Python

3강 – 함수, 파일 읽고 쓰기, GIT

함수란

- 반복적인 내용을 한 뭉치로 묶어서 사용하는 부분 + 어떤 입력값을 주었을 때 어떤 결과값을 돌려준다.
- def 함수명(입력 인수):
  - <수행할 문장>
  - <수행할 문장>
  - return (반환인수)
- Ex) def sum(a,b):
  - return a + b
- 다음과 같이 만든 함수 사용 법
- a=3, b=4, c=sum(a,b)
- print (c)
- 7

```
# 1. 입력값이 없는 함수
def say():
    return 'Q1 Hi'

a=say()
print a

# 2. 결과값이 없는 함수
def sum(a,b):
    print "Q2 %d + %d = %d" % (a,b,a+b)

a=3
b=4
sum(a,b)

# 3. 입력값도 결과값도 없는 함수
def say():
    print 'Q3 Hi'

say()

# 4. 입력값이 여러 개인 함수
def sum_many(*arg):
    sum=0
    for i in arg:
        sum = sum + i
    return sum

result=sum_many(1,2,3)
print "Q4 "+str(result)
result=sum_many(1,2,3,4,5)
print "Q4 "+str(result)

## (**)는 key형을 입력할 때 사용한다.

# 5. 입력 인수에 초깃값 미리 설정된 함수
def say_myself(name,old,man=True):
    print "Q5 My name is %s." % name
    print "My age is %d." % old
    if man:
        print "I'm a guy"
    else:
        print "I'm a girl"

say_myself("DongHyun",24)
say_myself("JungEun",22,False)
```

함수의 여러 형태

1. 입력값이 없는 함수
2. 결과값이 없는 함수
3. 입력값도 결과값도 없는 함수
4. 입력값이 여러 개인 함수
5. 입력 인수에 초깃값 미리 설정된 함수

함수 안에서 선언된 변수의 효력 범위

결과는???

```
>>> a=1
>>> def vartest(a):
        a=a+1


>>> vartest(3)
>>> print a
```

- 함수 안에서 함수 밖의 변수를 변경하기 위한 방법

1. return 사용

```
>>> a=1
>>> def vartest(a):
        a=a+1
        return a

>>> a=vartest(a)
>>> print a
```

2. global 명령어 사용

```
>>> a=1
>>> def vartest():
        global a
        a=a+1


>>> vartest()
>>> print a
```

예제

1. 짝수인지 홀수인지 프린트해주는 함수를 만드세요.

2. 두 리스트를 비교하고,같은 수의 요소를 새로운 리스트에 넣고 그 리스트를 리턴 하는 함수를 만드세요.

파일 읽고 쓰기

f.=open(“새파일.txt”,’w’)

f.close()

- 파일 객체=open(파일 이름, 파일 열기 모드) -

파일 열기 모드	설명
r	읽기 모드 – 파일을 읽기만 할 때 사용
w	쓰기 모드 – 파일에 내용을 쓸 때 사용
s	추가 모드 – 파일의 마지막에 새로운 내용을 추가할 때 사용

- 파일을 새로 만든 다음 프로그램에 의해 만들어진 결과값을 새 파일에 적어 보고, 파일에 적은 내용을 읽어보는 프로그램을 만든다.

파일 쓰기

```
# writedata.py
f=open("C:/Python35/새파일.txt",'w')
for i in range(1,11):
    data="%d번째 줄입니다.\n"%i
    f.write(data)
f.close()
```

- Python35 파일 위치에 새 파일이라는 텍스트 문서가 생성된다.

파일 읽기

- readlines() 함수 사용
  - f = open(“C:/Python35/새파일.txt”, ‘r’)
  - lines = f.readlines( )
  - for line in lines:
    print(line):
    f.close()
- 한 라인이 리스트 요소 하나로 읽힌다. 즉, lines는 [“1 번째 줄입니다.”,”2 번째 줄입니다”, … ,”10 번째 줄입니다.”] 써보고 index 로 나눠서 프린트해보면 이해가 빠를 것이다.

- read() 함수 사용
  - f = open("C:/Python35/새파일.txt","r")
  - data=f.read()
  - print(data)
  - f.close()
- 파일 전체 내용을 문자열로 리턴한다.

과제

1. 리스트에 있는 모든 수를 곱해서 그 수를 리턴 하는 함수를 만드세요.

Sample List: (8,2,3,-1,7)

2. 리스트에 있는 수에서 짝수만 프린트하는 함수를 만드세요.

Sample List: [1,2,3,4,5,6,7,8,9]

Expected Result: [2,4,6,8]

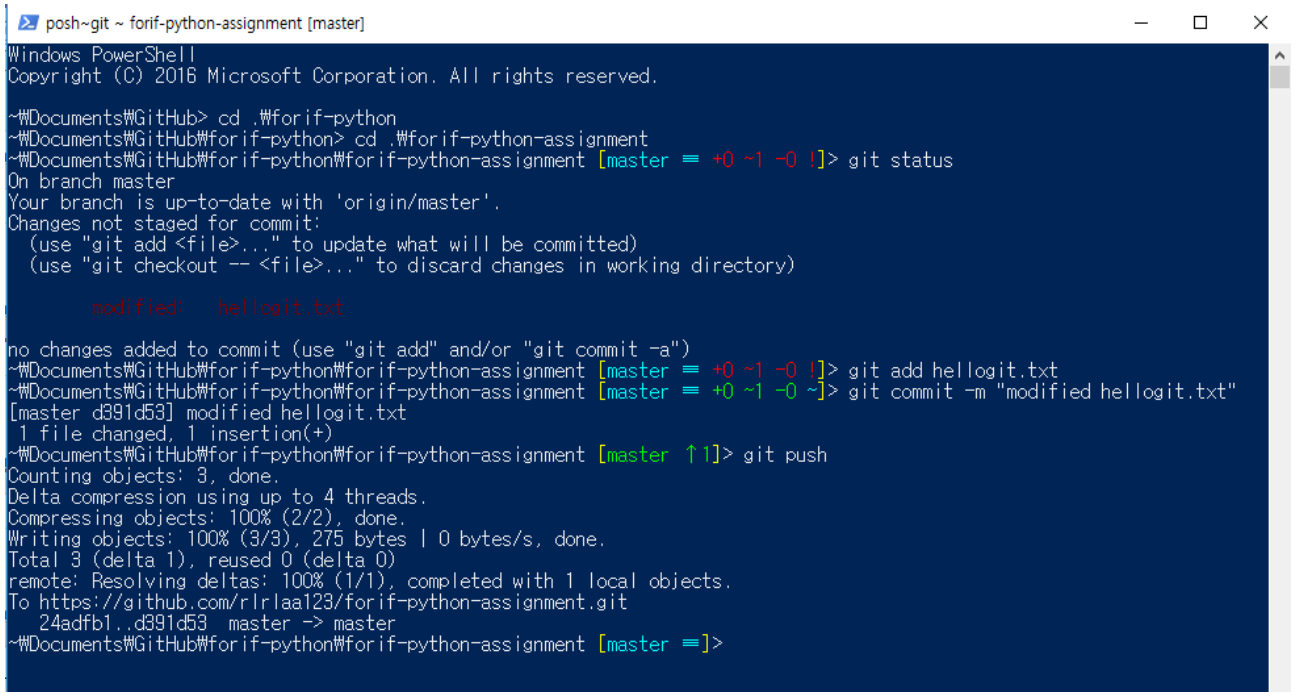
과제

- 3. 학생들의 이름,학번,전공을 각자 .txt 파일에 넣어서 보내 달라고 했는데 파일들을 일일이 열어서 확인하기가 힘들어 한 파일에 모두의 정보를 넣는 프로그램을 만들고 싶습니다. 파일명을 입력하면 그 파일에 있는 텍스트를 모두 읽어 들여 한 텍스트 파일에 쓰는 함수를 만드세요.
- * 함수의 입력 값으로는 파일명을 쓰는걸로 합니다.
  - 예) def collectInfo('김동현','강민주','김찬호'):
- 요 링크 참고해서 파일 읽고 쓰기를 익혀보세요. - link

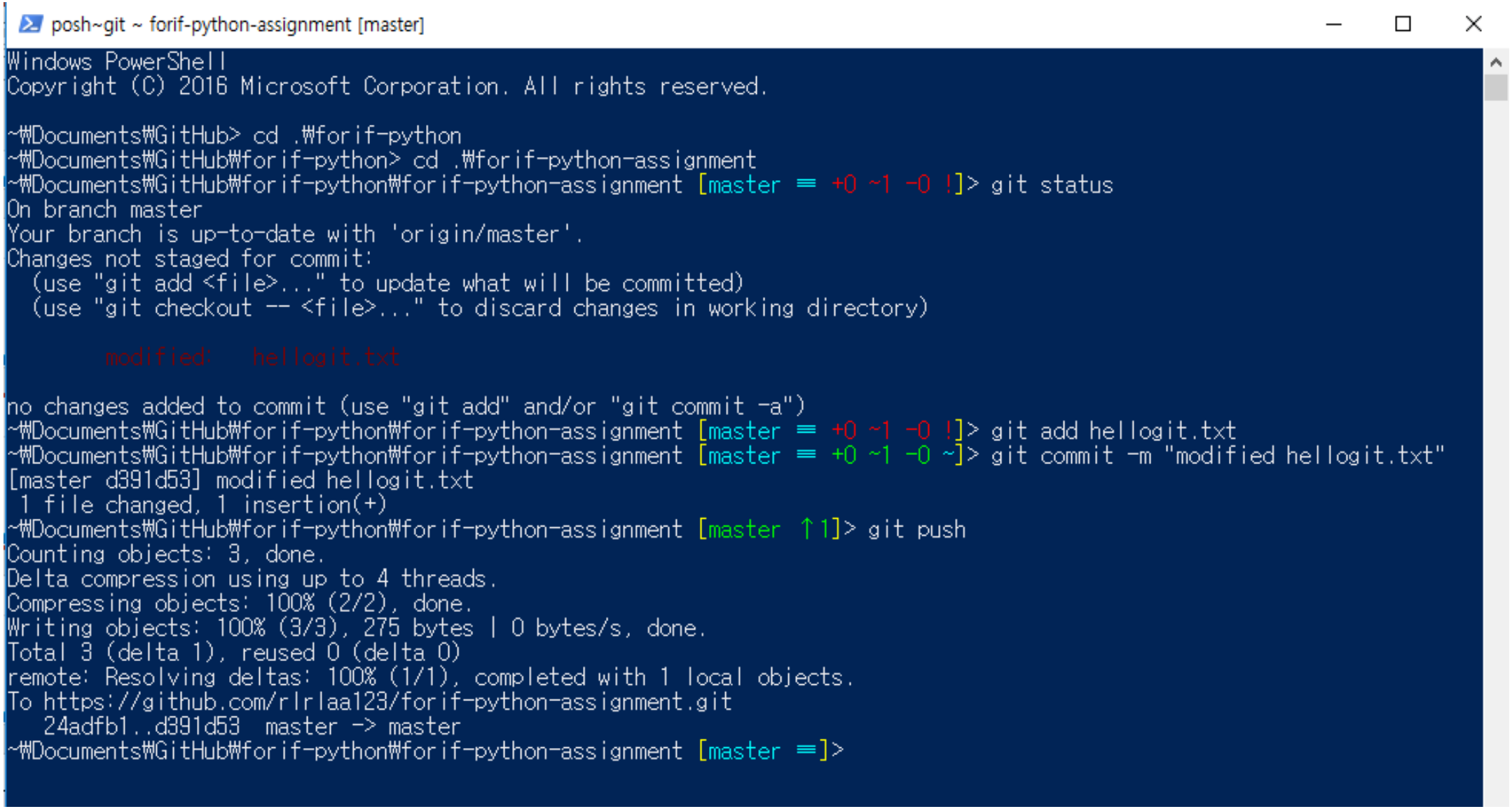
Git

*Git 다운로드 https://git-scm.com/ Git 튜토리얼 https://try.github.io/

- Git 은 협업용 소프트웨어라고 저는 부릅니다. 코드 수정 및 파일 업로드를 여러 명이 쉽게 서버에서 할 수 있고, 수정되거나 추가 및 삭제된 파일에 대해서는 모두 기록이 남습니다.



- hellogit.txt라는 파일에 텍스트를 변경하고 그 변한 코드를 git에 올리는 프로세스 입니다. 빨간색으로 된 modified: hellogit.txt 라는 텍스트와 같이 프로그램이 파일의 변화를 감지하고 알려주고, 다음과 같은 모든 프로세스는 모두 기록이 남습니다.


posh~git ~ forif-python-assignment [master]
Windows PowerShell
Copyright (C) 2016 Microsoft Corporation. All rights reserved.
~₩Documents₩GitHub> cd .₩forif-python
~₩Documents₩GitHub₩forif-python> cd .₩forif-python-assignment
~₩Documents₩GitHub₩forif-python₩forif-python-assignment [master ≡ +0 ~1 -0 !]> git status
On branch master
Your branch is up-to-date with 'origin/master'.
Changes not staged for commit:
(use "git add <file>..." to update what will be committed)
(use "git checkout -- <file>..." to discard changes in working directory)
modified: hellogit.txt
no changes added to commit (use "git add" and/or "git commit -a")
~₩Documents₩GitHub₩forif-python₩forif-python-assignment [master ≡ +0 ~1 -0 !]> git add hellogit.txt
~₩Documents₩GitHub₩forif-python₩forif-python-assignment [master ≡ +0 ~1 -0 ~]> git commit -m "modified hellogit.txt"
[master d391d53] modified hellogit.txt
1 file changed, 1 insertion(+)
~₩Documents₩GitHub₩forif-python₩forif-python-assignment [master ↑1]> git push
Counting objects: 3, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 275 bytes | 0 bytes/s, done.
Total 3 (delta 1), reused 0 (delta 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local objects.
To https://github.com/rlrlaa123/forif-python-assignment.git
24adfb1..d391d53 master -> master
~₩Documents₩GitHub₩forif-python₩forif-python-assignment [master ≡]>

Git 시작하기

- 우선, https://github.com 에 가서 회원가입을 하고
- New Repository 라는 버튼을 클릭해서 새로운 저장소를 만듭니다. 이 Repository는 github 서버에 내 파일들을 올릴 수 있는 공간 및 저장소라고 보시면 됩니다.
  - 그리고 만들때, **Initialize this repository with a README** 부분을 클릭해제 하시고 Repository를 만드세요.
- 이제, 방금 만든 레포지토리에 ReadMe.md 파일을 올리는 것을 같이 해볼 것입니다.

명령 프롬프트(cmd)

- 명령어 인터페이스란 텍스트 터미널을 통해 사용자와 컴퓨터가 상호작용하는 방식을 뜻한다고 합니다. 지금처럼 마우스로 드래그앤 클릭해서 파일을 관리하기 이전에는 이 방식만으로 컴퓨터와 상호작용 했습니다.
- git을 배우기 위해 필요한 간단한 cmd 명령어들만 알려드리겠습니다. 하지만 앞으로 개발을 하면서 cmd 는 많이 접하게 될 것이니 다른 명령어들도 찾아서 익혀 두면 좋을 것 같습니다.

명령어	기능
cd	디렉토리 변경
ls (dir)	파일 리스트 보기
mkdir	디렉토리 생성
rm	디렉토리 삭제
vi	vi 편집기로 들어감
touch	빈 파일을 생성

간단한 프로세스

These steps can be followed when you first make your repository.

1.Make Respository on Github
- •By doing this, you are creating your own space on Github’s server.

2.Create new directory that you want to share files with your Repository.
- •On your computer

3.In that directory, start git git init
- •command is

4.Create Readme.md file git touch Readme.md
- •you created Readme.md file on your computer but you should also upload it on your Repository so that you can share it with your team.

5.Track Readme.md file git add Readme.md OR git add *
- •(asterick ‘*’ means everything)
- •By doing so you are telling git to use or point at such file.

6.Check what the git is pointing or about to do. git status
- •This command will show what kind of changes it will perform.

7.Commit the change git commit -m "write_message"
- •Commit means you are performing the change. Think of all the open-source projects on Github, whenver people make
- •changes to the code, they are committing to the project. I think this is why Git calls it commit.

8.Add remote repository git remote add [repository_name] [repository_address]
- •You should tell the git where the repository you made on Github is and add the direction on your computer.

9.Check your remote repository git remote

10.Upload the committed changes to your repository git push -u [repository_name] [branch_name]

레포지토리 불러오기(Clone Repository)

1.This is how I clone other repository to mine

2.fork the repository
3.click clone and download button
4.copy url to request HTTP
5.clone it to your local directory git clone_ [request url]
6.Track everychange git add *
    •check every changes on the directory
7.commit
8.push

When you change something on your Github repository

Don’t forget to do ***git push*** This command will update every changes from your repository to your directory.

과제

오늘 만든 Repository는 여러분들이 어떻게 쓰냐에 달려있지만 제가 여러분들의 과제를 검사하는 공간이 될 것입니다.

1. Readme.md 파일에 자기 Repository에 대해 간단히 소개하고 저를 포함한 파이썬방 다른 사람들의 Repository 주소를 추가하세요.
2. 이번주 과제를 여기에 제출하세요.

참고

Python Docs-Function https://docs.python.org/3/tutorial/controlflow.html#defining-functions

함수 https://wikidocs.net/24

파일 입출력 https://wikidocs.net/26