

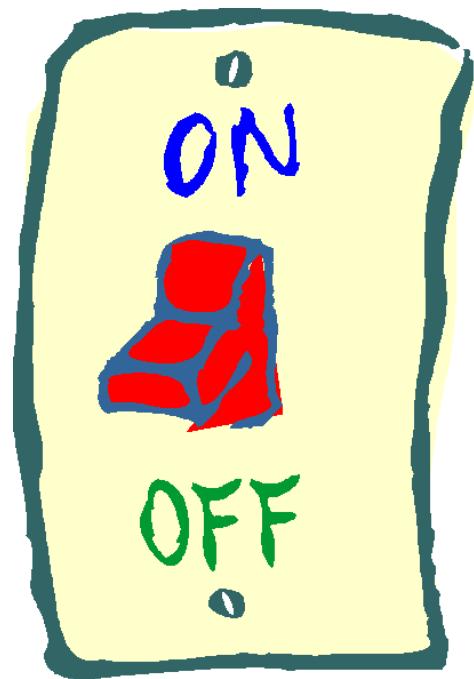
A+ Computer Science

IF ELSE IF

**What is
an if else
statement?**

The if else Statement

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



The if else Statement

```
int aplus = 990;  
if( aplus > 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 uiScore=200;  
if(uiScore>190)  
{  
    System.out.println("team");  
}  
else  
{  
    System.out.println("bench");  
}
```

OUTPUT

team

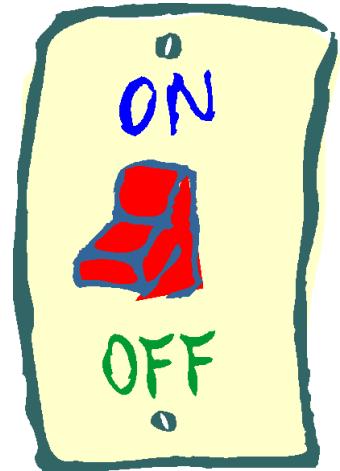
ifelse.java
ifelseUIL.java

If
Else

If

The if else if Statement

```
if( boolean condition placed here )  
{  
    do something 1;  
}  
else if( boolean condition placed here )  
{  
    do something 2;  
}  
else if( boolean condition placed here )  
{  
    do something 3;  
}
```



If Else If

```
int val = 58;  
int ans = 0;  
if( val > 80 )  
    ans +=25;  
else if( val > 60 )  
    ans += 15;  
else if( val > 40 )  
    ans += 5;  
System.out.println( ans );
```

OUTPUT
5



If Else If

```
int val = 98;  
int ans = 0;  
if( val > 80 )  
    ans += 25;  
else if( val > 60 )  
    ans += 15;  
else if( val > 40 )  
    ans += 5;  
System.out.println( ans );
```

OUTPUT
25



ifelseif.java

Common Errors

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

```
{  
}
```

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

```
{  
}
```



Common Errors

Never put a ;
before an open { brace

;
{ illegal

}; legal



Logical Operators

Logical

frequently used operators

Operator	Use
$x \mid\mid y$	either x or y must be true
$x \&\& y$	both x and y must be true
$!x$	true if x is false – false if x is true

Logical Operators

```
int one=6;
```

```
int two = 150;
```

```
if(one>3 || two>200)
```

```
{
```

```
    System.out.println("aplus");
```

```
}
```

```
else if(one<=6&&two<200)
```

```
{
```

```
    System.out.println("comp sci");
```

```
}
```

OUTPUT
aplus

Logical Operators

```
int one=6;  
int two = 150;  
  
if(one>6 | | two>200)  
{  
    System.out.println("aplus");  
}  
else if(one<=6&&two<200)  
{  
    System.out.println("comp sci");  
}
```

OUTPUT
comp sci

logical.java

Nested Ifs

Nested Ifs

```
num = 67;  
if(num>50)  
    if(num>60)  
        if(num>70)  
            System.out.println(">70");  
        else  
            System.out.println(">60 and <70");  
    else  
        System.out.println(">50 and <60");  
else  
    System.out.println("<50");
```

OUTPUT

>60 and <70

Nesting ifs

```
int p = 74;  
if(p > 30)  
{  
    System.out.println("aplus");  
    if(p > 40)  
        System.out.println("comp");  
}  
else  
    System.out.println("sci");
```

OUTPUT

aplus
comp



Nesting ifs

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

OUTPUT

<2



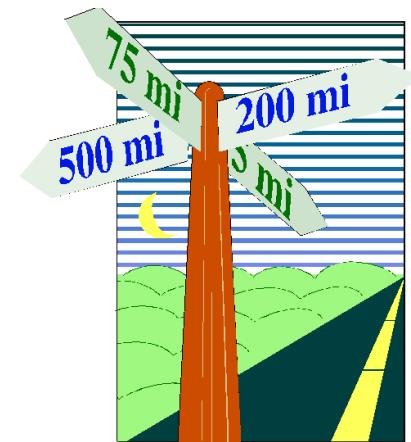
Nesting ifs

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

OUTPUT

<2

It is a good idea to use braces with ifs to indicate which statements are related.



ifnesting.java
danglingelse.java

Work on Programs!

Crank Some Code!

Comparing Strings

Comparing Objects

Object references can be compared with ==.

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

String References

```
String one = "compsci";  
String two = "compsci";
```

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

OUTPUT

==

== compares the String references which are the memory addresses of the actual String objects.

String References

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

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

OUTPUT

!=

== compares the String references which are the memory addresses of the actual String objects.

stringref.java

String

frequently used methods

Name	Use
equals(s)	checks if this string has same chars as s
compareTo(s)	compares this string and s for >,<, and ==
trim()	removes leading and trailing whitespace
replaceAll(x,y)	returns a new String with all x changed to y
toUpperCase()	returns a new String with uppercase chars
toLowerCase()	returns a new String with lowercase chars

The equals() method

```
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.

The equals() method

```
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.

The equals() method

```
String one = "big";
String two = "big";
if( one.equals( two ) )
{
    System.out.println("boom");
}
```

OUTPUT
boom

Equals is used to see if two strings contains the same letters in the same order in the same case.

The equals() method

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

OUTPUT
== big

The equals() method

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

OUTPUT

!= it

The equals() method

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

OUTPUT
!= it



compareTo()

```
String one = "region";
String two = "uilstate";
out.println(one.compareTo(two));
out.println(two.compareTo(one));
two = "region";
out.println(two.compareTo(one));
```

OUTPUT

-3
3
0

compareTo() returns the difference in ASCII value when comparing Strings.



compareTo()

```
Integer one = 90;
```

```
Integer two = 75;
```

```
out.println(one.compareTo(two));
```

```
out.println(two.compareTo(one));
```

```
two = 90;
```

```
out.println(two.equals(one));
```

```
out.println(two.compareTo(one));
```

OUTPUT

-1
1

true
0

compareTo() returns a negative value when A is less than B and a positive value when A is greater than B.

0 is returned with the A and B are the same.

equals.java
compareto.java

**Just
For
Fun**

Switch Case

```
int num = 30;  
switch (num)  
{  
    case 11 : out.println("num == 11"); break;  
    case 22 : out.println("num == 22"); break;  
    case 30 : out.println("num == 30"); break;  
    case 40 : out.println("num == 40"); break;  
    default : out.println("does not equal");  
}
```

OUTPUT

num == 30

Switch Case

If you have no break, every statement after the first true condition is executed until a break is encountered or the bottom of the switch case is reached.

Switch Case

```
int num = 30;  
switch (num)  
{  
    case 11 : out.println("num == 11");  
    case 22 : out.println("num == 22");  
    case 30 : out.println("num == 30");  
    case 40 : out.println("num == 40");  
    default : out.println("does not equal");  
}
```

OUTPUT

num == 30

num == 40

does not equal

switchcaseone.java

switchcasetwo.java
switchcasethree.java

Work on Programs!

Crank Some Code!

A+ Computer Science

IF ELSE IF