

JAVA - Metot

- Diğer dillerdeki fonksiyon kavramına denktir. Bir nesne bir metot çağırdığında o metodun nesne üzerinde bir iş yapması beklenir; değer atama, bir değer döndürme, dosyaya yazma vb.
 - Metoda gönderilmesi gereken veriler parametre listesi olarak methodda tanımlanır.
 - Metotlar kodun yeniden kullanılabilirliğini sağlar.
-
-

Metot İmzası

- Bir sınıf içerisinde bir metodun adı ve aldığı parametre listesi ile birlikte eşsizdir. Parametre listesindeki parametrelerinin sırasının farklı olması da bir farklılıktır. Örneğin;
 - **int ornekMetot(int x, String y)** ile
 - **int ornekMetot(String y, int x)** farklı metotlardır.
-
-

- [modifiers] return_type method_name
(parameter_list) [throws_clause] {

[statement_list]
}

- **Return Type (Döndürülen Tip)** : Metodun döndürdüğü tip geçerli bir Java tipi olmalıdır, primitive(ilkel) veya sınıf, ya da Bir şey döndürmeyecekse void olmalıdır. Eğer Return Type tanımlanmışsa metod return komutuyla bitirilmelidir.
 - **Method Name (Metot Adı)**: Metot ismi Java dili için tanımlı kurallara uygun karakterlere ve biçime sahip olmalıdır.
-
-

- **Parameter List (Parametre Listesi):** Bir metodun parametere listesi boş olabilir. Eğer birden fazla parametre tanımlanacaksa virgülle ayrılmalıdır.
 - **Curly Braces (Küme Parantezi):** Metodun gövdesi yani çalıştırılan kod tanımlamalarının yapıldığı kısım.
 - **Modifiers (Niteleyiciler) :** (public, protected, private), static
-
-

Örnek

- Java ilkel tipler için **değer geçirme (pass by value)** ve nesne tipleri için **referans geçirme (pass by reference)** kullanılır. İstisnai olarak String nesneler de değer olarak geçirilmektedir.
 - Bu bilgiyi kullanarak örnek programın ekran çıktısını yazın.
-
-

PassByProgram.java

```
public class PassByProgram {  
    public static void main(String[] args) {  
        int i = 25;  
        System.out.println(i);  
        iMethod(i);  
        System.out.println(i);  
        Foo foo = new Foo();  
        System.out.println(foo.toString());  
        fooMethod(foo);  
        System.out.println(foo.toString());  
    }  
    public static void iMethod(int i) {  
        i = 9;  
        System.out.println(i);  
    }  
    public static void fooMethod(Foo foo) {  
        foo.toString();  
        System.out.println(foo.toString());  
    }  
}
```

Foo.java

```
public class Foo {  
    public String toString() {  
        double x = Math.random() * 10;  
        return "x= " + x;  
    }  
}
```



Package Explorer

Hierarchy

- Veri Yapıları - Hafta 2
 - src
 - (default package)
 - Foo.java
 - PassByProgram.java
 - JRE System Library [java-6-sun-1.6.0.00]

```
PassByProgram.java Foo.java
public class PassByProgram {
    public static void main(String[] args) {
        int i = 25;
        System.out.println(i);
        iMethod(i);
        System.out.println(i);
        Foo foo = new Foo();
        System.out.println(foo.toString());
        fooMethod(foo);
        System.out.println(foo.toString());
    }

    public static void iMethod(int i) {
        i = 9;
        System.out.println(i);
    }

    public static void fooMethod(Foo foo) {
        foo.toString();
        System.out.println(foo.toString());
    }
}
```

Task List

Find: All

Uncategorized

Outline

- PassByProgram
 - main(String[])
 - iMethod(int)
 - fooMethod(Foo)

Problems @ Javadoc Declaration Console

<terminated> PassByProgram [Java Application] /usr/lib/jvm/java-6-sun-1.6.0.00/bin/java (Oct 17, 2007 2:4

```
25
9
25
x= 9.14509727100988
x= 5.755646228947146
x= 4.674596039769141
```

```
public class PassByProgram {  
    public static void main(String[] args) {  
        int i = 25;  
        System.out.println(i);  
        iMethod(i);  
        System.out.println(i);  
        Foo foo = new Foo();  
        System.out.println(foo.toString());  
        fooMethod(foo);  
        System.out.println(foo.toString());  
    }  
    public static void iMethod(int i) {  
        i = 9;  
        System.out.println(i);  
    }  
    public static void fooMethod(Foo foo) {  
        foo.toString();  
        System.out.println(foo.toString());  
    }  
}
```



Confirm Perspective Switch



This kind of launch is configured to open the Debug perspective when it suspends.

This Debug perspective is designed to support application debugging. It incorporates views for displaying the debug stack, variables and breakpoint management.

Do you want to open this perspective now?

☐ Remember my decision

Yes

No



Debug Servers

PassByProgram [Java Application]

PassByProgram at localhost:54189

Thread [main] (Suspended (breakpoint at line 3 in PassByProgram))

PassByProgram.main(String[]) line: 3

/usr/lib/jvm/java-6-sun-1.6.0.00/bin/java (Oct 17, 2007 2:46:38 PM)

Variables Breakpoints

Name	Value
args	String[0] (id=15)

PassByProgram.java Foo.java

```
public class PassByProgram {  
    public static void main(String[] args) {  
        int i = 25;  
        System.out.println(i);  
        iMethod(i);  
        System.out.println(i);  
        Foo foo = new Foo();  
        System.out.println(foo.toString());  
        fooMethod(foo);  
        System.out.println(foo.toString());  
    }  
    public static void iMethod(int i) {
```

Outline

PassByProgram

- main(String[])
- iMethod(int)
- fooMethod(Foo)

Console Tasks

PassByProgram [Java Application] /usr/lib/jvm/java-6-sun-1.6.0.00/bin/java (Oct 17, 2007 2:46:38 PM)



Writable

Smart Insert

3 : 1

PassByProgram.java Foo.java

```
public class PassByProgram {  
    public static void main(String[] args) {  
        int i = 25;  
        System.out.println(i);  
        iMethod(i);  
        System.out.println(i);  
        Foo foo = new Foo();  
        System.out.println(foo.toString());  
        fooMethod(foo);  
        System.out.println(foo.toString());  
    }  
    public static void iMethod(int i) {
```

Debug

Servers



PassByProgram [Java Application]

PassByProgram at localhost:54189

Thread [main] (Suspended (breakpoint at line 3 in PassByProgram))

PassByProgram.main(String[]) line: 3

/usr/lib/jvm/java-6-sun-1.6.0.00/bin/java (Oct 17, 2007 2:46:38 PM)

(x)= Variables

Breakpoints

Name	Value
args	String[0] (id=15)

xxxx
xxxx
xxxx

Örnek 2

```
1
2 public class Veri {
3
4     private int[] veriler ;
5
6     public void setVeriler(int [] tmp){
7         veriler = tmp;
8     }
9
10    public static double ortalama(int[] gecici) {
11
12        double toplam = 0;
13        for (int deger : gecici) {
14            toplam = toplam + deger;
15        }
16
17        return toplam / gecici.length;
18    }
19
20    public double ortalama() {
21
22        double toplam = 0;
23        for (int deger : this.veriler) {
24            toplam = toplam + deger;
25        }
26
27        return toplam / this.veriler.length;
28    }
29 }
30
31
```



```
2 public class Program {
3
4     /**
5      * @param args
6      */
7     public static void main(String[] args) {
8         int[] degerler = new int[5];
9         degerler[0] = 3;
10        degerler[1] = 6;
11        degerler[2] = 12;
12        degerler[3] = 8;
13        degerler[4] = 5;
14
15        System.out.println(Veri.ortalama(degerler));
16
17        Veri denemeVeri = new Veri();
18
19        denemeVeri.setVeriler(degerler);
20
21        System.out.println(denemeVeri.ortalama());
22
23    }
24 }
25
26 }
27
```

Breakpoint – (Satır 12)

```
2 public class Veri {  
3  
4     private int[] veriler ;  
5  
6     public void setVeriler(int [] tmp){  
7         veriler = tmp;  
8     }  
9  
10    public static double ortalama(int[] gecici) {  
11  
12        double toplam = 0;  
13        for (int deger : gecici) {  
14            toplam = toplam + deger;  
15        }  
16  
17        return toplam / gecici.length;  
18    }
```

Veri.javaProgram.java

1
2 public class Veri {
3
4 private int[] veriler ;
5
6 public void setVeriler(int [] tmp){
7 veriler = tmp;
8 }
9
10 public static double ortalama(int[] gecici) {
11
12 double toplam = 0;
13 for (int deger : gecici) {
14 toplam = toplam + deger;
15 }
16
17 return toplam / gecici.length;
18 }
19
20 public double ortalama() {
21
22 double toplam = 0;
23 for (int deger : this.veriler) {
24 toplam = toplam + deger;
25 }

VariablesExpressions

Name	Value
gecici	int[5] (id=12)
toplam	3.0
deger	6