

TeamLab/lab_for_gachon_cs50

Lab #3 - Vim Editor 연습하기

Copyright 2015 © document created by TeamLab.Gachon@gmail.com

Introduction

이제 수업에 있어 가장 큰 관문인 Vim Editor를 연습하는 Lab을 시작한다. Vim Editor는 호불호가 많이 갈리는 Editor이고 배우기가 쉽지 않기 때문에 입문자들이 기피하는 Editor이다. 그러나 프로그래밍을 배우면 배울수록 GUI 환경보다는 CLI(Command Line Interface)에서 보내는 시간이 늘어간다. 이를 위해서 최소한의 Vim Editor 활용법에 익숙해 지는 것이 장기적으로 중요하다. 물론 여러분들은 이미 앞에 두 Lab Assignment을 실행하면서 어느정도 Vim Editor에 익숙해졌을것으로 기대된다. 그러나 본 수업의 초반에 주로 활용되는 Vim Editor를 능숙하게 다룰려면 여러가지 Vim 명령어를 빠르게 사용하여야 한다. 특히 초반에는 명령어가 익숙하지 않아 명령어 자체를 찾기 위해 보내는 시간이 많을 것이다. 최대한 그 시간을 절약하여 효율적인 코딩을 하기위해 본 Lab에서는 Vim Editor의 기본적인 기능을 연습한다.

숙제 template 파일 다운로드

먼저 숙제 template 파일을 cs50 서버로부터 다운로드 받는다. 이미 해보았기 때문에 어렵지 않을 것이다. 명령은 아래와 같다.

```
python3.4 submit_assignment.py -get vimeditor
```

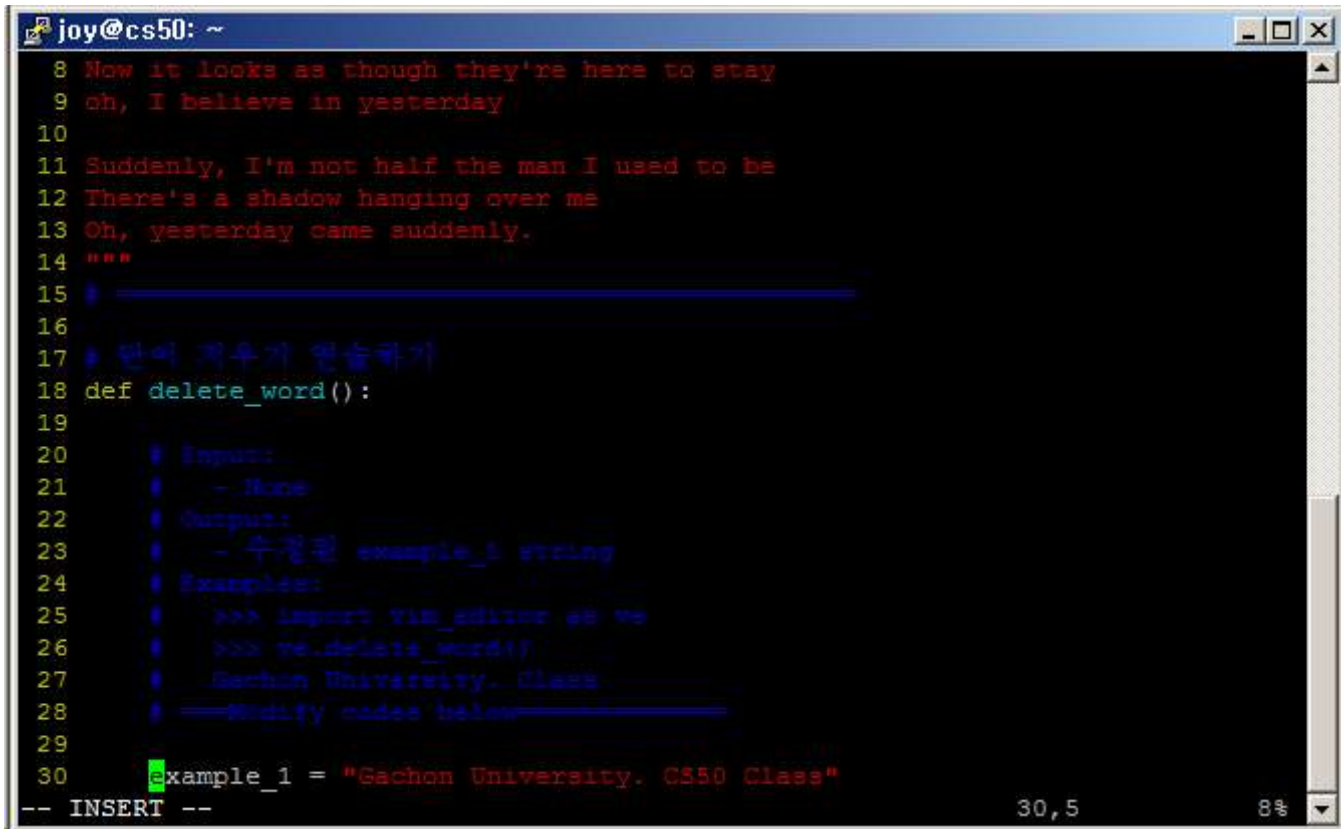
파일을 간단히 보면 최상단의 "절대 수정하지 말 것" 이라고 적힌 3개의 변수와 4개의 수정해야 할 함수로 이루어져 있다. 각각의 함수를 수정하기 위해서는 상단의 3개의 변수를 각 함수로 copy&paste 한 후 lab assignment를 진행하여야 한다.

delete_word() 함수 수정하기

첫 번째 수정해야 할 함수는 `delete_word()` 함수 이다. 이 함수는 `example_1` 이라는 변수를 `delete_word()`로 먼저 이동 한 후, `CS50` 이라고 적힌 글자를 삭제해 주면 된다. 이를 위해 아래의 순서로 파일을 수정해 보자

1. 커서를 활용하여 4번째 줄로 이동하고 y,y 를 눌러준다. 참고로 키보드 표시에서 `[`는 순서대로 입력을 의미하고, `+`는 동시에 입력을 의미한다. `yy`는 현재의 줄을 복사하라는 의미이다.
2. esc를 누르고 `:29`를 누른후 Enter를 치면 29번째 줄로 이동하게 된다.
3. 이동 후, p를 누르면, 복사한 `example_1="~~~"` 가 30번째 줄에 출력될 것이다. p는 paste 즉 붙여넣기를 의미한다.
4. 현재 붙여넣기는 되어 있지만 Indentation (들여쓰기)가 이루어지지 않았다. 다시 한번 기억해야 할 것은 파이썬은 들여쓰기를 잘못할 경우 제대로 동작하지 않는다는 것이다. 30번째 줄로 이동하여 esc를 누른 후, `shift+i` 또는 `i`를 입력하여 수정 가능 모드로 변경한다.
5. 마지막으로 커서가 30번째줄 맨 앞쪽에 위치한 상태에서 Tab 키를 눌러 `example_1`변수의 들여쓰기를 한 단계 이동해 주자.

6. 완성이 되었다면 아래와 유사한 화면을 볼 수 있을 것이다.



```

joy@cs50: ~
8 Now it looks as though they're here to stay
9 oh, I believe in yesterday
10
11 Suddenly, I'm not half the man I used to be
12 There's a shadow hanging over me
13 Oh, yesterday came suddenly.
14 ***
15
16
17
18 def delete_word():
19
20     # Input:
21     # - None
22     # Output:
23     # - 수정된 example_1 string
24     # Example:
25     # >>> import vim_editor as ve
26     # >>> ve.delete_word()
27     # Gachon University. Class
28     # ---Empty codes below---
29
30 example_1 = "Gachon University. CS50 Class"
-- INSERT --
30,5 8%
```

첫 번째 미션을 성공했다. 남은 미션은 간단한 Tip만 알려주겠다. 최선을 다해보자.

change_word() 함수 수정하기

두 번째 함수는 `example_2` 변수를 `change_word()` 함수에 복사 한 후 `ACE` 라는 단어를 `Gachon` 으로 변경 하는 것이다. 다양한 방법들이 가능하겠지만 본 lab assignment에서는 한 단어를 지우는 `d`, `w` 와 새로 글을 삽입할 때 쓰는 `i` 를 이용해서 `ACE`라는 단어를 지우고, `Gachon` 이라는 단어로 변경하기 바란다.

delete_five_lines() 함수 수정하기

세 번째 함수는 `example_3` 변수를 `delete_five_lines()` 함수에 복사 한 후 5문장을 지우는 것이다. 5문장이라고는 하지만 이것 저것 지워야 할 게 많다. 정확히는 마지막 문장만 남겨야 숙제로 인정된다. 숙제를 위해 알아야 할 것은 한 번에 여러 줄을 복사하고 붙여 넣기 하는 것이다. 이를 위해서 첫 번째로 `y,10,↓` 라고 누르면 현재 줄을 기준으로 총 11줄이 복사 된다. 본 예제는 6번째 줄로 이동하여 9줄을 복사하고, 63번째줄로 이동하여 붙여넣기를 해보자. `example_3` 변수의 들여쓰기를 맞춰주자. 이 때 따옴표 세개의 경우 긴 문장의 텍스트 데이터를 저장할 때 쓰는 문자로 첫 줄을 제외한 나머지 줄은 아래처럼 들여쓰기를 하지 않아도 된다.

```
example_3 = """
```

```
Yesterday, all my troubles seemed so far away
Now it looks as though they're here to stay
oh, I believe in yesterday
```

```
Suddenly, I'm not half the man I used to be
There's a shadow hanging over me
```

```
Oh, yesterday came suddenly.
'''
```

붙여 넣기 후 5문장을 삭제하자. `There's a~`으로 시작하는 줄로 이동하여 `d,5,↑` 라고 누르면 현재 줄부터 위로 6줄이 지워진다. `y`와 마찬가지로 `d` 입력후 `[숫자]`, `[화살표]`는 삭제 또는 복사할 문자수 또는 줄 수를 의미한다.

change_word_2() 함수 수정하기

마지막이다 `delete_five_lines` 함수와 마찬가지로 `example_3` 변수를 `change_word_2()` 함수로 이동한다. 이번에 할 것은 함수내에 있는 단어들을 변경하는 것이다. 워드에서는 `ctrl+h`를 누르는 것과 같다고 보면 된다. 이를 위한 명령어는 먼저 `esc`를 누르고 `:10,100s/abc/new/gi` 같이 입력하면 된다. 각각의 의미는 10과 100은 변경을 할 구역 즉 10번째줄 부터 100번째줄 까지라는 의미이며, `abc`는 변경될 기존 단어, `new`는 변경할 새 단어이다. `g`와 `i`는 option인데 `g`는 문장 전체를 다 찾아서 변경하는 것이고 `i`는 대소문자 구분없이 변경하는 것이다. `g`가 없으면 한 줄에 `abc abc abc`가 있을 경우 첫번째 `abc`만 변경한다. 본 함수에서는 `yesterday`를 `Today` 바꾸는 것이 숙제이다. 위에서 얘기한 명령어를 상황에 맞게 수정하여 해보자. `example_3` 변수를 보면 알겠지만, `yesterday`가 대소문자 구분하여 총 3번 나온다. 3번 나온 `yesterday` 단어를 모두 `Today`로 바꿔주기 바란다. 한가지 유의할 점은 `Today`는 대소문자를 구분하여 채점을 한다. 고로 `Today`는 `today`가 아닌 반드시 첫 자를 대문자로 표시해주자.

제출하기

길고 긴 하루였다. 다음과 같은 명령어로 먼저 본인의 파일이 제대로 작동하는 지 보자.

```
python3.4 vim_editor.py
```

그렇지 않다면 다시 글을 꼼꼼히 읽어보거나 질문을 통해 본인의 파일을 점검해 보기 바란다. 아래와 같은 명령어로 숙제를 제출하면 결과가 나올 것이다.

```
python3.4 submit_assignment.py -submit vim_editor.py
# 아이디와 패스워드 입력
```

Function Name	Passed?	Feedback
change_word_2	Not Yet	Check Your Logic
check_initial_values	PASS	Good Job
delete_word	Not Yet	Check Your Logic
delete_five_lines	PASS	Good Job
change_word	Not Yet	Check Your Logic

위에 숙제는 성공하지 못한 숙제일 경우이다. 성공했다면, 모든 것이 PASS로 나올 거고 Feedback도 Good Job으로 표시 될 것이다. 할 수 있다 믿고 해보자. 세상에 쉬운 건 하나도 없다.

Next Work

첫 번째 강의를 무리없이 따라온 것을 축하한다. 사실상 우리는 아직 제대로 된 프로그램을 만들어 본적이 없지만, 그래도 조금은 의미 있는 결과를 만들어 봤다고 생각한다. 아직 CLI가 익숙하지 않다면 집에가서 올려둔 자료들을 다시보면서 점검해 보길 바란다. 많이 따라 만들어보는 것 보다 좋은 연습은 없다. 다음 주에도 다시 볼 수 있길 간절히 원한다.

Human knowledge belongs to the world - from movie 'Password' -

Footnotes