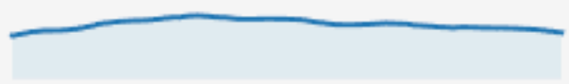
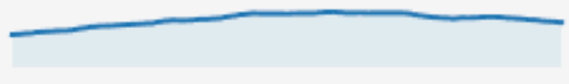
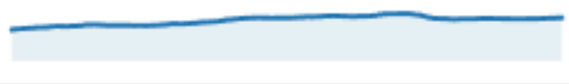
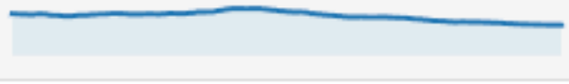
THE INTRODUCTION TO

JAVASCRIPT & VARIABLES

HISTORY

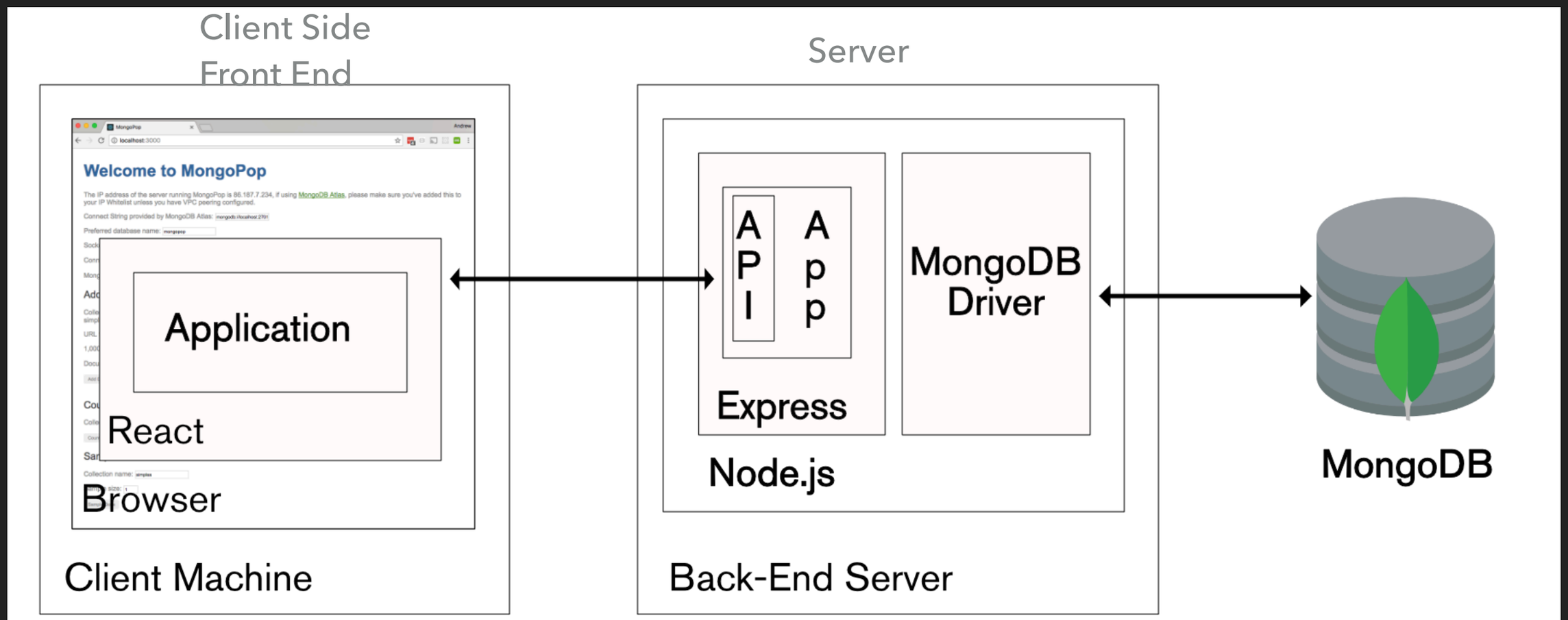
- Created by Brendan Eich in 1995 for Netscape Navigator 2 release.
- It only took Brendan Eich 10 days to create a working prototype.
- Originally Called Mocha - then LiveScript - then at the release of 1.1 it was changed to JavaScript.
- As JavaScript picked up popularity. A standardization of the language was needed to define the core implementation of the language.
- ECMA (European Computer Manufacturer's Association) or ECMAScript - defines the standards for scripting languages such as JavaScript.

WHY JAVASCRIPT

Rank	Language	Monthly Active Users	Trend
1	JavaScript	22.63%	
2	Python	14.75%	
3	Java	14.01%	
4	C++	8.45%	
5	C	6.03%	
6	PHP	5.85%	
7	C#	5.03%	
8	Shell	4.85%	
9	Go	4.10%	
10	TypeScript	3.89%	

WHY JAVASCRIPT (CONT)

- JavaScript was created out of the need to make websites more dynamic.
- It can be used across the full-stack.



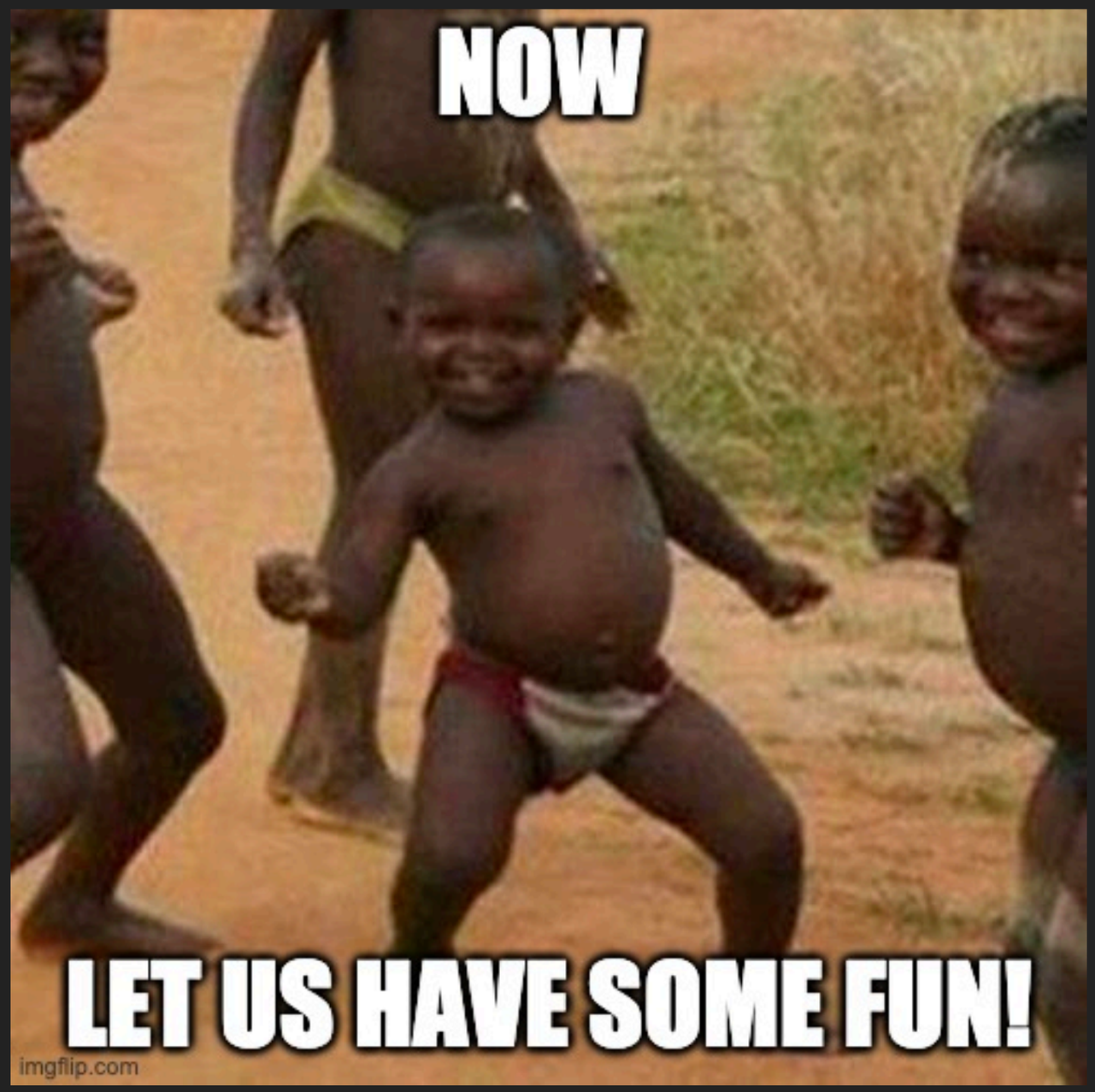
WHY JAVASCRIPT (CONT)

- Client Side - Makes requests, and displays what the users sees on a web page.
  - Examples of client side applications: Chrome - Safari - IE
- Server Side - Responds to requests made by the client, and sends the requested information to the client.
  - Examples of servers side applications: nodeJS - ExpressJS
- Database - Uses JSON (JavaScript Object Notation) documents to store information.
  - Examples of database applications that use JSON - PostgreSQL - mongoDB

WHY JAVASCRIPT (CONT)

- No context switching!!
  - This allows you to focus on using only JavaScript throughout the entire stack to create applications. This versatility makes JavaScript great for beginners.
- Web development has gravitated towards responsive, dynamic applications running in the browser.
- Examples: Gmail, Twitter, Facebook, Google Maps.

VARIABLES



DEFINITION:

Values / Variables are data types that have specific purpose and intrinsic behavior

PRIMATIVE		OBJECT	
Data Types	Syntax	Data Type	Syntax
STRING	“hello”	OBJECT	{key: ‘value’}
NUMBER	1234	ARRAY	[1,2,3]
BOOLEAN	true/false	DATE	New Date()
UNDEFINED	undefined	RegExp	/.*/g
NULL	null	FUNCTION	function(){}

- String data type:
  - Ability to manipulate the characters:

```
"hello world".toUpperCase()
// HELLO WORLD
```

  - toUpperCase is a method that can be used with String data type
- Number data type:
  - Ability to preform arithmetic operations

```
(1 + 2) * 5
// 15
```

JAVASCRIPT VARIABLES

- Variables:
  - Javascript gives us the ability to store data types with identifiers.

```
456789 * 456789 * 456789;
// 95311852611897070
```

```
var num = 456789;
num * num * num;
// 95311852611897070
```

```
var num = 456789;
var num2 = num * num * num;
num2;
// 95311852611897070
```

- The syntax behind Variables:
- Key word => label => assignment operator => value

```
var nameOfVariable = 'valueOfVariable';
```

- Behind the scene - Declaring a variable named nameOfVariable in memory (reserving some space)
- Next it assigns it to the String data type “valueOfVariable"

- CHALLENGE:
- Part 1: Declare four variables after your favorite animals
- Question… What are the variables' value?
- Part 2: After declaring all four variables, assign each one to a value

- Adding different data types will evaluate to different values
- Integers as we have seen before...

$$1 + 2 = 3$$

- Strings...

$$\text{“1”} + \text{“2”} = \text{“12”}$$

- This is called concatenation

CHALLENGE:

- Use four variables to concatenate four variables to create the string:

"1234567"

EXPRESSIONS:

- Any unit of code that can be evaluated to a value is an expression
- How Javascript interprets your code:

```
var randomNum = 10 + (15 * 2);
```

- Literal Expression = 3
- Arithmetic Expression = 2
- Assignment Expression = 1

Operator precedence

- Just like in mathematics, javascript has an order of how it evaluates each expression.
- To see all the different orders take a look here:
- http://www-lia.deis.unibo.it/materiale/JS/developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Operators/Operator_Precedence.html

Short Hand Syntax

	Shorthand	Standard
Increment	x++;	x = x + 1;
Decrement	x--;	x = x - 1;
Addition	x+= 5;	x = x + 5;
Subtraction	x-=5;	x = x - 5;