

크롬은 웹앱 외에도 네이티브와 유사한 형태의 크롬 앱을 지원한다. 물론, 개발 언어는 웹앱과 동일하게 HTML5, JavaScript, CSS 를 사용하지만, 크롬 API 와 기존 웹사이트에서 접근할 수 없는 서비스들을 사용할 수 있다. 크롬 앱으로 개발할 수 있는 것들로 다음과 같은 것이 있다.

- Shells (VMWare, Citrix, SSH, RDP or VNC clients)
- Music/video streaming
- Photo/video/music editing

크롬 앱은 설치 후, 로컬에서 실행되므로 네트워크에 의존적이지 않으며 오프라인 상태에서도 동작한다.

이 즈음 하여, 구글 홈페이지(https://developer.chrome.com/apps/first_app.html)에 나와있는 간단한 샘플을 작성해 보았다.

작성하여야 할 파일은 다음과 같다.

```
manifest.json  
background.js  
window.html
```

그리고 아이콘으로 사용할 아래 이미지 파일들을 같은 폴더에 위치시킨다.
(파일명 calculator-128.png, calculator-16.png)



manifest.json

```
{  
  "name": "Hello World!",  
  "description": "My first Chrome App.",  
  "version": "0.1",  
  "app": {  
    "background": {  
      "scripts": ["background.js"]  
    }  
  },  
  "icons": { "16": "calculator-16.png", "128": "calculator-128.png" }  
}
```

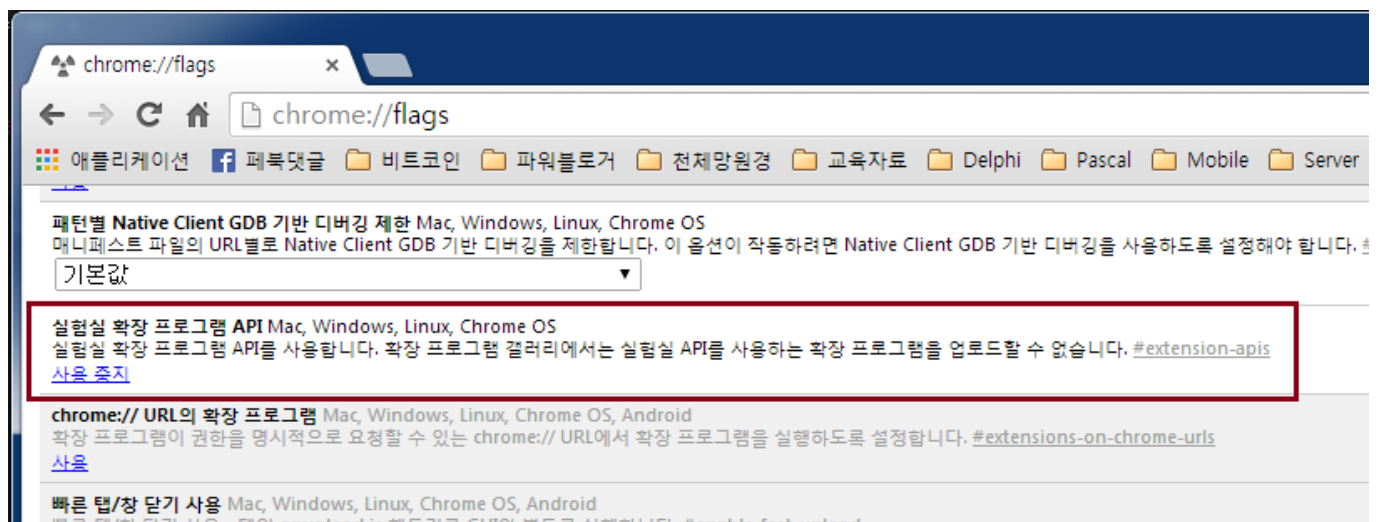
background.js

```
chrome.app.runtime.onLaunched.addListener(function() {
  chrome.app.window.create('window.html', {
    'bounds': {
      'width': 400,
      'height': 500
    }
  });
});
```

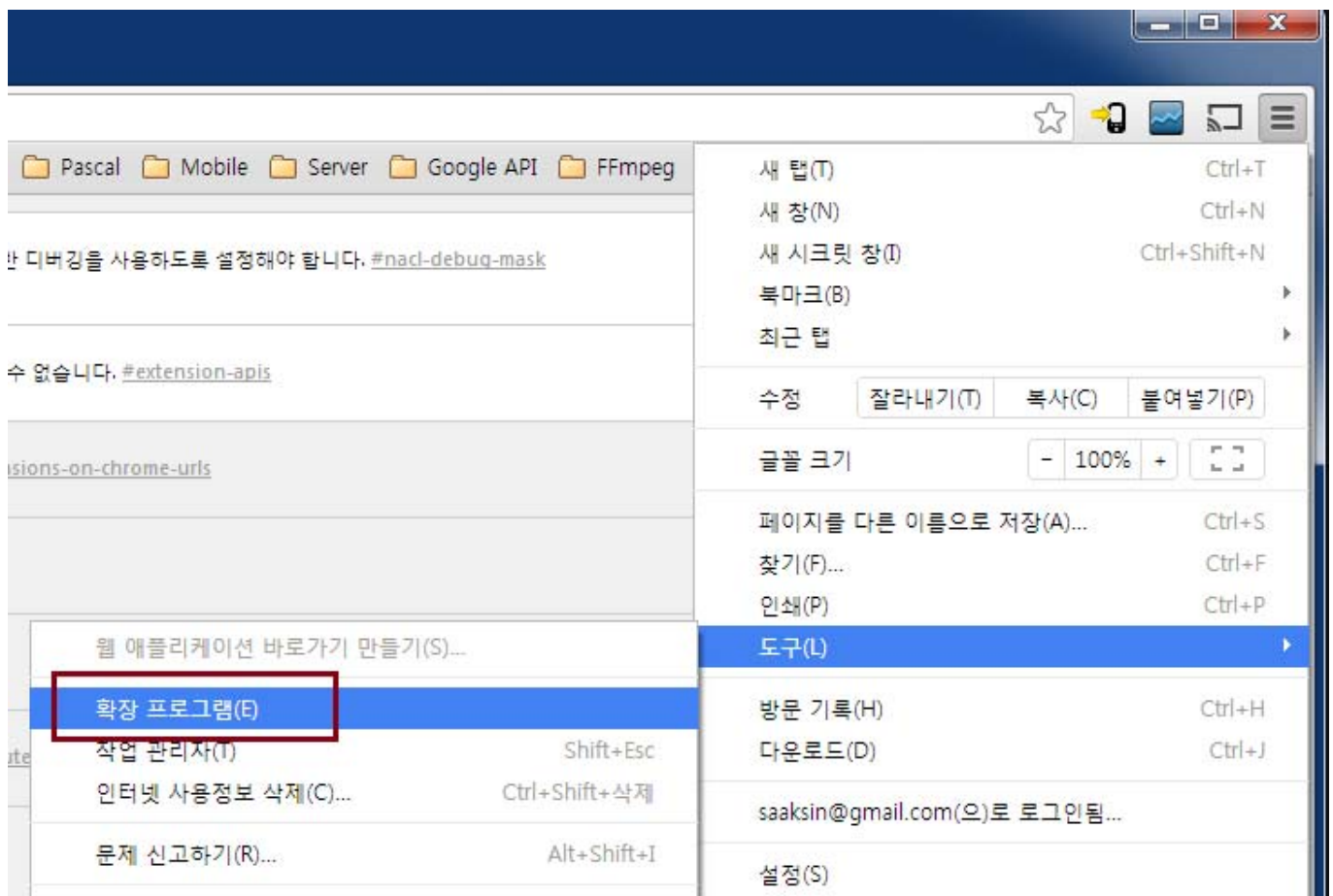
window.html

```
<!doctype html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Document</title>
</head>
<body>
  <div>Hello, world!!</div>
</body>
</html>
```

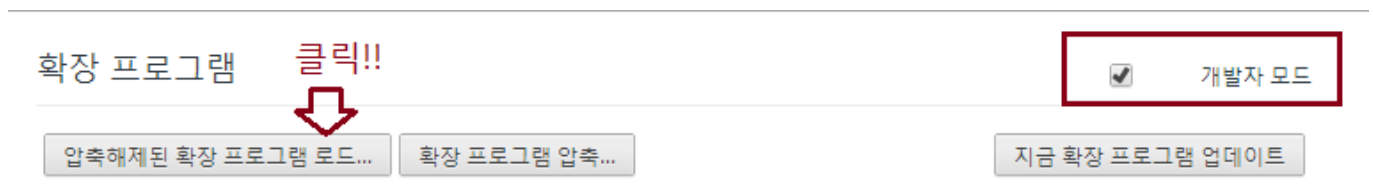
모든 파일을 생성하였으면, 이제 크롬 브라우저의 설정을 변경해준다. 먼저, 주소창에 `chrome://flags` 를 입력한 후 "**실험실 확장 프로그램 API**" 를 사용으로 변경한다.(아래 이미지는 이미 사용으로 전환한 상태임)



확장 프로그램 메뉴를 실행한다.



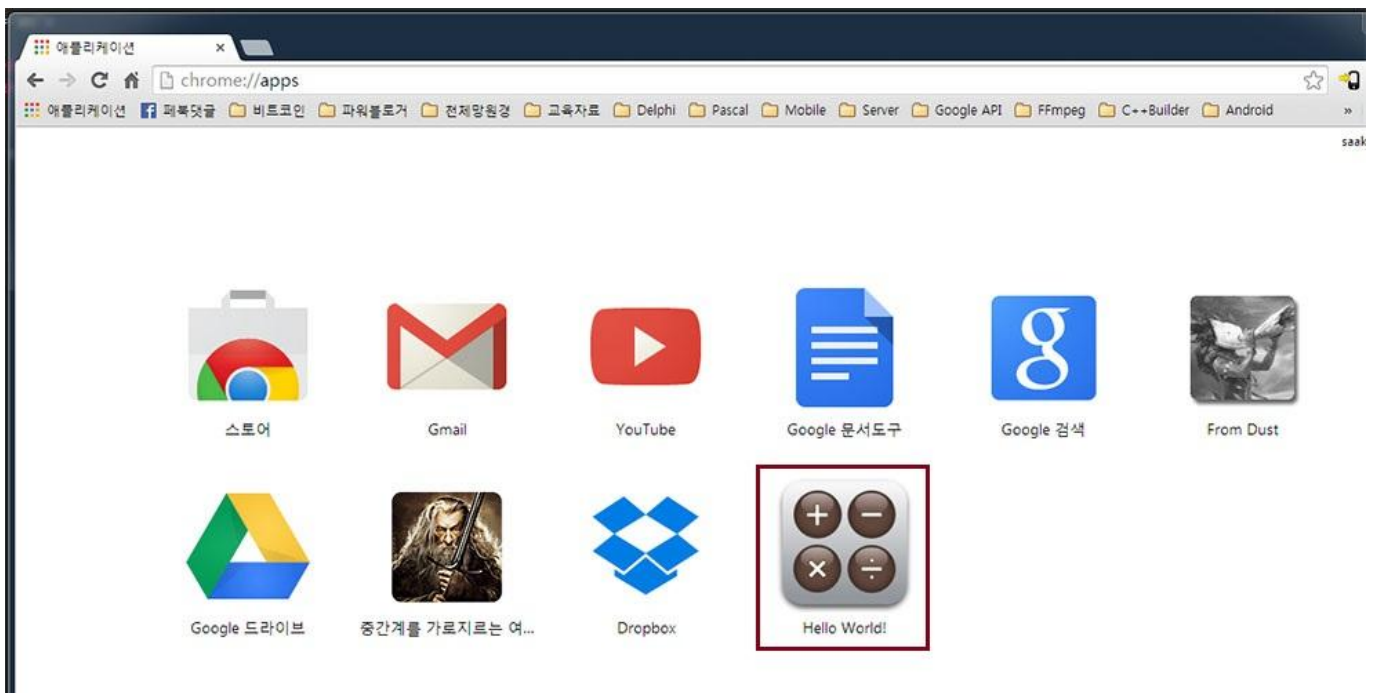
개발자 모드를 체크하고 "압축해제된 확장 프로그램 로드..." 버튼을 클릭한 다음, 상기 파일들이 위치하고 있는 폴더를 지정한다.



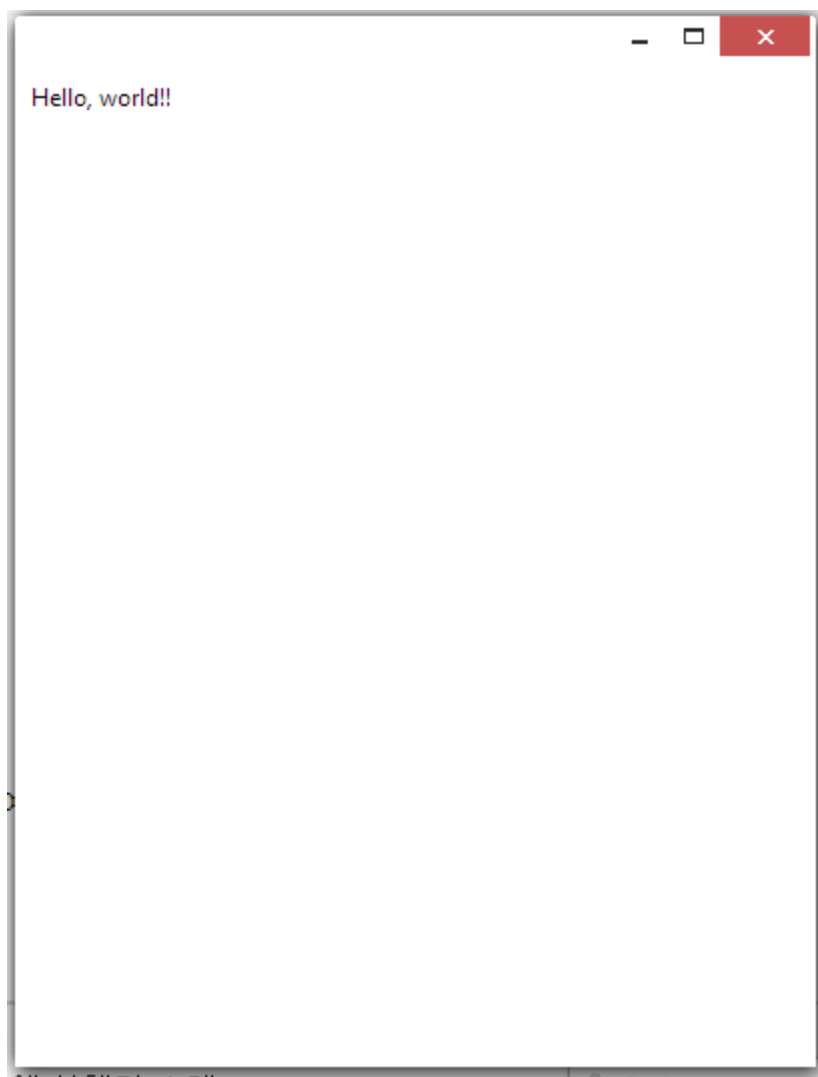
지정함과 동시에 다음과 같은 아이콘이 나타날 것이다.



주소창에 `chrome://apps` 를 입력하여 애플리케이션들 중 Hello World 앱이 추가되었는지 확인한다.



해당 앱을 클릭하여 실행하면...



마치 네이티브 앱처럼 프로그램이 실행된다. 윈도우 태스크바에도 해당 앱의 창이 나타난다.

이후 실행시 크롬에서 직접 실행하지 않고 명령 프롬프트에서 아래와 같은 방법으로 실행할 수 있다.

chrome --load-and-launch-app=설치된경로