

Nation

Code

Develop: Coding

Variables

{codenation}[®]

Learning Objectives

- To understand how variable works in JavaScript
- To understand and use 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

All Around the World

Display the 8th character in upper case on the console

First thing's first

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

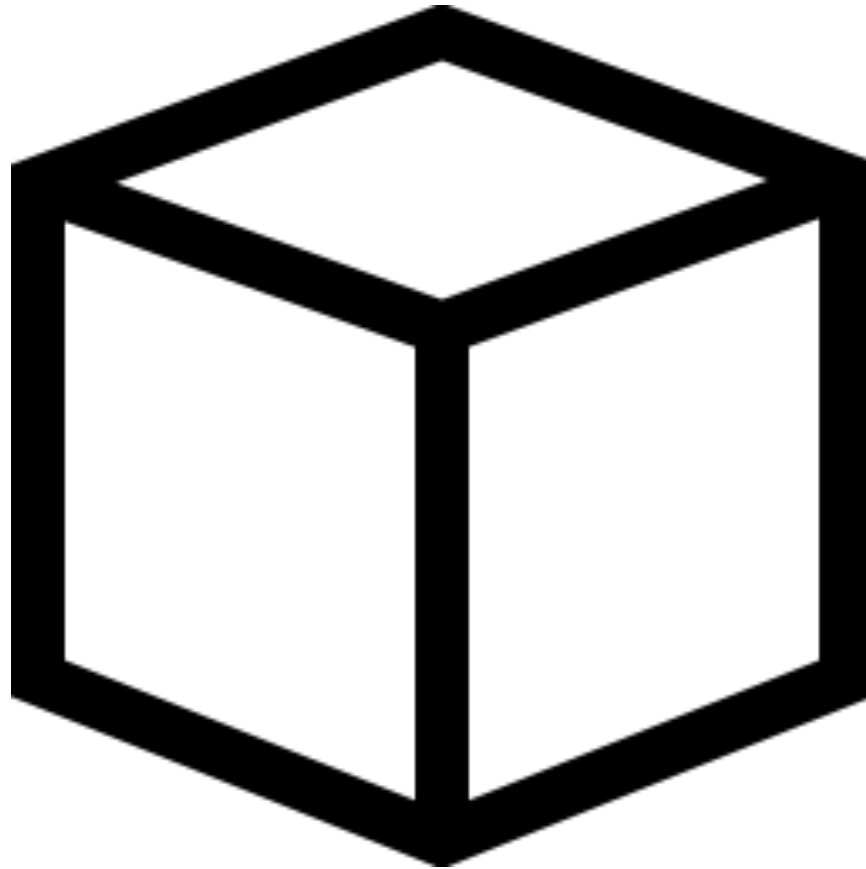
**Things are getting
interesting**

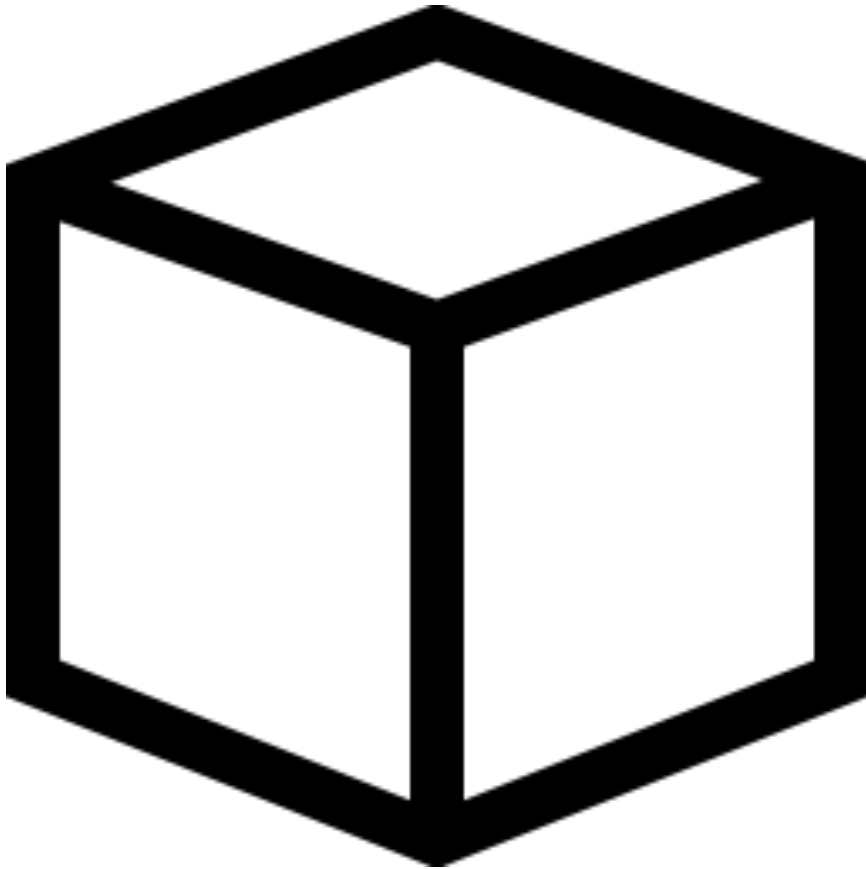
Introducing

Variable

They're like boxes.

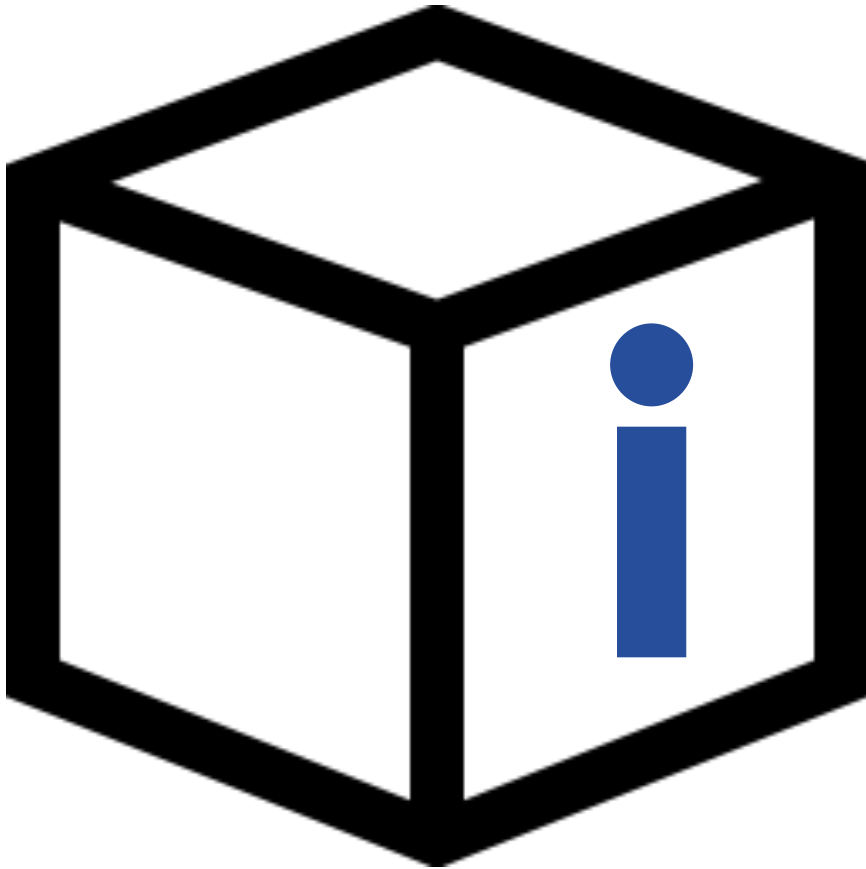
Not very technical that is it?





We store items in boxes to retrieve later

**Different items can be stored in the box
at different times**



We store items in boxes to retrieve later

Different items can be stored in the box at different times

In code, we give variables names so we can access things inside them. Exactly like saying "get me that thing from the blue box over there"

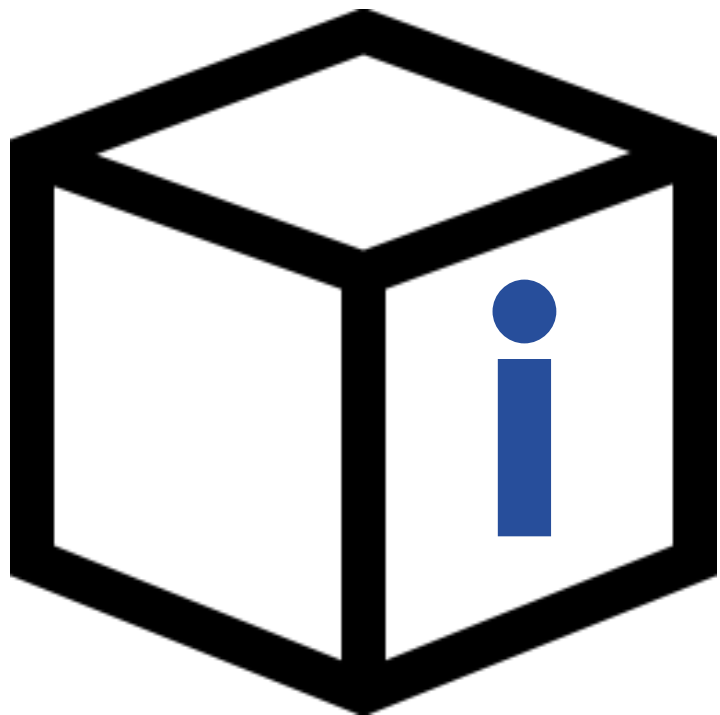
**Imagine a cash
machine**

WITHDRAW 10_POUNDS
FROM 82929201

should be

WITHDRAW AMOUNT
FROM ACCOUNTNUM

**We will be able to
reuse code**



So variables...

- 1) allow us to store data inside them
- 2) access them via a name
- 3) then place new data in them whenever we want

**We don't need to tell
JavaScript what kind of
data will be stored in a
variable**

Because it's a "loosely
typed" language

So if I want to store my name in a variable, what kind of data type would that be?

**Yes, string. You're
just too good.**

You can declare
variables using **let**,
const and **var**

**Let my favourite
drink get stored**

Create variable

Create a **constant** value
that won't change

```
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 and store a value of 10 in it.

Constant means to constant value and when something is constant, it doesn't change


```
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**



var **vs** let

let is more common these days, var is classic

What data types?

String

Boolean

Null

Number

Undefined

String : for representing text

Number : for representing numbers (decimal and integers)

Boolean : for true and false

Null : for nothing

Undefined : for when a data type isn't determined

Symbol : This data type is used as the key for an object property when the property is intended to be private, for the internal use of a class or an object type

Time for sum maths



=

*=

+=

/=

-=

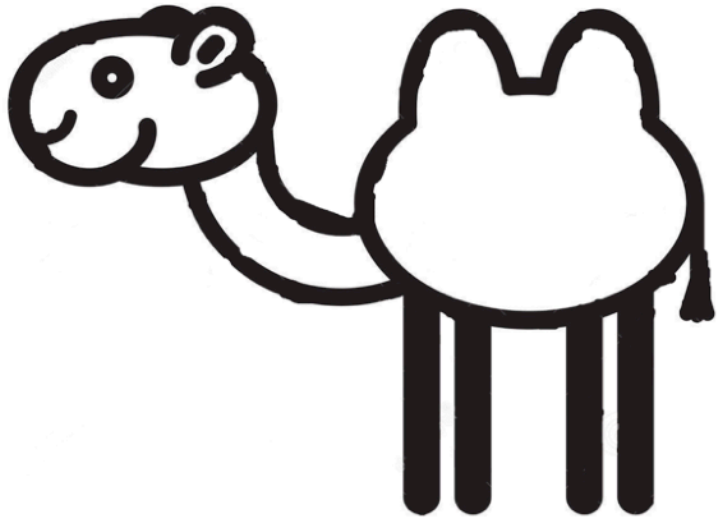
++

--

Operators to store values

**Don't get the
hump**

**Have you noticed we've
stuck to a particular
convention when
naming variables?**



favouriteDrink
thisNumber
firstName

...It's called
camelCase.

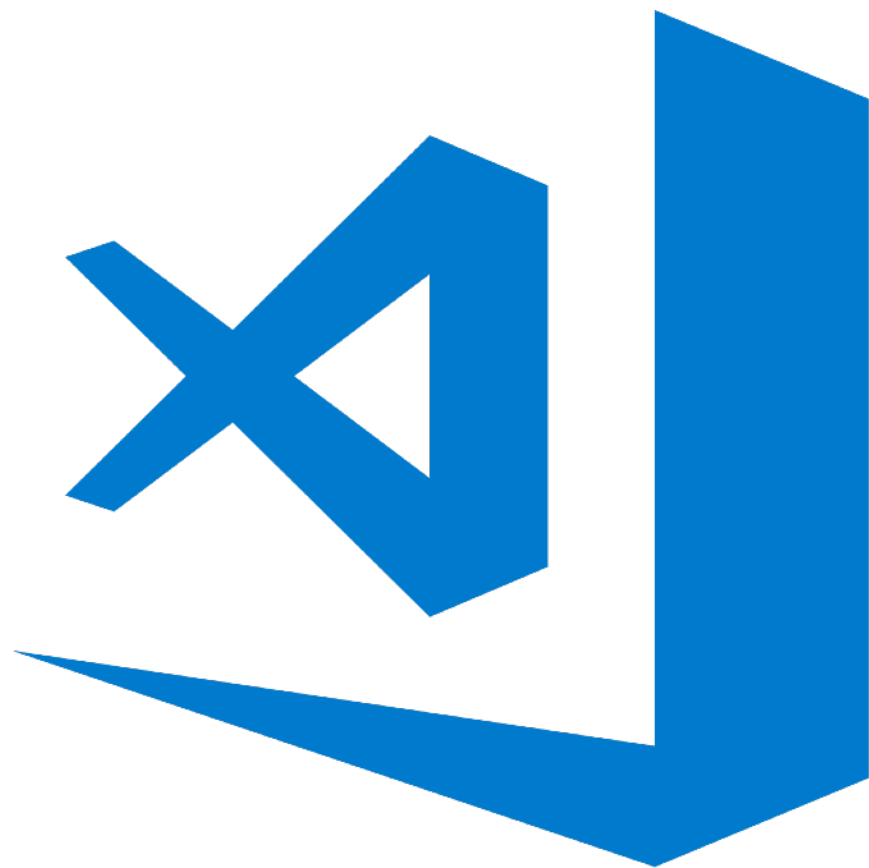
**It's best practice and
enhances code
readability**

How to access data in variables

console.log();

console.log(" ");

A large, thick red 'X' is drawn over the right portion of the code snippet, specifically covering the string and closing parenthesis, indicating that this part of the code is incorrect or being rejected.



To VS Code

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

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

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

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

Learning Objectives

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

Activity(1):

Create a program that stores someone's name, age and favourite colour that logs to the console in a complete sentence

Activity(2):



(1) Create a 9 variables space1, space2... space9

(2) Assign either the value 'x', 'o', ' ' to each of these variable

(3) Insert the variables into the board using the `${varName}` syntax and make your board look like the one displayed

