

آموزش Switch در جاوا

روش دیگر برای کنترل جریان برنامه های شما switch statement می باشد. یک switch statement به شما گزینه ی تست دامنه ای از مقادیر را برای متغیر هایتان ارائه می دهد. این عبارت می تواند به جای عبارت بلند و پیچیده ی if ... else استفاده شود. ساختار عبارت switch مانند زیر می باشد:

```
switch ( variable_to_test ) {  
  
    case value:  
        code_here;  
        break;  
  
    case value:  
  
        code_here;  
  
        break;  
  
    default:  
  
        values_not_caught_above;  
  
}
```

بنابراین شما با لغت switch شروع کرده اید که با یک جفت آکولاد دنبال می شود. تمام دیگر بخش های این عبارت در بین آکولادها قرار می گیرند. متغیری که می خواهید چک کنید بین آکولادهای switch قرار می گیرد. بنابراین شما یک جفت آکولاد دارید. بخش های دیگر عبارت switch نیز بین دو آکولاد قرار می گیرد. برای هر مقداری که می خواهید چک کنید نیاز به لغت case دارید. سپس مقداری را دارید که قصد بررسی آن را دارید:

case value:

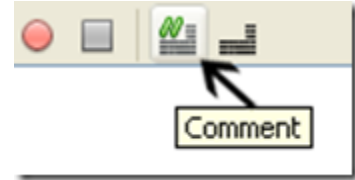
پس از case value: دو نقطه قرار می گیرد. سپس اگر مقدار هماهنگ باشد، آن چیزی را قرار می دهید که تمایل دارید اتفاق بیفتد. این کد شماست که می خواهید اجرا شود. برای گریز از هر مورد در عبارت switch، لغت break لازم است.

مقدار پیش فرض در انتها انتخابی می باشد. اگر مقادیر دیگری وجود دارند که می توانند در متغیر شما قرار بگیرند اما شما برای جای دیگری در عبارت switch امتحان نکرده اید، این مقادیر می توانند شامل شوند.

اگر همه ی آن گیج کننده می باشد در اینجا کد دیگری را مشاهده می کنید. می توانید برای این یا یک پروژه ی جدید ایجاد کنید و یا اینکه فقط روی کدی که دارید کامنت بدهید. یک راه سریع برای کامنت گذاشتن بر روی کد در NetBeans از نوار ابزار در بالا می باشد. ابتدا کدی را که می خواهید کامنت بگذارید مشخص کنید. سپس روی آیکن کامنت کلیک کنید:

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

88146323 - 88446780 - 88146330



اما در اینجا کد آن را مشاهده می کنید:

```
public static void main(String[] args) {  
  
    int user = 18;  
  
    switch ( user ) {  
        case 18:  
            System.out.println("You're 18");  
            break;  
        case 19:  
            System.out.println("You're 19");  
            break;  
        case 20:  
            System.out.println("You're 20");  
            break;  
        default:  
            System.out.println("You're not 18, 19 or 20");  
    }  
}
```

اولین کاری که کد انجام می دهد تنظیم مقداری برای امتحان کردن می باشد. مجدداً ما یک متغیر صحیح را تنظیم کرده و آن را user می نامیم. مقدار را بر روی 18 تنظیم کرده ایم. عبارت switch متغیر یوزر را چک کرده و آنچه در آن است را مشاهده خواهد کرد. سپس از طریق هر یک از عبارات مورد وارد خواهد شد. وقتی موردی را پیدا می کند که هماهنگ می باشد، متوقف خواهد شد و کد را برای آن مورد اجرا خواهد کرد. سپس از عبارت switch گریز خواهد داشت.

برنامه را امتحان کنید. مقادیر مختلفی برای متغیر یوزر وارد کرده و نتیجه را مشاهده کنید.

متأسفانه نمی توانید برای دامنه ای از مقادیر بعد از مورد امتحان کنید و تنها برای یک مقدار میسر می باشد. بنابراین نمی توانید این کار را انجام دهید:

```
case (user <= 18):
```

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/>

اما این کار را می توانید انجام دهید:

```
case 1: case 2: case 3: case 4:
```

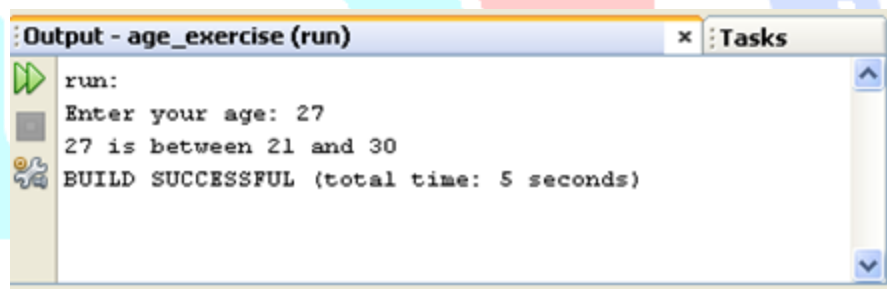
بنابراین خط بالا دامنه ی مقادیر را از 1 تا 4 تست می کند. اما شما باید هر مقدار را "spell out" کنید. (دقت داشته باشید که همه ی موردها و colon ها (دو نقطه) کجا قرار می گیرند).

در اتمام این بخش در مورد منطق شرطی، تمرین های زیر را امتحان کنید:

برنامه ای بنویسید که ورودی یوزر را از console قبول می کند. برنامه ها باید یک عدد گرفته و سپس محدوده های سنی زیر را امتحان کنید: 0 تا 10، 11 تا 20، 21 تا 30، 30 و بیشتر. یک پیغام در پنجره ی Output به فرمت زیر نمایش دهید:

```
user_age + " is between 21 and 30"
```

بنابراین اگر یوزر 27 را به عنوان سن وارد کند، پنجره ی Output مانند زیر خواهد بود:



اگر یوزر 30 یا بیشتر از 30 باشد، فقط می توانید پیغام زیر را نمایش دهید:

```
"Your are 30 or over"
```

کمک برای این تمرین:

برای دریافت مقادیر رشته از یوزر، این کار را انجام دادید:

```
String age = user_input.next( );
```

اما متود () next برای رشته ها استفاده می شود. سنی که از یوزر دریافت می کنید باید یک عدد صحیح باشد، بنابراین نمی توانید از () next استفاده کنید.

اگر می خواهید چک کنید که یک رشته مشابه رشته ی دیگر می باشد، می توانید از متودی به نام equals استفاده کنید.

```
String user_name = "Bill";
```

آدرس آموزشگاه : تهران - خیابان شریعتی - بالا تر از خیابان ملک - جنب بانک صادرات - پلاک 651 طبقه دوم - واحد7

88146323 - 88446780 - 88146330

```
if ( user_name.equals( "Bill" ) ) {  
  //DO SOMETHING HERE  
}
```

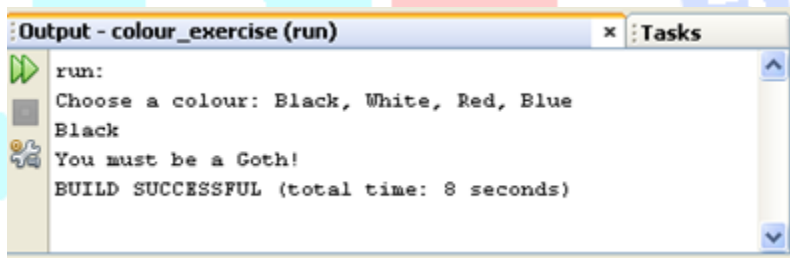
در کد بالا یک متغیر String را تنظیم کرده و آن را user_name نامیده ایم. سپس مقدار "Bill" به آن اختصاص داده ایم. بین آکولادهای IF، مجدداً نام متغیر داریم که با یک نقطه دنبال می شود. پس از این نقطه لغت "equals" قرار می گیرد. بین یک جفت آکولاد دیگر رشته ای را تایپ می کنید که سعی در امتحان آن دارید.

نکته: در هنگام بررسی اگر یک رشته همانند رشته ی دیگر می باشد، این دو رشته باید دقیقاً هماهنگ باشند. بنابراین "Bill" با "bill" متفاوت می باشد. مورد اول دارای B بزرگ و مورد دوم دارای b کوچک می باشد.

برای این تمرین برنامه ای بنویسید که از یک بوزر می خواهد تا از بین چهار رنگ انتخاب کند: سیاه، سفید، قرمز یا آبی. بسته به رنگ انتخاب شده، از عبارت های IF ... ELSE IF برای نمایش یکی از پیغام های زیر استفاده کنید:

"BLACK شما باید یک Goth باشید" WHITE " شما یک فرد پاک هستید" RED " شما یک شخص سرگرم کننده و برون گرا هستید" BLUE " شما طرفدار چلسی هستید، اینطور نیست؟ "

وقتی برنامه ی شما به پایان می رسد، پنجره ی Output باید شبیه به تصویر زیر باشد:



بسیار خوب، اجازه بدهید ادامه داده و نگاهی به loop داشته باشیم. در بخش بعد سرعت را کمی بالا می بریم.



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/>