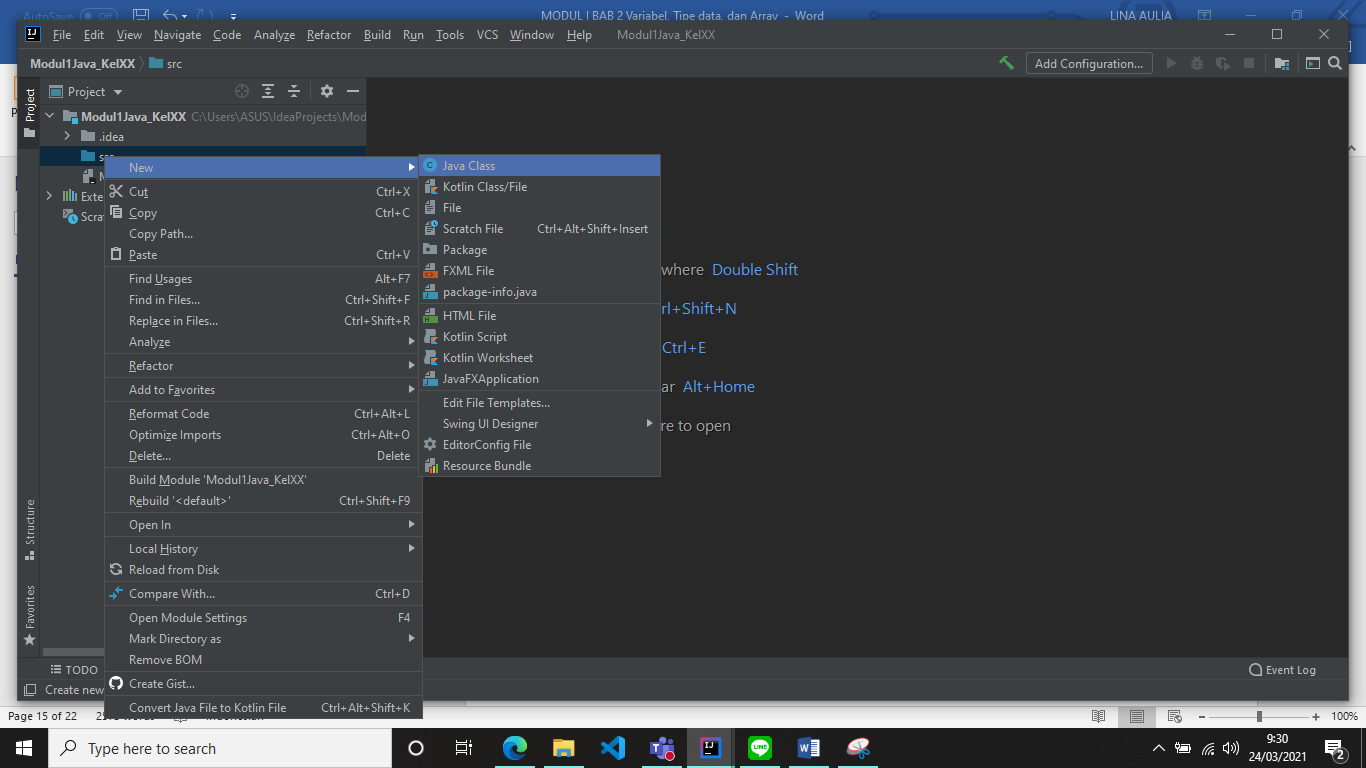


Gambar 2. 10 *New Project* Percobaan 1 Java



Gambar 2. 11 *project name* Java

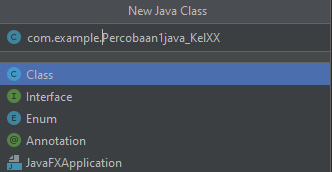
1. Setelah itu, pada bagian *project* (sebelah kiri) klik kanan pada folder src klik kanan *new*, kemudian pilih Java *Class*



Gambar 2. 12 *New Java Class*

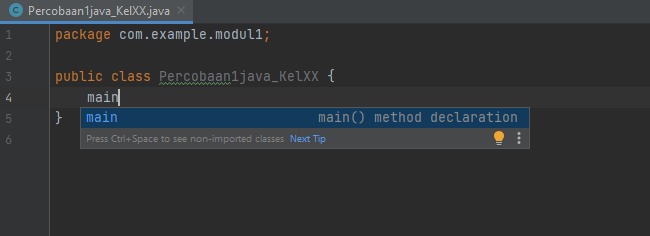
1. Berikan nama com.example.Percobaan1java\_KelXX kemudian

*Enter*



Gambar 2. 13 *Class name* percobaan 1 java

1. Kemudian ketik main kemudian klik *option* yang muncul dan klik *Enter*.

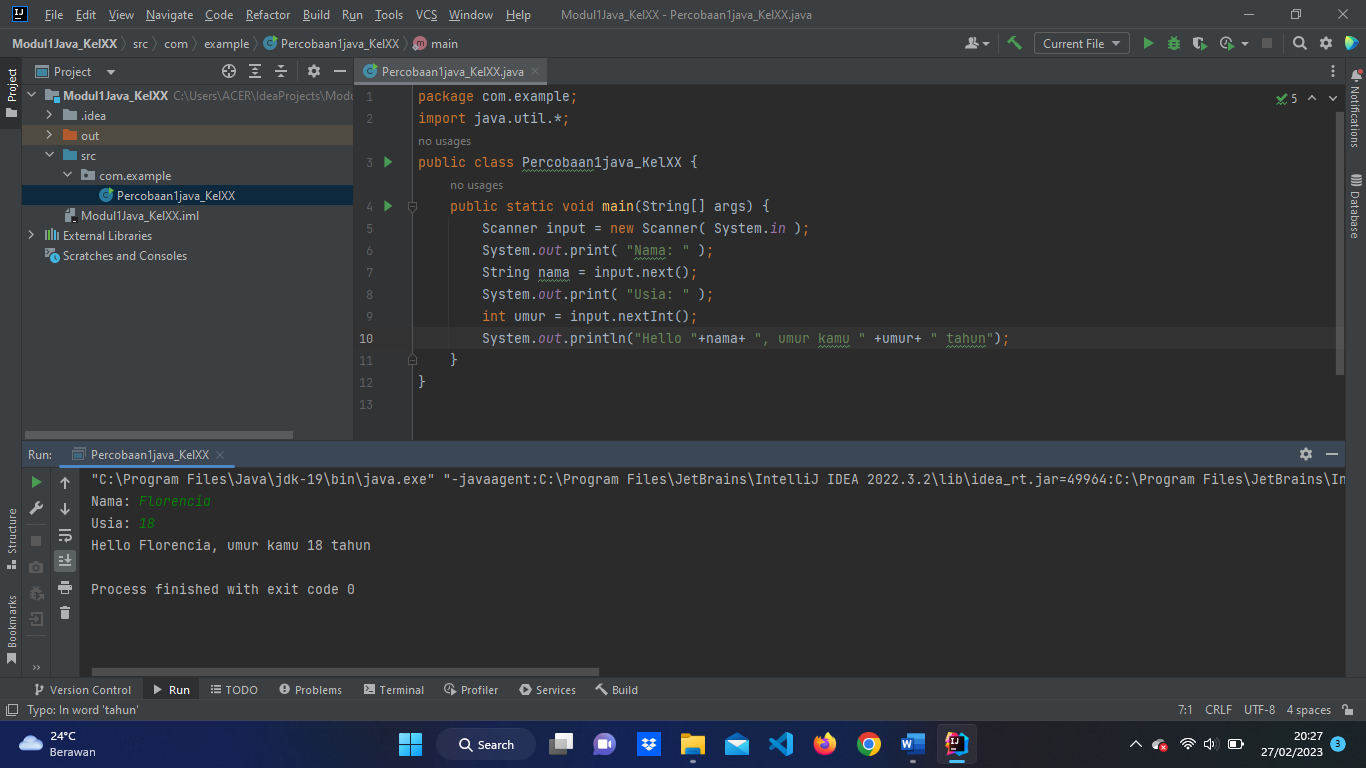


Gambar 2. 14 *main method*

1. Kemudian masukkan *source code* berikut

|  |
| --- |
| package com.example;  import java.util.\*;  public class Percobaan1java\_KelXX {  public static void main(String[] args) {  Scanner input = new Scanner( System.in );  System.out.print( "Nama: " );  String nama = input.next();  System.out.print( "Usia: " );  int umur = input.nextInt();  System.out.println("Hello "+nama+ ", umur kamu " +umur+ "tahun");  }  } |

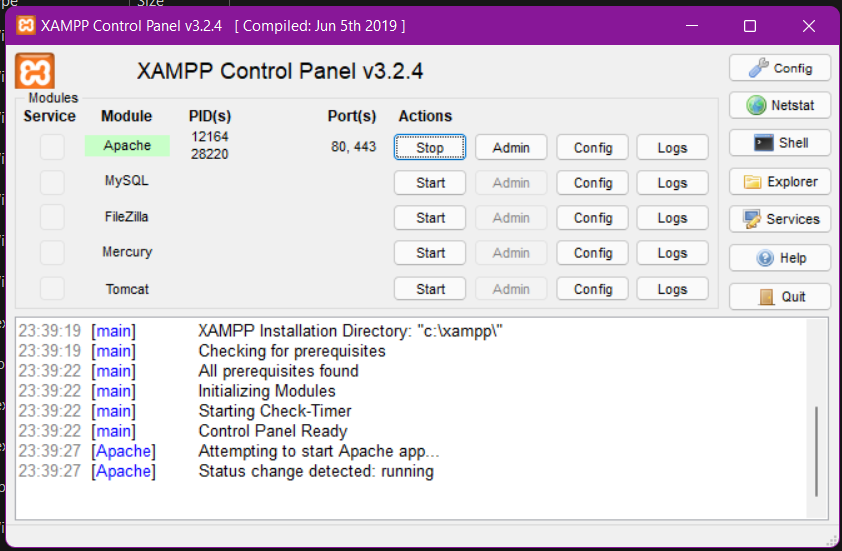
1. Kemudian *Run* dan amati hasilnya



Gambar 2. 15 *Output* Percobaan 1 Java

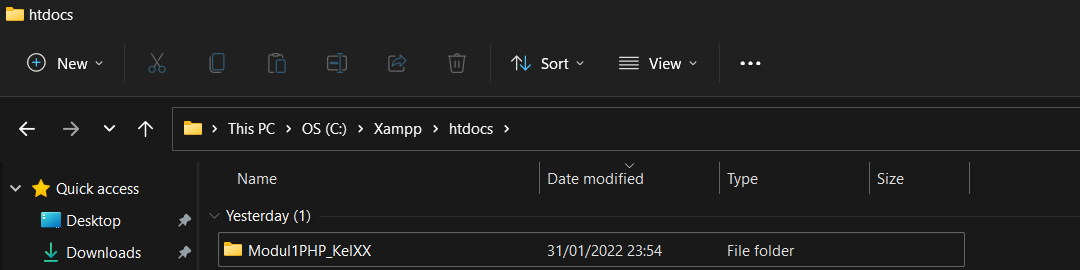
### **D. Deklarasi variabel dan tipe data pada PHP**

* + 1. Buka aplikasi xampp control panel, jalankan Apache.



Gambar 2. 16 menjalankan Apache

1. Buatlah sebuah folder baru bernama Modul1PHP\_KelXX pada direktoriC:/xampp/htdocs



Gambar 2. 17 *New* Folder

1. Kemudian buka Visual Studio Code kalian
2. Pilih menu *File* kemudian *new* -> *new file*

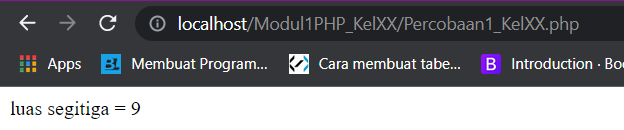


Gambar 2. 18 *New File*

1. Save *file* tersebut pada folder yang baru saja kalian buat dengan nama Percobaan1\_KelXX.php (path: C:/xampp/htdocs/Modul1PHP\_KelXX)
2. Masukkan *source code* berikut pada *file* yang telah kalian buat

|  |
| --- |
| <html>  <head>  <body>  <?php  $alas=6;  $tinggi=3;  $luas= 0.5 \* ($alas\*$tinggi) ;  echo "luas segitiga = $luas";  ?>  </body>  </head>  </html> |

1. Kemudian *save file* dengan *ctrl+s* kemudian buka *browser web* kalian dan masukkan *address* localhost/Modul1PHP\_KelXX/Percobaan1\_KelXX.php untuk melihat hasilnya.

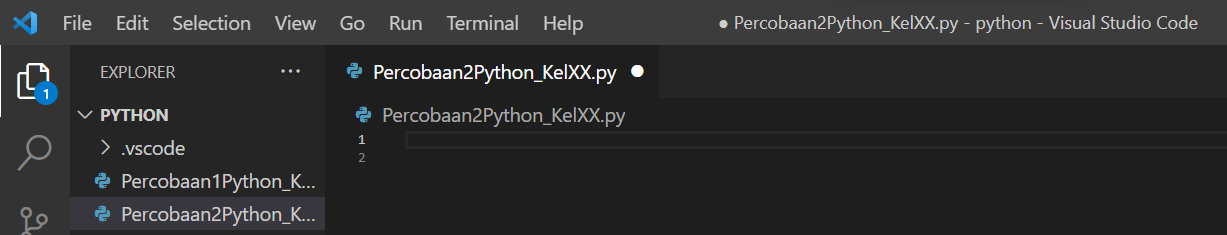


Gambar 2. 19 *Output* Percobaan 1 PHP

### **Percobaan 2 (Array)**

### **Array pada Python**

* 1. Buka kembali Visual Studio Code kalian, buat *file* python baru kemudian beri nama Percobaan2Python\_KelXX

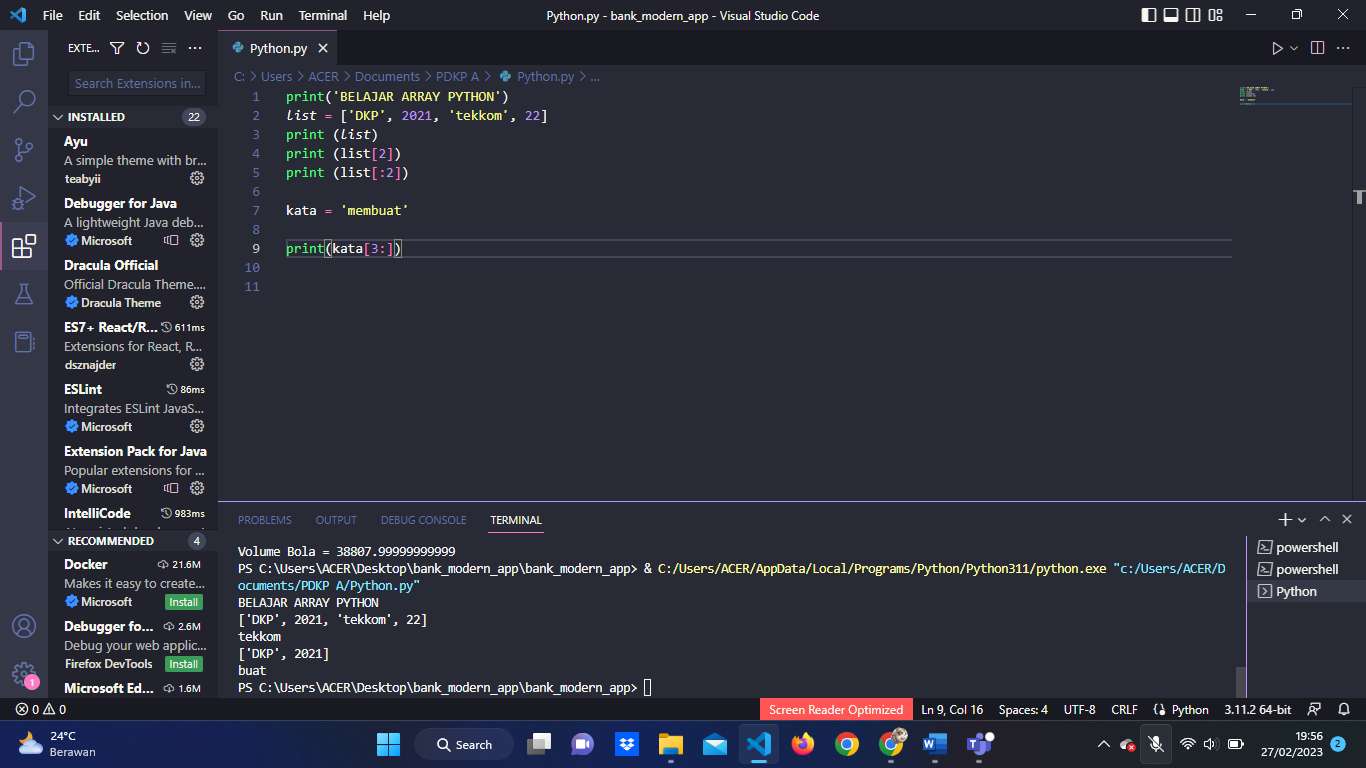


Gambar 2. 20 *File Name* Percobaan 2 Python

* 1. Kemudian masukkan *source code* berikut pada program

|  |
| --- |
| print('BELAJAR ARRAY PYTHON')  list = ['DKP', 2021, 'tekkom', 22]  print (list)  print (list[2])  print (list[:2])  kata = 'membuat'  print(kata[3:]) |

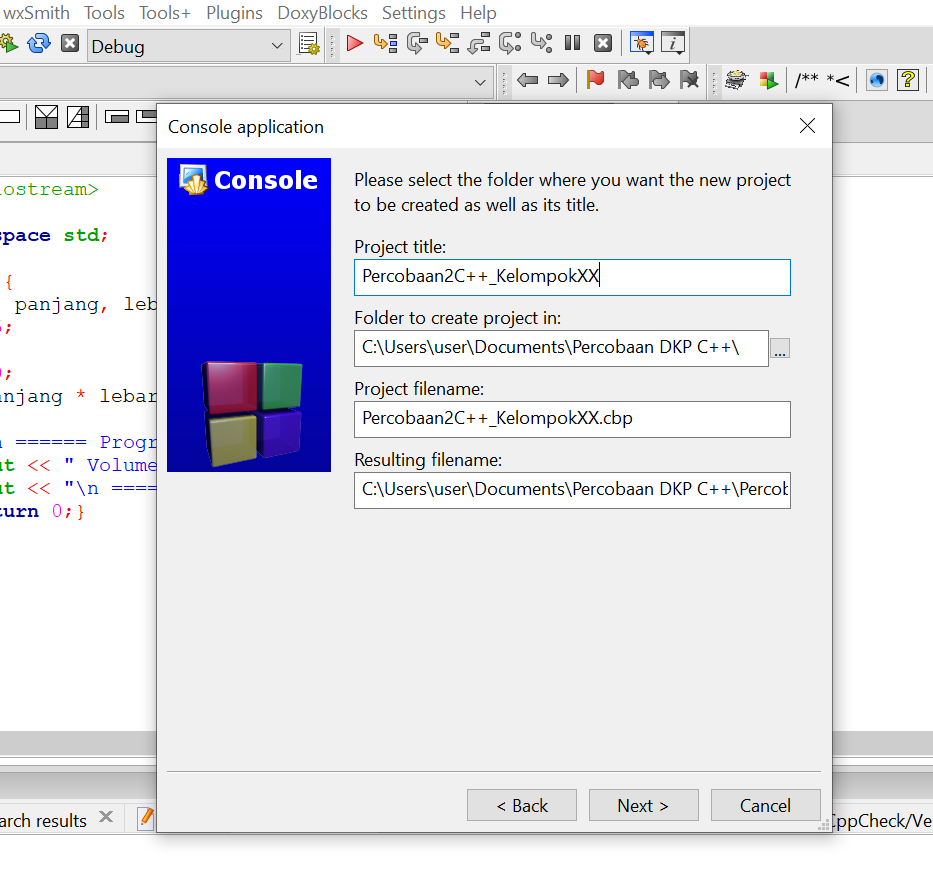
* 1. Klik *run* dan amati hasilnya



Gambar 2. 21 Hasil *running* Program

### **Array pada C++**

* + 1. Buat *File>New>Project*  di CodeBlocks , pilih bahasa C++ kemudian beri nama Percobaan2C++\_KelompokXX

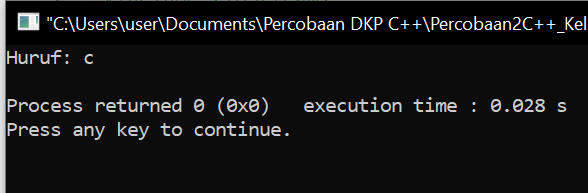


Gambar 2. 22 *Create Project* C++

* + 1. Kemudian klik dua kali pada bagian *sources* lalu klik dua kali pada file main.cpp lalu masukkan *source code* berikut jangan lupa *save* (CTRL+S)

|  |
| --- |
| #include <iostream>  #include <string>  using namespace std;  int main() {  char huruf[5] = { 'a', 'b', 'c','d', 'e' };  //mengambil data pada array  cout << "Huruf: " << huruf[2] << endl;  return 0;  } |

* + 1. Klik *Run* dan amati hasilnya



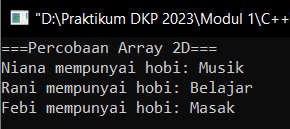
Gambar 2. 23 Hasil *Running* Program

Lalu untuk array 2 dimensi pada C++ maka lakukan langkah percobaan:

1. Buat *File>New>Project* dan beri nama Percobaan3C++\_KelXX
2. Masukkan *source code* berikut:

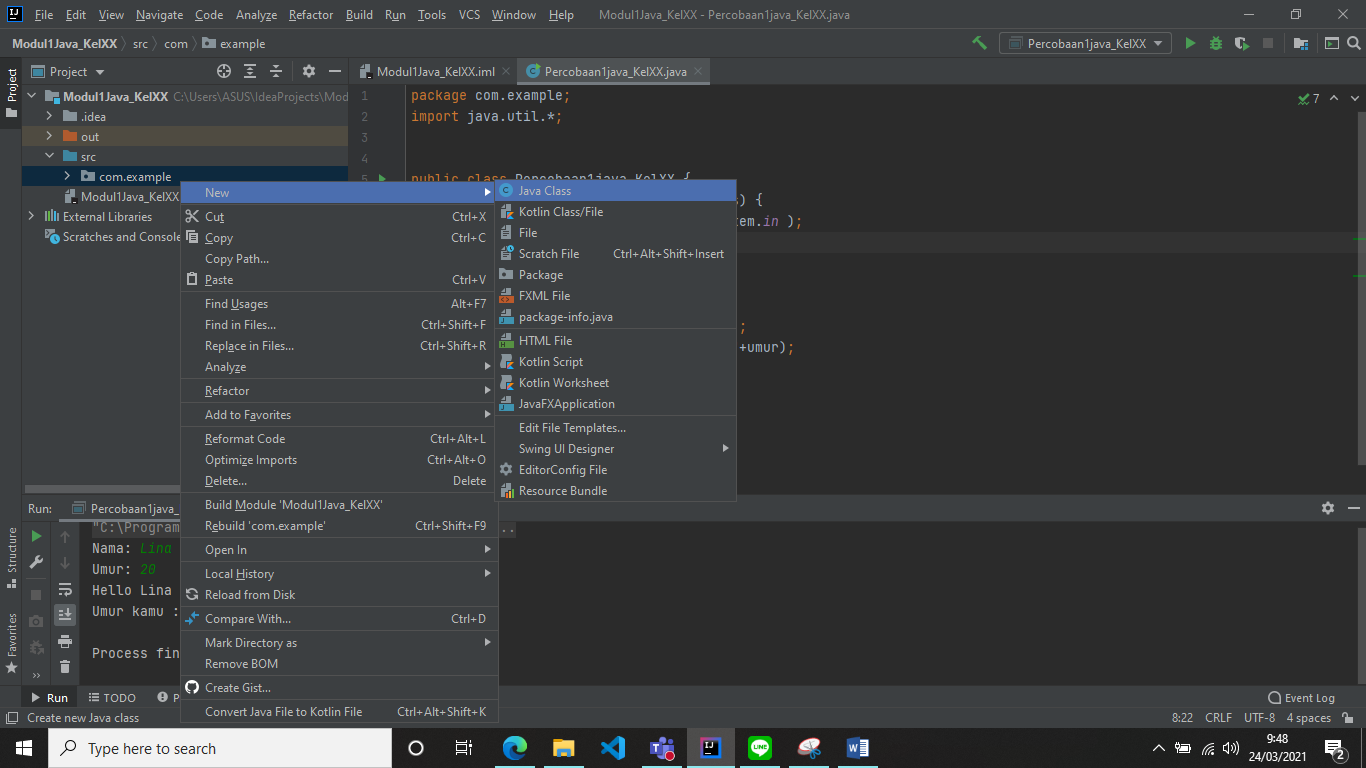
|  |
| --- |
| #include <iostream>  using namespace std;  int main()  {  string arr[2][3] = {{"Niana","Rani","Febi"},{"Musik","Belajar","Masak"}};  cout << "===Percobaan Array 2D===" << endl;  cout << arr[0][0] <<" mempunyai hobi: "<< arr[1][0] <<" "<<endl;  cout << arr[0][1] <<" mempunyai hobi: "<< arr[1][1] <<" "<<endl;  cout << arr[0][2] <<" mempunyai hobi: "<< arr[1][2] <<" "<<endl;  return 0;  } |

1. Maka *run code* dan hasilnya seperti ini

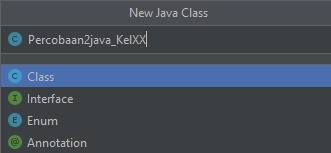


### ***Array* pada Java**

1. Buat *class* baru dengan cara klik kanan pada *pakage* com.example.modul 1 kemudian beri namabaru beri nama Percobaan2java\_KelXX



Gambar 2. 24 *New Class* Percobaan 2 Java

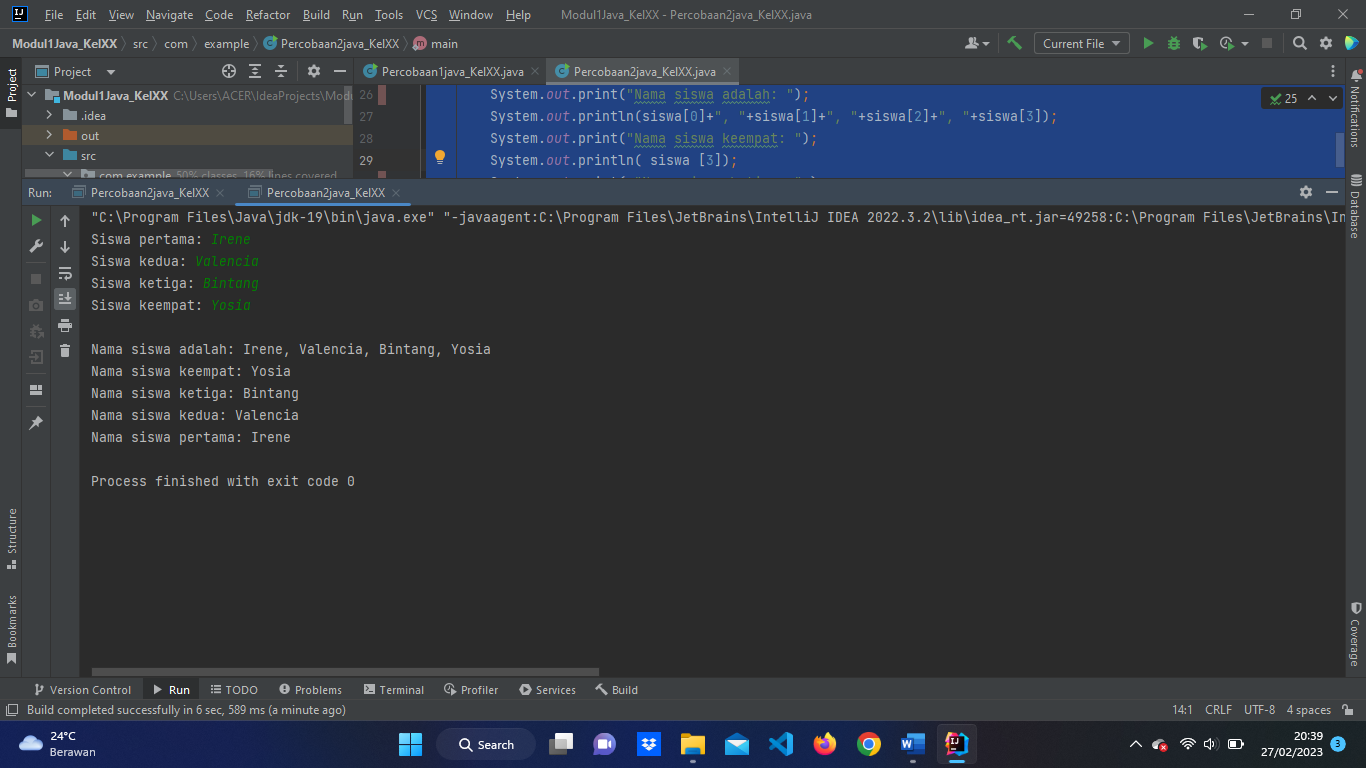


Gambar 2. 25 *Class Name* Percobaan2 Java

1. Masukkan *source code* berikut

|  |
| --- |
| package com.example;  import java.util.Scanner;  public class Percobaan2java\_KelXX {  public static void main(String[] args) {  Scanner input = new Scanner(System.in);  String[] siswa = new String[4];  System.out.print("Siswa pertama: ");  siswa[0] = input.nextLine();  System.out.print("Siswa kedua: ");  siswa[1] = input.nextLine();  System.out.print("Siswa ketiga: ");  siswa[2] = input.nextLine();  System.out.print("Siswa keempat: ");  siswa[3] = input.nextLine();  System.out.println();  System.out.print("Nama siswa adalah: ");  System.out.println(siswa[0]+", "+siswa[1]+", "+siswa[2]+", "+siswa[3]);  System.out.print("Nama siswa keempat: ");  System.out.println( siswa [3]);  System.out.print( "Nama siswa ketiga: " );  System.out.println( siswa[2] );  System.out.print( "Nama siswa kedua: " );  System.out.println( siswa[1] );  System.out.print( "Nama siswa pertama: " );  System.out.println( siswa[0] );  }  } |

1. Klik *Run* dan amati hasilnya



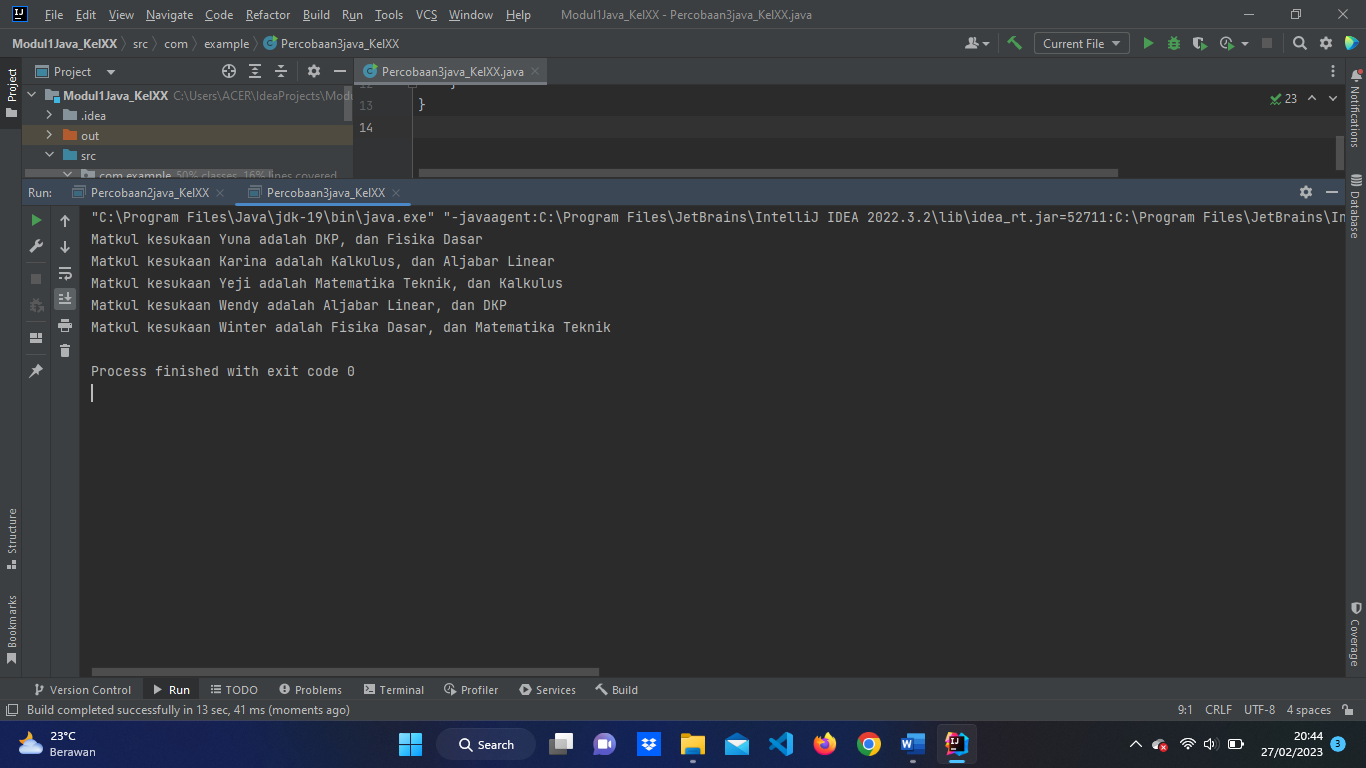
Gambar 2. 26 *Output* Percobaan 2 Java

Lalu bagaimana dengan array 2 dimensi? Lakukan percobaan sebagai berikut

1. Buat class baru beri nama Percobaan3java\_KelXX
2. Masukkan *Source Code* berikut

|  |
| --- |
| package com.example;  public class Percobaan3java\_KelXX {  public static void main(String[] args) {  String[][] array = {{"Yuna","Karina","Yeji","Wendy", "Winter"},{"DKP","Kalkulus","Fisika Dasar", "Aljabar Linear","Matematika Teknik",}};  System.out.println( "Matkul kesukaan " + array[0][0] + " adalah " + array[1][0] + ", dan " + array[1][2]);  System.out.println( "Matkul kesukaan " + array[0][1] + " adalah " + array[1][1] + ", dan " + array[1][3]);  System.out.println( "Matkul kesukaan " + array[0][2] + " adalah " + array[1][4] + ", dan " + array[1][1]);  System.out.println( "Matkul kesukaan " + array[0][3] + " adalah " + array[1][3] + ", dan " + array[1][0]);  System.out.println( "Matkul kesukaan " + array[0][4] + " adalah " + array[1][2] + ", dan " + array[1][4]);  }  } |

1. Run, lalu amati hasilnya



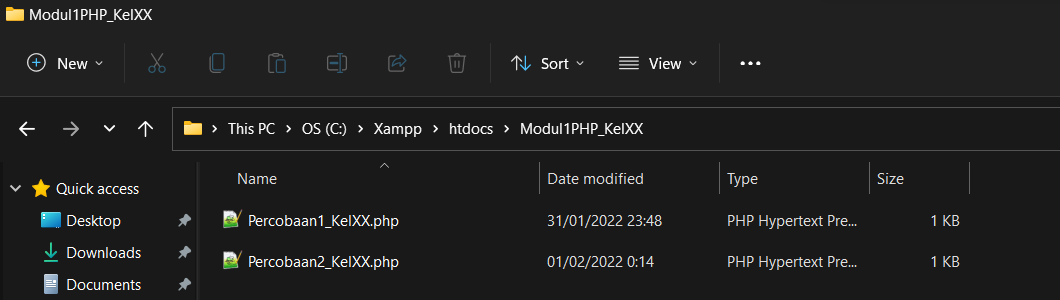
Gambar 2. 27 *Output* array 2 dimensi

### ***Array* pada PHP**

1. Buka Visual *studio code*
2. Klik *file*->*new file*
3. Masukkan *source code* berikut

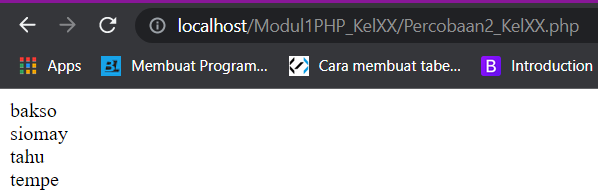
|  |
| --- |
| <?php  $makanan[0] = "bakso";  $makanan[1] = "siomay";  $makanan[2] = "tahu";  $makanan[3] = "tempe";  echo $makanan[0];  echo "<br>";  echo $makanan[1];  echo "<br>";  echo $makanan[2];  echo "<br>";  echo $makanan[3];  ?> |

1. Kemudian simpan *file* dengan nama Percobaan2\_KelXX.php dalam direktori C:/xampp/htdocs/Modul1PHP\_KelXX



Gambar 2. 28 *Save* Percobaan 2 PHP

1. Kemudian buka *browser web* kalian dan masukkan *address* localhost/Modul1PHP\_KelXX/Percobaan2\_KelXX.php amati hasilnya



Gambar 3. 29 *Output* Percobaan 2 PHP