Javascript Arrays

Our First Data Structure

Objectives

- Understand arrays conceptually
- Write code using JS arrays

Suppose I wanted to model a group of friends:

```
var friend1 = "Charlie";
var friend2 = "Liz";
var friend3 = "David";
var friend4 = "Mattias";
```

This is a lot of code, and it doesn't let us group the friends together

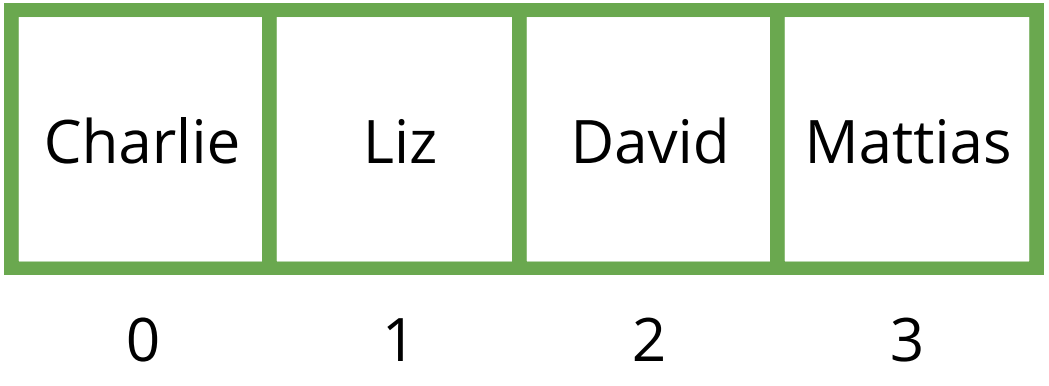
This is a perfect use case for an ARRAY

```
var friends = ["Charlie", "Liz", "David", "Mattias"];
```

Arrays

Arrays let us group data together in lists

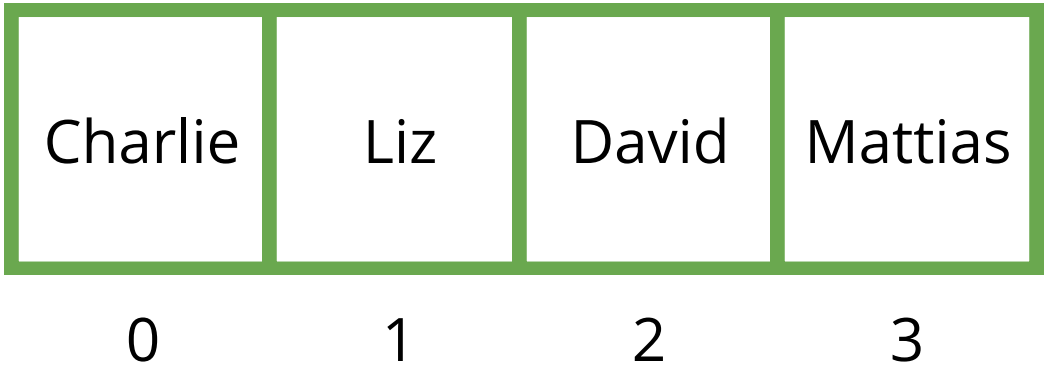
```
var friends = ["Charlie", "Liz", "David", "Mattias"];
```



Array are indexed starting at 0. Every slot has a corresponding number

Arrays

We can use those indices to retrieve data



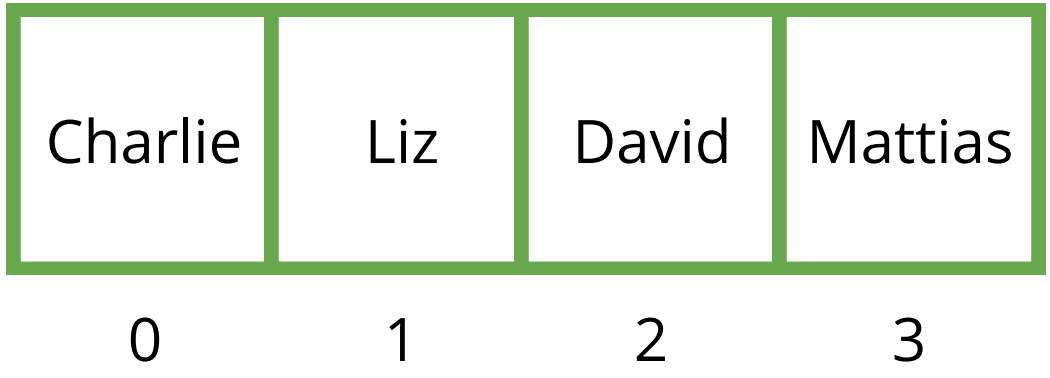

```
var friends = ["Charlie", "Liz", "David", "Mattias"];

console.log(friends[0])   //"Charlie"

friends[1] + " <3 " + friends[2]  //"Liz <3 David"
```

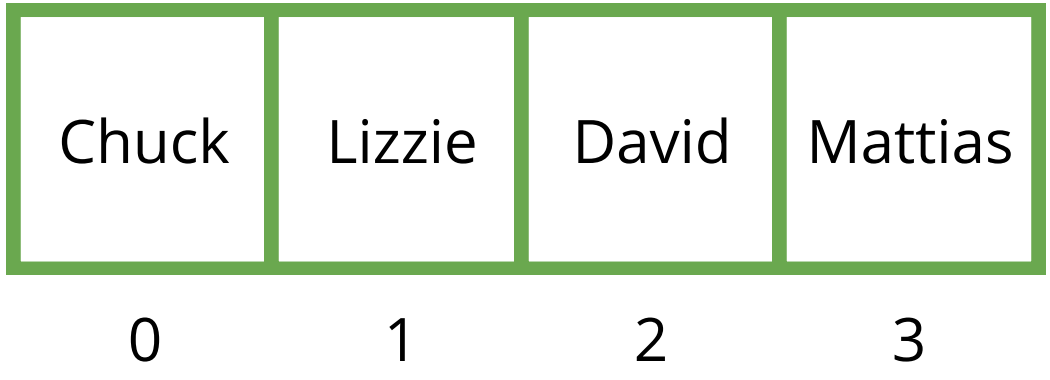
Arrays

We can also update arrays



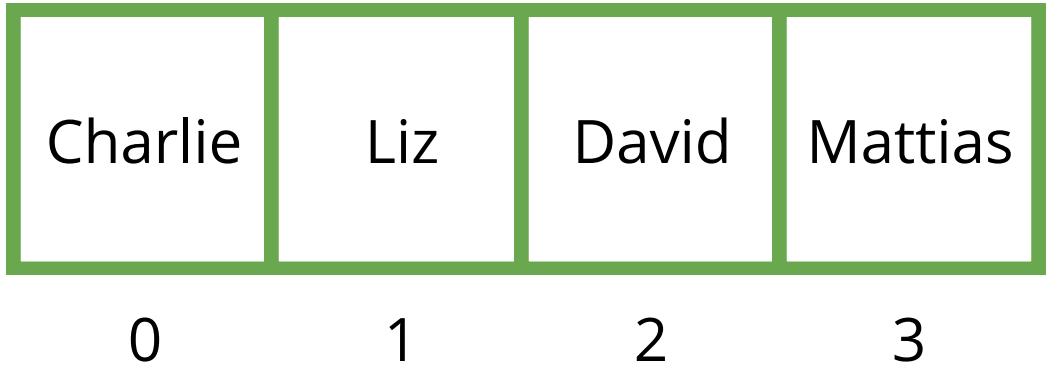

```
var friends = ["Charlie", "Liz", "David", "Mattias"];

friends[0] = "Chuck";
friends[1] = "Lizzie";
```

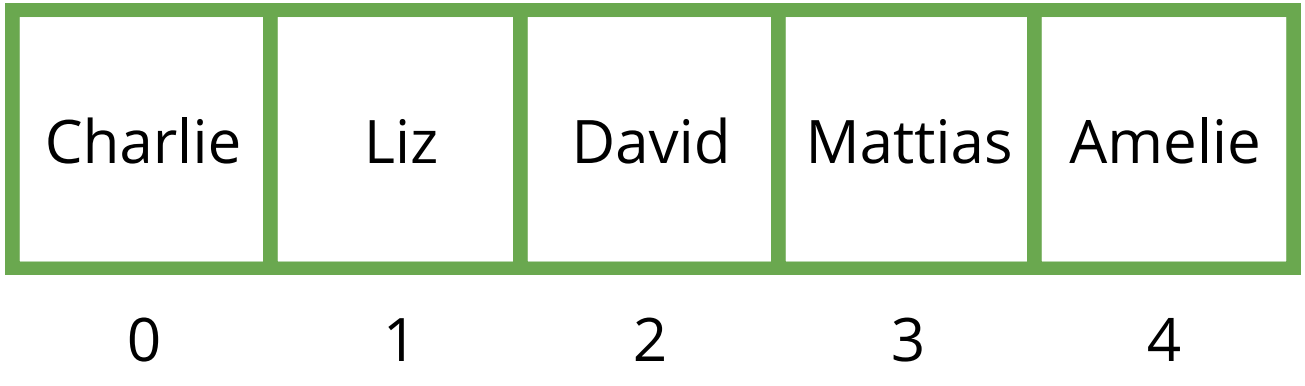


Arrays

We can also add new data




```
var friends = ["Charlie", "Liz", "David", "Mattias"];
friends[4] = "Amelie";
```



Arrays

Last few things

```
//We can initialize an empty array two ways:
var friends = []; //no friends :(
var friends = new Array() //uncommon

//Arrays can hold any type of data
var random_collection = [49, true, "Hermione", null];

//Arrays have a length property
var nums = [45,37,89,24];
nums.length   //4
```

Arrays

Exercise 1

```
var numbers = [22, 67, 33, 96, 88];

//What does the following line print?
console.log(numbers[numbers.length])
```

Arrays

Exercise 2

```
var friendGroups = [
  ["Harry", "Ron", "Hermione"],
  ["Malfoy", "Crabbe", "Goyle"],
  ["Mooney", "Wormtail", "Prongs"]
];

//What is the result of this line:
console.log(friendGroups[2][0]);
```

Strings

```
//Single or Double quotes OK
"hello world"
'hello world'

//Concatenation
"charlie" + "brown"  //"charliebrown"

//Escape Characters start with "\"
"Singin \"Do wah diddy, diddy, dum diddy do\" "
"This is a backslash: \\"

//Strings have a length property
"hello world".length  //11

//Access individual characters using [] and an index
"hello"[0]  //"h"
"hello"[4]  //"o"
```

Variables

```
//Variables are simply containers that store values
//They follow this pattern:
var yourVariableName = yourValue;

//They can store all of the values we've seen
var name = "Rusty";
var secretNumber = 73;
var isAdorable = true;

//Recall the stored value by calling the variable name
var name = "Rusty";
"hello there " + name    //"hello there Rusty"

var num = 37;
num + 3 + 10    //50

//We can also update existing variables
var name = "Bruce";
name = "Caitlyn";
```

Null and Undefined

```
//The two other primitives are null and undefined

//Variables that are declared but not
//initialized are undefined
//The following variables are undefined:
var name;
var age;

//null is "explicitly nothing"
var currentPlay = "charlie";
currentPlayer = null;   //game over
```