

# Ch7. 웹 개발: 통합하기 (Putting it all together)

- summarized by 전승일(2014/11/09)
- source code: <http://www.headfirstlabs.com/books/hfpython/>
- source code: \$ git clone <https://github.com/sijeon/HeadFirstPython>
- support site: <http://python.itcarlow.ie>

## 함께 하는 것은 좋은 것입니다

---

- 모든 사람이 감독의 데이터를 보고 싶어 하네요!

## 프로그램을 웹에 올릴 수 있어요

---

- 여러분의 프로그램은,

- \* 여러분의 웹사이트를 접속할 수 있는 사람은 누구나 사용할 수 있고
- \* 여러분의 웹 서버 한 곳에만 놓으면 되고
- \* 새로운 기능이 필요할 때 업데이트하기도 쉽습니다

- 웹 기반 애플리케이션: 웹앱 (참고: 252 ~ 253쪽)

## 여러분이 만들 웹앱은 뭘 해야 할까요?

---

- 무언가 보여 주는 멋지고 친근한 홈페이지
- 모든 선수의 이름을 보여주고 선택할 수 있는 페이지
- 선택한 선수의 기록을 보여주는 페이지

## MVC 모델로 웹앱 설계하기

---

- 이것들을 만드는 가장 좋은 방법은 무엇일까? 다양하다...
- 좋은 웹앱은 MVP 패턴을 따른다

```
* Model      : 웹앱의 데이터를 저장하거나 처리하는 코드
* View       : 웹앱의 UI를 만들고 출력하는 코드
* Controller : 웹앱의 뷰와 모델을 연결하고, 앱의 논리 처리를 담당하는 코드
```

## 데이터 모델링하기

---

- `get_coach_data()` 구현

```
* 텍스트 파일을 AthleteList 객체로 변환
```

- `put_to_store()` 구현

```
* file 목록을 받아 각 file별로 get_coach_data()를 호출하여 AthleteList를 결과로 받아서,
* 전체 선수 데이터를 선수 이름에 의해 색인된 dictionary로 변환하고,
* 이를 pickling 해서 파일에 저장
```

- `get_from_store()` 구현

```
* pickle에 저장된 데이터를 웹앱에 dictionary 객체로 불러들임
```

- 실습

```
webapp-ch07/cgi-bin/athletelist.py 와 이전 6장의 마지막 코드 비교
webapp-ch07/cgi-bin/athletemodel.py 확인
webapp-ch07/cgi-bin/athletelist_test.py로 개별 동작 시험 해 보기
```

## UI 살펴보기

---

- HTML 관련서: Head First HTML
- HTML helper module: `yate.py`, `yate_test.py`

## 코드 제어하기

---

- 폴더 구조: 270쪽 참조

## 웹서버는 CGI를 사용해서 프로그램을 실행합니다

---

- Common Gateway Interface
- web server에서 cgi를 활성화 해야 함
- 간단한 웹서버: simple\_httpd.py

## 선수 목록 출력하기

---

- 동적 콘텐츠 생성: generate\_list.py

## 또 다른 CGI 스크립트 만들기

---

- Form Data를 읽고 동적 콘텐츠 생성하기 - CGI script: generate\_timing\_data.py

```
import cgi
form_data = cgi.FieldStorage()
athlete_name = form_data['which_athlete'].value
```

## CGI 추적 기능을 활성화하면 에러를 찾기 쉽습니다

---

- CGI 스크립트에서 예외가 발생하면 Python program은 에러 메시지를 STDERR(표준 에러 출력 장치)로 보냄
- CGI 메커니즘은 표준 출력 장치로 보낸 메시지만 원하기 때문에 에러 메시지는 버림
- CGI 에러를 추적할 수 있는 기능

```
import cgitb
cgitb.enable()
```

# 작은 변화가 큰 차이를 만들 수 있습니다

---

- class property : class method를 class attribute 처럼 사용하게 해 주는 방법

```
@property
def top3(self):
    ...

athletes[name].top3
```

## Additional Information

---