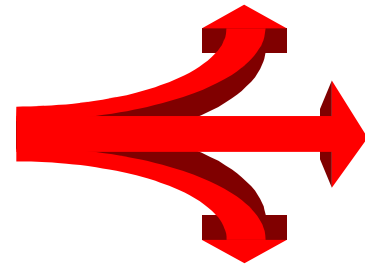


ifS



**What is
an if?**

the if statement

**if I am hungry
I eat something**



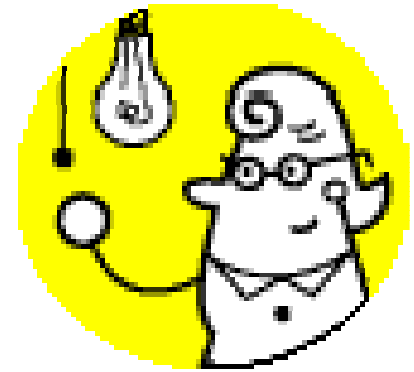
**if I am tired
I go to sleep**



the if statement

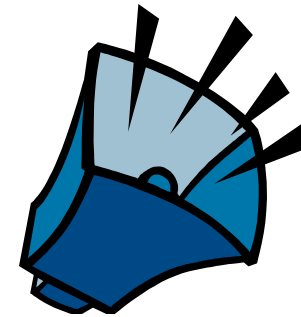
if it is dark

I turn a light on



if I can't hear the song

I make it louder



Logic



Relational

frequently used operators

Operator	Use
$x == y$	checks if x and y have the same value
$x > y$	checks if x is greater than y
$x < y$	checks if x is less than y
$x >= y$	checks if x is greater than or equal to y
$x <= y$	checks if x is less than or equal to y
$x != y$	checks if x is not equal to y

boolean

A boolean is any condition or variable that can be evaluated to true or false.

10 == 10

boolean isOdd = true;

boolean isEven = false;

The if statement!



If Definition

An if statement is a block of code that is associated with a condition. The block of code may execute once or not at all depending on the evaluation of the condition.



the if statement

```
if ( boolean condition placed here )  
{  
    do something 1;  
    do something 2;  
}
```



the if statement

```
int one = 109;  
if(one < 100)  
{  
    System.out.println("one < 100");  
}
```

```
if(one > 100)  
{  
    System.out.println("one > 100");  
}
```

OUTPUT

one > 100

the if statement

```
int uilScore = 240;
if(uilScore==240)
{
    System.out.println("state champ");
}
if(uilScore<100)
{
    System.out.println("work harder");
}
```

OUTPUT

state champ

the if statement

```
String stringOne = "big";  
if(stringOne.equals("it"))  
{  
    System.out.println("== it");  
}
```

OUTPUT
== big

```
if(stringOne.equals("big"))  
{  
    System.out.println("== big");  
}
```

the if statement

```
boolean isOdd = true;  
if(isOdd == true)  
{  
    System.out.println("isodd");  
}
```

```
if(isOdd == false)  
{  
    System.out.println("iseven");  
}
```

OUTPUT

isodd



nesting ifs

```
int num=7;  
if(num>2)  
{  
    if(num<10)  
        System.out.println(">2<10");  
    if(num>10)  
        System.out.println(">2>10");  
}
```

OUTPUT

>2<10



COMMON ERRORS

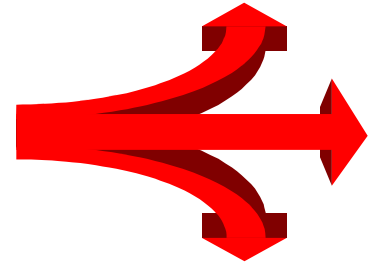
```
If(total >= 25);
```

```
{  
{
```

```
if(total = 10)
```

```
{  
{
```





if-else Strings

**What is
an
if else?**

the if-else statement

if I am tired
I go to sleep
else
I go for a run



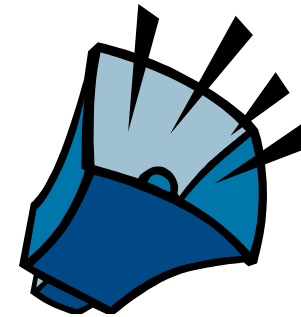
the if-else statement

if I like the current song

I make it louder

else

I change the song



the if-else statement

```
if( boolean condition placed here )  
{  
    do something 1;  
}  
else  
{  
    do something 2;  
}
```

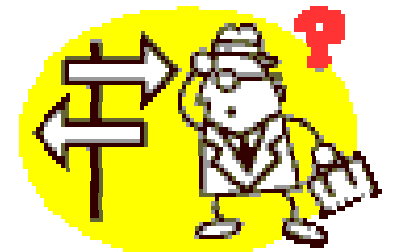


the if-else statement

```
int num=990;  
if(num>100)  
{  
    System.out.println("> 100!");  
}  
else  
{  
    System.out.println("! > 100!");  
}
```

OUTPUT

> 100!



the if-else statement

```
int num=50;  
if(num>100)  
{  
    System.out.println("> 100!");  
}  
else  
{  
    System.out.println("! > 100!");  
}
```

OUTPUT

! > 100!

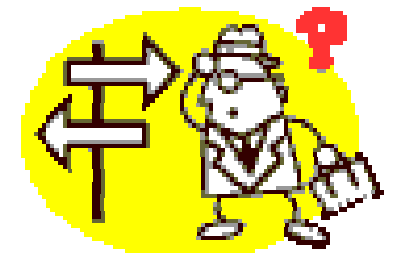


the if-else statement

```
int num=100;  
if(num>=100)  
{  
    System.out.println(">= 100!");  
}  
else  
{  
    System.out.println("! >= 100!");  
}
```

OUTPUT

>= 100!

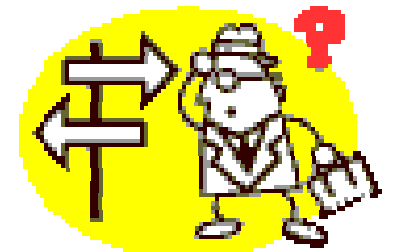


the if-else statement

```
int uilScore=200;  
if(uilScore>190)  
{  
    System.out.println("team");  
}  
else  
{  
    System.out.println("bench");  
}
```

OUTPUT

team

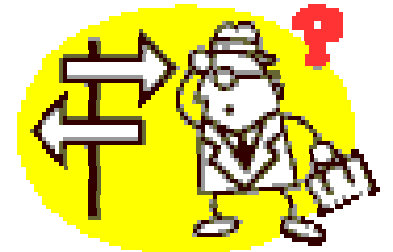


the if-else statement

```
String s = "one";  
if(s.equals("one"))  
{  
    System.out.println(s + " is one!");  
}  
else  
{  
    System.out.println(s + " is not one!");  
}
```

OUTPUT

one is one!



nesting ifs

```
int num=1;  
if(num>2)  
{  
    if(num<10)  
        System.out.println(">2<10");  
}  
else{  
    System.out.println("<2");  
}
```

OUTPUT

<2



nesting ifs

```
int num=11;  
if(num>2)  
    if(num<10)  
        System.out.println(">2<10");  
else  
    System.out.println("<2");
```

OUTPUT

<2

Always use braces with ifs to indicate which statements are related.



COMMON ERRORS

```
if(total >= 25)
```

```
{  
{
```

```
else(total = 10)
```

```
{  
{
```



Comparing Objects

Object references can be compared with `==`.

The actual object contents can be compared using `equals()` or `compareTo()`

Comparing Strings

String references can be compared with `==`.

The actual String contents can be compared using `equals()` or `compareTo()`

equals

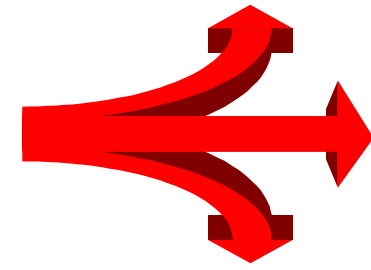
```
String one = new String("compsci");  
String two = new String("compsci");
```

```
if(one.equals(two))  
    System.out.println("equal");  
else  
    System.out.println("!equal");
```

OUTPUT

equal

equals() compares the values stored in the actual String objects.

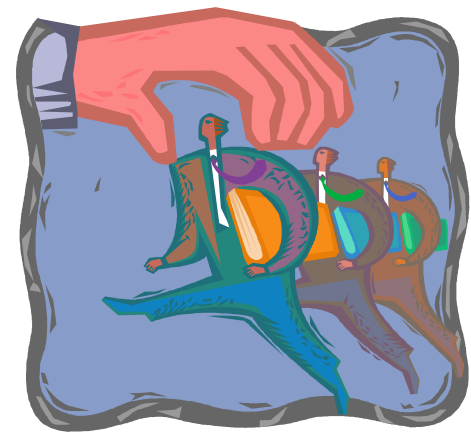


if-else if-else if

```
String letter = "C";
int ascii=0;
if(letter.equals("A")) {
    ascii=65;
}
else if(letter.equals("B")){
    ascii=66;
}
else if(letter.equals("C")){
    ascii=67;
}
else if(letter.equals("D")){
    ascii=68;
}
else{
    ascii=69;
}
out.println(ascii);
```

if
else if
else if

OUTPUT
67



```
int uilScore=200;
if(uilScore>220) {
    out.println("state bound");
}
else if(uilScore>200) {
    out.println("region bound");
}
else if(uilScore>180) {
    out.println("district bound");
}
else{
    out.println("take more tests");
}
```

if
else if
else if

OUTPUT

district bound

Only one condition can be found true!



logical operators



Logical frequently used operators

Operator	Use
<code>x y</code>	either x or y must be true
<code>x && y</code>	both x and y must be true
<code>!x</code>	true if x is false – false if x is true

logical operators

```
int height=6;  
int weight=150;
```

```
if(height>6 || weight>150)  
{  
    out.println("big un");  
}  
else if(height<=6&&weight<=150)  
{  
    out.println("little un");  
}
```

OUTPUT
little un

nested ifs

```
int num =75;
if(num>50)
{
    if(num>50&&num<100)
    {
        if(num>50&&num<150)
        {
            System.out.println(">50 && <150");
        }
    }
}
}
```

OUTPUT

>50 && <150

Dangling Else

```
int num=15;  
if(num>10){  
    if(num<25)  
        out.println("jump");  
}else  
    out.println("run");
```

OUTPUT

jump

Dangling Else

```
int num=35;  
if(num>10)  
    if(num<25)  
        out.println("jump");  
else  
    out.println("run");
```

OUTPUT

run