JavaScript I
Variables, operators and Pseudocode

Objective

- Using the JavaScript Console
- Variables and Declaring them
- Essential Data Types and Finding the Type
- Operators
- Creating a JS File
- Pseudocode

Introduction to JavaScript

- Usually used as a front-end scripting language.

- Can also be used for server-side programming.

- JavaScript is used to change web pages from static web pages to dynamic web pages.

HTML

HTML + CSS

HTML + CSS + JAVASCRIPT

JavaScript Background

- ECMAScript

  o ECMA is basically the organization that has designed JavaScript and ECMAScript is basically JavaScript

  o Based on technologies including JavaScript and JScript

  o Released in 1997 — a lot of changes have been made since

  o ES6 (aka ES2015 or Harmony) was the biggest revision

- ES6, ES7 and ES8

  o ES6 has introduced many changes to JavaScript (which basically is the same thing as ECMAScript) which improve the programming language greatly

  o ECMA now releases updates annually

👨‍💻👩‍💻 Using the JavaScript Console

- Open the Console

- Type the code `console.log("Hello World!!");` into the console

- Hit enter

```
    Elements   Console   Sources   Network   Timeline
🚫 🜏 top                      ▼  ☐ Preserve log
>  console.log("Hello, World!");
   Hello, World!
<- undefined
> |
```

- If everything isn't typed correctly, you may get an error, remember to follow your syntax rules!

```
>  console.log(Hello World);
⊗ Uncaught SyntaxError: missing ) after argument list
> |
```

Variables

- Programs usually process data that is put into the program and output the results of the processing.

- **Variables:** 'containers' to store the data we need to manipulate

- In calculations we use variables to hold values that can be changed

Declaring Variables

- to declare a variable is to use the keyword **'let'**. You can also use the keywords **'var'** and **'const'** to declare variables.

  - **var:** is the old way of creating a variable**.**

  - **const:** is used when you want to declare a variable that's value can't change

- To assign storage space in memory and give it a name we can reference it with:

  ```
  let variable_name = value_you_want_to_store;
  ```

Rules in naming variables

- Use camelCase style convention

- Contains only letters, numbers, underscores and dollar signs. No other characters can be used in variable names, including spaces. "My name" would thus not be an acceptable variable name.

  o Starts with a letter.

  o Is not a reserved word. In JavaScript, certain words are reserved. For example, you would not be able to name a variable "var", "console" or "log", because these are reserved words.
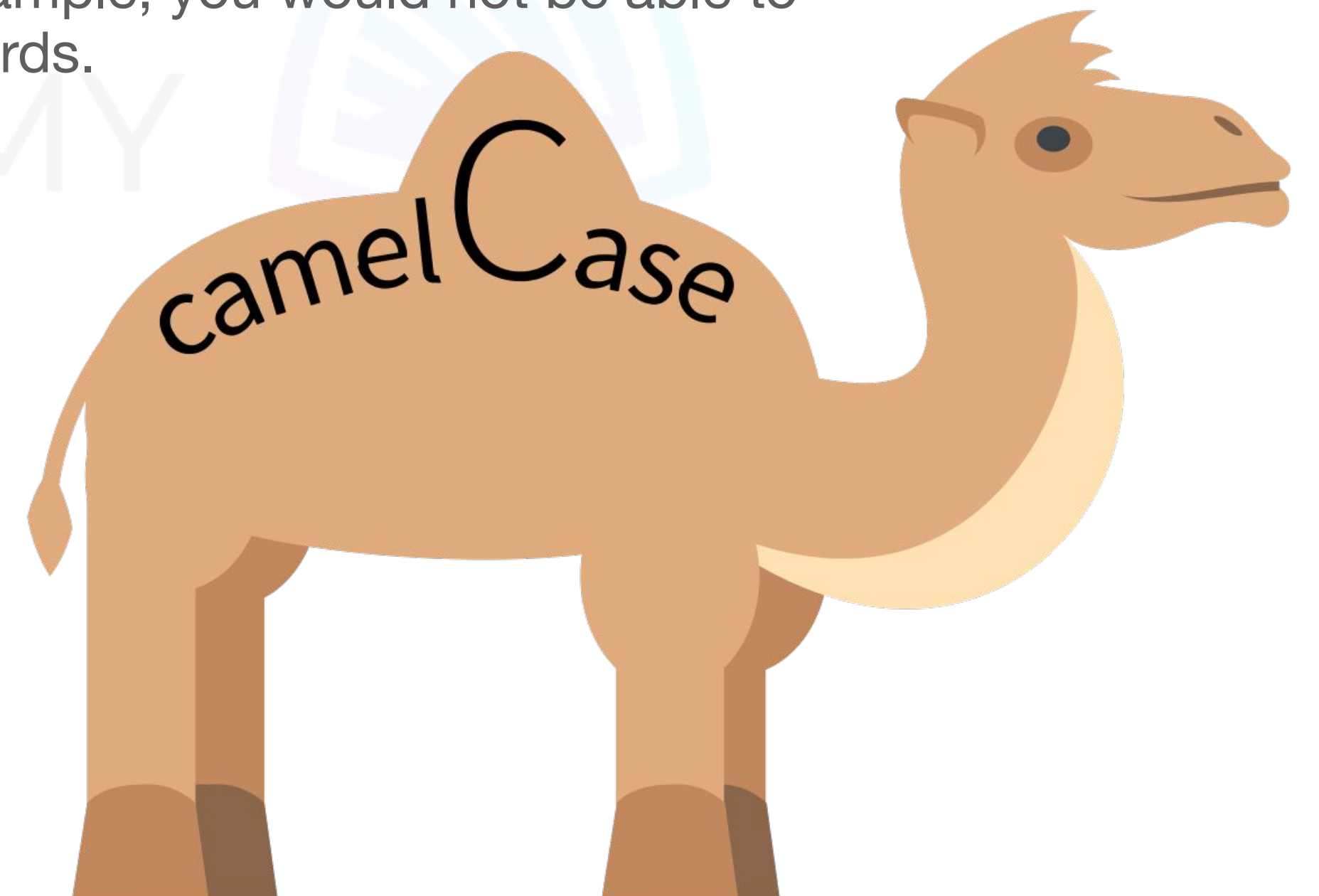
camelCase

Good✔ vs. bad ❌ naming conventions

Allowed		Not Allowed/ Not recommended	
`let userName`	Best Practice: camelCase	`let user_name`	Allowes but bad practice
`let ageGroup5`	Only letters and digits	`let 21Players`	Starting digits not allowed
`let $kindOfSpecial`	Starting with $ is allowed	`let user-b`	No special characters
`let userName`	Starting with _ is allowed	`let let`	Keywords not allowed

Google | 1000 طويق

Essential Data Types

- Data Type: the type of data that is stored by a variable

Data Type	Declaration
Numeric	`let someNumber = 25;`
String	`let someName = "Joe";`
Boolean	`let someBool = true;`
Array	`let someArray = [15, 17, 19];`
Object	`let someObject = {firstName: "James", lastName: "Bond"}`

Type Identification

- JavaScript is **Weakly typed programming language,** this mean its able to detect variable types automatically based on the value you assign to the variable

```
let myString = "12";


let myNumber = 12;
```

- 🤔So what will happen with this line of code?

```
let unknownType = 53 + "Bond";
```

Finding the Type

- We use the **typeof** built-in function

👨‍💻👩‍💻 **Enter the following code into your console and then press enter**

```
let myName = "Tom";
let num = 33.33;
let pass = true;

console.log(myName);
console.log(num);
console.log(pass);

let myDataType = typeof num;
console.log(myDataType);

console.log(typeof pass);
```

Mathematical Calculations with JavaScript

Arithmetic Operations	Symbol used in JavaScript
Addition	+
Subtraction	-
Multiplication	*
Division	/
Modulus (Divides left-hand operand by right-hand operand and returns remainder, e.g. 5%2 = 1)	%
Add one to a variable (e.g. 2++ = 3)	++
Subtract one from a variable (e.g. 2-- = 1)	--

👨‍💻👩‍💻 Type the following into your console.

```javascript
let num1 = 12;
let num2 = 34;

console.log("num1 = " + num2);
console.log("num2 = " + num2);
console.log("num1 + num2 = " + num1+num2);
console.log("num1 / num2 = " + num1/num2);
console.log("num2 % num1 = " + num2%num1);
console.log("num1++ = " + num1++);
console.log("num2-- = " + num2--);
```

👨‍💻👩‍💻 **Creating .js Files**

- Create a new file (using VS code)

- Enter the JavaScript instructions in this file one instruction per line

- Save this file with the .js extension.

- Save all your compulsory tasks as JavaScript files.

Pseudocode

- Pseudocode is a compact and informal high-level description of a program using the conventions of a programming language, but intended more for humans.

- Pseudocode is not an actual programming language. So it cannot be compiled into an executable program. It uses short terms or simple English language syntaxes to write code for programs before it is actually converted into a specific programming language.

- **Issue for driver license (Pseudocode If Else Example)**

```
1  BEGIN
2  NUMBER age
3
4  INPUT "Enter your age for driving licence"
5  OUTPUT age
6
7  IF age>=16  THEN
8      OUTPUT "You can take driving licence"
9  ELSE
10     OUTPUT "You can't take driving licence"
11 ENDIF
12
13 END
```

Pseudocode constructs

PSEUDOCODE CONSTRUCTS

SEQUENCE
Input: READ, OBTAIN, GET
Output: PRINT, DISPLAY, SHOW
Compute: COMPUTE, CALCULATE, DETERMINE
Initialize: SET, INIT
Add: INCREMENT, BUMP
Sub: DECREMENT

FOR
FOR iteration bounds
    sequence
ENDFOR

WHILE
WHILE condition
    sequence
ENDWHILE

CASE
CASE expression OF
condition 1: sequence 1
condition 2: sequence 2
...
condition n: sequence n
OTHERS:
default sequence
ENDCASE

REPEAT-UNTIL
REPEAT
    sequence
UNTIL condition

IF-THEN-ELSE
IF condition THEN
    sequence 1
ELSE
    sequence 2
ENDIF

EXTRA PSEUDOCODE CONSTRUCTS

CALLING CLASSES/ FUNCTIONS
CALL AvgAge with StudentAges
CALL Swap with CurrentItem and TargetItem
CALL getBalance RETURNING aBalance
CALL SquareRoot with orbitHeight RETURNING nominalOrbit

EXCEPTION HANDLING
BEGIN
statements
EXCEPTION
    WHEN exception
        statements to handle the exception
    WHEN another exception
        statements to handle the exception
END

Rules

1. Always capitalize the initial word (often one of the main 6 constructs).

2. Have only one statement per line.

3. Indent to show hierarchy, improve readability, and show nested constructs.

4. Always end multiline sections using any of the END keywords (ENDIF, ENDWHILE, etc.).

5. Use the naming domain of the problem, not that of the implementation. E.g., "Append the last name to the first name" instead of "name = first+ last."

6. Keep it simple, concise, and readable.

👨‍💻👩‍💻 **Write a pseudocode to Check a Number is Positive or Negative**

Resources

- [Pseudocode Examples](#)

- [Pseudocode 101: An Introduction to Writing Good Pseudocode | by Sara A. Metwalli](#)

- Academind

Summary

- Using the JavaScript Console
- Variables and Declaring them
- Essential Data Types and Finding the Type
- Operators
- Creating a JS File
- Pseudocode