QGIS Documentation Guidelines

QGIS Project

2024 년 10 월 11 일

Contents

1 단계별 공헌 방법 **3**
1.1 GitHub 웹 인터페이스 사용하기 4
1.2 Git 명령 프롬프트 도구 이용 10
1.3 더 읽어볼 거리 14

2 작성 지침 **15**
2.1 문서 작성 16
2.2 스크린샷 관리 23
2.3 공간 처리 알고리즘 문서 작성 24

3 PyQGIS 쿡북 코드 작성 **29**
3.1 테스트 가능한 코드 조각 작성 방법 30
3.2 사용자 로컬 환경에서 코드 조각을 테스트하는 방법 31

4 번역 지침 **33**
4.1 번역 과정 33
4.2 파일 번역 34

5 별명 **41**
5.1 사용법 42
5.2 일반 별명 42
5.3 툴바 버튼 아이콘 43
5.4 기타 기본 아이콘 51
5.5 속성 테이블 52
5.6 투영 및 지리참조자 52
5.7 인쇄 조판기 53
5.8 레이어 속성 54
5.9 플러그인 55

https://docs.qgis.org 에서 QGIS 문서를 볼 수 있습니다. 문서 작성이 진행되면서, 매일 (정확한 시간은 이 페이지 하단에서 확인할 수 있습니다) 모든 지원 버전 (테스트 버전, 장기 지원 버전 (LTR), 그리고 다음 LTR) 을 자동으로 빌드합니다.

https://github.com/qgis/QGIS-Documentation 에서 QGIS 문서 소스 파일을 찾을 수 있습니다. 대부분 reStructuredText(reST) 포맷 문법으로 작성되었으며, 일부 스크립트는 HTML 산출물을 후처리하는 Sphinx 도구 모음으로 작성되기도 했습니다. 이 도구에 대한 일반적인 정보를 알고 싶다면, https://docutils.sourceforge.io/docs/ref/rst/restructuredtext.html 또는 https://www.sphinx-doc.org/en/master/usage/restructuredtext/basics.html 을 참조하세요.

이 문서의 나머지 부분에서 다음과 같은 내용을 배울 수 있을 것입니다:

- git 시스템 및 문서 소스 파일이 저장된 GitHub 플랫폼을 사용해서 문서 소스 파일을 어떻게 관리하는지
- 규칙에 따라 텍스트를 어떻게 수정하고, 스크린샷을 어떻게 업로드할 수 있는지
- 사용자의 수정 사항을 어떻게 공유하고, 공식 문서로 푸시하도록 보장할 수 있는지

QGIS 프로젝트에 참여하는 방법에 대한 일반적인 정보를 알고 싶다면, QGIS 커뮤니티에 참여하기 웹페이지를 방문해주십시오.

CHAPTER 1

단계별 공헌 방법

- *GitHub* 웹 인터페이스 사용하기
  - *QGIS-Documentation* 포크 *(fork)*
  - 변경 사항 작성
    - 대안 *1:* `Edit on GitHub` 단축키 사용하기
    - 대안 *2:* 작성자의 문서 저장소에 애드혹 *(ad hoc)* 분기 생성하기
  - 파일 수정
  - 끌어오기 요청을 통해 변경 사항 공유
    - 새 끌어오기 요청 생성
    - 변경 사항 비교
    - 작성자의 끌어오기 요청을 설명
    - 끌어오기 요청 검토 및 논평
    - 수정 작업
  - 작성자의 병합된 분기 삭제
- *Git* 명령 프롬프트 도구 이용
  - 로컬 저장소
  - 또다른 원격 저장소 추가
  - 작성자의 기반 분기 업데이트
  - 작성자의 작업 분기로 기고
  - 작성자의 변경 사항 공유
  - 작성자의 로컬 및 원격 저장소 청소

- 더 읽어볼 거리

참고: 이 과정을 설명하는 데 QGIS 문서를 예로 들긴 했지만, 다음 단락부터 보여줄 모든 명령어 및 단계는 QGIS 웹사이트에도 적용할 수 있습니다.

이 문서를 읽고 있다는 사실은 당신이 QGIS 문서 작성에 공헌할 마음을 가지고 있으며, 그 방법을 찾고 있다는 것이겠죠. 잘 오셨습니다! 이 문서는 QGIS 문서 작성이라는 목표를 달성하기 위한 여러 가지 방법을 알려드리고, 따라야 할 주요 단계와 당신이 사용할 수 있는 요령들 및 주의해야 할 함정들을 보여드릴 것입니다.

도움이 필요하다면 주저하지 말고 당신이 고치려 하는 문제점 보고서에 코멘트를 남겨 질문하거나 QGIS 커뮤니티 팀 메일링 리스트 에 문의하십시오. 자세한 내용은 문서 작성하기 를 참조하세요.

이제 시작해보겠습니다.

문서 소스는 git 버전 제어 시스템을 통해 저장되며 GitHub 에서 사용할 수 있습니다. 고쳐야 할 문제 및 설명해야 할 기능들의 목록은 https://github.com/qgis/QGIS-Documentation/issues 에서 찾아볼 수 있습니다.

팁: 당신이 처음으로 공헌을 결심했기 때문에 어디에서부터 시작할지 잘 모르겠다면, 문제점 보고서 에 도전해보는 것도 좋을 겁니다.

문서 파일을 수정하는 데에는 결코 상호 배타적이 아닌 두 가지 주요 방법이 있습니다:

1. *GitHub* 웹 인터페이스 사용하기
2. *Git* 명령어줄 도구 사용하기.

1.1 GitHub 웹 인터페이스 사용하기

GitHub 웹 인터페이스에서 다음과 같은 작업을 실행할 수 있습니다:

- 파일 편집
- 변경 사항 미리보기 및 커밋 (commit)
- 주 저장소에 작성자의 변경 사항을 삽입하도록 끌어오기 요청 생성
- 분기 (branch) 생성, 업데이트, 또는 삭제

아직 git 및 GitHub 의 전문 용어에 익숙하지 않다면, 다음 단락부터 이용할 몇몇 기본 용어 및 액션을 숙지하기 위해 GitHub Hello-world 프로젝트를 읽어보시기 바랍니다.

참고: 보고된 문제점을 고치려면

문제점 를 고치기 위해 변경 사항을 생성하는 경우, 자신에게 해당 문제점을 할당하기 위해 문제점 보고서에 코멘트를 추가하십시오. 여러 사람이 동시에 동일한 문제를 작업하는 일을 방지할 것입니다.

1.1.1 QGIS-Documentation 포크 (fork)

당신이 이미 GitHub 계정 을 가지고 있다고 가정하고, 가장 먼저 해야 할 일은 문서 소스 파일을 포크 (fork) 하는 것입니다.

QGIS-Documentation 저장소 페이지로 가서 우상단에 있는 Fork 버튼을 클릭하십시오.

당신의 GitHub 계정에서 QGIS-Documentation 저장소 (`https://github.com/<YourName>/QGIS-Documentation`) 를 찾을 수 있을 것입니다. 이 저장소는 공식 QGIS-Documentation 저장소의 복사본으로, 공식 문서에 영향을 주는 일 없이 완전한 작성 권한으로 변경 사항을 작성할 수 있습니다.

1.1.2 변경 사항 작성

QGIS 문서에 기고하는 데엔 여러 가지 방법이 있습니다. 다음 단락에 각각의 방법을 소개하고 있기는 하지만, 작성자가 한 과정에서 다른 과정으로 어떤 문제도 없이 바꿀 수 있습니다.

대안 1: `Edit on GitHub` 단축키 사용하기

QGIS 문서 웹사이트에 있는 페이지의 우상단에 있는 `Edit on GitHub` 링크를 클릭하면 각 페이지를 쉽고 빠르게 편집할 수 있습니다.

1. 이때 열리는 파일은 `qgis:master` 분기에 있는 파일로, 페이지 상단에 작성자는 이 저장소에 대한 작성 권한이 없으며 작성자가 변경한 내용은 작성자의 저장소에 새로운 분기로 저장될 것이라는 메시지가 뜰 것입니다.
2. 변경 사항을 작성하십시오. reStructureText 문법을 사용해서 문서를 작성하기 때문에, 당신이 작성하는 변경 사항에 따라 문서 작성 지침 을 따라야 할 경우도 있습니다.
3. 수정 작업이 끝나면, 작성자가 변경한 내용에 대해 간단히 설명하는 코멘트를 작성한 다음 *Propose changes* 를 클릭하십시오. 작성자의 저장소에 새 분기 (`patch-xxx`) 를 생성할 것입니다.
4. *Propose changes* 를 클릭하면 GitHub 가 *Comparing changes* 페이지로 이동할 것입니다.
   - 변경 사항 작성을 완료했다면, 아래에 있는끌어오기 요청을 통해 변경 사항 공유 부분 아래 있는변경 사항 비교 로 건너뛰십시오.
   - 작성자가 QGIS 에 변경 사항을 커밋하기 전에 추가로 변경하고자 하는 사항이 있다면, 다음 단계를 따르십시오:
     1. 작성자의 QGIS-Documentation 포크 (`https://github.com/<YourName>/QGIS-Documentation`) 로 가십시오.
     2. Branch: master 버튼을 클릭한 다음 `patch-xxx` 분기를 검색하십시오. 해당 분기를 선택하십시오. Branch: master 버튼의 텍스트가 *Branch: patch-xxx* 로 변했을 것입니다.
     3. 아래 있는파일 수정하기 로 넘어가십시오.

참고: 왼쪽 사이드바 하단에 있는 드롭다운 메뉴에서도 `Edit on GitHub` 단축키를 사용할 수 있습니다.

대안 2: 작성자의 문서 저장소에 애드혹 (ad hoc) 분기 생성하기

QGIS 문서의 작성자 포크 (fork) 에서 파일을 직접 편집할 수 있습니다.

먼저, 사용자의 `master` 분기가 `qgis:master` 분기의 최신 상태와 일치하는지 확인하십시오. 이를 확인하려면:

1. 예를 들어 `https://github.com/<YourName>/QGIS-Documentation` 같은 사용자 저장소의 메인 페이지로 가십시오. `master` 분기가 활성화되어 `qgis/QGIS-Documentation:master` 분기의 최신 상태와 일치하는지를 알려줄 것입니다.

   업스트림 분기보다 앞서 커밋한 경우 사용자의 `master` 브랜치를 정렬할 때까지 이전:ref:`대안 단축키 버튼 <edit_on_github>``을 사용하는 편이 좋습니다.

   뒤쳐진 커밋만 있는 경우:

   1. 오른쪽에 있는 *Fetch Upstream* 드롭다운 메뉴를 펼치십시오. 다음 작업을 할 수 있습니다:
      - *Compare*: 분기들을 비교해서 메인 저장소의 새로운 변경 사항들을 볼 수 있습니다.
      - *Fetch and merge*: 업스트림 분기로부터 사용자 분기로 변경 사항들을 가져올 수 있습니다.
   2. *Fetch and merge* 를 클릭해봅시다. 처리 과정이 끝나면 사용자 분기가 `qgis/QGIS-Documentation:master` 분기의 최신 상태와 일치한다고 표시됩니다.

2. 작성자의 QGIS 문서 저장소의 좌상단에 있는 Branch: master 버튼을 클릭하고 텍스트란에 유일한 이름 (unique name) 을 입력해서 새 분기 (branch) 를 생성하십시오. 새 분기의 이름은 작성자가 해결하고자 하는 문제점과 관련이 있어야 합니다. 이제 Branch: master 버튼이 *Branch: branch_name* 으로 보일 것입니다.
3. 이제 사용자 분기에 새로운 변경 사항을 적용할 준비가 끝났습니다.

> **주의: 절대 `master` 분기를 변경하지 말고, 애드혹 분기를 변경하십시오.**
>
> `qgis/QGIS-Documentation` 의 `master` 분기의 수정 사항을 작성자의 QGIS 문서 저장소의 복사본으로 병합하는 경우를 제외하면 `master` 분기를 변경하지 마십시오. 개별 분기들을 구분하면, 다른 분기들을 건드리지 않고 동시에 여러 문제점을 작업할 수 있습니다. 실수가 발생한 경우 언제나 해당 분기를 삭제하고 마스터 분기에서 새 분기를 생성해서 다시 시작할 수 있습니다.

1.1.3 파일 수정

1. QGIS 문서의 작성자 포크의 소스 파일 가운데 수정해야 할 파일을 탐색하십시오.
2. :ref:` 문서 작성 지침 <QGIS-documentation-guidelines>` 에 따라 수정하십시오.
3. 수정 작업이 끝나면 페이지 하단에 있는 **Commit Changes** 프레임으로 가서 작성자가 수정한 사항에 대해 간단한 주석을 작성한 다음, *Commit Changes* 를 클릭해서 작성자 분기에 변경 사항을 직접 커밋하십시오. *Commit directly to the branch_name branch.* 옵션을 선택했는지 확인하세요.
4. 문제점을 해결하기 위해 업데이트해야 하는 다른 파일에 대해서도 이 단계들을 반복하십시오.

1.1.4 끌어오기 요청을 통해 변경 사항 공유

공식 문서에 작성자의 변경 사항을 통합시키려면 끌어오기 요청 (pull request) 을 생성해야 합니다.

참고: `Edit on GitHub` **링크를 사용한 경우**

작성자가 변경 사항을 커밋하면, 깃허브 (GitHub) 가 작성자의 `patch-xxx` 분기에서 변경된 사항을 `qgis/QGIS-Documentation` 마스터 분기와 비교하는 새 페이지를 자동적으로 열 것입니다.

다음두 번째 단계 로 넘어가십시오.

새 끌어오기 요청 생성

QGIS-Documentation 저장소의 메인 페이지로 가서 *New pull request* 를 클릭하십시오.

변경 사항 비교

하나는 `base:master` 이고 다른 하나는 `compare:branch_name` 인 2 개의 대화 상자가 나타난다면 (`id7` 참조) 작성자 분기 가운데 하나의 변경 사항을 작성자의 마스터 분기로 병합시킬 뿐입니다. 이를 바꾸려면 *compare across forks* 링크를 클릭하십시오.

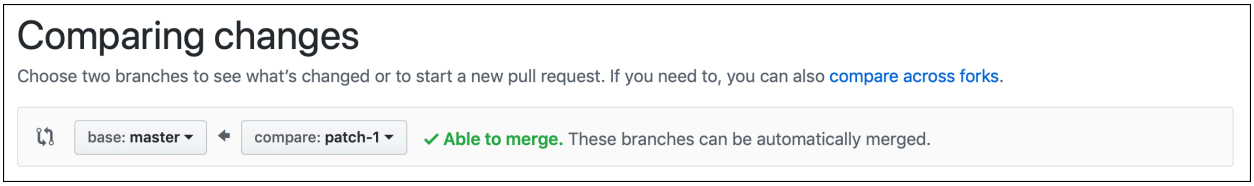

그림 1.1: 작성자의 *Comparing changes* 페이지가 이렇게 보일 경우 *compare across forks* 링크를 클릭

4 개의 드롭다운 메뉴가 나타날 것입니다. 이 메뉴들을 통해 작성자 분기의 변경 사항을, 병합시키고자 하는 마스터 분기와 비교할 수 있습니다. 이 드롭다운 메뉴는 다음과 같습니다:

- **base fork**: 작성자 변경 사항을 병합시키고자 하는 포크
- **base**: 작성자 변경 사항을 병합시키고자 하는 기반 포크 (base fork) 의 분기
- **head fork**: 작성자가 기반 포크로 통합하려 하는 변경 사항이 있는 포크
- **compare**: 해당 변경 사항을 가진 분기

`qgis/QGIS-Documentation` 을 **base fork** 로, `master` 를 **base** 로 선택하고, 작성자 저장소 `<YourName>/QGIS-Documentation` 을 **head fork** 로, 작성자가 변경한 분기를 **compare** 로 설정하십시오.

녹색 **Able to merge** 체크 표시는 작성자의 변경 사항을 문제없이 공식 문서로 병합시킬 수 있다는 사실을 나타냅니다.

Create pull request 버튼을 클릭하십시오.

경고: ✕ Can't automatically merge. **표시가 나타난 경우**

충돌 (conflict) 이 일어났다는 뜻입니다. 다른 누군가가 작성자의 변경 사항과 충돌을 일으키는 내용을 커밋했기 때문에 작성자가 수정한 파일이 병합시키고자 하는 분기와 버전이 맞지 않는 것이죠. 그래도 끌어오기 요청을 생성할 수는 있지만 병합을 완료하려면 모든충돌 을 해소해줘야 할 것입니다.

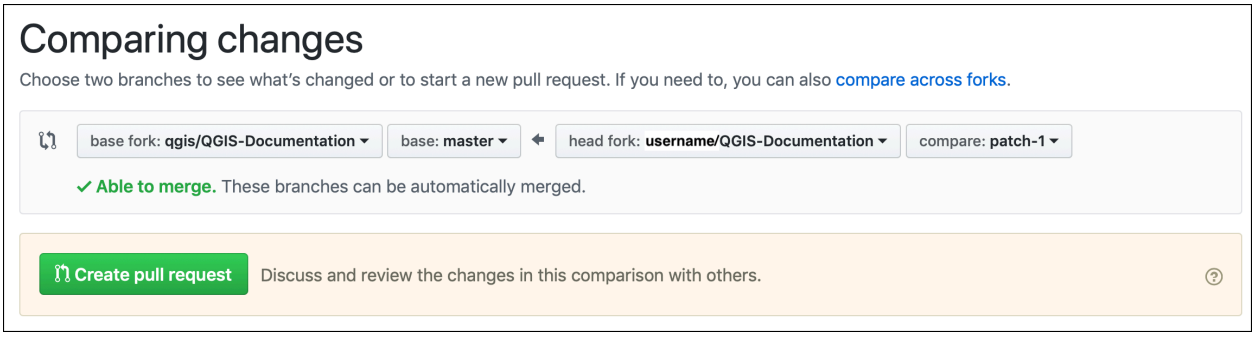

그림 1.2: `qgis/QGIS-Documentation` 과 작성자 저장소 사이의 변경 사항 비교

팁: 번역 중에도, QGIS 문서의 최신 버전 은 계속 유지보수되며 기존 문제점들을 해결하고 있습니다. 다른 배포판의 문제점을 해결하는 중이라면, 앞의 단계들에서 **base** 를 `master` 에서 적절한 `release_...` 분기로 바꾸십시오.

작성자의 끌어오기 요청을 설명

텍스트란이 나타날 것입니다. 작성자가 제기하는 문제점과 관련된 주석을 작성하십시오.

혹시 특정한 기존 문제점 과 관련된 내용일 경우, 주석에 문제점 번호를 넣어주세요. # 과 함께 문제점 번호를 (예: `#1234`) 입력하면 됩니다. `fix` 또는 `close` 같은 용어를 앞에 넣어주면, 끌어오기 요청이 병합되는 대로 해당 문제점은 종결될 것입니다.

작성자가 변경하는 모든 문서 페이지를 가리키는 링크들도 추가해주세요.

Create pull request 를 클릭하십시오.

끌어오기 요청 검토 및 논평

앞에서 설명한 대로, 누구나 끌어오기 요청을 통해 문서에 수정 사항을 제출할 수 있습니다. 마찬가지로 누구나 끌어오기 요청에 대해 검토하고 질문을 하거나 논평 할 수 있습니다. 문서 작성 스타일이 프로젝트의 지침과 맞지 않을 수도 있고, 변경 사항에 일부 중요한 세부 내용이나 스크린샷이 누락됐을 수도 있고, 또는 모든 것이 제대로 되어 있을 수도 있습니다. 검토 및 논평은 형식과 내용 모든 관점에서 기고 내용의 질을 향상시키는 데 도움이 됩니다.

끌어오기 요청을 검토하려면:

1. 끌어오기 요청 페이지 로 가서 작성자가 논평하고 싶은 PR(pull request) 을 클릭하십시오.
2. 페이지 하단에 작성자가 끌어오기 요청에 대해 일반적인 논평을 남길 수 있는 텍스트란이 있을 것입니다.
3. 특정 행에 대한 논평을 남기고 싶다면,
   1. Files changed 버튼을 클릭하고 논평하고자 하는 파일을 찾으십시오. 변경 사항을 보려면 *Display the source diff* 를 클릭해야 할 수도 있습니다.
   2. 작성자가 논평을 남기고 싶은 행으로 스크롤한 다음 + 아이콘을 클릭하십시오. 작성자가 논평을 남길 수 있는 텍스트란이 열릴 것입니다.

특정 행에 대한 논평은 다음 두 가지 방법으로 제출할 수 있습니다:

- 단일 논평이라면 *Add single comment* 버튼을 사용하십시오. 그대로 공개될 것입니다. 짧은 논평이거나, 다른 논평에 대한 댓글일 경우에만 이 버튼을 사용하세요.
- 검토의 일부라면, *Start a review* 버튼을 클릭하십시오. 작성자의 논평이 확인 (validation) 과정을 거친 후 자동으로 공개되지 않기 때문에, 버튼을 클릭한 다음에도 논평을 편집하거나 취소하거나 주요 검토 내용 또는 끌어오기 요청에 대한 전반적인 지침의 개요를 추가하거나 작성자가 해당 끌어오기 요청의 내용을 찬성하는지 반대하는지 밝힐 수도 있습니다. 작성자가 자신의 검토 내용을 더 잘 구성할 수 있고, 논평을 편집할 수 있고, 스스로 준비됐을 때 공개할 수 있으며 저장소 팔로워들에게 각 논평마다 알림을 보내는 것이 아니라 단일 알림을 보낼 수 있기 때문에 더 유연하고 편리한 방법입니다. 더 자세한 내용 을 참조하세요.

그림 1.3: 특정 라인에 변경 사항을 제안하는 논평

행 논평에는 끌어오기 요청 작성자가 끌어오기 요청에 적용할 수도 있는 제안을 내장시킬 수 있습니다. 제안을 추가하려면, 논평 텍스트란 위에 있는 Insert a suggestion 버튼을 클릭하고 제안 블록 안에서 텍스트를 수정하십시오.

팁: 작성자의 끌어오기 요청에 제안을 배치 (batch) 프로세스로 커밋하는 것을 선호

끌어오기 요청 작성자로서, 작성자의 끌어오기 요청에 검토자의 피드백을 직접 통합시킬 때 고려해야 할 제안들이 많아 배치 프로세스로 추가하는 편이 좋은 경우, 논평 텍스트란 하단에 있는 *Commit suggestion* 버튼을 사용하지 말고:

1. Files changed 탭으로 전환하십시오.
2. 작성자가 포함시키고 싶은 각 변경사항마다 *Add suggestion to batch* 를 클릭하십시오. 추가할 때마다 숫자가 올라가는 것을 볼 수 있습니다.
3. 작성자의 끌어오기 요청에 제안들을 적용할 준비가 끝나면 *Commit suggestions* 버튼을 누른 다음, 변경 사항에 대한 설명을 입력하십시오.

이렇게 하면 작성자 분기에 모든 수정 사항을 커밋 한번으로 추가할 것입니다. 변경 사항 이력도 더 알아보기 쉽고, 저장소 팔로워들에게 가는 알림도 줄어듭니다. 의도하지는 않았지만, 작성자가 클릭해야 하는 횟수도 줄어들 겁니다.

수정 작업

새 끌어오기 요청은 자동적으로 끌어오기 요청 (pull request) 목록 에 추가될 것입니다. 다른 편집자들과 운영자들이 작성자의 PR 을 검토해서 제안을 커밋하거나 수정을 요구할 수도 있습니다.

끌어오기 요청은 자동화된 빌드 점검도 (예: rst(reStructuredText) 서식 작업의 경우 파이썬 코드 문법을 점검) 촉발할 것입니다. 점검 보고서는 페이지 하단에 표시됩니다. 오류를 발견했을 경우, 작성자의 커밋 (commit) 옆에 빨간 가위표가 나타날 것입니다. 이 빨간 가위표 또는 끌어오기 요청 페이지 하단의 요약 부분에 있는 `Details` 를 클릭하면 오류의 세부 정보를 볼 수 있습니다. 보고된 모든 오류 또는 경고를 해결해야 `qgis/QGIS-Documentation` 저장소에 작성자의 변경 사항을 커밋할 수 있습니다.

작성자의 끌어오기 요청이 메인 저장소와 병합되기 전까지, 작성자의 끌어오기 요청을 향상시키거나 요구받은 수정 사항을 적용하거나 빌드 오류를 해결하기 위해 끌어오기 요청을 수정할 수 있습니다.

끌어오기 요청을 변경하려면 작성자의 끌어오기 요청 페이지의 Files changed 탭을 클릭한 다음 수정하고자 하는 파일명 옆에 있는 연필 버튼을 클릭하십시오.

작성자의 끌어오기 요청에서 커밋한 것과 동일한 분기를 변경하는 경우, 추가한 변경 사항이 끌어오기 요청에 자동적으로 추가될 것입니다. 따라서, 작성자가 해당 끌어오기 요청으로 해결하려는 문제에 관련되었을 경우에만 변경 사항을 추가해야 합니다.

또다른 문제점을 해결하고 싶다면, 해당 변경 사항을 위한 새 분기를 생성한 다음 앞의 단계를 반복하십시오.

모든 빌드 오류가 수정되고 작성자와 운영자가 작성자의 변경 사항에 대해 만족한 다음에야 운영자가 작성자의 기고 내용을 병합할 것입니다.

1.1.5 작성자의 병합된 분기 삭제

작성자의 변경 사항이 병합됐다면, 해당 분기를 삭제해도 됩니다. 목적을 다 한 분기를 삭제하면 사용자 저장소에 사용되지 않거나 이전 버전인 분기를 유지하지 않아도 됩니다.

1. QGIS 문서 저장소의 사용자 포크 (`https://github.com/<YourName>/QGIS-Documentation`) 로 가십시오.
2. *Branches* 탭을 클릭하십시오. *Your branches* 아래 사용자 분기 목록이 있을 겁니다.
3. Delete this branch 아이콘을 클릭하면 원하지 않는 모든 분기를 삭제할 수 있습니다.

1.2 Git 명령 프롬프트 도구 이용

깃허브 (GitHub) 웹 인터페이스는 QGIS 문서 저장소에 작성자의 기고 내용을 쉽게 업데이트하도록 도와주지만, 다음과 같은 도구들을 제공하진 않습니다:

- 작성자의 커밋을 그룹화하고 변경 이력을 삭제
- 메인 저장소와 일어날 수 있는 충돌을 해결
- 작성자의 변경 사항을 테스트하기 위한 문서를 빌드

더 강력하고 더 고급인 도구에 접근하려면 그리고 저장소의 로컬 사본을 보유하려면 먼저 작성자의 하드드라이브에 Git 을 설치 해야 합니다. 작성자가 자주 필요로 할 몇몇 기본 사항들을 다음 단락에서 소개합니다. 웹 인터페이스를 사용하기로 결정한 경우에도 신경 써야 할 규칙들도 함께 찾을 수 있을 겁니다.

다음에 나오는 코드 예시에서, `#` 로 시작하는 줄은 주석을 의미하지만 `$` 로 시작하는 줄은 작성자가 입력해야 하는 명령어를 나타냅니다.

1.2.1 로컬 저장소

이제 **작성자의** QGIS 문서 저장소의 로컬 사본을 받을 준비가 됐습니다.

다음 웹 URL 을 사용하면 작성자의 QGIS 저장소를 복제할 수 있습니다:

```
# move to the folder in which you intend to store the local repository
$ cd ~/Documents/Development/QGIS/
$ git clone https://github.com/<YourName>/QGIS-Documentation.git
```

이 명령줄은 예시일 뿐입니다. 작성자는 `<YourName>` 을 작성자의 깃허브 사용자명으로 대체하여 경로 및 저장소 URL 둘 다 작성자의 로컬 환경에 맞춰야 합니다.

다음을 점검하십시오:

```
# Enter the local repository
$ cd ./QGIS-Documentation
$ git remote -v
origin  https://github.com/<YourName>/QGIS-Documentation.git (fetch)
origin  https://github.com/<YourName>/QGIS-Documentation.git (push)
$ git branch
* master
```

- *origin* 은 작성자의 QGIS 문서 저장소의 원격 저장소의 명칭입니다.
- *master* 는 기본 주 분기입니다. 기고하는 데 절대로 이 분기를 이용해서는 안 됩니다! **절대로요!!**

다른 방법으로는 SSH 프로토콜을 사용해서 작성자의 QGIS 저장소를 복제할 수 있습니다:

```
# move to the folder in which you intend to store the local repository
$ cd ~/Documents/Development/QGIS/
$ git clone git@github.com:<YourName>/QGIS-Documentation.git
```

팁: 권한 거부 (퍼블릭키) 오류?

앞의 명령어로 권한 거부 (퍼블릭키) 오류가 반환됐다면 작성자의 SSH 키에 문제가 있을 수 있습니다. 자세한 내용은 깃허브 도움말 을 참조하십시오.

SSH 프로토콜을 사용한 경우 다음을 점검하십시오:

```
# Enter the local repository
$ cd ./QGIS-Documentation
$ git remote -v
origin  git@github.com:<YourName>/QGIS-Documentation.git (fetch)
origin  git@github.com:<YourName>/QGIS-Documentation.git (push)
$ git branch
* master
```

여기서 작업을 시작해도 괜찮지만, 길게 봤을 때 작성자의 작업 내용을 밀어낼 경우 (깃허브 처리 과정에서는 끌어오기 요청이라고 합니다) 작성자의 로컬/원격 저장소에서 `qgis/QGIS-Documentation` 저장소의 주 분기가 갈라져 나올 것이기 때문에 수많은 문제점이 발생할 것입니다. 그러면 메인 원격 저장소를 계속 추적하면서 분기와 작업해야 합니다.

1.2.2 또다른 원격 저장소 추가

주 프로젝트에서 작업을 추적할 수 있으려면, 작성자의 로컬 저장소에 새 원격 저장소를 추가하십시오. 이 새 원격 저장소가 QGIS 프로젝트에서 나온 QGIS 문서 저장소입니다.

```
$ git remote add upstream https://github.com/qgis/QGIS-Documentation.git
$ git remote -v
origin  https://github.com/<YourName>/QGIS-Documentation.git (fetch)
origin  https://github.com/<YourName>/QGIS-Documentation.git (push)
upstream        https://github.com/qgis/QGIS-Documentation.git (fetch)
upstream        https://github.com/qgis/QGIS-Documentation.git (push)
```

마찬가지로, SSH 프로토콜을 사용해서 사용자의 로컬 저장소에 원격 저장소를 추가할 수 있습니다:

```
$ git remote add upstream git@github.com:qgis/QGIS-Documentation.git
$ git remote -v
origin  git@github.com:<YourName>/QGIS-Documentation.git (fetch)
origin  git@github.com:<YourName>/QGIS-Documentation.git (push)
upstream        git@github.com:qgis/QGIS-Documentation.git (fetch)
upstream        git@github.com:qgis/QGIS-Documentation.git (push)
```

이제 두 원격 저장소들 사이에서 선택할 수 있습니다.

- **작성자** 의 원격 저장소에 작성자의 로컬 분기를 밀어내는 *origin*
- 공식 문서로 작성자의 작업 내용을 (그럴 권한이 있을 경우) 통합하거나, 공식 저장소의 마스터 분기에서 나온 작성자의 로컬 저장소의 마스터 분기를 업데이트하는 *upstream*

참고: *upstream* 은 라벨일 뿐으로, 일종의 표준 명칭이지만 원하는 대로 명명할 수 있습니다.

1.2.3 작성자의 기반 분기 업데이트

새 작성 작업을 시작하기 전에, 항상 작성자의 로컬 저장소에 있는 마스터 분기를 업데이트해야 합니다. 테스트 버전 문서에 변경 사항을 밀어내려 한다고 가정하고, 다음 명령어를 실행하십시오:

```
# switch to master branch (it is easy to forget this step!)
$ git checkout master
# get "information" from the master branch in the upstream repository
# (aka qgis/QGIS-Documentation's repository)
$ git fetch upstream master
# merge update from upstream/master to the current local branch
# (which should be master, see step 1)
$ git merge upstream/master
# update **your** remote repository (aka <YourName>/QGIS-Documentation)
$ git push origin master
```

이제 작성자는 QGIS 문서 저장소의 공식 `master` 분기와 동일한 버전의 `master` 분기를 가진 로컬 및 원격 저장소를 얻었습니다. 문서 작성을 시작할 수 있습니다.

참고: 배포판 문서에 기고하고자 할 경우 분기를 전환

테스트 버전 문서 작업과 동시에, 최신 배포판 의 문제점들도 계속 고쳐나가고 있습니다. 작성자가 배포판 문서에도 기고할 수 있다는 의미지요. 앞 단락에 나온 예시 코드에서 `master` 를 최신 문서의 대응하는 분기로 치환하면 됩니다.

1.2.4 작성자의 작업 분기로 기고

이제 작성자의 기반 분기가 업데이트됐으니, 작성 내용을 추가할 전용 분기를 생성해야 합니다. 언제나 기반 분기가 아닌 다른 분기에서 작업하십시오! 언제나요!

```
# Create a new branch
$ git checkout -b myNewBranch
# checkout means go to the branch
# and -b flag creates a new branch if needed, based on current branch
# Let's check the list of existing branches (* indicates the current branch)
$ git branch
master
release_2.18
...
* myNewBranch
# You can now add your contribution, by editing the concerned file(s)
# with any application (in this case, vim is used)
$ vim myFile
# once done
$ git add myFile
$ git commit
```

커밋/밀어내기 (commit/push) 명령어에 대해:

- 오직 하나의 (원자 단위의 변경) 내용만, 예를 들어 오직 문제점 하나만 다루도록 해보십시오.
- 작성자의 커밋 제목 및 설명에 무엇을 변경했는지 꼼꼼히 서술해보십시오. 첫 줄은 제목으로, 대문자로 시작해야 하고 문자 80 개의 길이 제한이 있으며 `.` 로 끝나서는 안 됩니다. 제목은 간결하게 적으십시오. 설명은 더 길어도 되고, `.` 로 끝나며, 더 상세하게 서술할 수 있습니다.
- 어떤 문제점인지 식별하기 위해 `#` 뒤에 문제점 번호를 적으십시오. 해당 버그 티켓을 해결했다면 그 앞에 `Fix` 를 입력하십시오. 작성자의 커밋이 해당 티켓을 폐지할 것입니다.

작성자의 로컬 분기에 변경 사항을 저장하고 커밋했으니, 이제 끌어오기 요청을 생성하기 위해 변경 사항을 작성자의 원격 저장소에 전달해야 합니다:

```
$ git push origin myNewBranch
```

1.2.5 작성자의 변경 사항 공유

이제 작성자의 GitHub 저장소로 가서 앞 단락에서 설명한 대로끌어오기 요청을 생성 할 수 있습니다. 작성자의 분기에서 목표인 공식 QGIS 문서 저장소에 있는 원격 분기로 끌어오기 요청을 생성했는지 확인하십시오.

1.2.6 작성자의 로컬 및 원격 저장소 청소

작성자의 PR 이 공식 QGIS 문서에 통합된 다음, 작성자의 분기를 삭제할 수 있습니다. 이런 방식으로 많은 작업을 하는 경우, 몇 주만 지나도 쓸데없는 분기들을 많이 보유하게 될 겁니다. 따라서 다음 방법을 통해 작성자의 저장소를 말끔히 유지하십시오.

```
# delete local branch
$ git branch -d myNewBranch
# Remove your remote myNewBranch by pushing nothing to it
$ git push origin :myNewBranch
```

그리고 작성자의 로컬 저장소에 있는 `master` 분기를 업데이트하는 것도 잊지 마십시오!

1.3 더 읽어볼 거리

- 앞에서 설명한 깃허브 웹 인터페이스와 git 명령 줄 도구 이외에도, 작성자가 문서에 기고 내용을 생성하고 관리하는 데 사용할 수 있는 GUI 응용 프로그램 도 있습니다.
- 끌어오기 요청의 변경 사항이 최근 대상 분기에 밀어낸 변경 사항과 충돌을 일으키는 경우, 병합할 수 있으려면 먼저 충돌을 해소해야 합니다:
  - 서로 경쟁 관계에 있는 몇몇 행에 관련된 충돌인 경우, 깃허브 끌어오기 요청 (pull request) 페이지에 있는 *Resolve conflicts* 버튼을 사용할 수 있습니다. 이 버튼을 클릭하면 깃허브에서 병합 충돌 해결 에서 설명하는 대로 문제점을 해결합니다.
  - 파일 재명명 또는 제거에 관련된 충돌인 경우, git 명령 줄을 사용해서 충돌을 해소해야 합니다. 일반적으로, 먼저 `git rebase targetBranch` 호출을 이용해서 대상 분기를 사용자 분기 기반으로 새로 설정하고 보고된 충돌을 해결해야 합니다. 명령줄을 사용하여 병합 충돌 해결 에서 더 자세한 내용을 읽어보십시오.
- 때때로, 퇴고 과정의 끝자락에서 그럴 필요가 없는 데도 커밋 여러 개로 갈라진 변경 사항을 발견할 수도 있습니다. 이런 커밋들을 더 적은 개수와 더 의미 있는 커밋 메시지로 압축할 수 있는 git 명령 줄 도구가 있습니다. 자세한 내용은 명령줄에서 깃 다시 지정 사용 을 확인하세요.

CHAPTER 2

작성 지침

- 문서 작성
  - 표제
  - 목록
  - 들여쓰기
  - 내부 처리 태그
  - 라벨/참조
  - 도표와 이미지
    - 그림
    - 대체물
    - 도형
    - 표
  - 색인
  - 특별 언급
  - 코드 조각
  - 주석
- 스크린샷 관리
  - 새 스크린샷 추가
  - 스크린샷 번역
- 공간 처리 알고리즘 문서 작성

일반적으로, QGIS 프로젝트를 위한 reST 문서를 작성할 때 파이썬 문서 작성 스타일 지침서 를 따라주시기 바랍니다. 편의를 위해, 다음 부분에서 QGIS 문서 작성시 따라야 할 일반 규칙 모음을 소개합니다.

2.1 문서 작성

2.1.1 표제

문서의 `.rst` 파일 하나가 웹페이지 하나와 대응합니다.

밑줄 (1 단계 제목의 경우 윗줄도 포함) 그어진 제목을 통해 텍스트의 구조를 구성하는 절 (section) 을 식별합니다. 동일한 단계의 제목은 반드시 동일한 특수문자로 밑줄을 작성해야 합니다. QGIS 문서의 경우, 장 (chapter), 절 (section), 항 (subsection), 목 (minisec) 에 대해 다음과 같은 스타일을 사용해야 합니다.

```
********
Chapter
********

Section
=======

Subsection
----------

Minisec
.......

Subminisec
^^^^^^^^^^
```

2.1.2 목록

목록은 텍스트를 구성하는 데 유용합니다. 모든 목록에 공통되는 간단한 규칙은 다음과 같습니다:

- 목록의 모든 항목을 대문자로 시작하십시오.
- 간단한 문장 한 줄만 담고 있는 목록 항목 뒤에 구두점을 찍지 마십시오.
- 복잡한 문장 하나 또는 문장 여러 개로 이루어진 목록 항목의 구두점으로 마침표 (`.`) 를 사용하십시오.

2.1.3 들여쓰기

ReStructuredText 의 들여쓰기는 목록 또는 마크업 마커 와 정렬되어야 합니다. 들여쓰기로 블록 인용 (block quotes) 을 할 수도 있습니다. 들여쓰기 사양 을 참조하세요.

```
#. In a numbered list, there should be
   three spaces when you break lines
#. And next items directly follow

   * Nested lists
   * Are also possible
   * And when they also have
     a line that is too long,
     the text should be naturally
     aligned
   * and be in their own paragraph

However, if there is an unindented paragraph, this will reset the numbering:
```

(다음 페이지에 계속)

(이전 페이지에서 계속)

```
#. This item starts at 1 again
```

2.1.4 내부 처리 태그

태그를 사용해서 항목을 강조할 수 있습니다.

- **메뉴 GUI** : 일련의 완전한 메뉴 선택 순서를 표시합니다. 하위 메뉴를 선택해서 특정한 작업을 수행하는 순서, 또는 해당 순서의 하위 순서를 포함합니다.

  ```
  :menuselection:`menu --> submenu`
  ```

- **대화창 및 탭 제목**: 창 제목, 탭 제목, 버튼 그리고 옵션 라벨을 포함하는 대화형 사용자 인터페이스의 일부로써 라벨을 표시합니다.

  ```
  :guilabel:`title`
  ```

- **파일명 또는 디렉터리**

  ```
  :file:`README.rst`
  ```

- **팝업 텍스트를 가진 아이콘**

  ```
  |icon| :sup:`popup_text`
  ```

  (다음*image* 를 참조하십시오)

- **키보드 단축키**

  ```
  :kbd:`Ctrl+B`
  ```

  `Ctrl+B` 와 같이 보일 겁니다.

  키보드 단축키를 설명할 때, 다음과 같은 관례를 따라야 합니다:

  - 문자 키는 대문자로 표시합니다: `S`
  - 특수 키는 첫 문자를 대문자로 표시합니다: `Esc`
  - 키 조합은 키 사이에 공백 없이 + 기호를 넣어 표시합니다: `Shift+R`

- **사용자 텍스트**

  ```
  ``label``
  ```

- **Layer names** When referring to layers, format as inline code:

  ```
  ``layer name``
  ```

2.1.5 라벨/참조

텍스트 안에 앵커 (anchor) 를 사용해서 절 (section) 또는 페이지로 연결되는 하이퍼링크를 생성합니다.

다음은 절 (예: 라벨/참조 제목) 의 앵커를 생성하는 예시입니다.

```
.. _my_anchor:

Label/reference
---------------
```

동일 페이지 안에서 참조를 호출하려면 다음과 같이 작성합니다.

```
see my_anchor_ for more information.
```

다음과 같은 내용을 반환할 것입니다.

see *my_anchor* for more information.

'앵커'가 가리키는 라인/객체로 넘어갈 것을 아시겠습니까? 아포스트로피를 쓸 필요는 없지만 앵커 다음에 반드시 빈 줄을 넣어야 합니다.

문서 어디에서든 동일한 장소로 넘어가는 또다른 방법은 `:ref:` 기능을 이용하는 것입니다.

```
see :ref:`my_anchor` for more information.
```

앵커 대신 캡션으로 (이 경우 현재 절의 제목으로) 링크를 생성할 것입니다:

see 라벨/참조 for more information.

이렇게 참조 1(*my_anchor*) 과 참조 2(라벨/참조) 라는 방법이 있습니다. 참조가 완전한 캡션을 표시하는 경우가 많기 때문에, 절 이라는 단어를 따로 작성할 필요는 없습니다. 또한 참조를 설명하는 데 사용자 지정 캡션을 쓸 수도 있습니다.

```
see :ref:`Label and reference <my_anchor>` for more information.
```

다음을 반환합니다:

see *Label and reference* for more information.

2.1.6 도표와 이미지

그림

다음은 이미지를 삽입하는 예시입니다.

```
.. figure:: /static/common/logo.png
   :width: 10 em
```

다음과 같은 그림을 반환합니다.

대체물

텍스트 안에 이미지를 삽입할 수도 있고, 또는 다른 곳에서도 쓰기 위해 별명 (alias) 을 추가할 수도 있습니다. 문단 안에 이미지를 삽입하려면, 먼저 `source/substitutions.txt` 파일에 별명을 생성하십시오:

```
.. |nice_logo| image:: /static/common/logo.png
               :width: 1 em
```

그 다음 사용자의 문장에 별명을 호출하십시오:

```
My paragraph begins here with a nice logo |nice_logo|.
```

이 예시는 다음과 같이 표시될 것입니다:

My paragraph begins here with a nice logo .

깃허브에서 최대한 HTML 렌더링에 가까운 미리보기 렌더링을 가능하게 하려면, 사용자가 변경한 파일의 마지막에 이미지 대체물 호출도 추가해야 합니다. `substitutions.txt` 에서 복사-붙여넣기 하거나, `scripts/find_set_subst.py` 스크립트를 실행하면 됩니다.

참고: 현재, 일관성을 보장하고 QGIS 아이콘 이용을 돕기 위해별명 장에 사용자가 쓸 수 있는 별명 목록을 생성해 놓았습니다.

도형

```
.. _figure_logo:

.. figure:: /static/common/logo.png
   :width: 20 em
   :align: center

   A caption: A logo I like
```

결과는 다음과 같습니다:

다른 참조와 충돌이 일어나는 것을 피하기 위해, 언제나 도형 앵커를 `_figure_` 로 시작하고 도표 캡션과 쉽게 연관지을 수 있는 용어를 사용하십시오. 이미지에 대해서는 필수적으로 중앙 정렬만 사용해야 하지만, 필요한 경우 도형을 위한 다른 (`width`, `height`, `scale` 등등 같은) 옵션들은 자유롭게 사용해도 됩니다.

스크립트가 문서 생성시 HTML 및 PDF 버전에 있는 도형의 캡션 앞에 자동으로 생성된 번호를 삽입할 것입니다.

캡션을 쓰려면 (*My caption* 참조) `.. figure::` 단락의 빈 줄 다음에 원하는 텍스트를 삽입하면 됩니다.

다음과 같이 참조 라벨을 사용해서 도형을 참조할 수 있습니다:

그림 2.1: A caption: A logo I like

```
see :numref:`figure_logo`
```

다음과 같이 렌더링됩니다:

see 그림 2.1

도형을 참조하는 데에는 이 방법이 더 낫습니다.

참고: `:numref:` 가 제대로 작동하기 위해서는 도형이 **캡션을 가지고 있어야만 합니다**.

참조의 경우 `:numref:` 대신 `:ref:` 를 사용할 수도 있지만, `:ref:` 는 이미지 캡션 전체를 반환합니다.

```
see :ref:`figure_logo`
```

다음과 같이 렌더링됩니다:

see *A caption: A logo I like*

표

간단한 테이블은 다음과 같이 코딩할 수 있습니다:

```
=======  =======  =======
x        y        z
=======  =======  =======
1        2        3
4                 5
=======  =======  =======
```

다음과 같이 렌더링될 것입니다:

x	y	z
1	2	3
4		5

공백을 남기려면 \ (백슬래시) 뒤에 공백을 삽입하십시오.

좀 더 복잡한 표를 만들어서 참조할 수도 있습니다:

```
.. _my_drawn_table:

+---------------+--------------------+
| Windows       | macOS              |
+---------------+--------------------+
| |win|         | |osx|              |
+---------------+--------------------+
| and of course not to forget |nix|  |
+-------------------------------------+

My drawn table, mind you this is unfortunately not regarded as a caption

You can reference it like this: my_drawn_table_.
```

결과물은 다음과 같습니다:

Windows	macOS
and of course not to forget	

My drawn table, mind you this is unfortunately not regarded as a caption

You can reference to it like this *my_drawn_table*.

더욱 복잡한 표를 원한다면, `list-table` 을 사용하는 편이 좋습니다:

```
.. list-table::
   :header-rows: 1
   :widths: 20 20 20 40

   * - What
     - Purpose
     - Key word
     - Description
   * - **Test**
     - ``Useful test``
     - complexity
     - Geometry.  One of:

       * Point
       * Line
```

결과물은 다음과 같습니다:

What	Purpose	Key word	Description
Test	`Useful test`	complexity	Geometry. One of: • Point • Line

2.1.7 색인

색인 (index) 은 독자가 문서에서 정보를 손쉽게 찾을 수 있게 해주는 방법입니다. QGIS 문서는 몇몇 필수적인 색인들을 제공합니다. 활용성이 매우 높은 (알기 쉽고 일관적이며 서로 긴밀하게 연결되어 있는) 색인 집합을 제공할 수 있도록 해주는 규칙이 몇 개 있습니다:

- 색인은 사람이 읽고 이해하고 번역할 수 있어야 합니다. 색인은 많은 단어들로 이루어질 수 있지만, 단어들 사이를 불필요한 _, - 등의 문자들로 연결해서는 안 됩니다. 예를 들면 `loading_layers` 또는 `loadingLayers` 보다는 `Loading layers` 를 사용해야 합니다.
- 단어의 철자가 특별히 그렇게 되어 있는 경우를 제외하고, 색인의 첫 글자만 대문자로 쓰십시오. 예: `Loading layers`, `Atlas generation`, `WMS`, `pgsql2shp`.
- 올바른 철자로 된 가장 편리한 표현을 재사용하고 불필요한 중복을 피하기 위해, 기존 색인 목록 을 살펴보십시오.

RST 에는 몇몇 색인 태그들이 있습니다. 일반 텍스트 안에서는 그때그때 즉시 처리되는 (inline) `:index:` 태그를 사용하면 됩니다:

```
QGIS can load several :index:`Vector formats` supported by GDAL ...
```

또는 다음 문장의 시작 부분을 링크하는 블록 수준 마크업 `.. index::` 를 사용해도 됩니다. 앞에서 언급한 규칙 때문에, 블록 수준 태그를 사용할 것을 권장합니다:

```
.. index:: WMS, WFS, Loading layers
```

`single`, `pair` 그리고 `see` 같은 색인 파라미터를 사용할 것도 권장합니다. 좀 더 구조화되고 상호 연결된 색인 테이블을 작성하기 위해서입니다. 인덱스 생성에 대해 더 자세히 알고 싶다면 색인 생성하기 를 참조하세요.

2.1.8 특별 언급

때때로, 사용자에게 경고하거나 상기시키거나 힌트를 주기 위해 설명의 몇몇 포인트를 강조하고 싶을 수도 있습니다. QGIS 문서에서는 `.. warning::`, `.. seealso::`, ``.. note::`` 그리고 `.. tip::` 같은 reST 특별 지시어 (special directive) 를 사용합니다. 이 지시어들은 작성자의 언급 사항을 강조하는 프레임을 생성합니다. 자세한 내용은 문장 수준 마크업 을 참조하세요. 경고 및 힌트 둘 다 명료하고 적절한 제목이 반드시 필요합니다.

```
.. tip:: **Always use a meaningful title for tips**

   Begin tips with a title that summarizes what it is about. This helps
   users to quickly overview the message you want to give them, and
   decide on its relevance.
```

2.1.9 코드 조각

예를 들기 위해 코드 조각 (code snippet) 을 삽입하고 싶을 수도 있습니다. 이런 경우, 한 줄 아래에 `::` 지시어를 삽입하고 코드를 작성하십시오. 더 나은 렌더링을 원한다면, 특히 코드 언어에 따라 색상 강조를 적용하고 싶다면, `.. code-block:: xml` 같은 코드 블록 지시어를 사용하십시오. 더 자세한 내용은 코드 표시하기 를 참조하세요.

참고: 참고, 힌트 및 경고 프레임 안의 텍스트는 번역할 수 있지만, 코드 블록 프레임은 번역할 수 없다는 점을 주의하십시오. 즉 코드에 관련되지 않은 언급은 자제하고 가능한 한 짧게 작성해야 한다는 뜻입니다.

2.1.10 주석

어떤 번역 소프트웨어도 주석을 식별하지 못 하며, 또 주석은 PDF 포맷으로 제대로 변환되지도 않습니다. 따라서 가능하다면 어떤 문서에도 주석을 쓰지 마십시오.

다음은 주석을 생성하기 위한 문법입니다 (이와 같이 표시됩니다[1])

```
blabla [1]_
```

다음을 가리키게 됩니다.

2.2 스크린샷 관리

2.2.1 새 스크린샷 추가

보기 좋은 새 스크린샷을 생성하기 위한 힌트를 몇 개 드리겠습니다. 스크린샷이 참조하는 `.rst` 파일과 동일한 폴더 안에 있는 이미지 폴더 (`img/`) 에 스크린샷 이미지를 저장해야 합니다.

- 스크린샷을 캡처하기 위해 사용된 기존 QGIS 프로젝트 몇 개를 이 저장소의 `./qgis-projects` 폴더에서 찾아볼 수 있습니다. QGIS 다음 버전에서 스크린샷을 다시 캡처하는 데 도움이 될 것입니다. 이 프로젝트들은 QGIS 샘플 데이터 (일명 알래스카 데이터셋) 을 이용하는데, QGIS 문서 저장소와 동일한 폴더에 이 데이터셋을 압축 해제해야 합니다.
- 기능을 보여주는 데 필요한 최소한의 공간만 차지하도록 창 크기를 줄입니다. (작은 대화창을 화면 전체 크기로 캡처할 필요는 없겠죠.)
- 직관적이고 단순할수록 좋습니다. (모든 툴바를 활성화시킬 필요는 없습니다.)
- 스크린샷의 크기를 이미지 편집기로 조정하지 마십시오. 필요할 경우 `.rst` 파일 내부에서 이미지 크기를 설정하면 됩니다. (해상도를 제대로 높이지 않고 크기를 줄일 경우 보기에 안 좋은 이미지가 됩니다.)
- 배경을 잘라내십시오.
- 배경이 하얀색이 아닐 경우 상단 모서리를 투명하게 조정하십시오.
- 인쇄 해상도를 `135 dpi` 로 조정하십시오. 예를 들어 김프 (GIMP) 를 사용할 경우 *Image ▯ Print size* 메뉴에서 인쇄 해상도를 조정한 다음 저장하십시오. 이렇게 하면 이미지가 HTML 에서는 원래 크기로, PDF 에서는 훌륭한 인쇄 해상도로 나올 겁니다. 다음과 같이 ImageMagick 변환 명령어를 써서 여러 이미지들을 배치 작업할 수도 있습니다:

  ```
  convert -units PixelsPerInch input.png -density 135 output.png
  ```

- 스크린샷을 `.png` 로 저장하십시오. (`.jpeg` 특허 침해를 피하기 위해서입니다.)

[1] 핵심 플러그인의 업데이트

- 스크린샷은 텍스트가 설명하는 내용을 보여주어야 합니다.

팁: 우분투를 사용한다면, 전체 수준 메뉴 기능을 제거해서 더 작은 응용 프로그램 스크린샷을 생성하려면 다음 명령어를 사용하십시오:

```
sudo apt autoremove appmenu-gtk appmenu-gtk3 appmenu-qt
```

2.2.2 스크린샷 번역

번역된 사용자 지침서를 위한 스크린샷을 생성하길 원한다면 다음 규칙을 따르세요:

번역 이미지는 `img/<your_language>/` 폴더에 있어야 합니다. 영어'원본'스크린샷과 동일한 파일명으로 저장하십시오.

2.3 공간 처리 알고리즘 문서 작성

공간 처리 알고리즘을 위한 문서를 작성하고자 할 경우, 다음과 같은 지침들을 고려하십시오:

- 공간 처리 알고리즘 도움말 파일은 QGIS 사용자 설명서의 일부입니다. 따라서 사용자 지침서 및 기타 문서들과 동일한 서식을 이용하십시오.
- 각 알고리즘 문서는 그에 대응하는 **제공자 (provider)** 폴더에 대응하는 **그룹 (group)** 파일로 저장돼야 합니다. 예를 들어 *Voronoi polygon* 알고리즘은 *QGIS* 제공자와 *vectorgeometry* 그룹에 속해 있습니다. 따라서 설명을 추가해야 할 정확한 파일은 다음과 같습니다: `source/docs/user_manual/processing_algs/qgis/vectorgeometry.rst`

  참고: 지침서 작성을 시작하기 전에, 이미 해당 알고리즘을 설명하고 있는지 확인하십시오. 이미 설명되어 있다면, 기존 설명을 강화할 수 있습니다.

- 각 알고리즘이 제공자 이름과 알고리즘 자체의 유일한 이름에 대응하는 앵커 *(anchor)* 를 가진다는 사실이 **극히** 중요합니다. 도움말 버튼이 도움말 페이지에서 정확한 부분을 열 수 있게 해주기 때문입니다. 이 앵커는 다음과 같이 제목 **위에** 배치시켜야 합니다 (라벨/참조 도 참조하세요):

  ```
  .. _qgisvoronoipolygons:

  Voronoi polygons
  ----------------
  ```

  알고리즘 이름을 알고 싶다면 공간 처리 툴박스에 있는 알고리즘 위에 마우스를 가져가기만 하면 됩니다.
- 알고리즘 설명의 첫 문장으로"이 알고리즘은 이런 기능을 하고 저런 기능도…"같은 문장을 피하십시오. 다음과 같이 좀 더 일반적인 표현을 사용하도록 하십시오:

  ```
  Takes a point layer and generates a polygon layer containing the...
  ```

- 알고리즘 이름을 반복하며 알고리즘의 기능을 설명하지 마세요. 특히 파라미터 자체를 설명할 때 파라미터 이름을 반복하지 마십시오. 예를 들어 알고리즘이 `Voronoi polygon` 일 경우 입력 레이어 를 폴리곤을 계산할 레이어 라고 설명하는 편이 좋습니다.
- 알고리즘 설명에서 해당 알고리즘이 QGIS 에서 단축키를 가지는지 또는 제자리 (in-place) 편집 작업을 지원하는지 명시하십시오.

- 이미지를 추가하십시오! 백문이 불여일견입니다! `.png` 포맷을 사용하고 문서를 위한 일반 지침을 따르십시오. (더 자세한 내용은도표와 이미지 를 참조하세요.) 정확한 폴더에 이미지 파일을 저장하십시오. 예를 들어 작성자가 편집중인 `.rst` 파일이 있는 폴더의 `img` 폴더에 이미지를 저장해야 합니다.
- 필요한 경우, 알고리즘에 대한 추가 정보를 제공하는"참조 (See also)"부분에 (예를 들면 출판물 또는 웹페이지를 가리키는) 링크를 추가하십시오. 정말로 참조해야 할 내용이 있을 경우에만"참조"부분을 추가하십시오. "참조"부분을 비슷한 알고리즘들을 가리키는 링크들로 채우는 것도 좋은 방법입니다.
- 알고리즘 파라미터와 산출물에 대한 명확한 설명을 하십시오: 기존 알고리즘들을 참고하세요.
- 알고리즘 옵션의 자세한 설명을 복제하지 마십시오. 파라미터 설명에 이 정보를 추가하세요.
- 알고리즘 또는 파라미터 설명에 벡터 도형 유형 정보를 추가하지 마십시오. 이미 파라미터 설명에서 해당 정보를 볼 수 있습니다.
- 파라미터의 기본값을 다음과 같이 추가하십시오:

```
* - **Number of points**
  - ``NUMBER_OF_POINTS``
  - [number]

    Default: 1
  - Number of points to create
```

- 파라미터가 지원하는 입력 유형 을 설명하십시오. 다음 몇몇 유형들 가운데 하나를 고를 수 있습니다:

파라미터/산출물 유형	Description	시각 지표
포인트 벡터 레이어	`vector: point`	
라인 벡터 레이어	`vector: line`	
폴리곤 벡터 레이어	`vector: polygon`	
일반 벡터 레이어	`vector: any`	
벡터 필드 숫자	`tablefield: numeric`	1.2
벡터 필드 문자열	`tablefield: string`	abc
벡터 필드 일반	`tablefield: any`	
래스터 레이어	`raster`	
래스터 밴드	`raster band`	
HTML 파일	`html`	
테이블 레이어	`table`	
표현식	`expression`	ε
포인트 도형	`coordinates`	
범위 (extent)	`extent`	
좌표계	`crs`	
열거 목록 (enumeration)	`enumeration`	
목록	`list`	
숫자	`number`	1,00
문자열	`string`	Display name lakes.shp
불 (Boolean)	`boolean`	
파일 경로	`folder`	
파일	`file`	
매트릭스	`matrix`	

다음 페이지에 계속

표 2.1 – 이전 페이지에서 계속

파라미터/산출물 유형	Description	시각 지표
레이어	`layer`	
입력 유형과 동일한 산출 유형	`same as input`	
정의	`definition`	
Point	`point`	
다중레이어 (MultipleLayer)	`multipleLayers`	
범위 (range)	`range`	
인증환경설정 (AuthConfig)	`authconfig`	
메시 (mesh)	`mesh`	
조판 (layout)	`layout`	
조판항목 (LayoutItem)	`layoutitem`	
색상	`color`	
축척	`scale`	

- 기존에 잘 문서화된 알고리즘을 공부해서 유용한 모든 조판을 복사하십시오.
- 작성을 완료하면, 단계별 공헌 방법 에서 설명하는 지침을 따라 작성자의 변경 사항을 커밋하고 PR(Pull Request) 을 생성하면 됩니다.

다음은 작성자에게 도움이 될 조판과 설명을 담고 있는 기존 알고리즘 의 예시입니다:

```
.. _qgiscountpointsinpolygon:

Count points in polygon
------------------------
Takes a point and a polygon layer and counts the number of points from the
point layer in each of the polygons of the polygon layer.
A new polygon layer is generated, with the exact same content as the input
polygon layer, but containing an additional field with the points count
corresponding to each polygon.

.. figure:: img/count_points_polygon.png
  :align: center

  The labels in the polygons show the point count

An optional weight field can be used to assign weights to each point.
Alternatively, a unique class field can be specified. If both options
are used, the weight field will take precedence and the unique class field
will be ignored.

``Default menu``: :menuselection:`Vector --> Analysis Tools`

Parameters
..........

.. list-table::
   :header-rows: 1
   :widths: 20 20 20 40

   * - Label
     - Name
     - Type
     - Description
   * - **Polygons**
```

(다음 페이지에 계속)

(이전 페이지에서 계속)

```
     - ``POLYGONS``
     - [vector: polygon]
     - Polygon layer whose features are associated with the count of
       points they contain
   * - **Points**
     - ``POINTS``
     - [vector: point]
     - Point layer with features to count
   * - **Weight field**

       Optional
     - ``WEIGHT``
     - [tablefield: numeric]
     - A field from the point layer.
       The count generated will be the sum of the weight field of the
       points contained by the polygon.
   * - **Class field**

       Optional
     - ``CLASSFIELD``
     - [tablefield: any]
     - Points are classified based on the selected attribute and if
       several points with the same attribute value are within the
       polygon, only one of them is counted.
       The final count of the points in a polygon is, therefore, the
       count of different classes that are found in it.
   * - **Count field name**
     - ``FIELD``
     - [string]

       Default: 'NUMPOINTS'
     - The name of the field to store the count of points
   * - **Count**
     - ``OUTPUT``
     - [vector: polygon]

       Default: [Create temporary layer]
     - Specification of the output layer type (temporary, file,
       GeoPackage or PostGIS table).
       Encoding can also be specified.

Outputs
.......

.. list-table::
   :header-rows: 1
   :widths: 20 20 20 40

   * - Label
     - Name
     - Type
     - Description
   * - **Count**
     - ``OUTPUT``
     - [vector: polygon]
     - Resulting layer with the attribute table containing the
       new column with the points count
```

CHAPTER 3

PyQGIS 쿡북 코드 작성

- 테스트 가능한 코드 조각 작성 방법
  - 문서 테스트 *Sphinx* 지시어
  - 그룹화 테스트
- 사용자 로컬 환경에서 코드 조각을 테스트하는 방법

PyQGIS-Developer-Cookbook 의 일부를 업데이트하거나 추가하려 한다면, 코드 조각을 자동적으로 테스트할 수 있도록 해주는 규칙 몇 가지를 따라야 합니다.

테스트 작업은 매우 중요합니다. 코드를 자동 확인해주는 과정이기 때문입니다. 오류가 있는 코드 조각 또는 구식 메소드를 사용하는 코드는 테스트를 통과하지 못 하고, 작성자가 문제점을 수정하도록 도와줄 알림이 뜰 것입니다.

테스트 작업을 위해, Sphinx 문서 테스트 확장 기능 을 사용합니다. 더 자세한 내용을 알고 싶다면 확장 기능 문서를 참조하세요.

3.1 테스트 가능한 코드 조각 작성 방법

테스트 가능한 코드 조각을 작성하는 것은 구식 방법과 크게 다르지 않습니다. 기본적으로, 다른 Sphinx 지시어 *(directive)* 를 이용해야 합니다.

3.1.1 문서 테스트 Sphinx 지시어

(자동적으로 코드 문법을 강조해주는) `.. code-block:: python` 지시어에 코드를 내장하는 대신, 이제 `.. testcode::` 에 코드를 내장해야 합니다. 그러니까 다음 대신:

```
.. code-block:: python

   crs = QgsCoordinateReferenceSystem("EPSG:4326")
   assert crs.isValid()
```

이제 이렇게 해야 합니다:

```
.. testcode::

   crs = QgsCoordinateReferenceSystem("EPSG:4326")
   assert crs.isValid()
```

예시 코드를 작성한 다음, 코드를 평가하고 자동으로 실행될 몇몇 언명 *(assertion)* 을 추가해야 합니다.

앞의 예시에서, 작성자는 좌표계를 생성하고 `assert crs.isValid()` 로 좌표계가 무결한지 **테스트** 합니다. 코드의 파이썬 문법이 틀렸거나 `crs.isValid()` 가 `False` 를 반환하는 경우, 테스트 작업 도중 이 코드는 통과하지 못 할 것입니다.

코드 조각에 대한 테스트를 성공적으로 실행하려면, 모든 클래스를 가져와서 코드 조각에 사용된 모든 변수를 선언해야만 합니다. 코드 조각 자체에 이들을 포함시킬 수도 있고 (이 경우 HTML 페이지에 보이게 됩니다) `.. testsetup::` 지시어에 추가시킬 수도 있습니다 (이 경우 HTML 페이지에 보이지 않습니다). `.. testsetup::` 지시어는 `.. testcode::` 앞에 와야 합니다:

```
.. testsetup::

   from qgis.core import QgsCoordinateReferenceSystem

.. testcode::

   crs = QgsCoordinateReferenceSystem("EPSG:4326")
   assert crs.isValid()
```

코드 조각이 객체를 생성하지 않는 (따라서 `assert object.isValid()` 같은 코드를 사용할 수 없는) 경우, `print()` 메소드를 사용해서 코드를 테스트할 수 있습니다. 이때 기대 산출물과 비교하기 위해 `.. testoutput::` 지시어에 기대 결과를 추가하십시오:

```
.. testcode::

   print("QGIS CRS ID:", crs.srsid())
   print("PostGIS SRID:", crs.postgisSrid())

.. testoutput::

   QGIS CRS ID: 3452
   PostGIS SRID: 4326
```

`.. testoutput::` 의 내용은 기본적으로 HTML 산출물에 보이게 됩니다. HTML 에서 이를 숨기려면 `:hide:` 옵션을 사용하십시오:

```
.. testoutput::
   :hide:

   QGIS CRS ID: 3452
   PostGIS SRID: 4326
```

참고: 코드 조각이 `print()` 선언을 하나라도 담고 있는 경우, 기대 산출물을 가진 `.. testoutput::` 을 추가해야만 합니다. 그렇지 않으면 테스트가 실패할 것입니다.

3.1.2 그룹화 테스트

각 reST 문서마다, 코드 조각들을 연속해서 테스트합니다. 즉 일련의 모든 코드 조각에 대해 하나의 `.. testsetup::` 을 사용할 수 있고, 그후 코드 조각들이 문서 앞부분에 선언된 변수들에 접근할 것이라는 뜻입니다.

아니면, 동일한 페이지에 있는 예시들을 그룹으로 분리해서 개별적으로 테스트를 실행할 수도 있습니다.

그룹에 코드 조각을 추가하려면 각각의 지시어에 (쉼표로 구분된) 하나 이상의 그룹명을 추가하면 됩니다.

```
.. testcode:: crs_crsfromID [, morenames]

   crs = QgsCoordinateReferenceSystem("EPSG:4326")
   assert crs.isValid()
```

`doctest` 가 각 코드 조각 그룹을 구분해서 개별적으로 테스트할 것입니다.

참고: 관련 내용과 어울리는 그룹명을 사용하십시오. <chapter>_<subchapter> 와 비슷한 이름을 사용하십시오. 예를 들어 crs_intro, crs_fromwkt 같은 이름 말이죠. 테스트에 통과하지 못 하는 경우, 이런 이름이 어디서 오류가 발생했는지 식별할 수 있게 해줄 것입니다.

어떤 그룹도 선언하지 않으면 코드 조각은 `default` 라는 그룹에 추가될 것입니다. 또는 그룹 이름으로 `*` 를 사용하는 경우, 그 코드 조각은 모든 그룹의 테스트에 포함될 것입니다. 이는 테스트 환경에서 사용하기에 일반적으로 유용한 방법입니다:

```
.. testsetup:: *

   from qgis.core import QgsCoordinateReferenceSystem
```

3.2 사용자 로컬 환경에서 코드 조각을 테스트하는 방법

참고: 다음은 리눅스 시스템에서 유효한 지침입니다.

파이썬 코드 조각을 테스트하려면, *QGIS* 가 설치되어 있어야 합니다. 다음과 같은 여러 가지 옵션이 있습니다:

- 파이썬 가상 환경에서 *Sphinx* 와 함께 설치된 사용자 시스템의 *QGIS* 를 사용할 수 있습니다:

```
make -f venv.mk doctest
```

- 직접 빌드한 *QGIS* 설치본을 사용할 수 있습니다. 다음을 해야 합니다:
  1. `venv.mk` 파일뿐만 아니라 사용자 지정 `Makefile` 확장자를 생성해야 합니다. 예를 들어 다음 내용을 가진 `user.mk` 파일처럼 말이죠:

     ```
     # Root installation folder
     QGIS_PREFIX_PATH = /home/user/apps/qgis-master

     include venv.mk
     ```

     또는:

     ```
     # build output folder
     QGIS_PREFIX_PATH = /home/user/dev/QGIS-build-master/output

     include venv.mk
     ```

  2. 그 다음, 대상 `doctest` 를 실행하는 데 사용하십시오:

     ```
     make -f user.mk doctest
     ```

- 공식 *QGIS* 도커 이미지 내부에서 대상 `doctest` 를 실행할 수 있습니다:

```
make -f docker.mk doctest
```

먼저 도커 (docker) 를 설치해야 합니다. 이 방법은 QGIS 를 내장한 도커 이미지를 사용하기 때문입니다.

CHAPTER 4

번역 지침

- 번역 과정
- 파일 번역
  - *Transifex* 에서 번역
  - *Qt Linguist* 로 번역
  - 사용자 지침서 번역
  - 번역 규칙 요약

이 지침은 번역자를 돕는 것이 목적입니다. 먼저 기술적으로 번역이 어떻게 이루어지는지 그 일반적인 과정을 설명합니다. 그 다음 실제로 네덜란드어로 번역된 영어 `.rst` 파일을 통해 번역 작업을 설명합니다. 마지막으로번역 규칙 을 요약할 것입니다.

참고: 이 지침들이 QGIS 문서에 중점을 두고 있긴 하지만, 다음 절부터 설명하는 방법 및 규칙은 QGIS 응용 프로그램이나 웹사이트 번역에도 적용할 수 있습니다.

4.1 번역 과정

QGIS 문서는 `.rst` 파일 포맷에 영어로 작성되었습니다. 번역을 제공하기 위해서는:

1. 미리 빌드된 스크립트로 `/QGIS-Documentation/locale/en` 폴더에 `.po` 확장자를 가진 영어를 위한 번역 파일을 생성합니다.
2. `.po` 파일에 있는 문장들이 Transifex 웹 플랫폼으로 푸시되어, 번역자가 편집기를 통해 영어에서 자신의 언어로 번역을 시작할 수 있게 됩니다.
3. 파일 번역율이 100% 에 도달하면, 번역된 문자열을 자동으로 `/QGIS-Documentation/locale/<language>` 에 있는 문서 저장소로 다시 끌어옵니다.

4. 다음 문서 빌드 시 (하루에 적어도 한 번은 빌드합니다–이 페이지 하단에서 시간을 확인해보세요) 스크립트가 문장들을 재사용해서 번역된 산출물을 생성합니다.
5. 100% 번역되지 않은 파일의 경우, 스크립트가 2 주마다 트랜시펙스 (Transifex) 에서 깃허브로 번역된 문자열들을 끌어와서 다음 빌드 시 번역된 문자열도 반영합니다.
6. `.rst` 파일이 업데이트될 때마다, 영문 `.po` 파일을 업데이트해서 Transifex 에 있는 대응 파일로 변경 사항을 밀어냅니다. 즉 이미 번역된 `.rst` 문서에 새로운 문단이 추가되는 경우, 새로운 또는 업데이트된 문장들만 번역된 `.po` 파일에 추가되고 번역되어야 한다는 뜻입니다.

참고: QGIS 데스크탑 번역의 특징

QGIS 응용 프로그램을 번역하는 데 있어 가장 중요한 차이점은, 응용 프로그램의 특정 버전을 형성하는 `.py`, `.cpp`, `.yaml` 파일에 들어 있는 번역할 수 있는 모든 문자열을 Transifex 에서 `.po` 파일들이 아니라 단일 `.ts` 파일 하나로 (예: `qgis-application/qgis_ko.ts (branch release-3_30)`) 밀어내고 끌어온다는 점입니다. (모든 배포된 버전의) 배포 시간에 (매일) 깃허브의 개발 분기로 번역된 문자열을 끌어옵니다.

현재 QGIS 를 번역하는 데 쓰이는 도구는 두 가지입니다.

- QGIS 를 번역하는 가장 쉽고 권장하는 방법인 Transifex 웹 플랫폼 은 앞에서 설명한 과정을 투명하게 보여주며, 번역할 수 있는 모든 텍스트를 번역자를 위해 한 곳에 모아 줍니다. 번역자는 원하는 파일을 골라 번역하기만 하면 됩니다. 번역된 파일은 다음 배포판이 푸시될 때까지 플랫폼에 저장됩니다.
- Qt 개발자 도구인 Qt Linguist 를 사용하면, 번역자가 로컬 환경에서 소스 코드로부터 `.po` (또는 `.ts`) 파일을 가져와 번역한 다음 다시 푸시해야 합니다.

어떤 도구를 선택하든, 번역 규칙은 동일하다는 사실을 명심하십시오.

4.2 파일 번역

어떻게 번역 작업을 해야 하는지 설명하기 위해, 열지도 플러그인을 예시로 이용하겠습니다. 이 예시를 영어에서 네덜란드 어로 번약할 텐데, 모든 언어의 다른 문서를 번역할 때도 실질적으로 동일합니다.

문서의 소스를 다음 위치에서 찾을 수 있습니다.

```
QGIS-Documentation/source/docs/user_manual/plugins/plugins_heatmap.rst
```

어째서 이 문서를 예시로 선택했을까요?

1. 이 문서는 이미지, 캡션, 헤더, 참조 및 별명 (replacement) 도 포함하고 있기 때문이고,
2. 내가 직접 작성했으니 번역하기도 쉽기 때문이지요;-)

빌드 과정에서 영어 `.po` 파일이 생성되는데, 다음 위치에서 이 파일을 찾을 수 있습니다:

```
QGIS-Documentation/locale/en/LC_MESSAGES/docs/user_manual/plugins/plugins_heatmap.po
```

이에 대응하는 네덜란드어 `.po` 파일은 (기본적으로 복사본입니다) 다음 위치에 있습니다:

```
QGIS-Documentation/locale/nl/LC_MESSAGES/docs/user_manual/plugins/plugins_heatmap.po
```

이 파일 옆에 아주 작은 `.mo` 파일도 함께 있을 것이며 이것은 어떤 번역도 이루어지지 않았다는 것을 뜻합니다.

4.2.1 Transifex 에서 번역

Transifex 를 통해 번역하기 위해 필요한 작업:

1. Transifex 계정을 생성하고 QGIS 프로젝트에 가입 해야 합니다.
2. 번역자가 언어팀의 일원이 되었다면, 대응하는 프로젝트를 (이 경우 `QGIS Documentation` 을) 클릭하십시오. 번역할 수 있는 언어들이 번역률과 함께 표시됩니다.

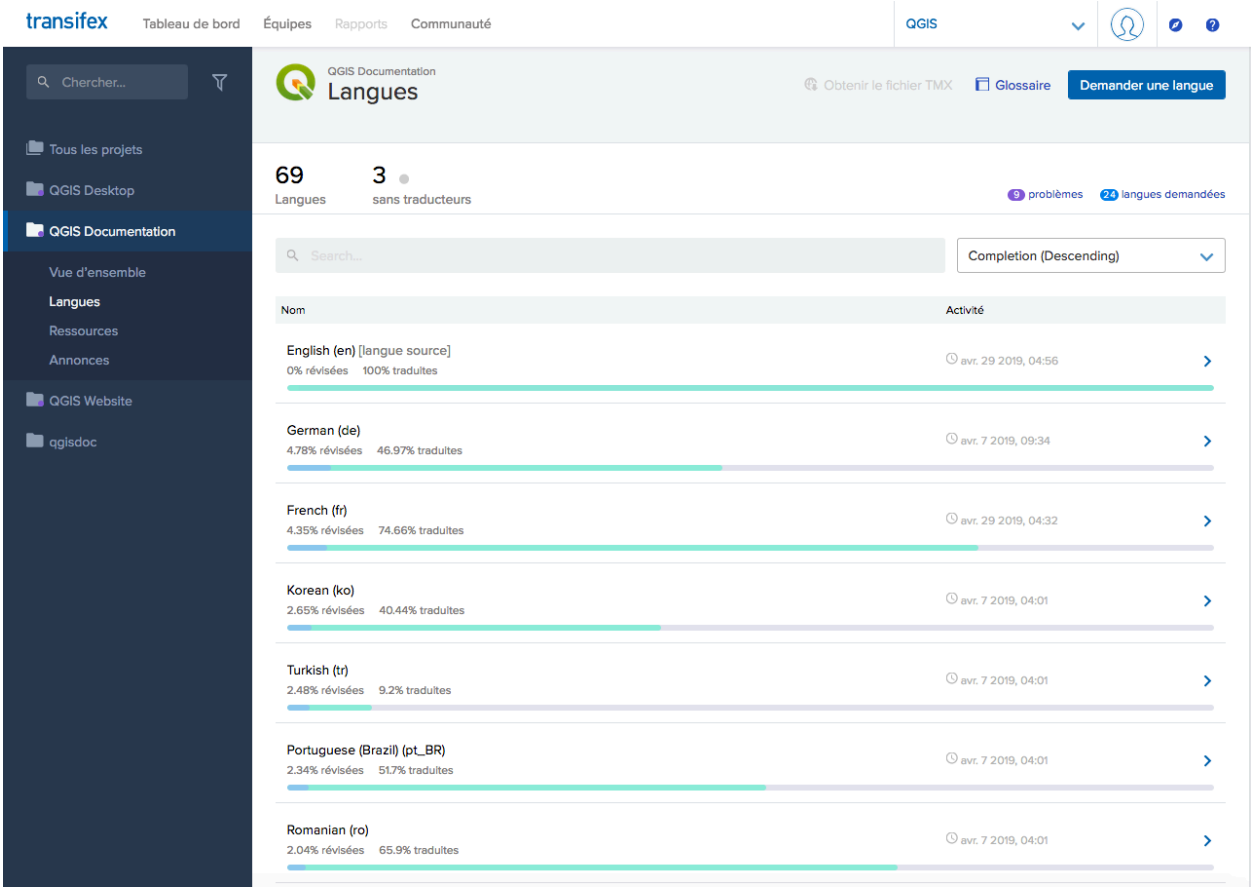

그림 4.1: Transifex 메뉴에서 번역할 언어 선택하기

3. 번역자의 언어 위에 마우스를 가져가서 클릭하면:
   - *View resources*: 번역 가능한 `.po` 파일이 번역률, 문자열 개수 및 기타 메타데이터와 함께 표시되거나
   - *Translate*: 번역 인터페이스가 번역할 수 있는 모든 `.po` 파일과 함께 열립니다.
4. 번역하고자 하는 파일 (이 경우 열지도 플러그인 파일인 `docs_user-manual_plugins_plugins-heatmap` 파일) 또는 번역이 완료되지 않은 아무 파일을 식별해서 클릭하십시오. 파일에 있는 문자열들을 불러와서 인터페이스를 통해 필터링하거나 번역하거나 번역을 제안하거나 등등의 작업을 할 수 있습니다.

   팁: 문서 또는 웹사이트의 경우, 페이지의 푸터 (footer) 에 있는 `Fix me` 링크를 클릭하면 Transifex 에 있는 해당 페이지의 번역 페이지로 직접 연결됩니다.

5. 번역자는 각 텍스트를 선택해서지침 을 따라 번역하기만 하면 됩니다.

Transifex 웹 편집기 사용에 대한 자세한 정보를 원한다면, https://help.transifex.com/en/articles/6318216-translating-with-the-web-editor 를 참조하십시오.

4.2.2 Qt Linguist 로 번역

Qt Linguist 를 사용하는 경우, 다음 작업이 필요합니다:

1. `.po` 또는 `.ts` 파일 (들) 을 직접 가져오십시오. Transifex 플랫폼 또는 (깃허브에 있는) 소스 저장소의 `locale/$language` 폴더에서 파일 (들) 을 다운로드할 수 있습니다.
2. 로컬 환경에서 번역을 진행하십시오.
3. 수정된 파일을 대응하는 소스로 (Transifex 또는 깃허브로) 업로드하십시오.

Transifex 에서 번역 가능한 파일을 다운로드/업로드할 수는 있지만, 이 과정을 권장하는 것은 아닙니다. Transifex 에는 버전 관리 시스템이 없기 때문에, 번역자가 업로드한 파일이 기존 파일을 대체할 뿐입니다. 번역자가 작업하는 동안 다른 사람들이 적용시켰을 수도 있는 수정 사항들을 덮어쓸 위험이 있습니다.

처음으로 Qt Linguist 에서 파일을 열었을 때 다음 대화창이 뜰 것입니다:

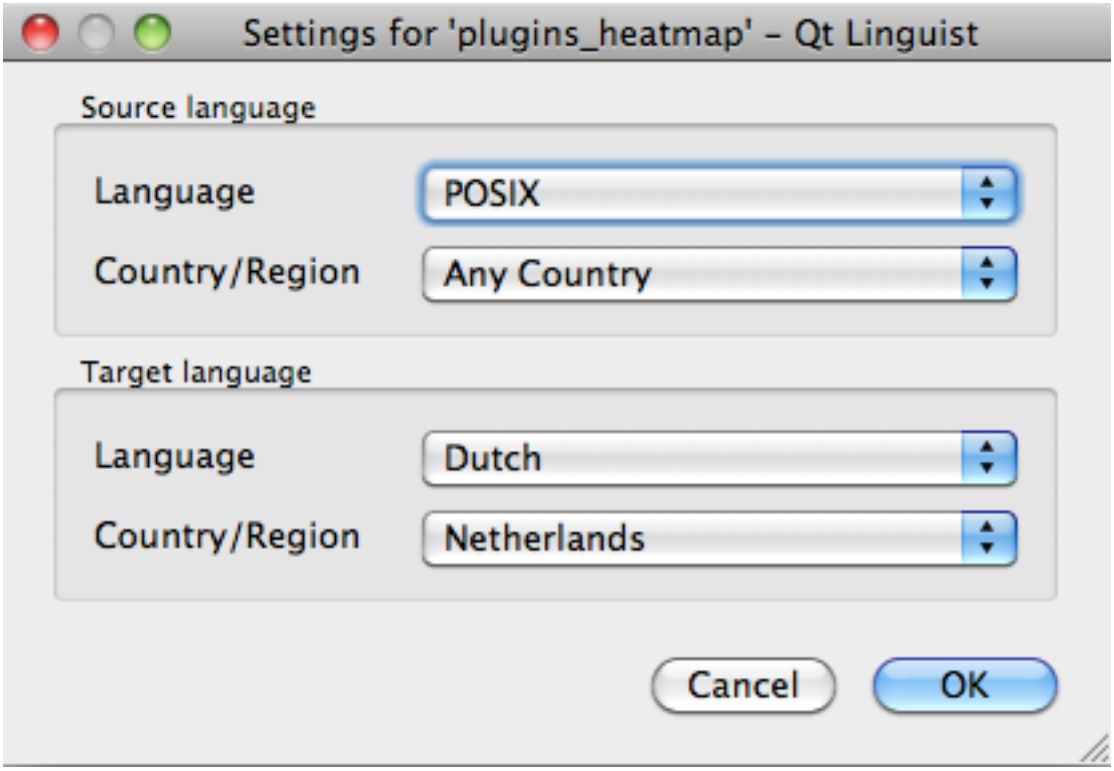

그림 4.2: Qt Linguist 메뉴에서 번역할 언어 선택하기

목표 언어를 정확하게 선택해야 합니다. 소스 언어는 언어 POSIX, 국가/지역 Any Country 를 그대로 놔두어도 상관없습니다.

OK 버튼을 클릭하면 Qt Linguist 에 문장이 채워져 번역을 시작할 수 있습니다. 그림 4.3 을 참조하십시오.

메뉴에 사용하기에 편리한 다음 버튼들이 있습니다.

- 번역 완료 다음 (Translation Done Next) 버튼은 가장 중요한 버튼입니다. 어떤 항목을 번역해야 할 경우, 텍스트 영역에 번역을 입력한 다음 이 버튼을 클릭하십시오. 번역해야 할 항목이 아니라면 번역 텍스트

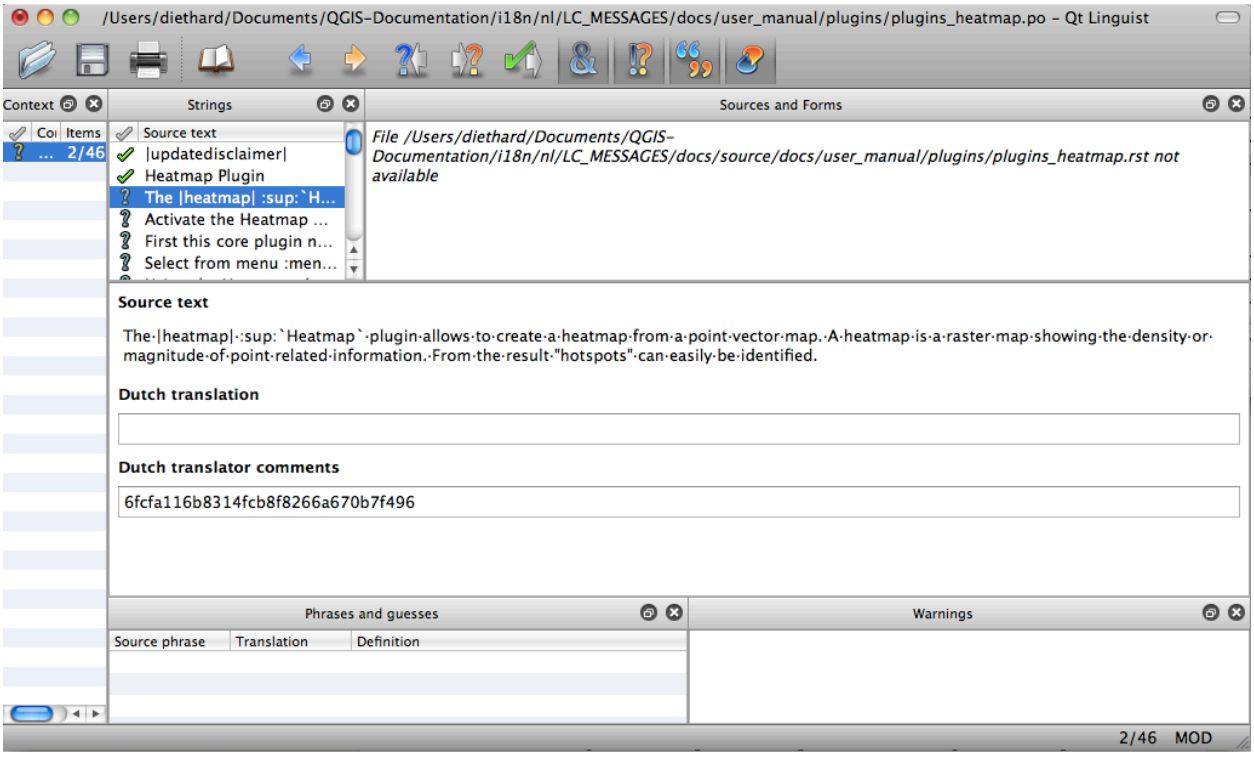

그림 4.3: Qt Linguist 메뉴를 사용해 번역하기

영역을 비워놓은 채 역시 이 버튼을 클릭하십시오. 항목 번역을 완료했고 다음 항목을 이어서 번역하겠다는 의미입니다.

- 이전 항목 (Goto Previous) 버튼은 이전에 번역한 항목으로 가는 데 쓰입니다.
- 다음 항목 (Goto Next) 버튼은 다음에 번역할 항목으로 가는 데 쓰입니다.
- 다음 할 일 (Next Todo) 버튼은 번역해야 할 다음 첫 번째 항목으로 넘어갑니다. 원본 문서가 변경돼 새로운/업데이트된 문장들만 번역해야 할 경우 유용합니다.
- 이전 할 일 (Previous Todo) 버튼은 이전 방향으로 검색해서 번역해야 할 첫 번째 항목으로 넘어갑니다.

Qt Languist 사용에 대한 자세한 내용은 https://doc.qt.io/qt-5/linguist-translators.html 을 보십시오.

> **경고:** 소스 저장소에서 번역할 내용을 다운로드하고 싶다면, `master` 분기에서 다운로드 하면 안 됩니다. 번역 작업의 경우 특정 버전에 대한 영어 문서가 완전히 업데이트되고 나면, 작업할 수 있는 번역 분기가 언제나 존재합니다. 예를 들어, QGIS 2.8 버전의 사용자 설명서를 번역하려면 manual_en_v2.8 분기를 이용해야만 합니다.

4.2.3 사용자 지침서 번역

이제 열지도 플러그인 설명서를 번역해봅시다!

대부분의 문장은 번역하는 데 복잡하지 않을 겁니다. 이 번역 과정에서는 어떤 부분 (reST 표현) 에 특별한 번역이 필요한지 짚어보겠습니다.

다음에 번역할 문단이 흥미로워 보이는군요.

```
The |heatmap| :sup:`Heatmap` plugin allows to create a heatmap from a
point vector map. A heatmap is a raster map showing the density or
magnitude of point related information. From the result "hotspots" can
easily be identified.
```

이 문단은 reST 표현을 두 개 담고 있습니다.

1. `|heatmap|` : `|` 사이에 있는 단어들은 별명으로, 이 단어들을 번역해서는 안 됩니다! 이 표현은 열지도 플러그인 아이콘으로 교체될 겁니다!
2. `:sup:`Heatmap` : :sup:` 표현은 중첩 선언으로 그 다음에 오는 텍스트를 좀 더 높게 표시합니다. 이 표현은 마우스 커서를 툴바 항목 위에 올렸을 때 나타나는 팝업 텍스트를 보여주기 위해 쓰입니다. QGIS 응용 프로그램에서 실제로 번역된 경우 이 텍스트도 번역해야 할 수도 있습니다만, 이번 네덜란드 어 예시에선 번역하지 않습니다!

이 문단의 다른 모든 평문은 번역할 수 있습니다!

다음 번역 항목은 `:ref:` 표현을 담고 있는데, 설명서 안 또다른 위치를 참조하는 데 흔히 쓰이곤 하죠! `:ref:` 표현 뒤에 나오는 텍스트는 유일 식별자이기 때문에 번역해선 안 됩니다!

```
First this core plugin needs to be activated using the Plugin Manager
(see Section :ref:`load_core_plugin`). After activation the heatmap icon
|heatmap| can be found in the Raster Toolbar.
```

이 경우 `load_core_plugin` 은 캡션을 보유한 reST 항목을 가리키는 유일 참조 식별자입니다. `:ref:` 표현은 헤더의 텍스트로 교체되어 하이퍼링크로 변할 것입니다. 이 참조가 참조하는 헤더를 번역할 경우, 이 헤더를 가리키는 모든 참조도 자동적으로 함께 번역될 겁니다.

다음 항목엔 QGIS 응용 프로그램의 메뉴에 실제로 표시되는 텍스트를 담고 있는 reST 태그 `:menuselection:` 이 있는데, 응용 프로그램이 번역된 경우 당연히 이 메뉴 텍스트도 번역해야겠죠.

```
Select from menu :menuselection:`View --> Toolbars --> Raster` to activate
the Raster Toolbar when it is not yet activated.
```

이 항목에서 실제로"View –>"를"Beeld –>"로 번역했는데, 네덜란드 어로 현지화된 QGIS 응용 프로그램에 쓰인 번역이기 때문입니다.

조금 더 가면 다음과 같은 까다로운 번역 항목을 보게 됩니다.

```
The |heatmap| :sup:`Heatmap` tool button starts the Dialog of the Heatmap
plugin (see :numref:`figure_heatmap_settings`).
```

이 항목은 도형 `figure_heatmap_settings_` 를 가리키는 참조를 담고 있는데, 위치 참조와 마찬가지로 이 참조를 변경해서는 안 됩니다!! reST 문서에 있는 참조 정의는 `.po` 파일에 포함되어 있지 않기 때문에 변경해서는 안 됩니다. 즉 도형을 가리키는 참조는 번역하면 안 된다는 뜻입니다. HTML 을 생성할 경우 `figure_heatmap_settings` 를 보게 될 겁니다. PDF 문서를 생성할 경우 `figure_heatmap_settings_` 가 도형 번호를 대체하게 됩니다.

reST 속성을 가진, 다음으로 번역할 항목은 다음과 같습니다.

```
**Input Point dialog**: Provides a selection of loaded point vector maps.
```

이 문장에서 별표를 제거하지 마십시오. 별표 사이에 담긴 텍스트는 볼드체로 표시될 것입니다. 텍스트 자체는 대화창에 포함된 텍스트일 경우가 많아, 응용 프로그램에서도 번역될 가능성이 큽니다.

다음은 `:guilabel:` reST 태그를 담고 있는 번역 항목입니다.

```
When the |checkbox| :guilabel:`Advanced` checkbox is checked it will
give access to additional advanced options.
```

`:guilabel:` 태그의 텍스트 `Advanced` 는 QGIS 응용 프로그램에서도 번역될 가능성이 크기 때문에 번역해야 할 필요가 있을 겁니다!

다음 번역 항목은 ``airports`` 를 포함하고 있습니다. 이 인용 부호는 텍스트에 또다른 텍스트 폰트 속성을 부여하는데 쓰입니다. 이 경우 ``airports`` 는 글자 그대로의 값으로 번역할 필요가 없습니다.

```
For the following example, we will use the ``airports`` vector point
layer from the QGIS sample dataset (see :ref:`label_sampledata`).
Another excellent QGIS tutorial on making heatmaps can be found on
`https://www.qgistutorials.com
<https://www.qgistutorials.com/en/docs/creating_heatmaps.html>`_.
```

또 이 항목은 URL 과 외부 표현 방식을 가진 하이퍼링크도 포함합니다. URL 은 당연히 그대로 두어야 하지만, 독자가 읽을 수 있는 외부 표현 텍스트인 `https://www.qgistutorials.com` 은 변경할 수 있습니다. 하이퍼링크 마지막에 붙는 언더바를 절대 제거하지 마십시오. 하이퍼링크의 필수적인 부분입니다!!

4.2.4 번역 규칙 요약

1. `|bronze|`, `|checkbox|`, `|labels|`, `|selectString|`, `|addLayer|` 와 같이 `|` 문자 두 개 사이의 텍스트를 변경하지 마십시오. 이미지를 대체하기 위해 쓰이는 특수 태그입니다.
2. `:ref:`, `:file:`, `:numref:` 같은 롤 (role) 로 시작하는 참조가 제목을 포함하지 않는 이상, 변경하지 마십시오. 제목을 포함한다면, 제목을 번역할 수는 있지만 링크 (즉: < 와 > 사이의 텍스트) 를 변경하지는 마십시오.

   팁: 참조에 대한 제목이 제공되는 경우, Transifex 는 영어 소스 텍스트에서 링크 부분 대신 숫자를 표시할 수도 있습니다. 소스 텍스트에 있는 숫자를 클릭하면 번역중인 제목 옆에 참조 링크를 추가합니다.

3. `figure_labels_1_` 처럼 언더바로 끝나는 참조를 변경하지 마십시오.
4. 하이퍼링크 내부에 있는 URL 을 변경하지 마십시오. 그러나 외부 설명 텍스트는 변경해도 됩니다. 하이퍼링크 마지막에 있는 언더바를 공백을 추가하지 말고 (>`_) 내버려두십시오.
5. `:index:`, `:sup:`, `:guilabel:` 그리고 `:menuselection:` 태그 뒤에 붙는 인용 부호 안의 텍스트를 번역하십시오. QGIS 응용 프로그램에서 해당 텍스트를 번역했는지, 했다면 어떻게 번역했는지 확인하십시오. 태그 자체는 변경하지 마십시오.
6. 이중 별표 그리고 이중 인용 부호 사이의 텍스트는 값 또는 파일명을 나타내는 경우가 많으며, 때로는 번역해야 하고 때로는 그대로 두어야 합니다.
7. `` `, ``, *, **, :: `` 과 같은 특수 문자를 소스 텍스트와 정확히 동일한 개수로 작성하도록 유의해야 합니다. 이런 특수 문자는 제공하는 정보의 스타일에 기여합니다.
8. 특수 문자 또는 태그에 담긴 텍스트는 공백으로 시작해서도 끝나서도 안 됩니다.
9. 번역된 문자열을 새 단락으로 끝내지 마십시오. 그러지 않으면 HTML 생성 시 텍스트가 번역되지 않을 것입니다.

앞에서 정리한 규칙을 준수한다면 보기 좋은 번역 문서를 얻게 될 겁니다!
질문이 있다면, QGIS 커뮤니티 팀 또는 QGIS 번역 팀 으로 연락해주십시오.

CHAPTER 5

별명

- 사용법
- 일반 별명
  - 플랫폼 아이콘
  - 메뉴 항목
- 툴바 버튼 아이콘
  - 레이어 및 미리보기 관리
  - 프로젝트
  - 편집
  - 식별 결과
  - 디지타이즈 작업 및 고급 디지타이즈 작업
  - 메시 *(mesh)*
  - 맵 조작 및 속성
  - 선택 및 표현식
  - 라벨 및 도표
  - 지도 장식
  - 도움말
  - 색상
- 기타 기본 아이콘
- 속성 테이블
- 투영 및 지리참조자

- 인쇄 조판기
- 레이어 속성
- 플러그인
  - 공간 처리
  - 여러 핵심 플러그인
  - *GRASS* 통합

5.1 사용법

QGIS 사용자 설명서에서 아이콘을 쉽게 사용할 수 있도록, QGIS 문서 저장소 의 `/source/substitutions.txt` 파일에 각 아이콘에 대한 별명을 정의해놓았습니다. 다음 단락부터 이 가운데 일부 별명을 정리했습니다. 즉, 문서에 QGIS 응용 프로그램의 아이콘을 삽입하고자 할 때 이미 이용할 수 있는, 이용해야 하는 별명이 있을 가능성이 큽니다.

별명이 존재하지 않을 경우:

1. `/static/common` 폴더에 이용할 수 있는 아이콘이 있는지 문서 저장소를 확인해보십시오. 이미지가 없다면, QGIS 저장소 에서 아이콘 이미지 파일을 찾아 (기본 테마 폴더에 있을 가능성이 큽니다) 복사해서 `/static/common` 폴더에 (`.png` 포맷으로) 붙여넣어야 합니다. 편의 및 업데이트를 고려하면 가능한 한 파일명을 동일하게 유지하는 편이 좋습니다.
2. 다음 예시를 따라 `/substitutions.txt` 파일에서 별명을 가리키는 참조를 생성하십시오. 별명 텍스트는 파일명으로부터 파생되어야 하며, 카멜 표기법 (camelCase, 여러 단어를 붙여서 한 단어로 표기할 때, 첫 단어의 첫 문자는 소문자로, 다음 단어부터 첫 문자를 대문자로 표기하는 표기법) 을 따라야 합니다:

```
.. |dataSourceManager| image:: /static/common/mActionDataSourceManager.png
   :width: 1.5em
.. |splitLayer| image:: /static/common/split_layer.png
   :width: 1.5em
```

3. 문서의 대상 부분 (들) 을 새로운 별명으로 업데이트하십시오.
4. (선택적이지만 권장 사항) 다음 목록에 별명을 추가하십시오.
5. 별명이 쓰이는 파일 (들) 의 마지막에 있는 별명 목록에 새로운 별명 참조 정보를 추가하거나, 편리한 `scripts/find_set_subst.py` 스크립트를 실행하십시오.

```
# from the repository main folder
python3 scripts/find_set_subst.py
```

5.2 일반 별명

다음은 문서 작성 시 사용할 수 있는 아이콘과 그 별명의 예시입니다. 사용자 설명서의 여러 군데에서 찾아볼 수 있고, 사용할 수 있습니다.

5.2.1 플랫폼 아이콘

아이콘	별명	아이콘	별명
	|logo|		
	|kde|		|nix|
	|osx|		|win|

5.2.2 메뉴 항목

아이콘	별명	아이콘	별명
	|checkbox|		|unchecked|
	|radioBut-tonOn|		|radioBut-tonOff|
1,00	|selectNum-ber|		|selectString|
	|selectColor|		|selectColor-Ramp|
	|tab|	°	|degrees|
Display name lakes.shp	|inputText|		|slider|
☰	|hamburger-Menu|		

5.3 툴바 버튼 아이콘

5.3.1 레이어 및 미리보기 관리

아이콘	별명	아이콘	별명
	|dataSourceManager|		
	|addOgrLayer|		
	|addRasterLayer|		|addMssqlLayer|
	|addDelimitedTextLayer|		|addSpatiaLiteLayer|
	|addPostgisLayer|		|addOracleLayer|
	|addAfsLayer|		|addMeshLayer|
	|addVectorTileLayer|		|addXyzLayer|
	|addVirtualLayer|		|addWmsLayer|

다음 페이지에 계속

표 5.1 – 이전 페이지에서 계속

아이콘	별명	아이콘	별명
	`|addWcsLayer|`		`|addWfsLayer|`
	`|addPointCloudLayer|`		`|addGpsLayer|`
	`|addTiledSceneLayer|`		`|addHanaLayer|`
	`|newVectorLayer|`		`|newSpatiaLiteLayer|`
	`|newGeoPackageLayer|`		`|createMemory|`
	`|newVirtualLayer|`		`|newMeshLayer|`
	`|newGpx|`		
	`|dbManager|`		`|gdal|`
	`|geoPackage|`		`|spatialite|`
	`|virtualLayer|`		`|wms|`
	`|wcs|`		`|wfs|`
	`|pointCloudLayer|`		`|gps|`
	`|tiledSceneLayer|`		`|hana|`
	`|dbSchema|`		
	`|inOverview|`		`|addAllToOverview|`
	`|removeAllFromOverview|`		`|removeLayer|`
	`|showAllLayers|`		`|hideAllLayers|`
	`|showPresets|`		`|showSelectedLayers|`
	`|hideSelectedLayers|`		`|hideDeselectedLayers|`
	`|toggleAllLayers|`		`|toggleSelectedLayers|`
	`|addLayer|`		
	`|indicatorTemporal|`		`|indicatorNonRemovable|`
	`|indicatorEmbedded|`		`|indicatorFilter|`
	`|indicatorMemory|`		`|indicatorNoCRS|`
	`|indicatorBadLayer|`		`|favourites|`
	`|indicatorLayerError|`		`|indicatorNotes|`
	`|indicatorLowAccuracy|`		`|indicatorOffline|`

5.3.2 프로젝트

아이콘	별명	아이콘	별명
	|fileNew|		|fileOpen|
	|fileSave|		|fileSaveAs|
	|fileExit|		|user|

5.3.3 편집

아이콘	별명	아이콘	별명
	|undo|		|redo|
	|editCopy|		|editPaste|
	|editCut|		|saveEdits|
	|editableEdits|		
	|circle2Points|		|circle2TangentsPoint|
	|circle3Points|		|circle3Tangents|
	|circleCenterPoint|		|ellipseCenter2Points|
	|ellipseCenterPoint|		|ellipseExtent|
	|ellipseFoci|		|rectangle3PointsDistance|
	|rectangle3PointsProjected|		|rectangleCenter|
	|rectangleExtent|		|regularPolygon2Points|
	|regularPolygonCenterCorner|		|regularPolygonCenterPoint|

5.3.4 식별 결과

아이콘	별명	아이콘	별명
	|expandTree|		|collapseTree|
	|expandNewTree|		|formView|
	|deselectAll|		|editCopy|
	|filePrint|		
	|identifyByRectangle|		|identifyByPolygon|
	|identifyByFreehand|		|identifyByRadius|

5.3.5 디지타이즈 작업 및 고급 디지타이즈 작업

아이콘	별명	아이콘	별명
	|cad|		|cadConstruction|
	|cadParallel|		|cadPerpendicular|
	|floater|		
	|toggleEditing|		|allEdits|
	|tracing|		|snapping|
	|snappingVertex|		|snappingSegment|
	|snappingArea|		|snappingCentroid|
	|snappingMiddle|		|snappingEndpoint|
	|capturePoint|		|capturePolygon|
	|captureLine|		|captureCurveFromFeature|
	|deleteSelectedFeatures|		
	|circularStringCurvePoint|		|circularStringRadius|
	|vertexTool|		|vertexToolActiveLayer|
	|digitizeWithSegment|		|digitizeShape|
	|streamingDigitize|		|digitizeWithCurve|
	|moveFeature|		|moveFeatureCopy|
	|moveFeatureLine|		|moveFeatureCopyLine|
	|moveFeaturePoint|		|moveFeatureCopyPoint|
	|rotateFeature|		|rotatePointSymbols|
	|scaleFeature|		
	|offsetCurve|		|offsetPointSymbols|
	|simplify|		|reshape|
	|addRing|		|addPart|
	|fillRing|		
	|deleteRing|		|deletePart|
	|mergeFeatures|		|mergeFeatureAttributes|
	|splitFeatures|		|splitParts|
	|reverseLine|		
	|allowIntersections|		|avoidIntersectionsCurrentLayer|
	|avoidIntersectionsLayers|		|snappingSelf|

5.3.6 메시 (mesh)

아이콘	별명	아이콘	별명
	|meshDigitizing|		|meshReindex|
	|meshSelectExpression|		|meshSelectPolygon|
	|meshTransformByExpression|		|meshEditForceByVectorLines|
	|vertexCoordinates|		

5.3.7 맵 조작 및 속성

아이콘	별명	아이콘	별명
	|pan|		|panToSelected|
	|zoomIn|		|zoomOut|
	|zoomActual|		|zoomFullExtent|
	|zoomToLayer|		|zoomToSelected|
	|zoomLast|		|zoomNext|
	|zoomInXAxis|		|refresh|
	|identify|		|mapTips|
	|showBookmarks|		|newBookmark|
	|measure|		|measureArea|
	|measureBearing|		|measureAngle|
	|newMap|		|new3DMap|
	|tiltUp|		|tiltDown|
	|3dNavigation|		|play|
	|temporal		|temporalNavigationOff|
	|temporalNavigationFixedRange|		|temporalNavigationAnimated|
	|newElevationProfile|		

5.3.8 선택 및 표현식

아이콘	별명	아이콘	별명
	`|selectRectangle|`		`|selectPolygon|`
	`|selectFreehand|`		`|selectRadius|`
	`|selectAll|`		`|deselectAll|`
	`|invertSelection|`		`|expressionSelect|`
	`|deselectActiveLayer|`		
	`|selectDistance|`		`|selectLocation|`
	`|selectAllTree|`		`|select|`
	`|selectAdd|`		`|selectRemove|`
	`|formSelect|`		`|dataDefine|`
	`|expression|`		`|dataDefineOn|`
	`|dataDefineExpressionOn|`		`|dataDefineError|`
	`|dataDefineExpressionError|`		
	`|addExpression|`		
	`|expressionFilter|`		`|filterMap|`

5.3.9 라벨 및 도표

아이콘	별명	아이콘	별명
	|labelingSingle|		|labelingNone|
	|labelingRuleBased|		|labelingObstacle|
	|piechart|		|diagramNone|
	|text|		|histogram|
	|stackedBar|		
	|createAnnotationLayer|		|annotationLayer|
	|textAnnotation|		|svgAnnotation|
	|formAnnotation|		|htmlAnnotation|
	|actionText|		|textAlongLine|
	|labelbackground|		|labelbuffer|
	|labelformatting|		|labelplacement|
	|labelshadow|		|render|
	|labelcallout|		
	|labelAnchorCenter|		|labelAnchorCustom|
	|labelAnchorEnd|		|labelAnchorStart|
	|pinLabels|		|showHideLabels|
	|moveLabel|		|rotateLabel|
	|showPinnedLabels|		|showUnplacedLabel|
	|changeLabelProperties|		|autoPlacementSettings|

5.3.10 지도 장식

아이콘	별명	아이콘	별명
	|copyrightLabel|		|addGrid|
	|titleLabel|		|northArrow|
	|scaleBar|		|addMap|
	|addImage|		

5.3.11 도움말

아이콘	별명	아이콘	별명
	`|helpContents|`		`|qgisHomePage|`
	`|success|`		
	`|helpSponsors|`		`|contextHelp|`

5.3.12 색상

아이콘	별명	아이콘	별명
	`|colorBox|`		`|colorPicker|`
	`|colorSwatches|`		`|colorWheel|`

5.4 기타 기본 아이콘

아이콘	별명	아이콘	별명
	|arrowLeft|		|arrowRight|
	|arrowDown|		|arrowUp|
	|symbologyAdd|		|symbologyRemove|
	|projectProperties|		|options|
	|interfaceCustomization|		|keyboardShortcuts|
	|copyrightLabel|		|northArrow|
	|scaleBar|		|tracking|
	|gpsTrackBarChart|		
	|gpsConnect|		|gpsDisconnect|
	|gpsDestinationLayer|		|addTrackPoint|
	|recenter|		|reset|
	|folder|		|extents|
	|settings|		|start|
	|properties|		|deleteSelected|
	|browserExpand|		|browserCollapse|
	|codeEditor|		|add|
	|relations|		|layoutItem3DMap|
	|stopwatch|		|sensor|
	|clearItem|		

5.5 속성 테이블

아이콘	별명	아이콘	별명
	|openTable|		|openTableSelected|
	|openTableVisible|		|openTableEdited|
	|selectedToTop|		
	|selectAll|		|invertSelection|
	|panToSelected|		|zoomToSelected|
	|copySelected|		|editPaste|
	|expressionSelect|		|deleteSelectedFea-tures|
	|newAttribute|		|deleteAttribute|
	|editTable|		
	|newTableRow|		|calculateField|
	|refresh|		|formView|
	|conditionalFormatting|		|multiEdit|
	|dock|		|actionRun|
	|duplicateFeature|		|zoomTo|
	|panTo|		|highlightFeature|
	|handleStoreFilterExpressionChecked|		
	|handleStoreFilterExpressio-nUnchecked|		

5.6 투영 및 지리참조자

아이콘	별명	아이콘	별명
	|geographic|		|crs|
	|customProjection|		|setProjection|
	|projectionDisabled|		|projectionEnabled|
	|transformation|		|gdalScript|
	|georefRun|		|pencil|
	|linkQGisToGeoref|		|linkGeorefToQGis|
	|fullHistogramStretch|		

5.7 인쇄 조판기

아이콘	별명	아이콘	별명
	|newLayout|		|layoutManager|
	|duplicateLayout|		
	|newReport|		|newPage|
	|atlasSettings|		|atlas|
	|filePrint|		|saveMapAsImage|
	|saveAsSVG|		|saveAsPDF|
	|addBasicShape|		|addBasicCircle|
	|addBasicTriangle|		|addBasicRectangle|
	|addNodesShape|		|editNodesShape|
	|addPolygon|		|addPolyline|
	|addArrow|		|northArrow|
	|add3DMap|		|addMap|
	|elevationProfile|		|copyProfileSettings|
	|addLegend|		|addHtml|
	|addManualTable|		|addTable|
	|addImage|		|addMarker|
	|label|		|scaleBar|
	|select|		|moveItemContent|
	|setToCanvasScale|		|setToCanvasExtent|
	|viewScaleInCanvas|		|viewExtentInCanvas|
	|raiseItems|		|lowerItems|
	|moveItemsToTop|		|moveItemsToBottom|
	|alignLeft|		|alignRight|
	|alignHCenter|		|alignVCenter|
	|alignTop|		|alignBottom|
	|distributeLeft|		|distributeRight|
	|distributeTop|		|distributeBottom|
	|distributeHCenter|		|distributeVCenter|
	|distributeHSpace|		|distributeVSpace|
	|resizeShortest|		|resizeTallest|

다음 페이지에 계속

표 5.3 – 이전 페이지에서 계속

아이콘	별명	아이콘	별명
	|resizeNarrowest|		|resizeWidest|
	|resizeSquare|		|groupItems|
	|lockItems|		|unlockAll|
	|locked|		|unlocked|
	|lockRepeating|		|lockedGray|

5.8 레이어 속성

아이콘	별명	아이콘	별명
	|symbology|		|labelingSingle|
	|sourceFields|		|general|
	|metadata|		|action|
	|display|		|rendering|
	|join|		|diagram|
	|labelmask|		|temporal|
	|legend|		|dependencies|
	|3d|		|system|
	|elevationscale|		|layerTree|
	|editMetadata|		|overlay|
	|digitizing|		|auxiliaryStorage|
	|history|		|stylePreset|
	|search|		|pyramids|
	|transparency|		|rasterHistogram|
	|singleSymbol|		|nullSymbol|
	|graduatedSymbol|		|categorizedSymbol|
	|25dSymbol|		|ruleBasedSymbol|
	|invertedSymbol|		|heatmapSymbol|
	|pointDisplacementSymbol|		|pointClusterSymbol|
	|mergedFeatures|		
	|meshcontours|		|meshcontoursoff|
	|meshvectors|		|meshvectorsoff|

다음 페이지에 계속

표 5.4 – 이전 페이지에서 계속

아이콘	별명	아이콘	별명
	|meshframe|		|meshaveraging|
	|singleColor|		|paletted|
	|singlebandPseudocolor|		|multibandColor|
	|pointCloudExtent|		
	|sum|		|sort|
	|paintEffects|		|mapIdentification|
	|styleManager|		|iconView|
	|joinNotEditable|		|joinedLayerNotEditable|
	|joinHasNotUpsertOnEdit|		|filterTableFields|
	|symbologyEdit|		
	|sharingImport|		|sharingExport|

5.9 플러그인

5.9.1 공간 처리

아이콘	별명	아이콘	별명
	|processingAlgorithm|		|processingModel|
	|processingHistory|		|processingResult|
	|menu|		
	|processSelected|		|editHelpContent|
	|saveAsPython|		|modelOutput|
	|qgsProjectFile|		|addToProject|
	|fieldInteger|		
	|meanCoordinates|		|extractLayerExtent|
	|selectRandom|		|vectorGrid|
	|convexHull|		|buffer|
	|intersect|		|union|
	|symmetricalDifference|		|clip|
	|difference|		|dissolve|
	|checkGeometry|		|exportGeometry|
	|delaunay|		|centroids|

다음 페이지에 계속

표 5.5 – 이전 페이지에서 계속

아이콘	별명	아이콘	별명
	|polygonToLine|		|extractVertices|
	|lineToPolygon|		|nearestNeighbour|
	|splitLayer|		|heatmap|
	|showRasterCalculator|		|showMeshCalculator|
	|regularPoints|		|addGeometryAttributes|
	|basicStatistics|		|uniqueValues|
	|collect|		|simplify_2|
	|createGrid|		|distanceMatrix|
	|lineIntersections|		|mergeLayers|
	|sumPoints|		|sumLengthLines|
	|randomPointsInPolygons|		|randomPointsWithinPolygon|
	|randomPointsOnLines|		|randomPointsWithinExtent|
	|multiToSingle|		
	|grid|		|tiles|
	|merge|		|rasterClip|
	|contour|		|proximity|
	|polygonize|		|rasterize|
	|sieve|		|nearblack|
	|projectionAdd|		|projectionExport
	|8To24Bits|		|24To8Bits|
	|rasterInfo|		|rasterOverview|
	|vrt|		|voronoi|
	|translate|		|warp|
	|iterate|		|terminal|

5.9.2 여러 핵심 플러그인

기본 설치 시 제공되는 표준 플러그인이지만, 첫 설치 시 자동으로 로드되지 않는 플러그인입니다.

아이콘	별명	아이콘	별명
	`|showPluginManager|`		`|installPluginFromZip|`
	`|pythonFile|`		`|runConsole|`
	`|showEditorConsole|`		`|clearConsole|`
	`|offlineEditingCopy|`		`|offlineEditingSync|`
	`|plugin|`		`|metasearch|`
	`|geometryChecker|`		`|topologyChecker|`
	`|fromSelectedFeature|`		`|sqlQueryBuilder|`

5.9.3 GRASS 통합

아이콘	별명	아이콘	별명
	`|grassLogo|`		`|grassRegion|`
	`|grassTools|`		`|grassNewMapset|`
	`|grassOpenMapset|`		`|grassCloseMapset|`