

بسم الله الرحمن الرحيم

آموزشگاه تحلیل داده

تخصصی ترین مرکز برنامه نویسی و دیتابیس در ایران

انتقال مقادیر چندگانه به متدها

گردآورنده : مهندس افشین رفوآ

انتقال مقادیر چندگانه به متدها

شما می توانید بیشتر از یک مقدار را به متدهای خود انتقال دهید. متدهای زیر را به `your MyMethods` `class` اضافه کنید.

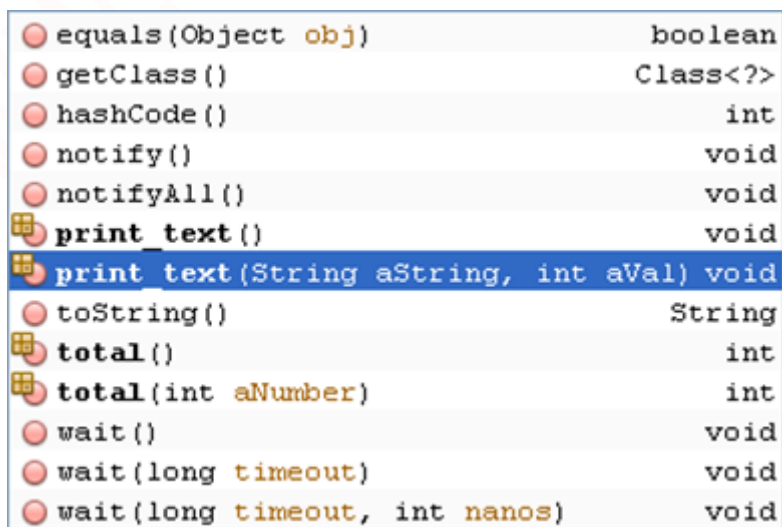
```
void print_text(String aString, int aVal) {  
    System.out.println( aString + aVal );  
}
```

تمام آنچه این متد انجام می دهد، چاپ کردن یک مورد می باشد. بین پیرانتزهای مربوط به نام متد، دو مقدار وجود دارد، یک متغیر `String` به نام `aString` و یک متغیر `int` به نام `aVal`. وقتی این متد را فرا می خوانیم، ابتدا نیاز به رشته و سپس نیاز به یک عدد داریم. سعی کنید آن را به روش دیگری انجام دهید و پیغام خطا دریافت خواهید کرد.

به گروه `TestMethods` خود بازگردید و فراخوانی زیر را به متد انجام دهید.

```
test1.print_text( "The value was ", aVal2 );
```

دوباره متد `print_text` باید در لیست `NetBeans` نمایش داده شود.



<code>equals(Object obj)</code>	<code>boolean</code>
<code>getClass()</code>	<code>Class<?></code>
<code>hashCode()</code>	<code>int</code>
<code>notify()</code>	<code>void</code>
<code>notifyAll()</code>	<code>void</code>
<code>print_text()</code>	<code>void</code>
<code>print_text(String aString, int aVal)</code>	<code>void</code>
<code>toString()</code>	<code>String</code>
<code>total()</code>	<code>int</code>
<code>total(int aNumber)</code>	<code>int</code>
<code>wait()</code>	<code>void</code>
<code>wait(long timeout)</code>	<code>void</code>
<code>wait(long timeout, int nanos)</code>	<code>void</code>

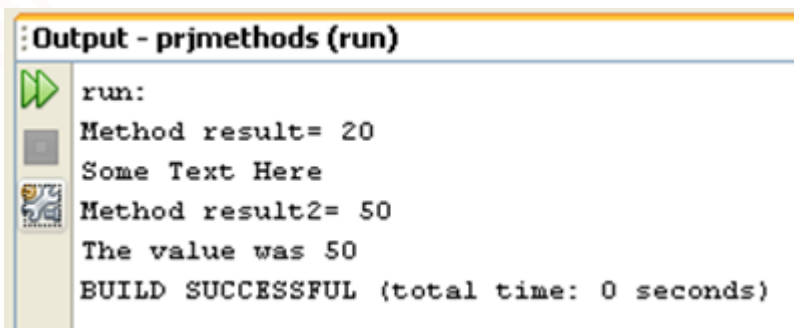
مقادیری (پارامترها) که تعیین می کنیم، همراه با متد بازگشتی، **void**، بین پرانتزها می باشند.

اما پنجره ی برنامه نویسی اصلی شما اکنون شبیه به تصویر می باشد.

```
public static void main(String[] args) {  
  
    MyMethods test1 = new MyMethods();  
  
    int aVal = test1.total();  
  
    System.out.println( "Method result= " + aVal );  
  
    test1.print_text();  
  
    int aVal2 = test1.total(30);  
    System.out.println( "Method result2= " + aVal2 );  
  
    test1.print_text("The value was ", aVal2);  
  
}
```

دو مقداری که انتقال می دهیم، توسط یک کاما مجزا می شوند. توجه داشته باشید که مقدار داخل **aVal2** انتقال داده شده باشد. به هر حال نام متغیر داخل پرانتزهای **print_text** در واقع **aVal** می باشد. گرچه نام های دو متغیر متفاوت می باشند، اما این مسئله مشکلی ایجاد نمی کند. آنچه شما انجام می دهید انتقال مقادیر به متد می باشد. بنابراین متغیر **aVal** با همان مقدار **aVal2** به پایان می رسد.

اکنون اگر برنامه ی خود را اجرا کنید، باید پنجره ی **Output** زیر را مشاهده کنید.



```
Output - prjmethods (run)  
run:  
Method result= 20  
Some Text Here  
Method result2= 50  
The value was 50  
BUILD SUCCESSFUL (total time: 0 seconds)
```

ما متدها را رها می کنیم. همانطور که پیش می رویم، متدهای بیشتری خواهیم نوشت، بنابراین شما باید زیاد تمرین کنید. بخش بعد در مورد گروه ها می باشد – برنامه نویسی آبجکت محور.

www.tahildadeh.com