

플레이 프레임워크 시작하기 2

- 왜 플레이 프레임워크인가?
- 플레이 프레임워크 설치 및 통합 개발 환경
- 플레이 프레임워크 시작하기 1
- 플레이 프레임워크 시작하기 2

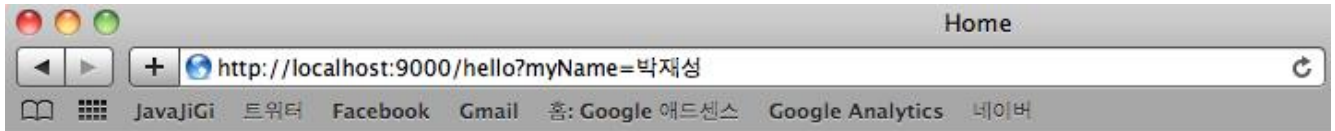
플레이 프레임워크 시작하기 1 글에 이어서 다음 과정을 진행한다.

지금까지 Hello World 애플리케이션을 구축을 완료했다. 여기서 몇 가지 개선 사항을 구현해 보자. 먼저 URL이 /application/sayhello에서 /hello로 개선해 보자. conf/routes 파일에서 다음 매핑을 추가한다.

```
GET / Application.index
```

```
GET /hello Application.sayHello
```

위와 같이 /hello 매핑을 추가한 후 위 테스트 과정을 다시 진행해 보자.



박재성~ Hello World

[Back to form](#)

위 화면과 같이 /hello URL을 통해 정상적으로 접근할 수 있다는 것을 확인할 수 있다. 여기서 눈여겨 볼 부분은 URL을 변경하기 위해 html 템플릿 파일에서 어떠한 변경 작업도 하지 않았으며, 단순히 conf/routes 파일에 /hello 매핑을 추가했을 뿐이라는 것이다.

다음은 레이아웃을 변경해 보자. 현재 index.html과 sayHello.html 파일은 모두 main.html 파일을 상속하고 있다. 따라서 레이아웃을 변경하는 작업은 main.html 파일을 변경하면 된다.

```
...
<body>
  Hello World .
  <hr />

  #{doLayout /}
</body>
...
```

app/views/main.html 파일을 위와 같이 수정한 후 다시 테스트를 해보자.



Hello World 애플리케이션.

지금까지 테스트를 진행하면서 의아한 부분이 있다. 지금까지 테스트 과정 중에서 한번도 서버를 재시작하지 않았다. 서버를 재시작하지 않은 것이 필자의 실수가 아니다. 플레이는 애플리케이션을 개발하는 과정에 서버를 재시작하지 않아도 모든 변경 사항이 자동으로 반영된다. 이 기능은 애플리케이션을 개발하는 과정에서 생산성을 향상하는데 크게 기여한다.

플레이 장점 중의 다른 하나는 쉬운 단위 테스트, 통합 테스트를 할 수 있도록 지원한다는 것이다. Hello World 프로젝트를 다음과 같이 다시 시작한다.

```
~$ play test
```


단계적으로 학습하는 실제 애플리케이션 개발

앞에서 살펴본 과정은 플레이의 기본적인 내용만 학습했다. 이 내용만으로는 플레이를 제대로 파악하기 힘들다. 실제 애플리케이션을 개발할 때 플레이를 어떻게 활용해 나갈 수 있는지에 대한 모습을 보려면 좀 더 큰 예제를 가지고 학습을 하는 것이 바람직할 것으로 판단된다.

플레이는 이를 위한 좋은 예제를 제공하고 있다. <http://www.playframework.org/documentation/1.2.4/home>에서 제공하는 “Tutorial — Play guide, a real world app step-by-step” 문서는 플레이의 전체적인 모습을 파악하는데 가장 좋은 예제이다. 단계별로 블로그 애플리케이션을 만들어 가는 과정을 통해 플레이가 어떤 형태로 동작하고 어느 곳에 활용하면 좋겠다는 아이디어를 얻을 수 있을 것이다.

플레이의 전체적인 모습을 보고 싶다면 <http://www.playframework.org/documentation/1.2.4/guide1> 문서부터 시작해 <http://www.playframework.org/documentation/1.2.4/guide12> 문서까지 따라해 보기 바란다. 몇 시간의 투자를 통해 플레이의 전체적인 모습을 파악할 수 있을 것이다.