

Nation Code

JavaScript Fundamentals

Variables



Learning Objectives

- } To understand and use variables and operators to store values and do calculations
- } To use camelCase when naming variables
- } To understand how to access data in variables

First thing's first... I'm the realest

"All Around the World"

Display the 8th character in upper case in the console



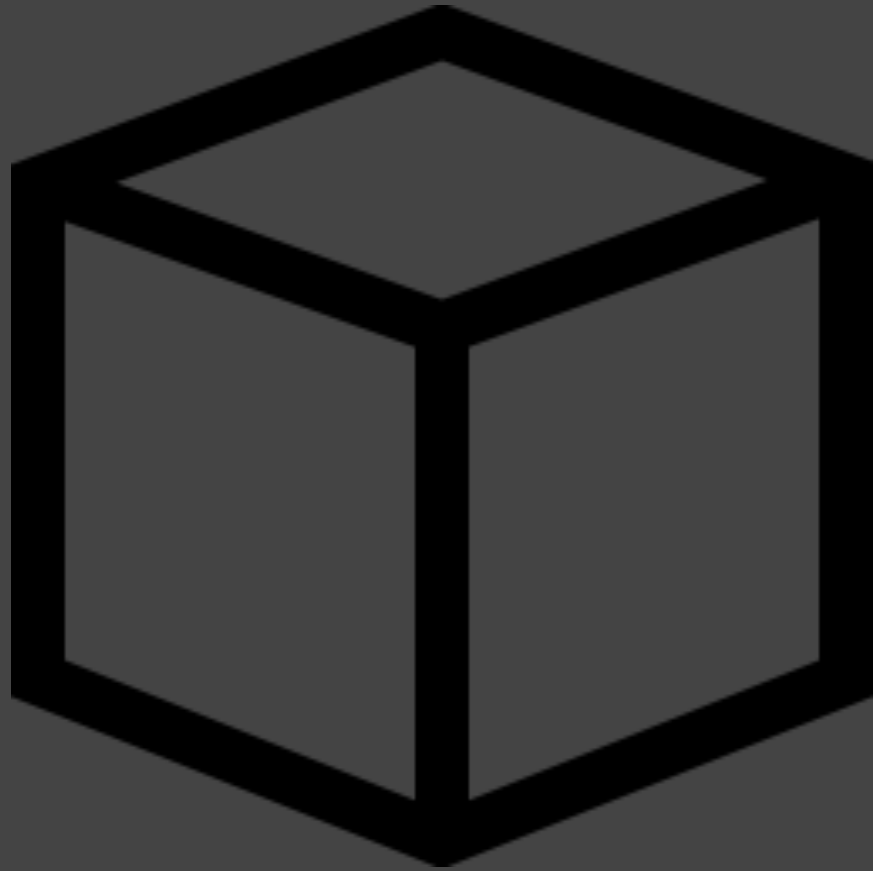
```
console.log("All around the  
world".charAt(7).toUpperCase());
```

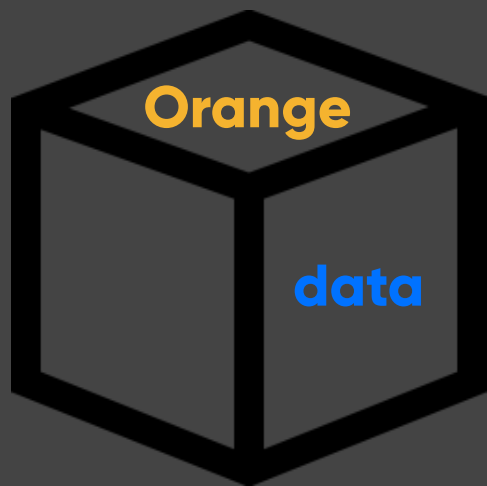


```
console.log("All around the  
world".toUpperCase().charAt(7));
```

Introducing

Variables





- } We store items in boxes to retrieve later
- } In JavaScript different items can be stored in the box as different times
- } In code these 'items' are variable data, we give variables names so we can access their data. Just like saying "hey, get me that **thing** from that **orange** box over there"



Imagine a cash machine

Code needs to be **reusable**

WITHDRAW 10_POUNDS
FROM 82929201

should be

WITHDRAW AMOUNT
FROM ACCOUNTNUM



In JavaScript we don't need to 'declare' what our data type is

It's a dynamically typed language



We can declare variables
using **var**, **let**, and **const**.

var is old (legacy), try to leave it alone.



```
var i = 10;
```

Create a variable called `i` which holds values that **can** be changed whenever the code is running, and store a value of 10 in it



```
let i = 10;
```

Create a variable called *i* which holds values that **can** be changed whenever the code is running, and store a value of 10 in it



```
const i = 10;
```

Create a variable called `i` which holds values that **cannot** be changed whenever the code is running, and store a value of 10 in it.

Const means **constant**.



Remember the titans data types?



Boolean

String

Number

Null

Undefined



Time for **sum** maths



+

-

*

**

/

%

++

--

Arithmetic Operators for calculations



=

*=

+=

/=

-=

++

--

Assignment Operators to store values

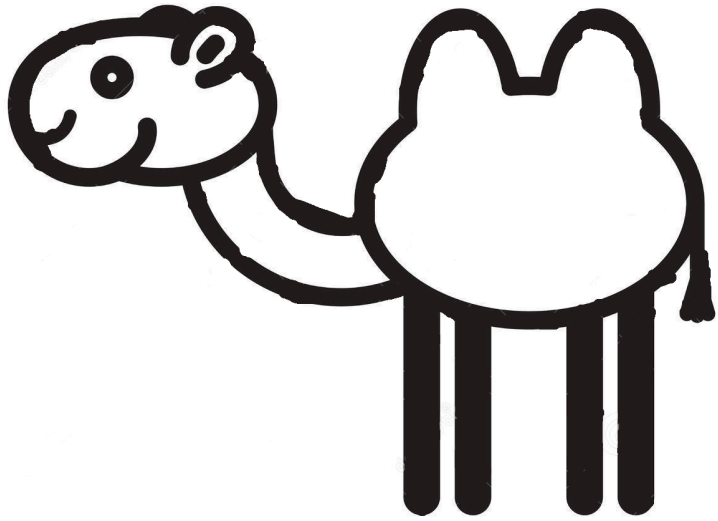


**Primary Operator
for assignment**



Don't get the **hump**

Let's talk **camelCase**



favouriteDrink
thisNumber
firstName



**It's best practice and enhances
code readability**



Let's **access**, and play with
some **data**.

```
let favouriteDrink = "coffee";  
console.log(favouriteDrink);
```

```
let favouriteDrink = "coffee";  
console.log("My favourite drink  
is " + favouriteDrink);
```



You can put **variables** inside
strings to create **sensible**
outputs

We call these **template literals**

```
let favouriteDrink = "coffee";  
console.log(`My favourite drink is  
${favouriteDrink}`);
```



We can also **update**, or **change**,
a **value** to adjust our outputs at
different stages

```
let favouriteDrink = "coffee";
```

```
console.log(`My favourite drink is  
${favouriteDrink}`);
```

```
favouriteDrink = "tea";
```

```
console.log(`My favourite drink is  
now ${favouriteDrink}`);
```

Learning Objectives

- } To understand and use variables and operators to store values and do calculations
- } To use camelCase when naming variables
- } To understand how to access data in variables

Activity 1

- } Write code that stores your name, age, and favourite colour
- } The stored data must be output to the console in a complete sentence
- } **Why not stretch:** use different method's to manipulate your output

Activity 2

- } Write code which stores what you may eat today for breakfast, lunch, and dinner
- } This must also be output to the console as a complete sentence
- } Update the stored data and output the same sentence again to display the changes
- } **Why not stretch:** use different method's to manipulate your output

Activity 3

- } Write code which will calculate how many days from today's date to your birthday
- } This must also be output to the console as a complete sentence
- } **Why not stretch:** why not calculate how many day's since your birth?

Activity 3

- } Write code which will calculate how many days from today's date to your birthday
- } This must also be output to the console as a complete sentence
- } **Why not stretch:** why not calculate how many day's since your birth?

Activity 4 (remember this?) :

- } Create 9 variables – space1, space 2... space 9
- } Assign either the value of 'x', 'o', ' ' to each of the variables
- } Using template literals inject the variable data into your board
- } Output this to the console

x		o	

x		x	

o			