

# Opera Branding Guidelines - (OPG)

Editor : Chanuim

2020.04.09

extension -> 확장프로그램으로 번역했습니다.

## Introduction

오페라 확장프로그램 만드는건 ㄹㅇ 쉽습니다. 오페라는 크로미움 확장프로그램 구조를 사용하기 때문에 크롬 확장프로그램 만드는 것에 익숙하다면 님은 집에 있을거임(?)  
↳ then you will be right at home. 이라는데 무슨 문맥으로 받아들여야 되지.  
혹여나 그렇지 않더라도 오페라 확장프로그램은 여전히 쉬울거예요, 오픈 웹 기술을 사용하고, 브라우저 기능을 활용할 수 있는 특정 API도 제공되기 때문이죠.

## What's in an Opera extension?

오페라 확장프로그램은 확장프로그램의 이름, 제작자, 기타등등과 같은 메타데이터를 정의하는 manifest파일을 포함하고 있습니다.  
그리고 확장프로그램이 가질 수 있는 다양한 API 권한들을 나열했습니다.  
그리고 브라우저UI와 커뮤니케이션하는 background 페이지 또는 background script가 있습니다.  
이것과는 별개로, 웹 페이지 변경을 다루는content script가 있을 수도 있습니다.  
버튼 팝업 혹은 옵션 페이지를 위한 html(과 연관된 css와 js)파일이 필요할 수 도 있습니다.

모든 JS와 HTML을 빼고, 확장프로그램 아이콘에 달 이미지도 필요합니다.

이건 .crx파일확장자로 zip파일 포맷이 되어있어야 됨.  
이거에 관한 오페라 확장프로그램의 구조를 더 알고싶다면 상세사항을 설명한 연관된 아티클(associated article)을 읽어보세요.

## Your first extension

이제 이게 어떻게 작동하는지 기본은 알게 됐으니 확장프로그램을 만들어 봅시다.  
브라우저 톨바에 버튼 붙인 간단한 확장프로그램을 만들어 봅시다.  
이 확장프로그램이 클릭 되면 새탭을 열고 dev.opera.com을 로-드합니다. 좀 간단한 예제인데 기본에 익숙해질거임.

## Step 1: Defining the extension with an extension manifest

우리가 할 첫번째 단계는 확장프로그램 manifest를 정의하는 것입니다. manifest에다가 이름, 설명, 작성자, 버전 넘버, 기타 세부사항들을 적을 수 있습니다.

manifest는 확장프로그램이 제대로 실행되기 위해 필요한 권한들을 정의한다는 다른 중요한 측면이 있습니다.

예를 들어 탭이 필요하다고 할 때, manifest파일에 그것을 명시해주어야만 합니다.

확장프로그램의 manifest는 json형식으로 쓰여지며 이걸 나중에 자세히 설명해드리겠습니다.  
지금은 그냥 바로 에디터나 열어서 manifest.json파일 만들고 이렇게나 쓰라고 ㅎㅎㅎ

## Step 2: Communicating with the browser: the background script

background 스크립트는 매우 중요합니다. 왜냐하면 browser UI를 조작하는 관련된 모든것이 background스크립트에 있습니다.

탭으로 작동할 것이므로, 스크립트의 Tabs API 방법을 사용할 거임

나중에 tabs에 관해 알아보겠지만 지금은 background.js를 작성하고 코드나 치라우

## Step 3: Including icons and other assets

툴바 버튼과 다른 곳에서 확장프로그램을 보여주기 위해 아이콘이 필요함.

또한, 이미지, 폰트, 비디오 기타등등이 필요할 수 있습니다. 부모 디렉토리에 같이 포함 시키거나, 미디어라는 폴더를 만들어 거기에 넣는 등 폴더를 만들어 분리할 수 있습니다.

## Step 4: Testing your extension

예아 우리의 확장프로그램을 테스트할 시간입니다. 최종 확장프로그램이면 extension 세팅 페이지에서 패키징하고 서명해야됩니다.

좋은 소식은 모든 테스트를 할 때마다 이 짓거리를 할 필요는 없다는 것이죠. 디렉토리에서 바로 확장프로그램을 테스트할 수 있습니다.

요렇게 말이죠

- 1) 브라우저 주소창에다가 opera:extensions 치기
- 2) developer mode 버튼 허용 체크
- 3) load unpacked Extension 버튼 클릭
- 4) extension 경로 선택하기

네 확장프로그램은 개발자 모드에서 로드되어야 합니다. 개발자 모드는 브라우저의 개발자 툴을 사용해 확장프로그램의 여러 부분을 분석할 수 있는 능력을 줍니다.

이걸로 네 확장프로그램의 코드를 바꿀 수 있고 리로드 버튼을 통해 빠르게 결과를 볼 수 있음.

만약 모든게 잘 됐다면, 브라우저 창의 오른쪽위의 주소창쪽에 아이콘을 볼 수 있을 거임  
클릭하면 새 탭이 열리며 Dev.Opera(오페라의 개발자 튜토리얼 사이트)로 감.

## Step 5: Packing it all up!

확장프로그램이 완료된 것에 만족했으면, 일단 CRX 파일로 패키징해야 됨.

요렇게 하셈

- 1) 브라우저 주소창에다가 opera:extensions 치기
- 2) 개발자 모드로 체크
- 3) 페이지의 위에 있는 Pack Extension 버튼 누르기
- 4) 확장프로그램의 경로 선택
- 5) "OK" 클릭

님의 crx 패키지는 님이 선택했던 경로의 부모 디렉토리에 생성될거임 ㅎㅎ

## What now?

보면 알겠지만, 오페라에서 확장프로그램 만드는건 ㄹㅇ쉬움. 이 튜토리얼을 통해, 간단한 확장을 만들고, 개발자 모드에서 로드하고 테스트 해보고 패키지 하는 걸 배우게 되었습니다.

여기서부터 님은 확장기능의 다른 부분을 자세히 다룬 우리가 작성했던 다른 튜토리얼(버튼, 탭, 메세징 등등)을 봐야됨.

순수 참조 가이드가 필요하면 사이드바의 API Docs 섹션을 확인해보셈.