**cara menulis / membuat method di java**

|  |
| --- |
| [**Daftar Materi Java**](https://alqaa.wordpress.com/2011/10/16/daftar-belajar-table-of-contents/) |

sebuah method menurut saya adalah kode yang menjelaskan sesuatu yang dapat dikerjakan oleh sebuah kelas / object, misalkan anda memiliki kelas Penjumlahan maka bisa diperkirakan salah satu methodnya adalah menjumlah(). langsung saja  
(kode tepat dibawah ini tidak bisa dicompile, hanya untuk menjelaskan saja)

* penulisan method adalah

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23 | tipeAkses TipeKembalian namaMethod(){}  atau  tipeAkses TipeKembalian namaMethod(tipeParameter1 parameter1, tipeParameter2 parameter2){}  tipeParameter bisa int, String, double, float, Object, Kelas, array  atau  final tipeAkses TipeKembalian namaMethod(){}  atau   tipeAkses static TipeKembalian namaMethod(){}  contoh1  public void menjumlah(){} -&gt;&gt; tipe kembali void==tidak kembali apa2  private String menjumlah(){} -&gt;&gt; mengembalikan String  int menjumlah(){} -&gt;&gt; mengembalikan integer    contoh2  public void menjumlah(int a, int b){} -&gt;&gt; dengan paramether  public String menjumlah(int a, int b){} -&gt;&gt; dengan parameter dan tipe kembali String    contoh3   public final void menjumlah(){} --&gt; dengan keyword final    contoh4  public static void main(String[] args) dengan keyword static,  tanpa tipe kembali dan parameter array dari String |

* method harus ditulis dalam kurung kurawal {}  kelas,  misalnya kita punya kelas dengan nama Penjumlahan dan method menjumlah() maka cara penulisannya adalah

|  |  |
| --- | --- |
| 1  2  3  4  5  6 | class Penjumlahan{       void menjumlah(){      }  } |

tidak boleh ditulis

|  |  |
| --- | --- |
| 1  2  3  4 | class Penjumlahan{  }  void menjumlah(){    } |

* method dapat diberi tipe akses (private,protected, public ) juga keyword final.  jika method diberi tipe akses public maka method ini dapat dipanggil dimana saja oleh object dari kelasnya, jika protected hanya bisa dipanggil oleh object dimana kode object berada dalam satu package / paket / folder dengan file kelasnya. jika private maka hanya bisa dipanggil di dalam kelasnya. langsung saja ke contoh pemanggilan method lewat object

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17 | class Penjumlahan{    //bisa private void menjumlah()  //bisa protected void menjumlah()  //bisa juga hanya void menjumlah() -> artinya tipe aksesnya default       public void menjumlah(){      }  public static void main(String[] args){    Penjumlahan objectPenjumlahan = new Penjumlanan(); //menciptakan object    objectPenjumlahan.menjumlah(); //memanggil method    //atau bisa juga langsung menciptakan object tanpa nama lalu langsung memanggil method  new Penjumlahan().menjumlah();  }  } |

pemanggilan method dilakukan di dalam kelas yang sama, jika dilakukan di kelas yang berbeda caranya pun sama dapat menggunakan **object dot namaMethod();** terdapat method main yang berfungsi sebagai method yang harus ada jika ingin file java dapat di eksekusi, penciptaan object  juga bisa dilakukan di method main yang jelas ketika ingin mengeksekusi sebuah file java haruslah ada method main.

* method memiliki tipe kembalian (return value) kita telah menulis method dengan return value **VOID**atau tidak mengembalikan apapun. bagaimana jika mengembalikan sesuatu misal kita buat method menjumlah() mengembalikan nilai berupa integer.

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7 | public int menjumlah(){  int i = 2;  int j = 3;  int c = 2+3;    return c;  } |

bisa dilihat ada kata**return**, maka dia akan mengembalikan nilai yang ada setelah kata return, dalam contoh return mengembalikan nilai c yaitu 5. Selain tipe kembalian void, perintah return harus ditulis, dan harus sesuai, jika ditulis String maka kembaliannya pun harus String, karena di contoh ditulis int maka kembaliannya pun harus int juga. bagaimana kalau method menjumlah() kita buat mengembalikan nilai String, sangat bisa!!! ini kodenya

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9 | public String menjumlah(){  int i = 2;  int j = 3;  int c = 2+3;    String s = "hasil dari 2 + 3 adalah "+c;    return s;  } |

maka method menjumlah() akan mengembalikan nilai String yaitu s. selanjutnya bagaimana penggunaan method yang me-**return value**selain void ? Nah… nilai dari return dapat digunakan untuk proses selanjutnya , contohnya adalah dimasukkan ke dalam variabel lain yang tipenya sama , misal method menjumlah() yang mengembalikan **int** dapat dipanggil sebagai berikut

Penjumlahan p = new Penjumlahan();

int a = p.menjumlah();

System.out.println(a);

kode diatas akan mencetak nilai a, nilai a adalah return value dari menjumlah() yaitu c = 5, maka akan dicetak 5;

selain int dan String, sebuah method juga dapat mengembalikan tipe yang lain.

* bila diberi kata2 **static**, maka pemanggilan method harus lewat kelas, bukan lewat object, jika method menjumlah dirubah menjadi
* public static int menjumlah(){
* int i = 2;
* int j = 3;
* int c = 2+3;
* return c;

}

maka pemanggilan method menjadi

Penjumlahan.menjumlah();

bukan melalui object seperti ini

Penjumlahan objectP = new Penjumlahan();

objectP.menjumlah();

//bukan juga dengan cara seperti ini

new Penjumlahan().menjumlah();