

Opera Branding Guidelines - (OPG)  
Editor : Chanuim  
2020.04.10

extension -> 확장프로그램으로 번역했습니다.

## Introduction

architecture overview에서 오페라 확장프로그램에 포함될 수 있는 UI 요소를 살펴 보았는데 이는 Page Actions와 Browser Actions라는 두 가지 카테고리에 속합니다.

이번 아티클에서는 확장프로그램에 action으로 UI 요소를 어떻게 붙이는가에 대해 보도록 할거예요.

브라우저에 이런 UI 요소를 나타내게 하는 건 엄청나게 간단해요. 당신이 할 건 확장프로그램의 manifest에서 필요한 세부사항을 채우고, 확장프로그램 패키지에 준비된 적절한 기능들을 갖고 있는지 확인하면 돼요. (아이콘 파일, 팝업 HTML, 백그라운드 js, 기타등등)

## Browser Actions

브라우저 액션은 확장프로그램 툴바에 UI 요소를 배치하는데 사용됩니다. 이런 UI 요소들은 버튼(벤티를 포함하거나 그렇지 않거나), 팝업이 될 수 있어요. 항상 UI 요소를 나타내게 해야 된다면 브라우저 액션을 사용하세요.

하지만, 항상 UI 요소를 나타내게 할 필요가 없거나, 특정 페이지에서만 나타내게 해야 된다면 페이지 액션을 사용하는걸 추천해요. 페이지 액션은 이 아티클의 밑부분에 설명이 있어요.

명심하세요! 브라우저 액션과 페이지 액션은 같이 사용할 수 없어요!

## Buttons

브라우저 UI에 버튼을 나타내려면 확장프로그램의 manifest 파일에 이렇게 적어주시면 돼요.

```
"Browser_action": {  
  "default_icon": "icon.png",  
  // 툴팁을 나타내는 건 옵션이에요 :D  
  "default_title": "My Sample Extension"  
}
```

기본 아이콘은 19px 혹은 38px로 제공되고, 브라우저가 알아서 사용자 디스플레이의 화소밀도(pixel density)에 맞춰서 아이콘 사이즈를 조절할 거예요.

manifest에 위의 내용을 적는건, 브라우저에 버튼을 나타내게 하는거예요. 클릭을 해보면 아무 결과도 나오지 않아요. 하지만, 버튼을 눌러 함수를 실행하기 위해 이게 필요할 거예요. 그걸 위해 클릭 이벤트에 이벤트 리스너를 달 수가 있습니다. 이렇게요

```
chrome.browserAction.onClicked.addListener(function() {  
  console.log("This button was clicked!");  
});
```

## Popups

팝업을 만들려면 HTML파일을 만들어야 돼요. (이걸 popup.html이라 부르죠!) 이 파일은 팝업이 열렸을 때, 웹 페이지를 보여줄거예요. 확장프로그램의 manifest에 이렇게 적으면 된답니다.

```
"browser_action" {
```

```

    "default_icon": "icon.png",
    // 토틱을 나타내는 건 옵션이에요 :D
    "default_title": "My Sample Extension",
    "default_popup": "popup.html"
}

```

주의 : 팝업을 정의하고 background.js에서 클릭 이벤트를 추가 할 경우 그 클릭 이벤트는 작동을 안해요.

## Badges

뱃지는 버튼 다음에 위치해 있는 작은 정보 조각이에요. 뱃지는 가끔씩 부가적인 정보를 보여줘요. (예를 들면 당신의 이메일사이트에 관한 확장프로그램이라면 읽지 않은 메일 수를 보여줄 수 있겠죠!)

뱃지는 매우 작은 양의 정보를 보여주도록 되어있어요. 그래서 4글자 제한이 걸려있어요. ;;; 4글자 보다 더 많이 적으면 크기 때문에 알아서 잘려요!

그럼 어떻게 뱃지를 만들고 텍스트를 설정하는지 보도록 할까요?

뱃지는 background 스크립트의 js언어로 설정할 수 있어요. (확장프로그램의 manifest에 background script를 꼭 적으세요.)

당신이 알아야 될 두 가지 함수가 있어요:

- chrome.browserAction.setBadgeBackgroundColor(), 이걸 뱃지의 배경색을 설정하는거예요
- chrome.browserAction.setBadgeText(), 이걸 뱃지의 문구를 설정해요.

우리의 예제에서는 빨간 배경색과 Hey 문구를 적은 뱃지를 만들어볼거예요. background script에 이렇게 적어주세요.

```

chrome.browserAction.setBedgeBackgroundColor({
    color: '#ff0000'
});
chrome.browserAction.setBadgeText({
    text: 'hey'
});

```

바로 그거예요! 이렇게 크롬 확장프로그램에 뱃지를 설정하는 법을 알아봤답니다. 그리고, 16진수 컬러 코드 대신에 RGBa 값을 넣어서 배경의 투명도를 조절할 수 있어요!

또. 일정 시간 간격에 맞추어서 업데이트 하거나 어떤 액션이 수행된 뒤에 업데이트하게끔 설정할 수 있다는걸 잊지마세요!



뱃지와 팝업의 모습.

## Page Actions

페이지 액션은 액션을 실행하기 위해 주소 창 안쪽에 아이콘을 배치해서 사용합니다. 이 '액션'은 현재 보이는 페이지와 관련성이 있어야 해요. 하지만, 브라우저에 열려있는 모든 페이지에 필요한 건 아니에요.

페이지 액션은 거의 같은 방법으로 브라우저 액션과 같이 버튼과 팝업을 만들 수 있어요. 하지만, 페이지액션에는 뱃지를 달 수 없어요.

## Buttons

버튼을 달기 위해, 단순히 manifest파일에 이렇게 적어주기만 하면 돼요.

```
"page_action": {  
  "default_icon": "icon.png",  
  // 툴팁을 나타내는 건 옵션이에요 :D  
  "default_title": "My Sample Extension",  
}
```

## Popups

팝업을 만들려면 HTML파일을 만들어야 돼요. (이걸 popup.html이라 부르죠!) 이 파일은 팝업이 열렸을 때, 웹 페이지를 보여줄 거예요. 확장프로그램의 manifest에 이렇게 적으면 된답니다.

```
"page_action": {  
  "default_icon": "icon.png",  
  // 툴팁을 나타내는 건 옵션이에요 :D  
  "default_title": "My Sample Extension",  
  "default_popup": "popup.html"  
}
```

## Displaying a Page Action

페이지 액션은 브라우저 액션과 달리 기본적으로 표시되지 않아요. `show()` 함수를 써서 명백히 보여줘야 돼요. 또, `hide()` 함수를 사용해 숨길 수 있어요.