

Process 객체

2022년 10월 8일 토요일 오후 6:06

[전역 변수]

: 아무런 변수를 선언하지 않고 사용 할 수 있는 것

- 문자열 자료형의 전역 변수

변수	설명
__filename	현재 실행 중인 코드의 파일 경로 나타낸다
__dirname	현재 실행 중인 코드의 폴더 경로를 나타낸다

[process 객체의 속성과 이벤트]

- Process 객체의 속성

속성	설명
Env	컴퓨터 환경 정보 나타낸다
Version	Nnode.js 버전을 나타낸다
Versions	Node.js와 종속된 프로그램 버전 나타낸다
Arch	프로세서의 아키텍처 나타낸다
Platform	플랫폼을 나타낸다

- Process 객체의 메소드

메소드	설명
Exit([exitCode = 0])	프로그램을 종료
memoryUsage()	메모리 사용 정보 객체를 리턴
Uptime()	현재 프로그램이 실행된 시간 리턴

[process 객체와 이벤트 개요]

- Node.js의 이벤트 연결 메소드

메소드	설명
On(이벤트이름, 이벤트핸들러)	이벤트 연결

이벤트핸들러: 이벤트가 발생했을 때 호출할 함수 의미

- Process객체의 이벤트

이벤트	설명
Exit	프로세스가 종료될 때 발생
uncaughtException	예외가 일어날 때 발생

Os 모듈

2022년 10월 9일 일요일 오후 7:06

- 앱을 만들 때 활용하지는 않는다.

```
Const os = require('os');
```

- Os 모듈의 메소드

메소드	설명
Hostname()	운영 체제의 호스트 이름 리턴
Type()	운영체제의 이름 리턴
Platform()	운영 체제의 플랫폼 리턴
Arch()	운영체제의 아키텍처를 리턴
Release()	운영체제의 버전리턴
Uptime()	운영체제가 실행된 시간을 리턴
Loadavg()	로드애버리지 정보를 담은 배열리턴
Totalmem()	시스템의 총메모리를 리턴
Freemem()	시스템의 사용 가능한 메모리를 리턴
Cpus()	Cpu의 정보를 담은 객체를 리턴
getNetworkInterfaces()	네트워크 인터페이스의 정보를 담은 배열을 리턴.

Url 모듈

2022년 10월 9일 일요일 오후 7:13

모듈 추출

```
Const url = require ('url');
```

- 메소드

메소드	설명
Parse(urlStr [, parseQueryString = false, slashesDenoteHost =false])	URL 문자열을 URL 객체로 변환해 리턴
Format(urlObj)	URL 객체를 URL 문자열로 변환해 리턴
Resolve(from, to)	매개 변수를 조합하여 완전한 URL 문자열을 생성해 리턴

이중 parse() 메소드 자주 사용함

모듈

2022년 10월 9일 일요일 오후 7:18

File system

Const fs = require('fs');

- 파일 읽기 메소드

메소드	설명
fs.readFileSunc(파일이름)	동기적으로 파일을 읽는다
fs.readFile(파일이름, 콜백함수)	비동기적으로 파일을 읽는다

- 파일 쓰기

메소드	설명
fs.writeFileSync(파일 이름, 문자열)	동기적으로 파일 쓴다
fs.writeFile(파일이름, 문자열, 콜백함수)	비동기적으로 파일 쓴다

** 비동기 처리 기억하기

[Request 모듈]

-
- 웹 요청을 쉽게 만들어 주는 모듈
- 개인이 제공하는 외부 모듈
- npm install request 로 설치

```
> const request = require('request');
Uncaught Error: Cannot find module 'request'
Require stack:
- <repl>
    at Function.Module._resolveFilename (node:internal/modules/cjs/loader:956:15)
    at Function.Module._load (node:internal/modules/cjs/loader:804:27)
    at Module.require (node:internal/modules/cjs/loader:1028:19)
    at require (node:internal/modules/cjs/helpers:102:18) {
  code: 'MODULE_NOT_FOUND',
  requireStack: [ '<repl>' ]
}
```

```
> const request = require('request');
Uncaught SyntaxError: Identifier 'request' has already been declared
>
```

[노드 패키지 매니저 npm]

- 프로그래밍 플랫폼이 기본적으로 제공하는 모듈을 내부 모듈이라고 한다
- 개인이 내부 모듈을 조합해서 사용하기 쉬운 형태로 만들거나 새로운 기능 구현해서 제공하는 것을 외부 모듈이라고 한다
- Npm은 node가 설치될 때 함께 설치 됨

npm install 모듈이름

예 : npm install express@4

[Cheerio모듈]

- Request 모듈로 가져온 웹 페이지는 단순한 HTML 문자열이다
- 여기서 정보를 추출해야 단순 데이터가 정보가 된다.
- 이 과정을 파싱이라고 한다.
- 명령어 : npm install cheerio

// 모듈 추출

Const cheerio = require('cheerio');

위 모듈을 사용하면 자신이 원하는 정보 쉽게 가져올 수 있음

날씨정보 확인, 주식정보 가져오는 활용 범위가 넓음

[async 모듈]

- Node.js 는 대부분의 메소드가 비동기적으로 구성되어 실행순서를 정의하기 어렵고 들어쓰기가 많다 이문제를 해결해 주는 모듈
- Npm install asvnc
- // 모듈을 추출
- Const async= require('async');
- Parallel()메소드
 - 문제확인하기: 자바스크립트 대부분의 비동기적 수행으로 콜백 지옥 발생 함수를 여러 개 들어 쓰기 하여 코드를 보기가 힘든 상태 의미
 - 문서 확인하기 : parallel()메소드 사용하면 자바스크립트 문제로 지적되는 콜백지옥 해결 가능
 - 코드 실행 전 npm install async 입력해 모듈 설치