under certain conditions the css must be changed

@media *\_\_*\_\_\_\_\_\_ {

()style that must be added

}

Eg.

@media print {

h1{

color: blue;

}

}

@media screen (min-width: 800px) {

h1{

color: blue;

}

}

readability

modullarity

efficiency

length

heirarchy

selector1 selector2{}

same level

selector1.selector2{}

string functions:

string.length

string.slice(x,y)

toUpperCase()

toLowerCase()

Incrementation:

X++

Decrementation:

x--

x += y

FUNCTIONS

function \_\_\_\_\_\_(parameters){

}

Math.random(); -> between 0 and 1

&& -> and

|| -> or

! -> not

ARRAYS

Array.includes()

Array.push()

LOOPS

while (condition that is true){

}

For (var i= 0; i<2;rob i++){

}

DOM

* Tree model
* Graphical user interface, text

  Description automatically generated

Inline

<body onload="alert('hello')">

Internal

<script>

alert("hello");

</script>

External file

To select objects in the DOM tree

document.firstElementChild.firstElementChild.\_\_\_\_\_\_\_

to change the dom

ele.innerHTML = "sample text"

Gives list of elements of the same tag:

document.getElementByTagName(“\_\_\_\_\_”)

to target a specific element from the retrived list

document.getElementsByTagName("li")[1].innerHTML = "hello"

Document.querySelector(“\_\_\_\_”).\_\_\_\_\_

document.querySelector("h1").style.color = "green";

for multiple elements

querySelectorAll

To add new class to a selected element:

document.querySelector("button").classList.add("new")

OBJECT:

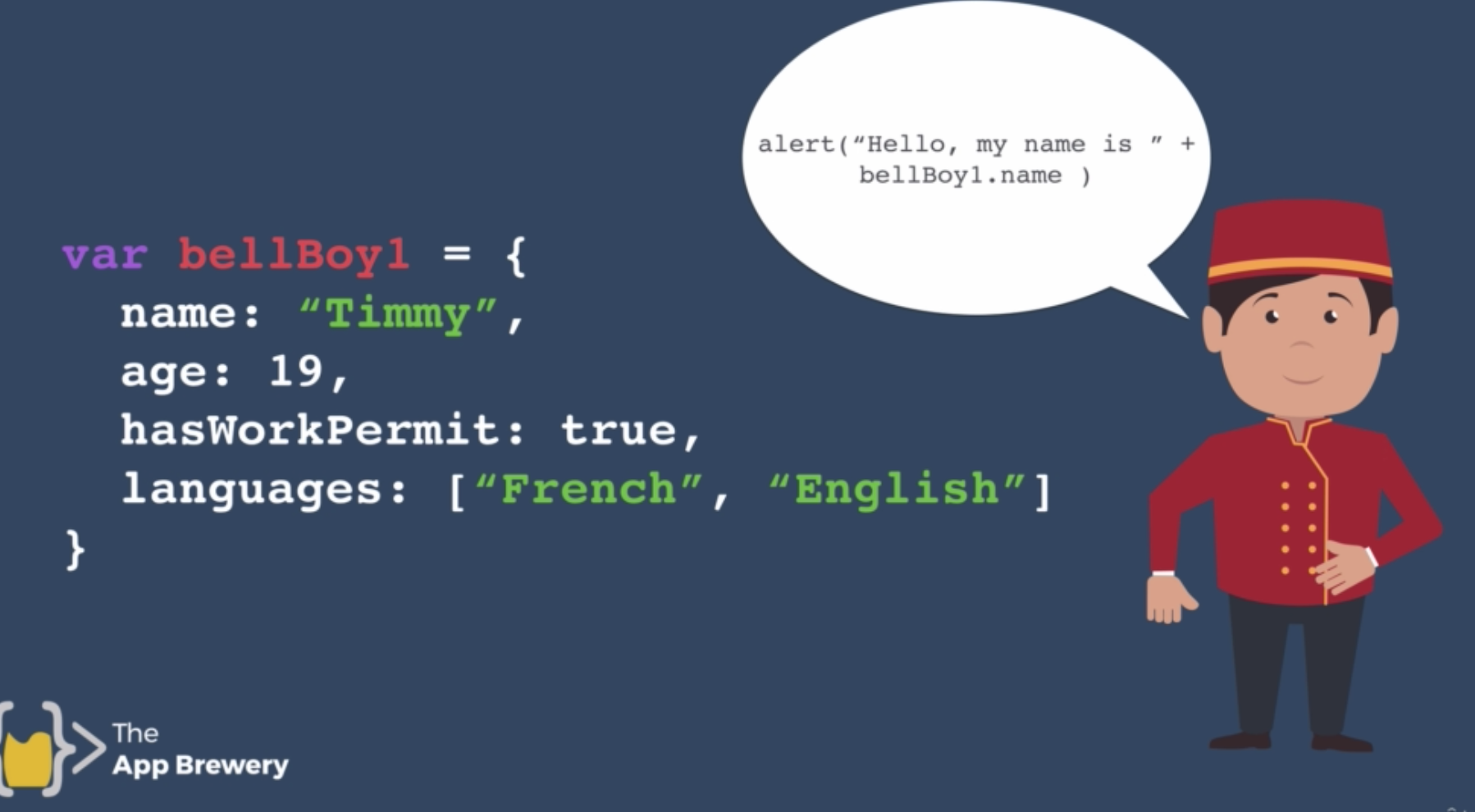
* Properties -> describe something about the object
  + Get
  + Set
* Methods -> something that the object can do
  + Call
  + Have parenthesis

EVENT LISTENERS

document.querySelector("button").addEventListener("click", buttonClick)

to find out the identity of the element that triggered the event -> this

OBJECT:



Graphical user interface, text

Description automatically generated