

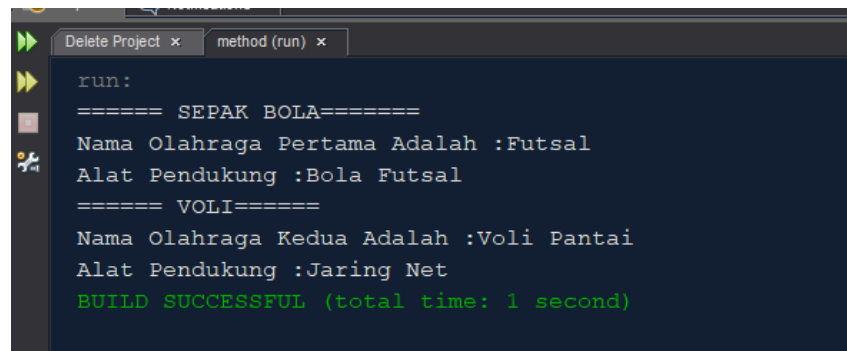
Nama : Miftakhuddin Falaki

NPM : 201843501551

Kelas : r5w

```
12 public class olahraga{
13     String nama, alat;
14
15     public static void main(String[] args){
16         // Membuat Objek dari Class olahraga
17         olahraga Bola = new olahraga();
18         System.out.println("===== SEPAK BOLA=====");
19         Bola.SepakBola();
20         System.out.println("===== VOLI=====");
21         Bola.Voli();
22     }
23     void SepakBola(){
24         nama = "Futsal";
25         alat = "Bola Futsal";
26         System.out.println("Nama Olahraga Pertama Adalah :"+nama);
27         System.out.println("Alat Pendukung :"+alat);
28     }
29     void Voli(){
30         nama = "Voli Pantai";
31         alat = "Jaring Net";
32         System.out.println("Nama Olahraga Kedua Adalah :"+nama);
33         System.out.println("Alat Pendukung :"+alat);
34     }
35 }
```

Output :



```
run:
===== SEPAK BOLA=====
Nama Olahraga Pertama Adalah :Futsal
Alat Pendukung :Bola Futsal
===== VOLI=====
Nama Olahraga Kedua Adalah :Voli Pantai
Alat Pendukung :Jaring Net
BUILD SUCCESSFUL (total time: 1 second)
```

### Penjelasan method void:

terdapat 2 buah method yang berbeda, masing-masing method mempunyai atribut yang berbeda, jadi method tersebut digunakan untuk mengkategorikan statement atau algoritma yang kita buat lalu memanggilnya pada method main menggunakan objek dari class.

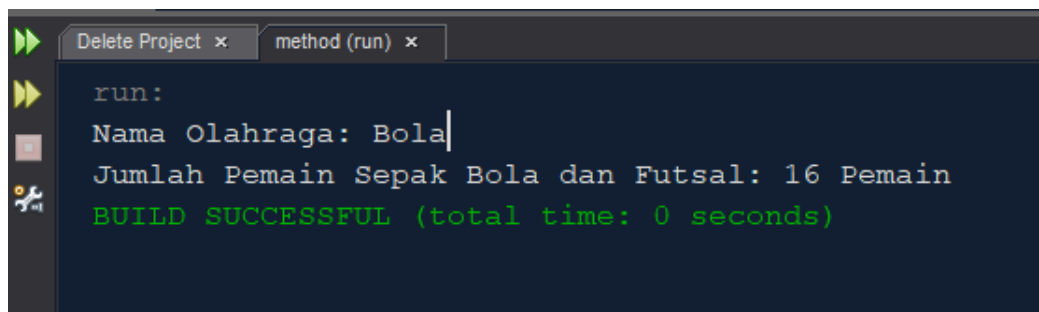
```

11
12 public class cont_ret {
13
14     String olahraga = "Bola";
15     int jml1 = 11, jml2 = 5, hasil;
16
17     String nama(){
18         //Mengembalikan Nilai olahraga
19         return olahraga;
20     }
21
22     int Jumlah(){
23         hasil = jml1 + jml2;
24         return hasil; //Mngembalikan Nilai dari Hasil
25     }
26     public static void main(String[] args){
27         // Membuat Objek dari Class olahraga
28         cont_ret data = new cont_ret();
29         System.out.println("Nama Olahraga: "+data.nama());
30         System.out.println("Jumlah Pemain Sepak Bola dan Futsal: "+data.Jumlah()+ " Pemain");
31     }
32 }

```

(Gambar cont\_ret)

Output :



```

Delete Project x method (run) x
run:
Nama Olahraga: Bola
Jumlah Pemain Sepak Bola dan Futsal: 16 Pemain
BUILD SUCCESSFUL (total time: 0 seconds)

```

### Penjelasan method return :

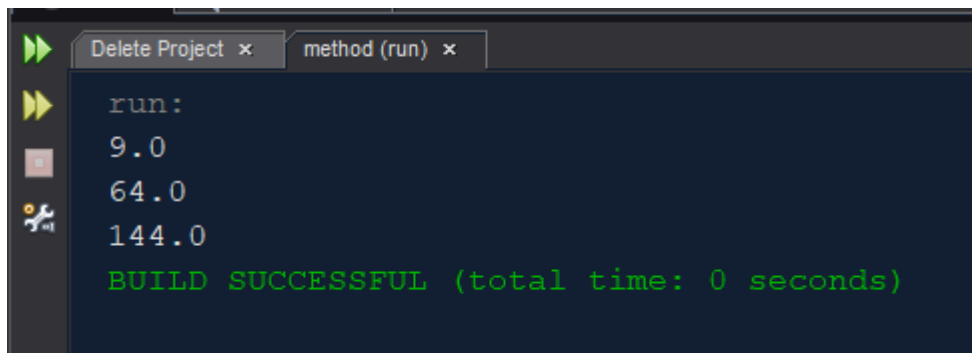
Intinya method return “mengembalikan nilai secara langsung atau sebuah nilai dari variable” algoritmanya ada di Gambar con\_ret

```

12 class stat1{
13     public static double nilaiku(double nilai){
14         //Mengembalikan Nilai nilai
15         return nilai*nilai;
16     }
17 }
18 public class stat{
19     public static void main(String[] args){
20         // Membuat Objek dari Class stat1
21         double akar=Math.sqrt(81);
22         double pangkat=Math.pow(8,2);
23         double kuadrat=stat1.nilaiku(12);
24         //menampilkan objek dari class stat1
25         System.out.println(akar);
26         System.out.println(pangkat);
27         System.out.println(kuadrat);
28     }
29 }
30

```

Output :



```

run:
9.0
64.0
144.0
BUILD SUCCESSFUL (total time: 0 seconds)

```

### Penejelasan method static :

Metode static yang melakukan kalkulasi normal untuk akar, pangkat dan kuadrat

```

10  * @author kikiKel
11  */
12  public class mainClass {
13      static void nama() {
14          System.out.println("Saya adalah salman");
15      }
16      void tampil() {
17          mainClass.nama();
18      }
19      public static void main(String[] args) {
20          mainClass main = new mainClass();
21          main.tampil();
22          mainClass.nama();
23      }
24  }

```

Output :

```

Delete Project x  method (run) x
run:
Saya adalah salman
Saya adalah salman
BUILD SUCCESSFUL (total time: 0 seconds)

```

### Penjelasan method static :

Dari output tersebut anda bisa memahami bahwa class dan object berbagi salinan yang sama dari variabel static, oleh karena itu mereka menampilkan hasil berupa nilai yang sama dari nama.

Ada 3 hal yang perlu diketahui dalam inisialisasi variabel static, yaitu:

- Variabel static akan diinisialisasi ketika class di-load.
- Variabel static dalam suatu class diinisialisasi sebelum objek apapun dari class tersebut dapat dibuat
- Variabel static dalam suatu class diinisialisasi sebelum method static apapun dalam class dijalankan.