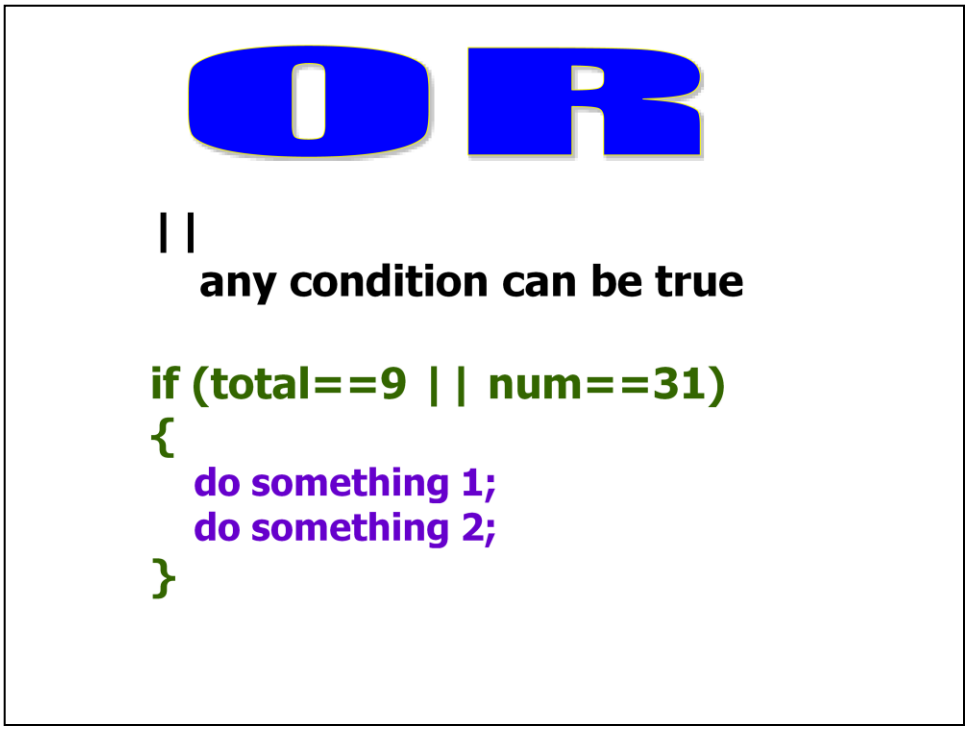
Jeroo Compound Conditionals



OR

||

any condition can be true

```
if (total==9 || num==31)
{
   do something 1;
   do something 2;
}
```

Or is used to see if any part is true. In some languages, or is actually written as a word. In other languages, or is written as a symbol, like `||` or `|`.

`||` evaluates as true if any part connected by `||`s is true.

```
if(A or B)
```

This condition is true if A or B is true. If A and B are both true, the condition is still true.

Or – ||

```
if (bob.isWater(AHEAD) ||
                 bob.isWater(RIGHT))
{
  bob.turn(LEFT);
}
```

In this example, bob will turn left if there is water either ahead or water to the right.

Or – ||

```
if ( bob.isNET(AHEAD) ||
                bob.isWater(AHEAD) )
{
  bob.turn(RIGHT);
  bob.turn(RIGHT);
}
```

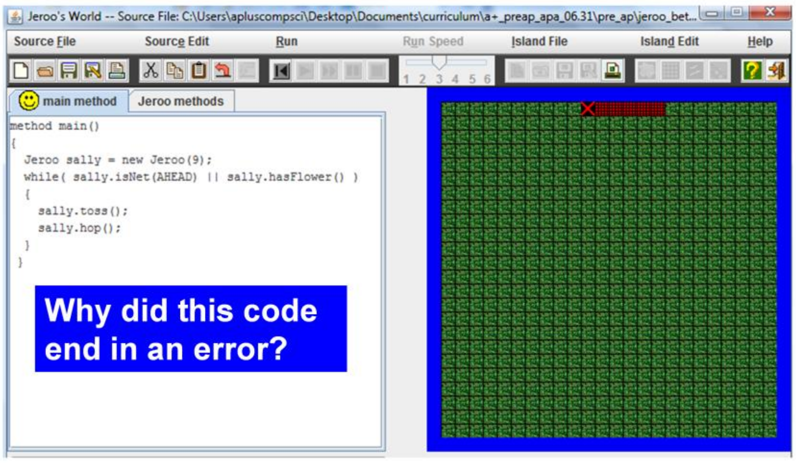
In this example, bob turns around if there is a net ahead OR if there is water ahead.

Or – ||

```
while ( bob.isNet(AHEAD) ||
                    bob.hasFlower())
 {
   bob.toss();
   bob.hop();
}
```

The power of repetition may become a problem since bob will toss flowers and hop as long as there are flowers OR there is a net ahead. If there is a net ahead after he runs out of flowers, he will hit the net.

Or – ||


Jeroo's World -- Source File: C:\Users\apluscompsci\Desktop\Documents\curriculum\a+_preap_apa_06.31\pre_ap\jeroo_bet...
Source File
Source Edit
Run
Run Speed
Island File
Island Edit
Help
1 2 3 4 5 6
main method
Jeroo methods

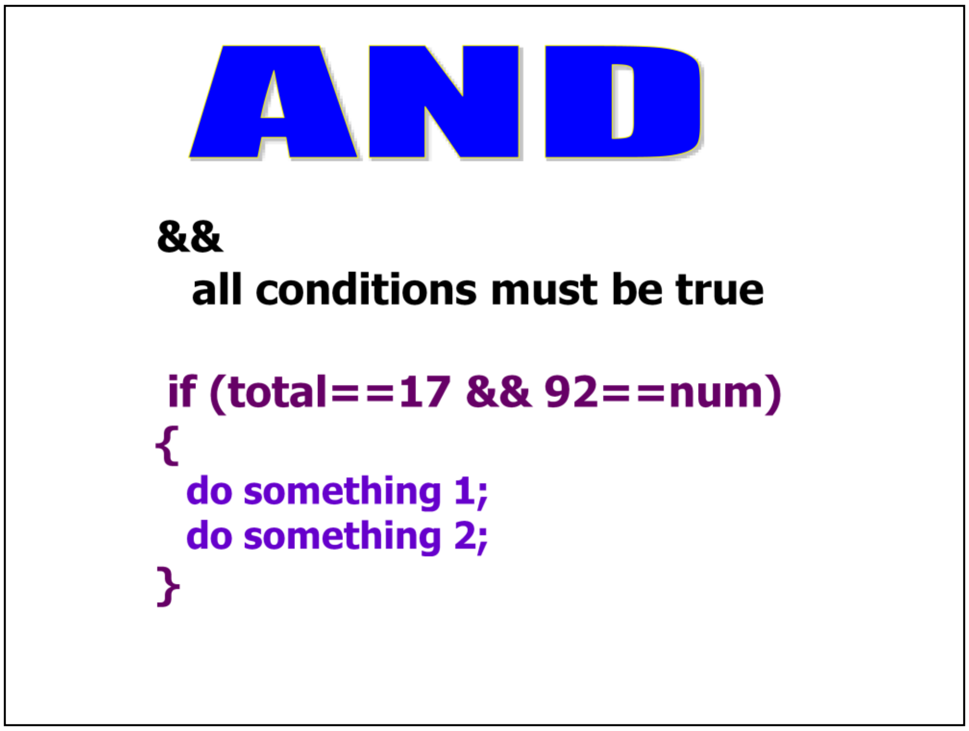
```
method main()
{
  Jeroo sally = new Jeroo(9);
  while( sally.isNet(AHEAD) || sally.hasFlower() )
  {
    sally.toss();
    sally.hop();
  }
 }
```

Why did this code end in an error?

Open

or.jsc

or.jev



AND

&&

all conditions must be true

```
if (total==17 && 92==num)
{
  do something 1;
  do something 2;
}
```

And is used to see if all parts are true. In some languages, and is actually written as a word. In other languages, and is written as a symbol, like `&&` or `&`.

`&&` evaluates as true if all parts connected by `&&`s are true.

```
if(A and B)
```

This condition is true if A and B are both true. If either A or B is false, the condition is false as both parts must be true in order for the condition to be true.

And – &&

```
if(bob.isWater(AHEAD) &&
                 bob.isWater(RIGHT))
{
  bob.turn(LEFT);
}
```

In this example, Bob only turns left when there is water both to his right and in front of him. If water is not in both places, Bob does not turn.

And – &&

```
if( bob.isNET(AHEAD) &&
              bob.hasFlower() )
{
  bob.toss();
  bob.hop();
}
```

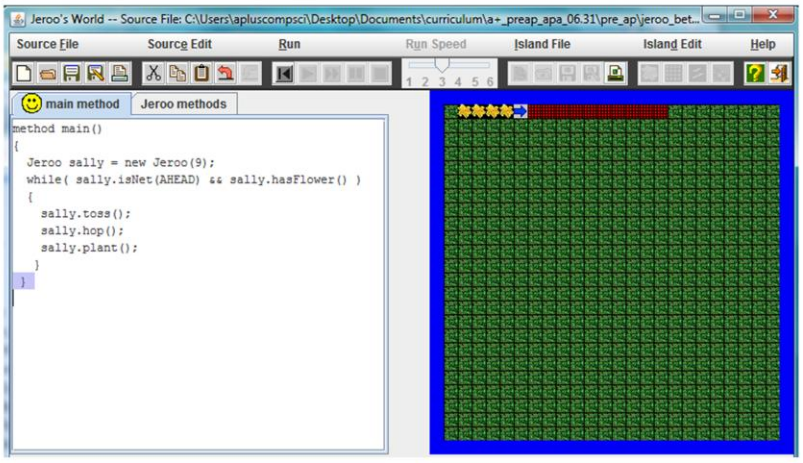
In this example, bob only tosses the flower and hops when he is facing a net AND is carrying a flower with him.

And – &&

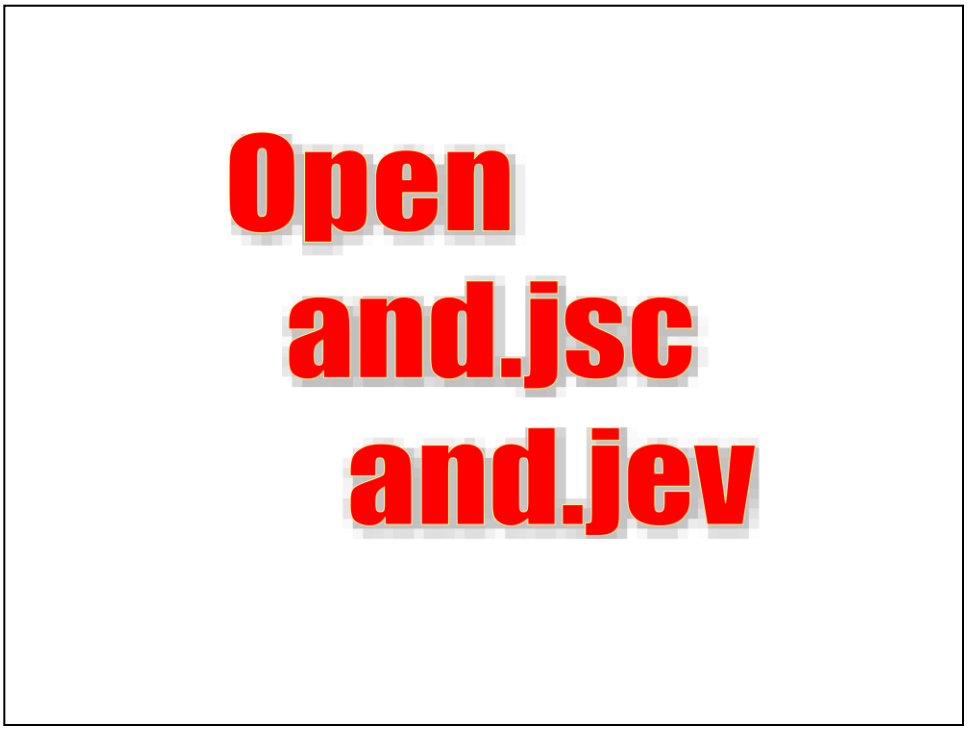
```
while (bob.isNet(AHEAD) &&
               bob.hasFlower())
{
  bob.toss();
  bob.hop();
}
```

The power of repetition will allow bob to toss flowers and clear nets as long as he is facing a net AND is carrying a flower with him. If he runs out of flowers, this code will stop him from hitting the net.

And – &&


Jeroo's World -- Source File: C:\Users\apluscompsci\Desktop\Documents\curriculum\a+_preap_apa_06.31\pre_ap\jeroo_bet...
Source File
Source Edit
Run
Run Speed
Island File
Island Edit
Help
main method
Jeroo methods

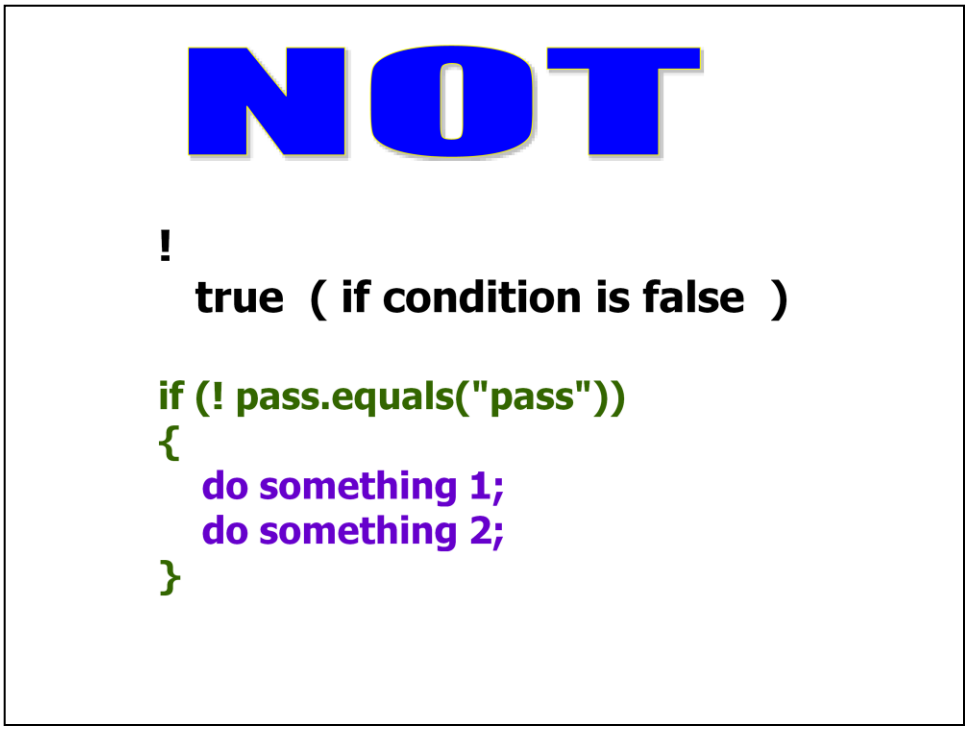
```
method main()
{
  Jeroo sally = new Jeroo(9);
  while( sally.isNet(AHEAD) && sally.hasFlower() )
  {
    sally.toss();
    sally.hop();
    sally.plant();
   }
 }
```



Open

and.jsc

and.jev



NOT

!

true (if condition is false)

```
if (! pass.equals("pass"))
{
   do something 1;
   do something 2;
}
```

Not is used to negate a boolean value. In some languages, not is actually written as a word. In other languages, not is written as a symbol, like `!`.

!true is false

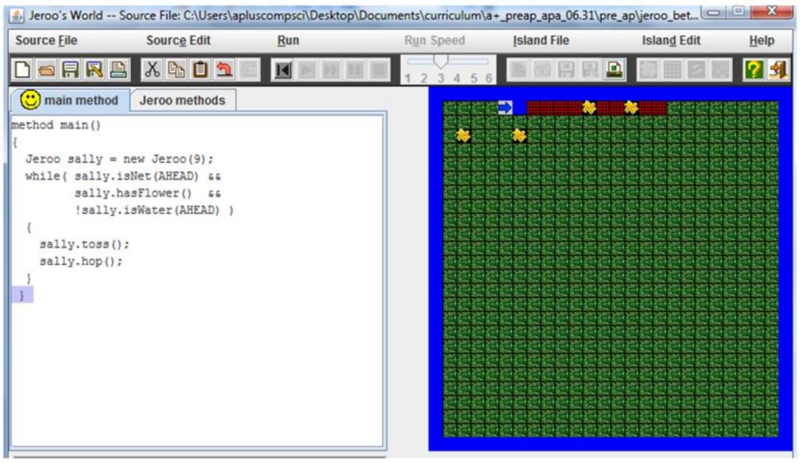
!false is true

Fundamental Boolean Logic

true and false = false
false and true = false
false and false = false
true and true = true

false or true = true
true or false = true
true or true = true
false or false = false

Not – !


Jeroo's World -- Source File: C:\Users\apluscompsci\Desktop\Documents\curriculum\a+_preap_apa_06.31\pre_ap\jeroo_bet...
Source File
Source Edit
Run
Run Speed
Island File
Island Edit
Help
main method
Jeroo methods

```
method main()
{
  Jeroo sally = new Jeroo(9);
  while( sally.isNet(AHEAD) &&
         sally.hasFlower()  &&
         !sally.isWater(AHEAD) )
  {
    sally.toss();
    sally.hop();
  }
}
```

Open

not.jsc

not.jev

Start work on Boolean Labs