

آموزش متغیرهای جاوا

برنامه ها با دستکاری داده ی واقع در حافظه کار می کنند. داده می تواند عدد، متن، آبجکت، اشاره گرهایی به دیگر بخش های حافظه، و موارد دیگر باشد. داده نام گذاری می شود، طوری که در صورت نیاز می تواند مجدداً فراخوانده شود. نام و مقدار آن با عنوان متغیر (Variable) شناخته می شوند. ما با مقادیر عددی شروع می کنیم.

برای ذخیره ی یک عدد در جاوا، باید گزینه های زیادی داشته باشید. تمام اعداد مانند 8، 10، 12 و غیره با استفاده از متغیر `int` ذخیره می شوند (`int`). مخفف `integer` (عدد صحیح) می باشد. اعداد ممیز شناور مانند 8.4، 10.5، 12.8 و غیره با استفاده از متغیر دوگانه ذخیره می شوند. شما ذخیره سازی را با یک علامت تساوی (=) انجام می دهید. اجازه بدهید چند مثال را بررسی کنیم (می توانید از کد `FirstProject` برای این مثال ها استفاده کنید).

برای تنظیم یک عدد کامل (عدد صحیح)، موارد زیر را به متود `main` از بخش قبل اضافه کنید:

```
public static void main(String[ ] args) {  
  
    int first_number;  
  
    first_number = 10;  
  
    System.out.println("My First Project");  
  
}
```

بنابر این این برنامه به جاوا می گوید که می خواهید مقداری از 10 را در متغیر عدد صحیح `first_number` نامیده ایم، ذخیره کنید.

اگر تمایل دارید، می توانید تمام این کار را در یک خط انجام دهید:

```
public static void main(String[ ] args) {  
  
    int first_number = 10;  
  
    System.out.println("My First Project");  
  
}
```

برای اینکه تمام موارد را در عمل ببینید، متود `println` را به این شکل تغییر دهید:

```
System.out.println("First number = " + first_number );
```

آنچه ما اکنون در بین آکولادهای `println` داریم، متن های مستقیمی هستند که در علامت نقل قول محدود شده اند:

```
("First number = "
```

آدرس آموزشگاه : تهران - خیابان شریعتی - بالا تر از خیابان ملک - جنب بانک صادرات - پلاک 651 طبقه دوم - واحد 7

88146323 - 88446780 - 88146330

ما یک علامت به علاوه نیز داریم که با نام متغیرمان دنبال می شود:

```
+ first_number );
```

علامت به اضافه به معنای اتصال به یکدیگر "join together" می باشد. بنابراین ما متن و نام متغیر خود را به یکدیگر متصل می کنیم. این عملکرد به عنوان concatenation الحاق (مطرح است).

پنجره ی شما اکنون باید مانند زیر به نظر برسد (دقت داشته باشید که هر خط با نقطه ویرگول به پایان می رسد):

```
package firstproject;

public class FirstProject {

    public static void main(String[] args) {

        int first_number;
        first_number = 10;

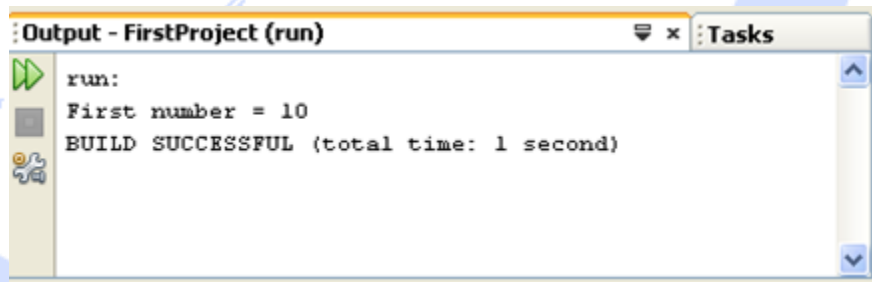
        System.out.println("First number = " + first_number );

    }

}
```

برنامه ی خود را اجرا کنید و باید صفحه ی زیر را در پنجره ی Output در پایین مشاهده کنید:

بنابراین عددی را که در متغیر ذخیره کرده ایم، به نام first_number، خروجی می باشد پس از علامت تساوی.



```
Output - FirstProject (run)
run:
First number = 10
BUILD SUCCESSFUL (total time: 1 second)
```

اجازه بدهید چند جمع ساده را امتحان کنیم. چند متغیر int به کد خود اضافه کنید، یکی برای ذخیره ی رومین عدد و دیگری برای ذخیره ی پاسخ.

```
int first_number, second_number, answer;
```

دقت کنید که چگونه سه متغیر نام روی یک خط داریم. اگر متغیرها از یک نوع باشند، شما می توانید این کار را در جاوا انجام دهید (برای مورد ما نوع int). بنابراین این نام هر متغیر با کاما (ویرگول) جدا می شود.

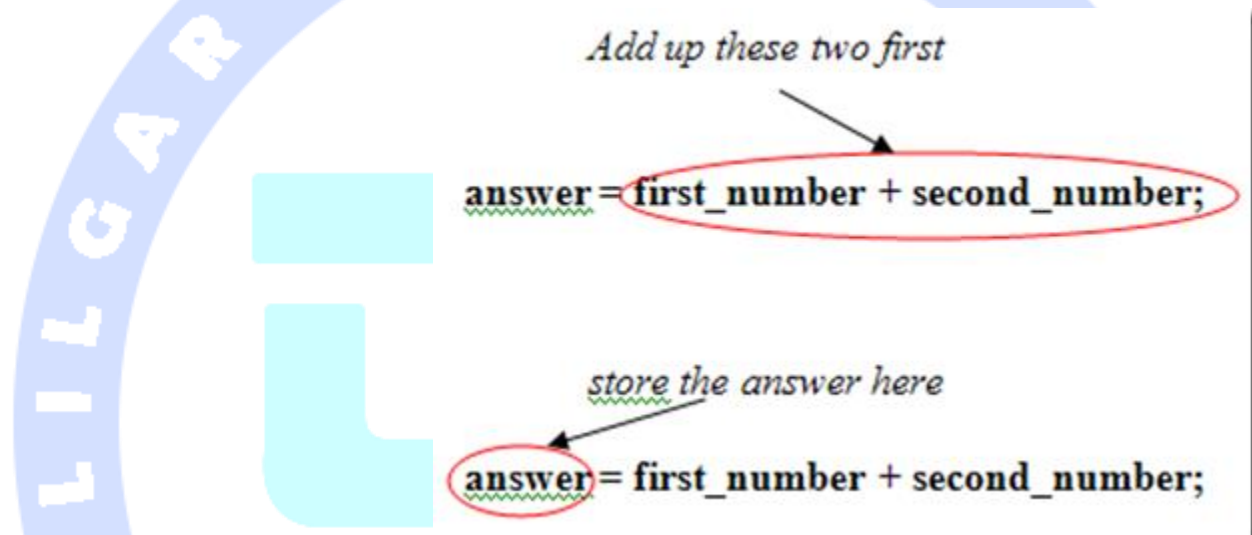
آدرس آموزشگاه : تهران - خیابان شریعتی - بالا تر از خیابان ملک - جنب بانک صادرات - پلاک 651 طبقه دوم - واحد 7

88146323 - 88446780 - 88146330

بنابراین می توانیم در متغیرهای جدید چیزی اضافه کنیم:

```
first_number = 10;  
second_number = 20;  
answer = first_number + second_number;
```

برای متغیر پاسخ، می خواهیم اولین عدد را به دومین عدد اضافه کنیم. جمع کردن با علامت به اضافه (+) انجام می شود. بنابراین در Java مقادیر را در `first_number` و `second_number` به یکدیگر اضافه می کند. وقتی این کار تمام شد، مجموع را در متغیر واقع در سمت راست تساوی ذخیره می کند. بنابراین به جای اختصاص دادن 10 یا 20 به نام متغیر، عمل جمع را انجام خواهد داد و سپس اختصاص می دهد. در موردی که واضح نیست، اتفاقی مانند زیر رخ خواهد داد:



مورد بالا سازگار با این مورد می باشد:

```
answer = 10 + 20;
```

اما جاوا تقریباً می داند که در داخل دو متغیر `first_number` و `second_number` چه چیزی وجود دارد، بنابراین شما می توانید فقط از نام ها استفاده کنید.

اکنون متود `println` خود را مانند زیر تغییر دهید:

```
System.out.println("Addition Total = " + answer );
```

مجدداً یک متن مستقیم را که با علامت نقل قول احاطه شده، با نام متغیر ترکیب می کنیم. پنجره ی برنامه نویسی شما باید مانند زیر باشد:

آدرس آموزشگاه : تهران - خیابان شریعتی - بالا تر از خیابان ملک - جنب بانک صادرات - پلاک 651 طبقه دوم - واحد 7

88146323 - 88446780 - 88146330

<http://www.tahlildadeh.com/>

```

public static void main(String[] args) {

    int first_number, second_number, answer;

    first_number = 10;
    second_number = 20;
    answer = first_number + second_number;

    System.out.println("Addition Total = " + answer );
}

```

Output - FirstProject (run)

```

run:
Addition Total = 30
BUILD SUCCESSFUL (total time: 1 second)

```

بنابراین برنامه ی ما موارد زیر را انجام داده است:

یک عدد ذخیره کرده است. یک عدد دوم ذخیره کرده است. این دو عدد را با یکدیگر جمع کرده است. نتیجه ی جمع را در یک متغیر سوم ذخیره کرده است. نتیجه را چاپ کرده است. شما مستقیماً نیز می توانید از اعداد استفاده کنید. خط پاسخ را مانند زیر تغییر دهید: `answer = first_number + second_number + 12`; برنامه ی خود را مجدداً اجرا کنید. چه چیزی چاپ شده است؟ همان موردی است که انتظار داشتید؟ می توانید اعداد کاملاً بزرگی در متغیر نوع `int` ذخیره کنید. حداکثر مقدار 2147483647 می باشد. اگر یک عدد منفی مورد نظر شماست، کمترین مقداری که می توانید داشته باشید -2147483648 می باشد. اگر اعداد بزرگتر و یا کوچکتر می خواهید، می توانید از متغیر عددی دیگری استفاده کنید: `double`، که آنها را در بخش بعدی مورد بررسی قرار می دهیم.



آدرس آموزشگاه : تهران - خیابان شریعتی - بالا تر از خیابان ملک - جنب بانک صادرات - پلاک 651 طبقه دوم - واحد 7

88146323 - 88446780 - 88146330

<http://www.tahlildadeh.com/>