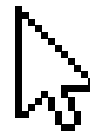




Computer Science : JavaScript

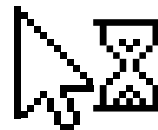


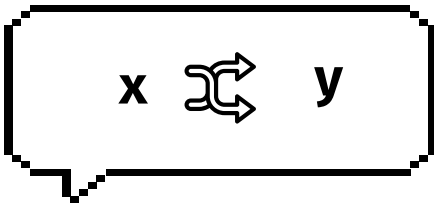
Week 3: Conditional Statements



Conditional Statements

Conditionals check given condition(s) and perform a tasks relative to the condition(s).





01

if ... else ...



if ...

```
let apples = true;  
if (apples) {  
  console.log('There are  
  apples')  
};
```



if (): inside the parentheses you provide a condition that reads as *true* or *false*

If the condition is true the {} will execute.

Syntax:

```
if (condition) {  
  console.log('statement to print')  
};
```



if ... else

```
let apples = true;
apples = false;
if (apples) {
  console.log('There are apples')
} else {
  console.log ('No apples')
};
```



You can set a code to execute in the case your condition is false,

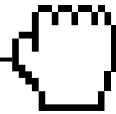
Syntax:

```
if (condition) {
  console.log('statement to print')
} else {
  console.log('statement to print if false')
};
```



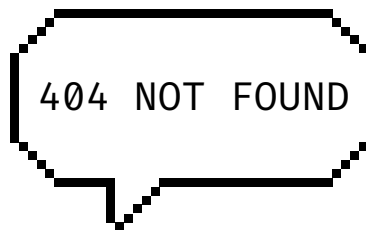
Practice

Create a variable and your own if else statement





02



Comparison Operators

Compare Values



Comparison Operators

Less than: <

Greater than: >

Less than or equal to: <=

Greater than or equal to: >=

Equal to: ==

Not equal to: !=





Practice

```
let students = 18;
```

Set condition: students
greater than or equal to 19

The statement should read
“Full House!” otherwise it
should read “Available spots”





IDENTIFIED

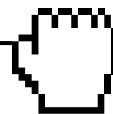
03

Logic Operators



Logic Operators

- “and” operator: `(condition A) && (condition B)`
 - both conditions must be true
- “or” operator: `(condition A) || (condition B)`
 - one must be true, and if the first condition is true the second condition will not be checked
- “not” operator: `!(condition A)`
 - negates the value (produces the opposite boolean)





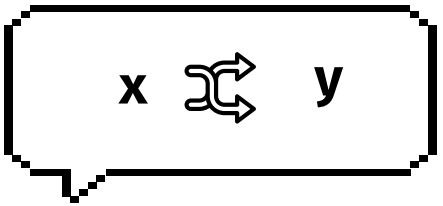
Practice

```
let emotion = "happy";  
let hungerLevel = 7;
```

Check if emotion is "Happy" or
hungerLevel is not greater than
equal to 7

If it executes true print "Go
treat yourself!" otherwise "Stay
home and wait."





04

Non-Boolean False

Not explicitly true or false.



Executes False

if statements run when given a true value even though it is not an explicit boolean. Here are some non-boolean false.

Number: 0 or NaN (Not a number)

Empty String: ' ', " "

null

undefined





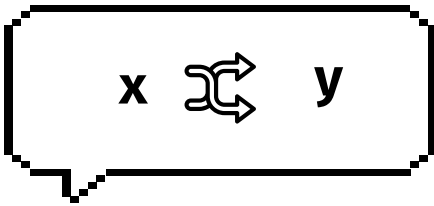
Practice

```
let sentence = "Its cold.";
```

Reassign to an empty string

```
If it executes true print "I'll  
turn on the heat." otherwise "The  
string is empty."
```





05

Ternary Operator



Ternary Operator

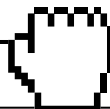
Shortcut to writing the if...else... statement: ?

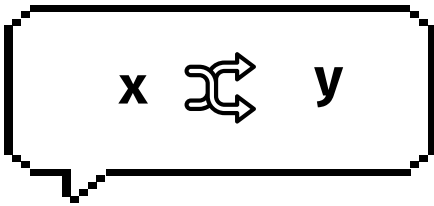
Long Version:

```
if (season === 'winter') {  
  console.log ('It's snowing')  
} else {  
  console.log ('No Snow')  
};
```

Shortcut:

```
season === 'winter' ?  
  console.log ('It's snowing') :  
  console.log ('No Snow');
```





06

else if statements

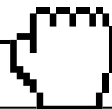


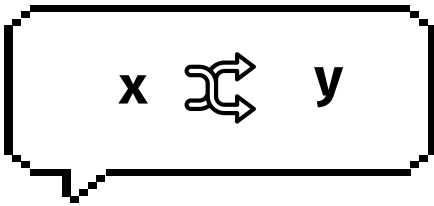
else if

If you would like to add more conditions with different execution use else if

Example:

```
if (season === 'winter') { // Condition 1
  console.log ('It's snowing')
} else if (season === 'spring') { // Condition 2
  console.log ('Its raining')
} else {
  console.log ('No Snow')
};
```





07

Switch

Option instead of else if



switch

If there are many conditions and writing the else if statements become too much its is easier to use switch as a shortcut

```
switch (season) {  
  case Condition 1 :  
    console.log ( );  
    break;  
  case Condition 2 :  
    console.log ( );  
    break;  
  default:  
    console.log ( );  
    break;
```





Example: let season = 'winter'

```
switch (season) {  
  case 'spring':  
    console.log ('Its raining');  
    break;  
  case 'winter':  
    console.log ('It's snowing');  
    break;  
  case 'fall':  
    console.log ('It's windy');  
    break;  
  default:  
    console.log ('It's hot');  
    break;  
}
```