

# Built-ins and globals

node built in modules and global

# Lesson Plan

- node globals
  - **global**
  - **timers**
  - **errors**
  - **process**
- node built in modules
  - **fs**
  - **path**
  - **EventEmitter**
  - **http**

Summary of this lesson is in the following URL:

<https://www.nerdeez.com/articles/node/builtin-modules-globals>

# node globals

- globals are function, classes, and objects that can be used without importing any module
- Since there are quite a few we will go over the common ones

# global

- an object which contain all the globals
- when using a global function or class, node will look for its presence in the **global** object
- you can attach global variables by adding keys to the **global** object.
- in functions **this** is set to be **global** unless **use strict**
- in modules this is equal to **module.exports**

# Javascript

- **this**
- **use strict**

# Timers

- on the global scope we have timer functions
- timer will execute a function sometime in the future.
- The timer functions are:
  - **setTimeout**
  - **setInterval**
  - **clearTimeout**
  - **clearInterval**
- Lets try and use those functions.
- What is **this** when using timer
- lets try and use timer in a class and examine this...

# Javascript

- lambda functions
- this in lambda function vs this in regular functions.

# process

- supply data of the current node process
- **process.cwd()** - working dir from which the process invoked
- **process.env** - environment variable object

environment variables are critical for transferring sensitive information.

Lets try to create a program that prints and environment variable



# Errors

- errors extend the base class **Error**
- you can use **throw** and **try... catch** blocks
- There are common errors that extends the **Error** class like **SyntaxError** which you can use.
- You can create your own errors by extending the base **Error** class
- throwing an error without catching it will terminate the node process

Lets try and write a function which divides  $a/b$  and throws an error if  $b$  is 0.

# Built in modules

Build in modules are modules that are compiled in node binary and are available to use when we install node.

- Lets cover the most used built in modules.

# fs

- this module is in charge of dealing with the file system
- Among the things we can do:
  - read files
  - check if file exists
  - check directory and files in directory
  - delete file
- docs for module: <https://nodejs.org/dist/latest-v10.x/docs/api/fs.html>
- Most of the methods in this module come in an async and sync methods
- It's recommended to use the async methods
- Lets try and use this module to read a text file

# path

- provide utilities for working with file and directory paths
- lets try to use this module to create a string of the absolute path to a file

# EventEmitter

- you can import this class with **require('events')**
- instance of this class can shout when an event happens
- you can attach listeners to an event
- **this** in listener will be the instance of the EventEmitter
- common pattern in node is to extend EventEmitter when your object release async events
- common pattern is to throw an error event on errors
- if an error event is caught then it won't terminate the process otherwise it will terminate the process

# http

- with this module we can send requests and listen for requests
- We usually won't use this module to listen for requests rather use express or other framework which will use it.
- lets use this module to query nerdeez todo rest server at the URL:
  - <https://nztodo.herokuapp.com/api/task/?format=json>
- since it's https we better use the https module
- we will use the get method with a callback function to query the server
- the callback will get a response event emitter with a **data** event we can attach to and get the body of the response.

# Student EX

- create a text file with a url to nereez rest server.
- usefs module to grab the text content asynchronously from that file
- the text content is the url to query so use **https** module to query the server with the url and print the result to the console

# Summary

Although there are many globals and built in modules we tried to cover the most used ones.

More data on them are available on the official docs:

<https://nodejs.org/dist/latest-v10.x/docs/api/>