Chapters 15-17

Scope, More types, and Exceptions
OH MY

CHAPTER BREAKDOWN

CHAPTER 15 SCOPE

SCOPE

BLOCK/LOCAL SCOPE
GLOBAL SCOPE
EXECUTION CONTEXT

BLOCK/LOCAL SCOPE: Variables that are initialized in a block or function have this scope. Can only be seen inside of that block or function

Global Scope: Variable declared in the main body of code. Also any variable not assigned let/const will be global because Javascript is default global scoped.

Execution Context: Condition under which the variable is executed aka Scope.

USING SCOPE

Variable Shadowing
Variable Hoisting

Variable Shadowing: Don’t do it. Try to keep your names specific to what you are using them for.

Variable Hoisting: Does not occur when using const or let. Var is the only variable type that gets hoisted. This is why we use let instead!

CHAPTER 16 MORE ON TYPES

Primitive Data Types

Undefined
Null

undefined is a primitive data type in JavaScript which is assigned to declared variables, which have *not* been initialized.

null is assigned to values that the programmer wishes to keep empty.

Chapter 17 EXCEPTIONS

17.1 Exception Intro

EXCEPTION INTRO

EXCEPTIONS AND ERRORS
ERROR OBJECT
COMMON EXCEPTIONS

EXCEPTION: A runtime error that returns an object with a name and message that supply information about why the error occurred

runtime errors and exceptions are interchangable

logic errors are not considered exceptoins

COMMON EXCEPTIONS:

- SyntaxError
- ReferenceError
- TypeOfError

17.2 Throw

THROW

You can throw a default error by using the throw statement.
You can also throw pre-existing errors

```
throw Error("You cannot divide by zero!");
```

```
Error: You cannot divide by zero!
at evalmachine.<anonymous>:1:7
at Script.runInContext (vm.js:133:20)
at Object.runInContext (vm.js:311:6)
at evaluate (/run_dir/repl.js:133:14)
```

```
throw SyntaxError("That is the incorrect syntax");
```

```
SyntaxError: That is the incorrect syntax
```

17.3 Exceptions as Control Flow

Exceptions as Control Flow

Control Flow
Catching Exceptions
Finally

Control Flow: The order in which statements are executed

Catching Exceptions: Utilize a try/catch statement to handle known exceptions. Catch Block only executes if there is an exception

Finally: Added at the end of a try/catch. Always executes no matter if an exception occurs or not.

QUESTIONS?