

بسم الله الرحمن الرحيم

## آموزشگاه تحلیل داده

تخصصی ترین مرکز برنامه نویسی و دیتابیس در ایران

### آموزش عبارات if

مدرس : مهندس افشین رفوآ

## آموزش عبارات if

برنامه نویسی که در حال حاضر در حال انجام آن هستید یک برنامه نویسی پی در پی می باشد، به این معنا که کد از بالا به پایین اجرا می شود. این برنامه نویسی خطی می باشد که در آن هر خط با شروع از اولین خط و پایان آخرین خط از کد خوانده خواهد شد.

اما همیشه تمایل ندارید برنامه هایتان اینگونه اجرا شوند. اغلب تمایل دارید کدها تنها تحت شرایط خاص اجرا شوند. برای مثال ممکن است بخواهید یک پیام در صورتی نمایش داده شود که یوزر کمتر از 18 سال سن داشته باشد و یا بالعکس پیام خاصی تنها در صورتی نمایش داده شود که یوزر بیشتر از 18 سال سن داشته باشد. شما می خواهید که جریان برنامه نویسی را برای خود کنترل کنید. این کار را می توانید با منطق شرطی انجام دهید.

منطق شرطی اساسا در مورد لغت IF می باشد: اگر یوزر کمتر از 18 سال سن دارد این پیغام را نمایش دهید، اگر یوزر بیشتر از 18 سال سن دارد، این پیغام را نمایش بده. خوشبختانه استفاده از منطق شرطی در جاوا ساده می باشد. اجازه بدهید با عبارات IF آغاز کنیم.

### عبارات IF (IF Statements)

در برنامه نویسی که عبارت IF وجود دارد، اجرای یک مورد به جای مورد دیگر در هنگام اجرای کد بسیار متداول است. ساختار عبارت IF در جاوا به شکل زیر است:

```
if ( Statement ) {  
  
}
```

عبارت را با لغت if (با حروف کوچک) و یک جفت آکولاد شروع می کنید. این کد، کدی می باشد که تنها زمانی می خواهید اجرا کنید که شرایط شما برقرار باشد. شرایط مربوط در داخل آکولادها قرار می گیرد:

```
if ( user < 18 ) {  
}
```

این شرط به این معناست " اگر یوزر کمتر از 18 سال سن دارد." اما به جای عبارت کمتر از نشان ( < ) استفاده کرده ایم. اگر یوزر کمتر از 18 سال باشد، می خواهیم که برنامه ای اجرا شود، برای مثال پیغامی نمایش داده شود:

```
if ( user < 18 ) {  
  
    //DISPLAY MESSAGE  
  
}
```

اگر یوزر کمتر از 18 سال نباشد، پیغام بین آکولادها اجرا نخواهد شد و برنامه به مسیر خود، به سمت خطوط پایین کد، ادامه خواهد داد. هر آنچه در داخل آکولادها تایپ می کنید، تنها در صورتی اجرا خواهند شد که شرایط درست باشند. این شرایط بین آکولادها قرار می گیرند.

قبل از امتحان این مورد، نماد تندنویسی دیگر > می باشد، این نماد به معنای بزرگتر از می باشد. عبارت IF در بالا برای یوزرهای بزرگتر از 18 سال قابل اصلاح می باشد:

```
if ( user > 18 ) {  
  
    //DISPLAY MESSAGE  
  
}
```

تنها مورد جدید در این کد نماد > می باشد. اکنون شرایط مربوط به یوزرهای بالاتر از 18 سال می باشد.

اما شرایط برای یوزرهایی با سن دقیقاً 18 سال درست نیست. اگر می خواهید شرایط در مورد افراد 18 سال یا بالای 18 سال درست باشد، می توانید از عبارت بزرگتر یا مساوی استفاده کنید. نمادها برای این شرایط بزرگتر از ( > ) و مساوی با ( = ) می باشند.

```
if ( user >= 18 ) {  
  
    //DISPLAY MESSAGE  
  
}
```

همچنین می توانید این شرایط را برای کوچکتر یا مساوی با به روش زیر چک کنید:

```

if ( user <= 18 ) {

    //DISPLAY MESSAGE

}

```

کد بالا حاوی یک نماد ( < ) می باشد که با علامت مساوی دنبال می شود.

اجازه بدهید این مورد را در یک برنامه ی ساده امتحان کنیم.

یک پروژه ی جدید را با کلیک بر روی **File > New Project** از نوار منو در **NetBeans** آغاز کنید. شما می توانید پوشه و گروه خود را با هر نامی که می خواهید نام گذاری کنید. کد زیر را وارد کنید ( نام پوشه ی ما **conditionallogic** و گروه ما نیز **IFStatements** نامیده می شود):

```

package conditionallogic;

public class IFStatements {

    public static void main(String[] args) {

        int user = 17;

        if (user < 18) {
            System.out.println("User is less than 18");
        }

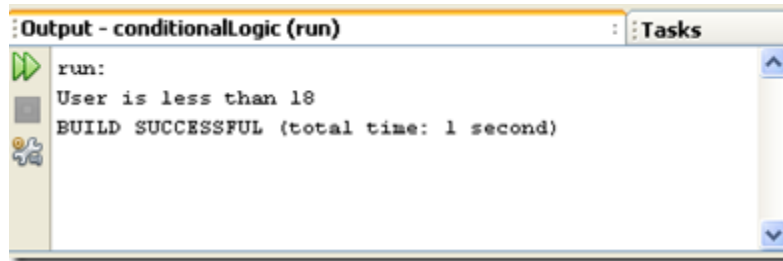
    }

}

```

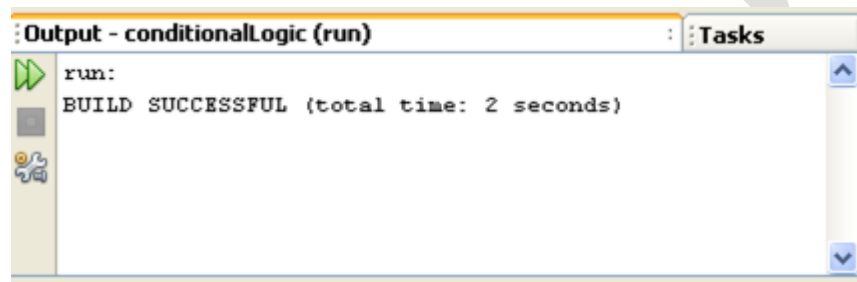
یک متغیر صحیح را تنظیم کرده ایم و مقداری برابر **17** به آن اختصاص داده ایم. عبارت **IF** مورد " کمتر از **18** سال" را چک می کند. بنابراین پیغام بین آگولادها باید چاپ شود.

برنامه ی خود را اجرا کرده و آن را بررسی کنید. (**NetBeans** معمولاً برنامه را در متن بولد شده در پنجره ی **Projects** اجرا می کند و نه آن کدی که شما نمایش داده اید. برای اجرای کد در پنجره ی **coding** در قسمتی از کد کلیک راست کنید. از منوی ظاهر شده **Run File** را انتخاب کنید). تصویر زیر را در پنجره ی **Output** باید مشاهده کنید:



```
Output - conditionalLogic (run) | Tasks
run:
User is less than 18
BUILD SUCCESSFUL (total time: 1 second)
```

اکنون مقدار را برای متغیر یوزر از 17 به 18 تغییر دهید. برنامه ی خود را مجددا اجرا کنید. باید تصویر زیر را مشاهده کنید:



```
Output - conditionalLogic (run) | Tasks
run:
BUILD SUCCESSFUL (total time: 2 seconds)
```

بنابراین برنامه به خوبی و بدون پیغام خطا اجرا خواهد شد. اینجاست که چیزی چاپ نمی شود. دلیل آن این است که کد پیغام بین آکولادهای عبارت **IF** وجود دارد و عبارت **IF** در حال بررسی مقادیر کمتر از 18 می باشد. اگر شرایط برقرار نباشد، جاوا آکولادها را کلاً نادیده گرفته و به کار خود ادامه می دهد.

تمرین:

نماد "کوچکتر از یا مساوی با" را با جایگزین نماد کوچکتر از کنید. پیغام خود را طوری تغییر دهید تا با مورد "یوزر 18 سال یا کمتر از 18 سال است" هماهنگ باشد. برنامه ی خود را مجددا اجرا کنید. آیا پیغام را مشاهده می کنید؟

تمرین:

مقدار یوزر را به 20 تغییر دهید. برنامه ی خود را مجددا اجرا کنید. آیا هنوز پیغام را مشاهده می کنید؟

شما می توانید بیشتر از یک عبارت **IF** در کد خود داشته باشید. کد زیر را امتحان کنید:

```

public static void main(String[] args) {

    int user = 18;

    if (user <= 18) {
        System.out.println("User is 18 or younger");
    }

    if (user > 18) {
        System.out.println("User is older than 18");
    }

}

```

این بار دو عبارت **IF** داریم. اولین عبارت مقادیر کمتری از یا برابر با **18** را بررسی می کند. دومین عبارت مقادیر بزرگتر از **18** را بررسی می کند. وقتی که کد با مقداری از **18** یا کمتر از **18** برای تغییر یوزر اجرا می شود، خروجی مانند زیر می باشد:

```

Output - conditionalLogic (run)
run:
User is 18 or younger
BUILD SUCCESSFUL (total time: 0 seconds)

```

تغییر مقدار متغیر یوزر به **20** نیز نتیجه ی زیر را به دنبال خواهد داشت:

```

Output - conditionalLogic (run)
run:
User is older than 18
BUILD SUCCESSFUL (total time: 0 seconds)

```

بنابراین تنها یکی از عبارت های **IF** خروجی خط چاپی خواهد داشت و تمام آن بستگی به مقدار متغیر یوزر دارد.  
در قسمت بعدی با **Conditional Logic** ادامه خواهیم داد.

Tahildadeh