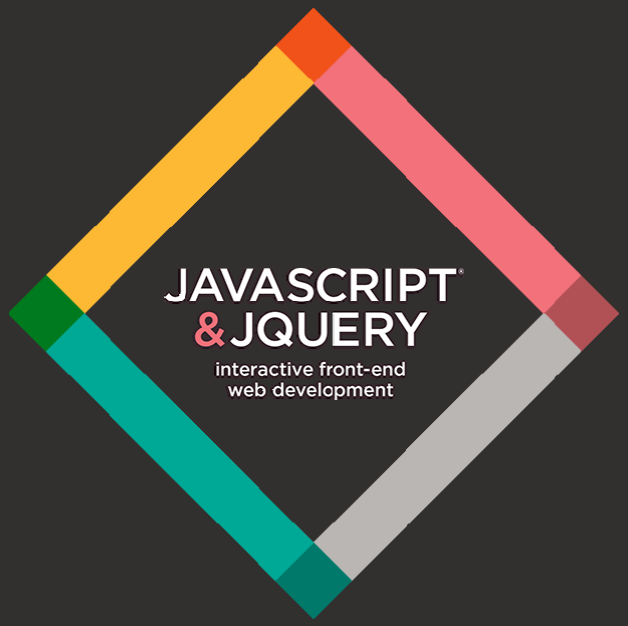


CHAPTER 2

# BASIC JAVASCRIPT INSTRUCTIONS

## STATEMENTS

Each individual step in a script is known as a **statement**.

Each statement should end with a semi-colon.

```
document.write('Welcome!');
```

SEMI-COLON

# COMMENTS

You should use **comments** to explain what your code does.

They help you remember it and others understand it.

# MULTI-LINE COMMENTS

```
/* Anything between
   these characters is
   a comment and will
   not be processed. */
```

```
/* Anything between
   these characters is
   a comment and will
   not be processed. */
```

# SINGLE-LINE COMMENTS

```
// Anything after the two
// forward slashes is also
// a comment and will not
// be processed.
```


```
// Anything after the two
// forward slashes is also
// a comment and will not
// be processed.
```


# VARIABLES

Scripts often need to store bits of information temporarily in order to achieve their tasks.

These bits of information - or data - are stored in **variables**.

# DECLARING A VARIABLE

```
var quantity;
```

```
var quantity;
```

KEYWORD

```
var quantity;
```

VARIABLE NAME

# ASSIGNING A VALUE TO A VARIABLE

```
quantity = 3;
```

```
quantity = 3;
```

VARIABLE NAME

```
quantity = 3;
```

ASSIGNMENT OPERATOR

```
quantity = 3;
```

VALUE

# DATA TYPES

JavaScript distinguishes between **numbers**, **strings**, and `true` or `false` values known as **Booleans**.

1

1

NUMBERS

0.75

NO QUOTES

1

NUMBERS

0.75

NO QUOTES

2

1

NUMBERS

0.75

NO QUOTES

2

STRINGS

'Hi Ivy!'

ENCLOSED IN QUOTES
WHICH CAN BE SINGLE
OR DOUBLE QUOTES,
BUT MUST MATCH

1

NUMBERS

`0.75`

NO QUOTES

2

STRINGS

`'Hi Ivy!'`

ENCLOSED IN QUOTES WHICH CAN BE SINGLE OR DOUBLE QUOTES, BUT MUST MATCH

3

1

NUMBERS

`0.75`

NO QUOTES

2

STRINGS

`'Hi Ivy!'`

ENCLOSED IN QUOTES WHICH CAN BE SINGLE OR DOUBLE QUOTES, BUT MUST MATCH

3

BOOLEAN

`true`

EITHER TRUE OR FALSE

# ARRAYS

An **array** is a special type of variable. It doesn't just store one value; it stores a list of values.

```
colors = [                    ];
```

```
colors = ['pink'              ];
```

```
colors = ['pink','yellow'     ];
```

```
colors = ['pink','yellow','green'];
```

```
colors = ['pink','yellow','green'];

colors[ ];
```

```
colors = ['pink','yellow','green'];

colors[0];
```

```
colors = ['pink','yellow','green'];

colors[1];
```

```
colors = ['pink','yellow','green'];

colors[2];
```

# ARITHMETIC OPERATORS

JavaScript uses mathematics to get some tasks done.

```
var width = 3;
```

```
var width = 3;
var height = 2;
```

```
var width = 3;
var height = 2;

area = width * height;
```

```
var width = 3;
var height = 2;

area = width * height;
```

```
var width = 3;
var height = 2;

area = width * height;
```

```
var width = 3;
var height = 2;

area = width * height;
```

6

# CONCATENATING STRINGS

There is just one string operator: the **+** symbol.

It is used to join strings on either side of it.

```
var greeting = 'Howdy ';
```

```
var greeting = 'Howdy ';
var name = 'Molly';
```

```
var greeting = 'Howdy ';
var name = 'Molly';

var message = greeting + name;
```

```
var greeting = 'Howdy ';
var name = 'Molly';

var message = greeting + name;
```

```
var greeting = 'Howdy ';
var name = 'Molly';

var message = greeting + name;
```

```
var greeting = 'Howdy ';
var name = 'Molly';

var message = greeting + name;
```

Howdy Molly