

CodeIgniter와 Composer 를 이용한 프로젝트 개발

코드이그나이터 한국 사용자 포럼
한대승(불의회상)



발표자 소개

- 한대승(불의회상)
- Codeigniter 한국 사용자 포럼 운영진
- 두 아이의 아빠, 주말부부 10년차
- PHP, JAVA, NODE JS, GO, JAVASCRIPT
- 다양한 프레임워크

Index

- Codeigniter란?
- Composer란?
- Codeigniter와 Composer 연동
- 연동 시연
- 마치며

Codeigniter란?

Codeigniter란?

프레임워크 (Framework)

프레임워크란?

- '뼈대나 근간'을 이루는 코드들의 묶음



- 프로그래밍에 필요한 응용 프로그램 표준 구조를 구현하는 클래스와 라이브러리 모음

프레임워크? 라이브러리?

특징	프레임워크	라이브러리
유저코드의 작성	프레임워크 클래스를 서브클래싱 해서 작성	독립적으로 작성
호출흐름	프레임워크코드가 유저코드를 호출	유저코드가 라이브러리를 호출
실행흐름	프레임워크가 제어	유저코드가 제어
객체의 연동	구조프레임워크가 정의	독자적으로 정의

〈표 1〉 프레임워크와 라이브러리의 비교

왜 프레임워크를 사용하는가?

- 재사용할 수 있는 수많은 코드를 통합, 개발자가 새로운 애플리케이션을 위한 표준 코드를 다시 작성하지 않아도 됨 (재사용성의 용이)
- 교육이 용이함 (이미 알고 있다면 땡큐!)
- 일정한 품질의 결과물을 얻을 수 있음 (기업의 입장)
- 완성된 구조에 자신이 맡은 부분만 개발함 (개발자의 입장)
- 표준화된 구조를 통한 협업과 분업 (개발시간의 단축)

PHP 프레임워크



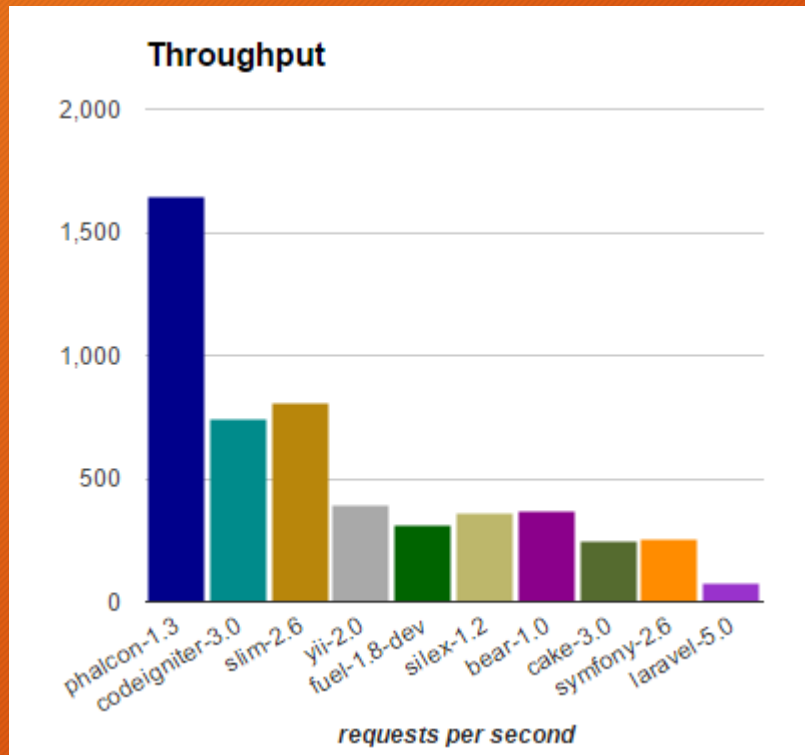
왜 Codeigniter 인가?

- 최소노력으로 사용할수있는 프레임워크
- 거의 설정이 필요없는 프레임워크
- 커맨드라인에서 뭔가를 입력할 필요가 없는 프레임워크
- 제한적인 코딩 규칙이 없는 프레임워크
- 템플릿 언어를 억지로 배우고싶지 않다.
- 복잡한것 대신 심플한 솔루션을 좋아한다.
- 명확하고 전체내용을 몽땅 다 설명 해주는 설명서를 원한다.
- 뭔가 문제가 생겼을때 도움을 받길 원한다.

Codeigniter에 적용된 디자인패턴

- Factory 패턴
- Singleton 패턴
- Strategy 패턴
- Front Controller 패턴
- MVC(Model-View-Controller) 패턴
- Active record 패턴

놀라운 프로그램 성능



- CentOS 6.5 64bit (VM)
- PHP 5.5.23
- Zend OPcache v7.0.4-dev
- Apache 2.2

- Phalcon 1.3.4
- CodeIgniter 3.0-rc3
- Slim 2.6.2
- Yii 2.0.4
- FuelPHP 1.8-dev
- Silex 1.2.3
- BEAR.Sunday 1.0.0-rc2
- CakePHP 3.0.0
- Symfony 2.6.5
- Laravel 5.0.20

Composer란?

Composer란?

패키지 관리 시스템

훌륭한 PHP 의존성 관리자

```
▼ codeigniter
  ▼ framework
    ▶ application
    ▶ system
    ▶ user_guide
    ▶ composer.json
    ▶ contributing.md
    ▶ index.php
    ▶ license.txt
    ▶ readme.rst
```

```
▼ meenie
  ▼ javascript-packer
    ▶ class.JavaScriptPacker.php
    ▶ class.JavaScriptPacker.php4
    ▶ composer.json
    ▶ example-file.php
    ▶ example-inline.php
    ▶ readme.txt
```




composer.json

```
▼ websharks
  ▼ css-minifier
    ▶ src
    ▶ CHANGELOG.md
    ▶ composer.json
    ▶ composer.lock
    ▶ LICENSE.txt
    ▶ README.md
  ▼ js-minifier
    ▶ src
    ▶ CHANGELOG.md
    ▶ composer.json
    ▶ composer.lock
    ▶ LICENSE.txt
    ▶ README.md
```

어떻게 관리 되는가?

Packagist *The PHP Package Repository* Browse Submit hoksi



Packagist is the main [Composer](#) repository. It aggregates public PHP packages installable with Composer.

Getting Started

Define Your Dependencies

Put a file named `composer.json` at the root of your project, containing your project dependencies:

```
{
  "require": {
    "vendor/package": "1.3.2",
    "vendor/package2": "1.*",
    "vendor/package3": "^2.0.3"
  }
}
```

For more information about packages versions usage, see the [composer](#)

Publishing Packages

Define Your Package

Put a file named `composer.json` at the root of your package, containing this information:

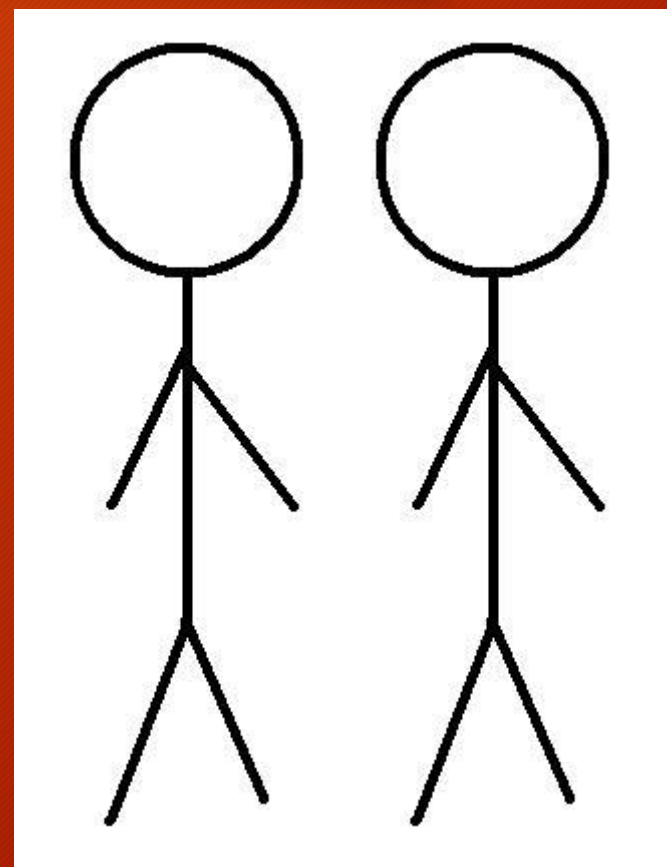
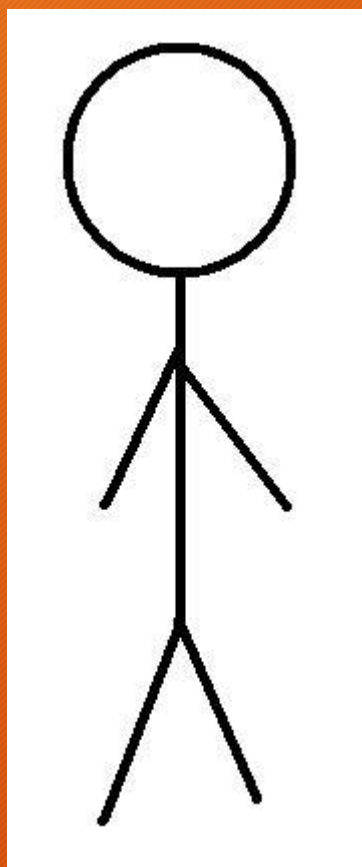
```
{
  "name": "your-vendor-name/package-name",
  "description": "A short description of what your package does",
  "require": {
    "php": ">=5.3.0",
    "another-vendor/package": "1.*"
  }
}
```


프로젝트에 어떻게 사용 하는가?

```
hoksi:~/workspace $ composer require twig/twig:~1.8
You are running composer with xdebug enabled. This has a
./composer.json has been updated
Loading composer repositories with package information
Updating dependencies (including require-dev)
```

```
hello.php
1 <?php
2 require_once "vendor/autoload.php";
3
4 |
```

패키지를 공유하고 싶다면?



의존성 업데이트는 어떻게?

```
hoksi:~/workspace/_core $ composer update
```

```
You are running composer with xdebug enabled. This has a major impact
```

```
Loading composer repositories with package information
```

```
Updating dependencies (including require-dev)
```

- Removing matthiasmullie/path-converter (1.0.7)
- Removing codeigniter/framework (3.0.3)
- Installing codeigniter/framework (3.0.4)

```
Downloading: 100%
```

```
Writing lock file
```

```
Generating autoload files
```

```
hoksi:~/workspace/_core $
```

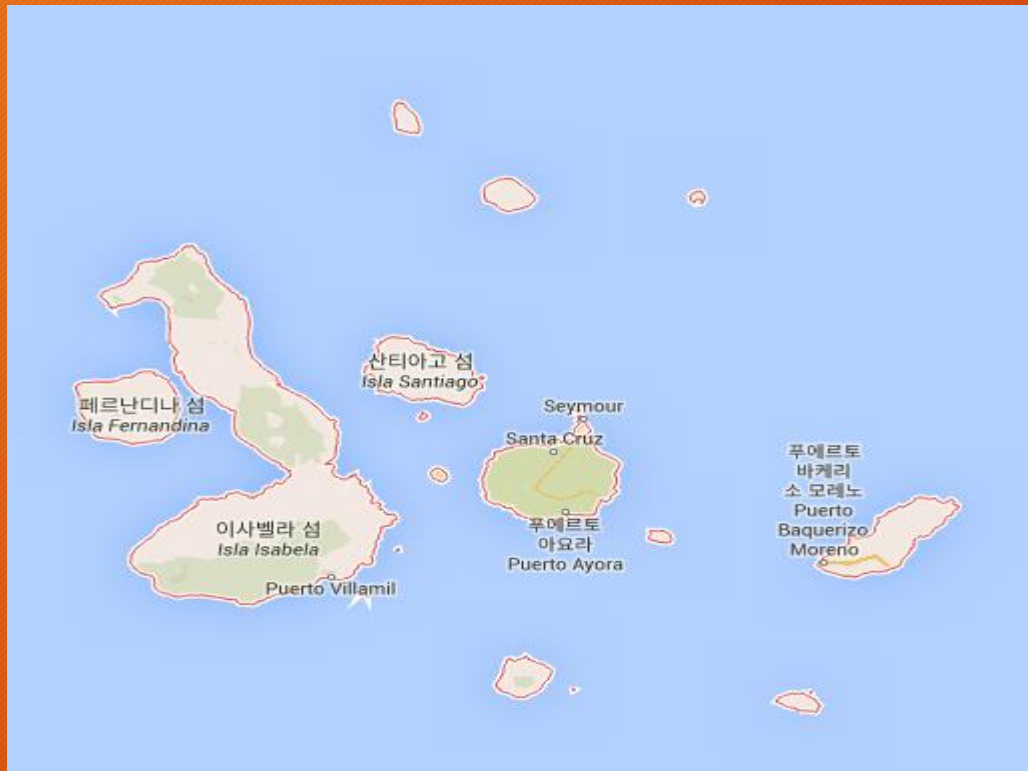
Composer가 궁금하다면?

- Composer 공식 사이트
- <https://getcomposer.org/>

- Composer 한글 메뉴얼 사이트
- <https://xpressengine.github.io/Composer-korean-docs/>

Codeigniter와 Composer 연동

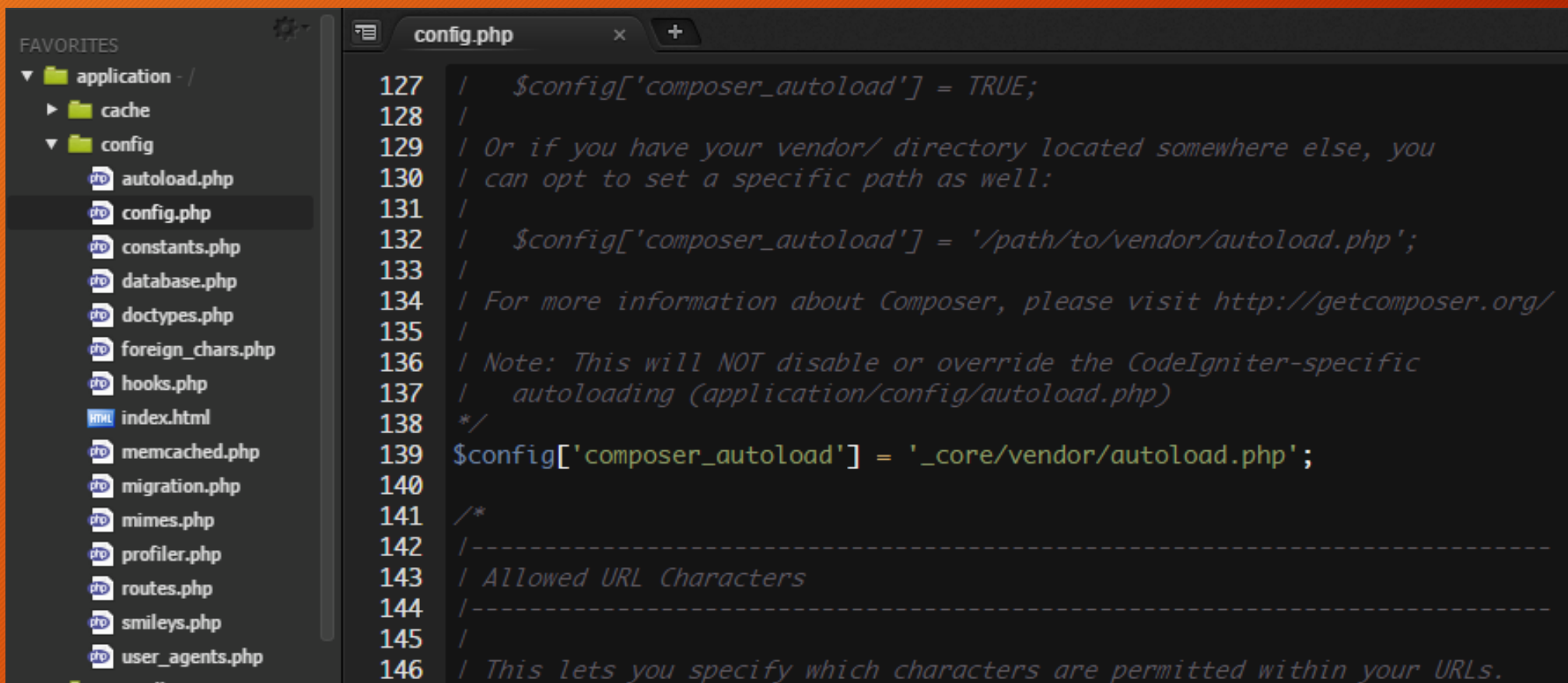
왜? 연동하려 하는가?



갈라파고스

에콰도르

어떻게 연동 하는가?




The image shows a code editor window with a sidebar on the left displaying a file tree. The main window shows the content of the `config.php` file. The file tree includes folders for `application`, `cache`, and `config`, with various PHP files listed under `config`. The code in the main window is as follows:

```
127 | $config['composer_autoload'] = TRUE;
128 |
129 | / Or if you have your vendor/ directory located somewhere else, you
130 | can opt to set a specific path as well:
131 |
132 | $config['composer_autoload'] = '/path/to/vendor/autoload.php';
133 |
134 | / For more information about Composer, please visit http://getcomposer.org/
135 | /
136 | / Note: This will NOT disable or override the CodeIgniter-specific
137 | autoloading (application/config/autoload.php)
138 | */
139 | $config['composer_autoload'] = '_core/vendor/autoload.php';
140 |
141 | /*
142 | -----
143 | Allowed URL Characters
144 | -----
145 | /
146 | / This lets you specify which characters are permitted within your URLs.
```

Composer를 통한 설치

Packagist Browse Submit

 ↓ ★

Packagist is the main Composer repository. It aggregates public PHP packages installable with Composer.

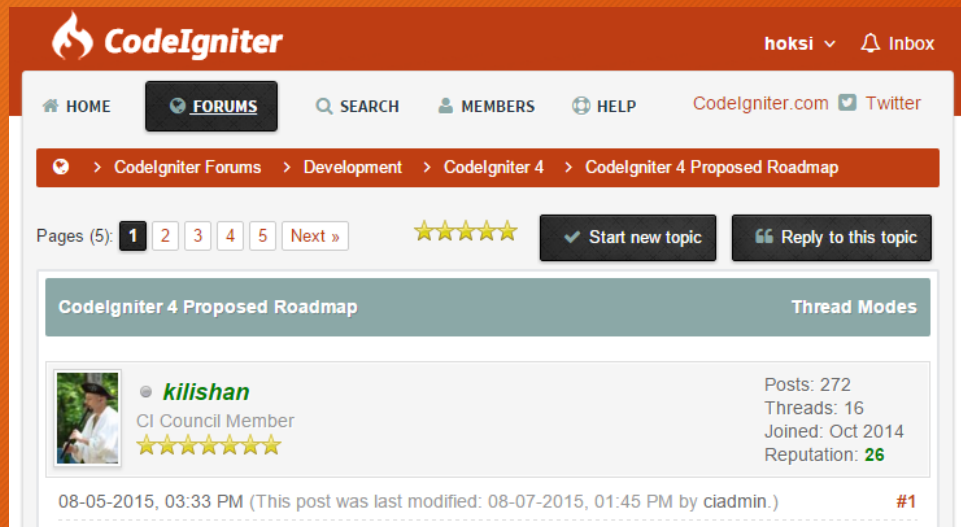
codeigniter/framework	PHP	↓ 11 070
The CodeIgniter framework		★ 11 450

★ **codeigniter/framework**

↓ `composer require codeigniter/framework`

The CodeIgniter framework

Codeigniter 4



The screenshot shows a forum post on the CodeIgniter website. The page header includes the CodeIgniter logo, the user name 'hoksi', and an 'Inbox' notification. The navigation bar contains links for HOME, FORUMS, SEARCH, MEMBERS, and HELP, along with links to CodeIgniter.com and Twitter. The breadcrumb trail indicates the path: CodeIgniter Forums > Development > CodeIgniter 4 > CodeIgniter 4 Proposed Roadmap. Below the breadcrumb, there are page navigation options (Pages 5: 1, 2, 3, 4, 5, Next), a star rating of five stars, and buttons for 'Start new topic' and 'Reply to this topic'. The main content area shows the thread title 'CodeIgniter 4 Proposed Roadmap' and a 'Thread Modes' button. The post is by user 'kilishan', a CI Council Member with a five-star reputation. The user's profile statistics are: Posts: 272, Threads: 16, Joined: Oct 2014, Reputation: 26. The post content is: '08-05-2015, 03:33 PM (This post was last modified: 08-07-2015, 01:45 PM by ciadmin.) #1'.

The application and system directories will both work with PSR-4 autoloading. CodeIgniter will use our own autoloader, as well as integrate Composer for those times you need it.

시연

클라우드 IDE 구름(https://www.goorm.io)

구름이란? 개발용 교육용 Contact 도움말 한국어

Email Password 아이디 기억 로그인 비밀번호 재설정 회원가입

오픈 베타 사용자를 모집 중입니다! 지금 회원가입하세요! - 바로가기

구름IDE v2.2 - 160112이 릴리즈 되었습니다! 확인해 보세요! - 확인하기

언제 어디서나 프로그래밍을 배우세요!
소프트웨어를 개발하세요!

이제, 때와 장소를 가리지 않고 프로그래밍을 배우고, 소프트웨어를 개발하세요!
클라우드 코딩 서비스 구름이 여러분들로부터 섬뚱는 창의적인 아이디어들을
언제 어디서서나 현실로 만들 수 있도록 도와드릴 것입니다.
구름EDU를 이용하여 초등학교/중학교/고등학교/대학교를 비롯한 다양한 교육 기관에서 프로그래밍 언어를 효율적으로 교육하는데 사용하세요!
구름을 이용하여 다른 개발자와 효율적으로 협업하세요!
더 적은 시간으로 더 재미있고 효과적인 개발을 하실 수 있습니다.

온라인 빌드/컴파일/디버깅
다양한 언어 지원
실시간 협업
교육용 버전

웹 브라우저만 있으면 언제 어디서나 클라우드를 직접하고 싶으신가요?
C/C++, JAVA, nodejs, JSP, PHP,
다른 개발자들과 힘을 모으고 싶으신가요?
코딩 교육을 위한 전용 기능을 갖추고 있습니다.

시연순서

- 연동 결과 보기
- 컴포저 설치
- 컴포저를 이용한 Codeigniter 설치
- Codeigniter와 Composer 연동
- QR code 컴포넌트 설치
- QR code 샘플 프로그램 작성
- Zend Bar code 컴포넌트 설치
- Bar code 샘플 프로그램 작성

마치며