

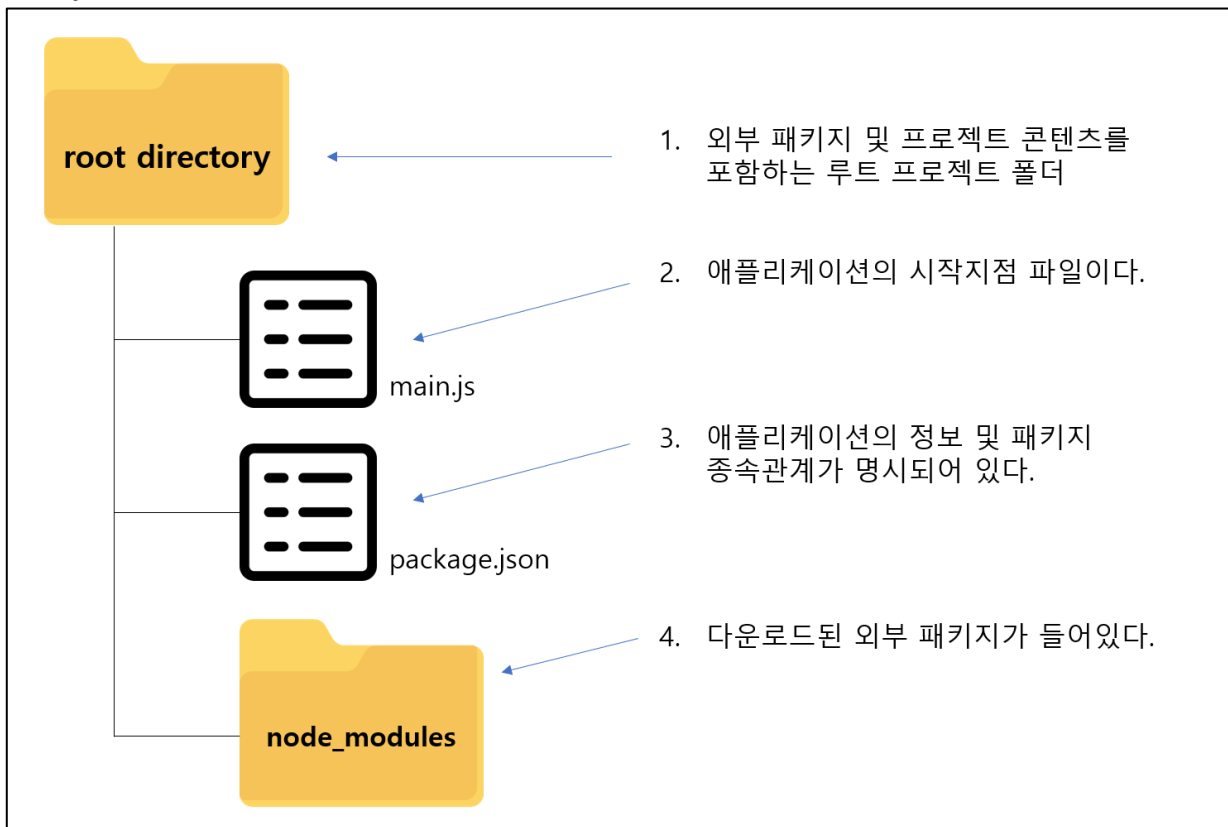
온라인 모듈

Node.js에서의 모듈이란?

- 앱을 체계적이고 효율적으로 유지하기 위한 라이브러리내 최소단위 파일.
즉, 모듈은 단 하나의 콘텐츠 및 기능을 동작하는 JavaScript 파일이다.
- Node.js에서는 npm을 사용하여 온라인으로 패키지 및 모듈, 라이브러리를 다운/업로드를 할 수 있다.
(npm은 Python의 pip와 비슷한 역할을 한다.)

애플리케이션 초기화

- Node.js는 앱이나 모듈은 프로젝트의 특성을 기술하고 있는 [package.json](#)파일을 포함한다.
- Node.js는 앱을 만드는 초기단계에서 [npm init](#)라는 명령어를 사용하여 프로젝트 이름, 애플리케이션 버전, 간단한 설명, 앱 시작지점 파일명, 테스트 파일, Git 저장소, 작성자명, 라이선스 코드를 명시한다. 명시를 하게 되면 해당 디렉터리에는 [package.json](#)파일과 [node_module](#)디렉터리가 생성된다.
- [node_module](#)디렉터리에는 온라인에서 다운로드 받은 모든 외부 패키지가 들어있다.
- Node.js 애플리케이션 구조



예제1) 애플리케이션 초기화 및 다운받은 외부 패키지 사용.

1. 애플리케이션 초기화를 위한 애플리케이션 루트 디렉터리 생성
: mkdir (프로젝트 or 애플리케이션 이름)
2. 생성된 디렉터리로 이동
: cd (1.에서 생성한 디렉터리 명)
3. 애플리케이션 초기화
: npm init
4. npm init를 입력하게 되면 프로젝트의 특성을 입력하라는 프롬프트가 나온다.
 - 4.1 그 중에서 entry point와 author만을 입력한다.
 - 4.2 entry point는 main.js를 기입하고
 - 4.3 author에는 자신의 이름을 기입한다.
 - 4.4 두 항목을 제외한 나머지 항목은 모두 디폴트값 Enter로 그냥 넘어가면 된다.
5. npm init를 실행 후에 해당 디렉터리에는 package.json이라는 파일이 생성된다.

✓ npm init 실행화면

```
wlgns12www@nodejs-test:~/nodejs/2_module/application_test$ npm init
This utility will walk you through creating a package.json file.
It only covers the most common items, and tries to guess sensible defaults.

See `npm help json` for definitive documentation on these fields
and exactly what they do.

Use `npm install <pkg>` afterwards to install a package and
save it as a dependency in the package.json file.

Press ^C at any time to quit.
package name: (application_test)
version: (1.0.0)
description: app init
entry point: (index.js) main.js
test command:
git repository:
keywords:
author: INLAB_Ji-Hun
license: (ISC)
About to write to /home/wlgns12www/nodejs/2_module/application_test/package.json:
{
  "name": "application_test",
  "version": "1.0.0",
  "description": "app init",
  "main": "main.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "author": "INLAB_Ji-Hun",
  "license": "ISC"
}
```

→ package.json

※ 여기까지 npm init를 이용한 애플리케이션 초기화가 완료되었다.

이후에는 온라인으로 패키지를 다운받아 사용해보는 예제를 수행하는 내용이다.

6. 온라인으로 cities라는 외부 패키지를 다운로드

: npm install cities --save

✓ npm install cities --save 실행화면

```
wlgns12www@nodejs-test:~/nodejs/2_module/application_test$ npm install cities --save
npm notice created a lockfile as package-lock.json. You should commit this file.
npm WARN application_test@1.0.0 No repository field.

+ cities@2.0.0
added 3 packages in 0.937s
wlgns12www@nodejs-test:~/nodejs/2_module/application_test$ ls
node_modules package-lock.json package.json
```

: 다운받은 외부 패키지를 저장하는 node_modules라는 디렉터리가 새로 생성되는 것을 확인가능.

: node_modules에는 다운받은 cities패키지가 저장되어 있다.

: 또한, package.json파일에는 애플리케이션이 종속하는 패키지 이름 및 버전이 추가로 명시되어 있다.

7. vi main.js

: 프로젝트의 root directory에 main.js를 생성

8. 아래 박스의 내용과 같이 main.js에 입력 후 저장

```
"use strict"

const cities = require("cities");           // cities 패키지 요청
var myCity = cities.zip_lookup("10016");    // zip_lookup메소드의 결과를 myCity에 할당
console.log( myCity );                     // myCity의 내용 출력
```

9. main.js를 node명령으로 실행

✓ 실행화면

```
wlgns12www@nodejs-test:~/nodejs/2_module/application_test$ ls
main.js node_modules package-lock.json package.json
wlgns12www@nodejs-test:~/nodejs/2_module/application_test$ node main.js
{ zipcode: '10016',
  state_abbr: 'NY',
  latitude: '40.746180',
  longitude: '-73.97759',
  city: 'New York',
  state: 'New York' }
```

참고문헌

- 조나단 웨슬러, 김성준. (2020.01.31). NODEJS로 프로그래밍 시작하기. 74-77.