

데이터 처리 실습

인공지능 플랫폼 설계 - 13주차

학습 내용

1. 데이터 처리 실습
2. Git 활용



Python 프로그래밍

데이터 정제

- 결측치 처리 : 빠진 데이터 처리
- 이상치 처리 → 너무 크거나 작은 값
- 오류 데이터 처리 → 잘못 된 값
- 중복 데이터 처리

데이터 정제 : 결측치 처리

1. 결측치 삭제 : dropna()

- dropna() 메서드는 기본적으로 NaN이 하나 이상 포함된 행이나 열을 삭제
- axis 매개변수를 1로 지정하여 데이터프레임에서 NaN이 포함된 열을 삭제하기
- 모든 값이 NaN인 열을 삭제하려면 dropna() 메서드에 how 매개변수를 'all'로 지정

df.dropna(axis=0) → 행 제거

df.dropna(axis=1) → 열 제거

2. 결측치 대체 : fillna()

- fillna() 함수 사용
- 인자로는 대체할 값을 지정

df.fillna(0)

3. 결측치 정보 : info(), isnull().sum()

- df.info() 함수
- isnull()을 sum()으로 합산

df.info()

df.isnull().sum()

_데이터 정제 : 이상치 처리

- 이상치를 찾는 방법 이상치에 대한 조건을 부여해서 셀을 찾고 수정
- 형식

`df.loc[<이상치 조건> , <열이름>] = <수정값>`

ex) `df.loc[(df['열3'] > 100) | (df['열3'] < 0) , '열3'] = 0`

```
import pandas as pd
```

```
data = {'열1': [1, 2, 2, 3, 4],  
        '열2': ['A', 'B', 'B', 'C', 'D'],  
        '열3': [-10, 20, 30, 40, 150],  
        '열4': ['A', 'B', 'B', 'Z', 'Z']}
```

```
df = pd.DataFrame(data)
```

```
df.loc[df['열4'] == 'Z', '열4'] = 'F'
```

```
df.loc[(df['열3'] > 100) | (df['열3'] < 0) , '열3'] = 0
```

	열1	열2	열3	열4
0	1	A	-10	A
1	2	B	20	B
2	2	B	30	B
3	3	C	40	Z
4	4	D	150	Z

_데이터 정제 : 중복행 처리

중복행 처리 단계

- `df.drop_duplicates(subset=[<중복을 체크할 열의 조합>], keep='first/last', inplace=True)`
- `keep` 옵션은 중복된 행 중에서 어떤 것을 선택할 것인지 지정

`df.drop_duplicates(subset=['열2','열4'], keep='first', inplace=True)`

	열1	열2	열3	열4
0	1	A	0	A
1	2	B	20	B
2	2	B	30	B
3	3	C	40	F
4	4	D	0	F

_데이터 실습

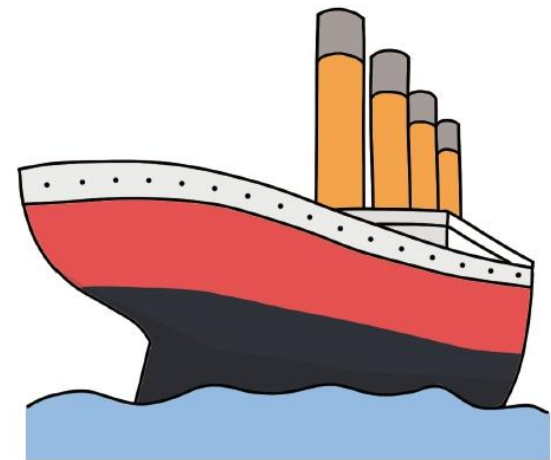
[문제]

타이타닉호 침몰은 영화로도 제작된 유명한 사건입니다.

1912년 4월 10일 유람선 타이타닉호가 영국 사우샘프턴을 떠나 미국 뉴욕으로 향하는 첫 항해 중 빙산과 충돌하여 침몰했습니다.

당시 배에 2,200여 명이 승선하였으나 그 중 1,500여 명이 사망하였습니다.

이번 예제에서는 타이타닉 탑승자 데이터에 결측치가 있는지 살펴보고 요금 열의 이상치를 확인해 봅시다.



_데이터 실습

1. 데이터를 읽어와 행과 열 수를 확인. 'titanic.csv' 파일 이용.

```
import pandas as pd
df = pd.read_csv('./titanic.csv')
df.shape
```

(891, 12)

승객이 891명이고 열은 12개.

2. isnull() 함수로 각 열의 결측치 수를 확인.

```
df.isnull().sum()
```

```
PassengerId 0
Survived 0
Pclass 0
Name 0
Sex 0
Age 177
SibSp 0
Parch 0
Ticket 0
Fare 0
Cabin 687
Embarked 2
```

나이 177건, 방 호수 687건, 탑승지 2건이 누락.



_데이터 실습

3. 요금 열에서 이상치 개수를 확인

상위 25%, 75% 기준 범위를 벗어난 데이터를 확인하기

```
q1 = df['Fare'].quantile(.25)
q3 = df['Fare'].quantile(.75)
IQR = q3 - q1
print('하한값:', q1 - 1.5 * IQR, '상한값:', q3 + 1.5 * IQR)

out1 = df[df['Fare'] < (q1 - 1.5 * IQR)]
out2 = df[df['Fare'] > (q3 + 1.5 * IQR)]
len(out1), len(out2)
```

```
하한값: -26.724 상한값: 65.6344
(0, 116)
```

하한값보다 작은 이상치는 없으나 상한값보다 큰 이상치는 116개나 있음. 예상한 것과 일치함,

4. 성별 구분

```
sum(out2['Sex'] == 'male')
```

46

요금을 특별히 많이 낸 승객 116명 중 46명이 남성이고 나머지 70명은 여성.



IT 플랫폼 실습

_Git 시스템

<https://git-scm.com/downloads/win>

Download for Windows

Click here to download the latest (2.45.1) 64-bit version of Git for Windows recent maintained build. It was released 5 days ago, on 2024-05-14.

Other Git for Windows downloads

Standalone Installer

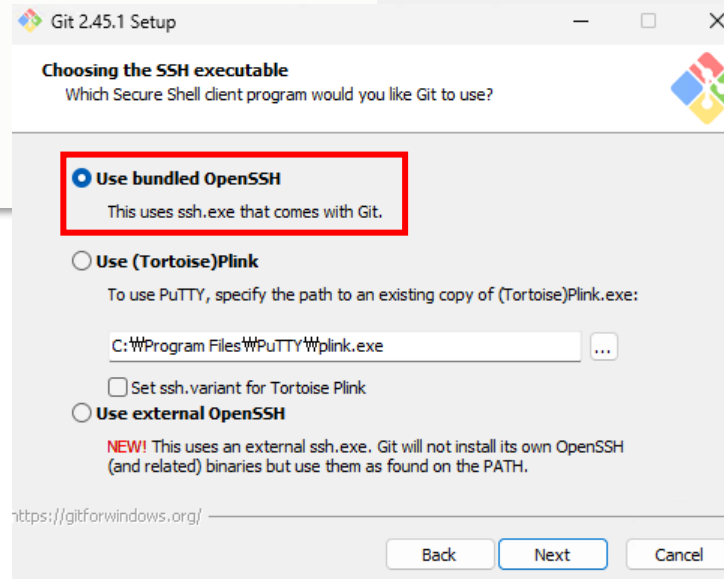
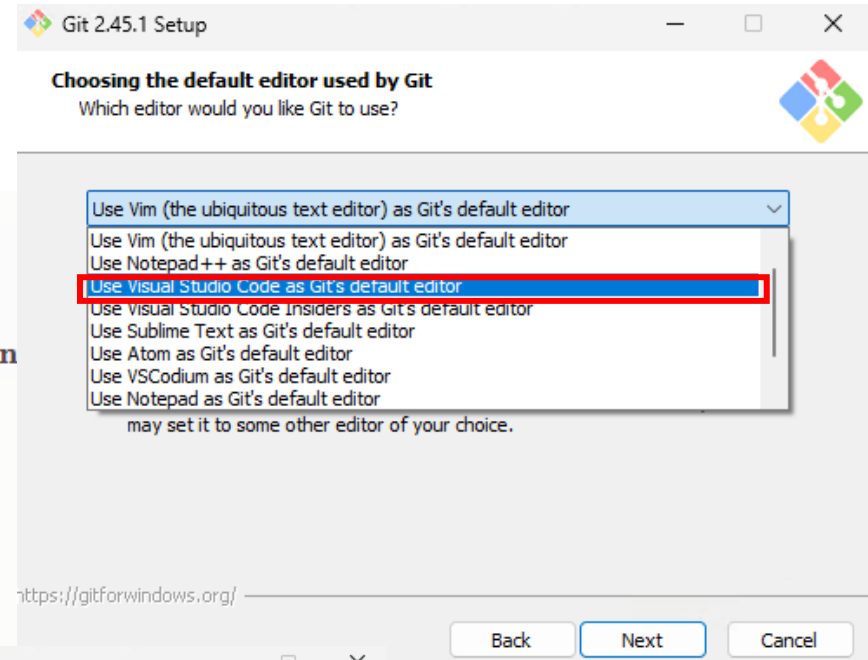
32-bit Git for Windows Setup.

64-bit Git for Windows Setup.

Portable ("thumbdrive edition")

32-bit Git for Windows Portable.

64-bit Git for Windows Portable.

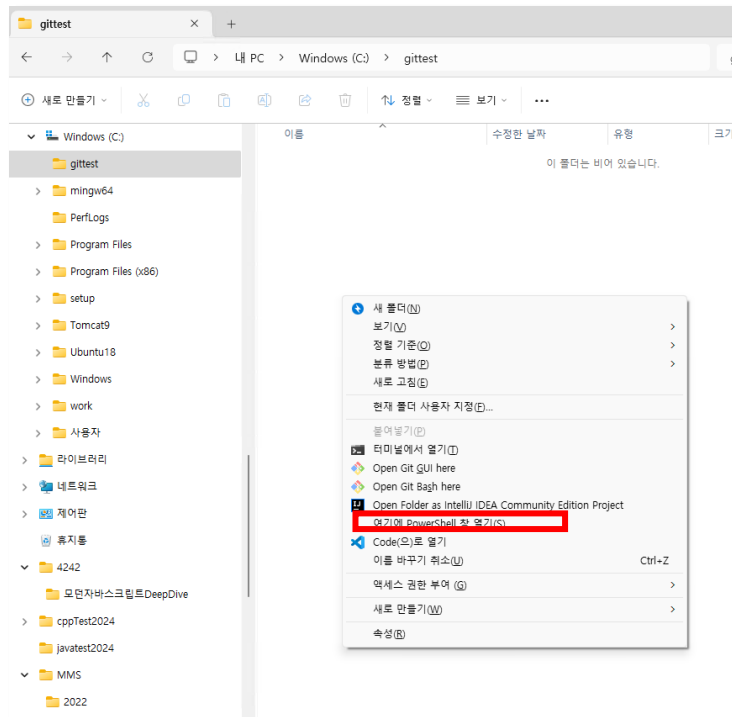


_Git : Local에 git 생성하기

- GIT를 사용할 디렉토리 생성
C:\Wgittest
- GIT 실행 (windows 파워셸)

git init

git status



```
관리자: Windows PowerShell
PS C:\gittest> git init
Initialized empty Git repository in C:/gittest/.git/
PS C:\gittest> git status
On branch master

No commits yet

nothing to commit (create/copy files and use "git add" to track)
PS C:\gittest> |
```



_Git 기본 사용

- 기본 설정

`git config --global user.email "홍길동@naver.com"`

`git config --global user.name "홍길동"`

- git에 올릴 파일의 편집

파일 편집

- staging area는 commit을 하기 전에 commit할 파일들을 골라놓는 곳입니다.

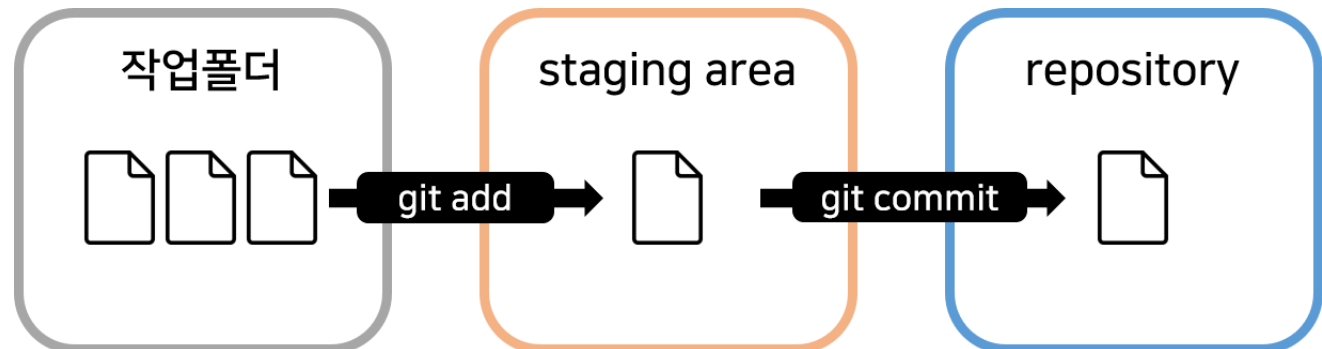
`git add .`

`git status`

- repository는 commit된 파일의 버전들을 모아놓는 곳 (로컬)

`git commit -m '메세지'`

`git log --stat`



_Git 기본 실습

git 로그 : 커밋 이력을 확인

```
git log
```

```
git log --stat
```

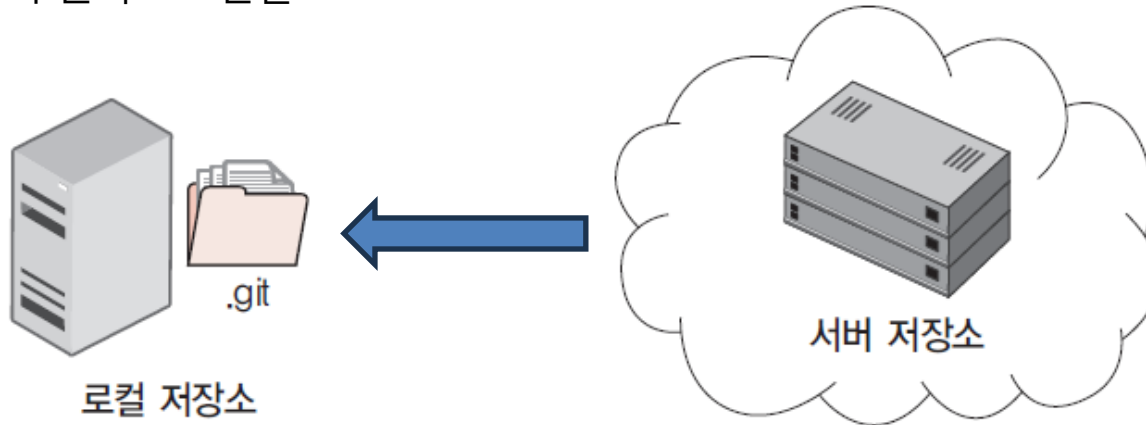
```
git log --pretty=format:"%h - %an, %ar : %s"
```

```
git log --graph
```

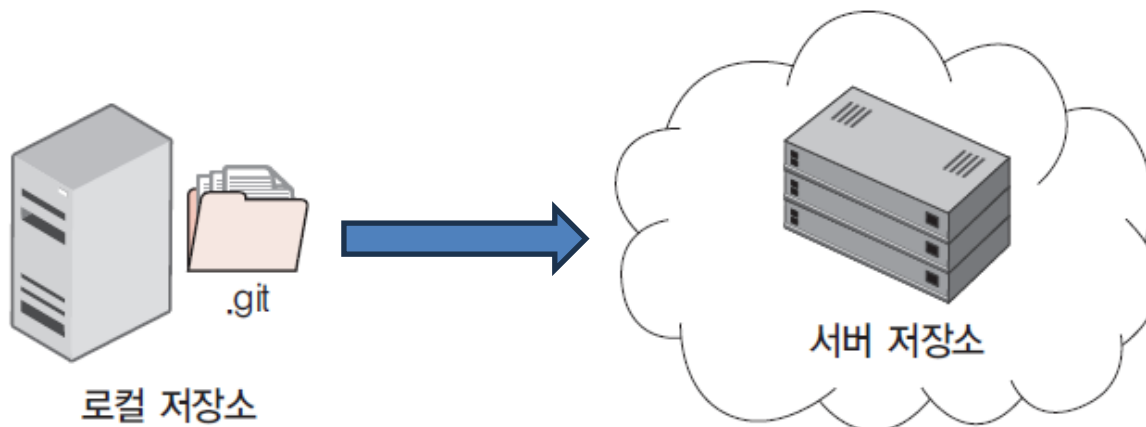


_Git : 원격과 연결하기

- 로컬 GIT과 원격 GIT 연결



`git clone https://github.com/사용자이름/저장소이름.git`



`git remote add origin https://github.com/사용자이름/저장소이름.git`



_Git : 원격과 연결하기

- 원격 GIT 생성 (깃허브)
- 원격 repo를 로컬로 가져 오기 (clone)

`git clone https://github.com/사용자이름/저장소이름.git` 저장소이름

```
PS C:\gittest> git clone https://github.com/topmentor/edutest.git edutest
Cloning into 'edutest'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (3/3), done.
PS C:\gittest>
```

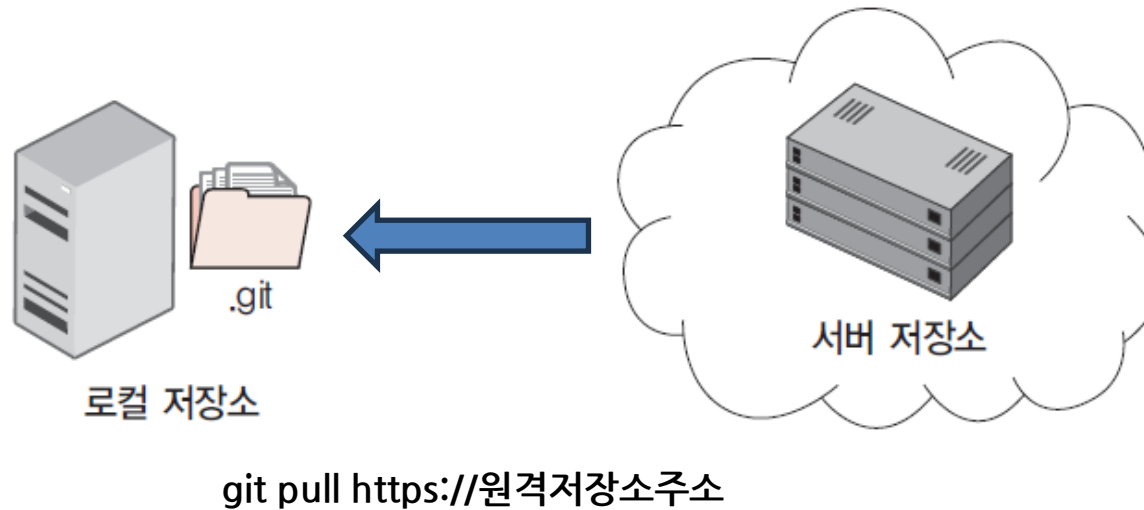
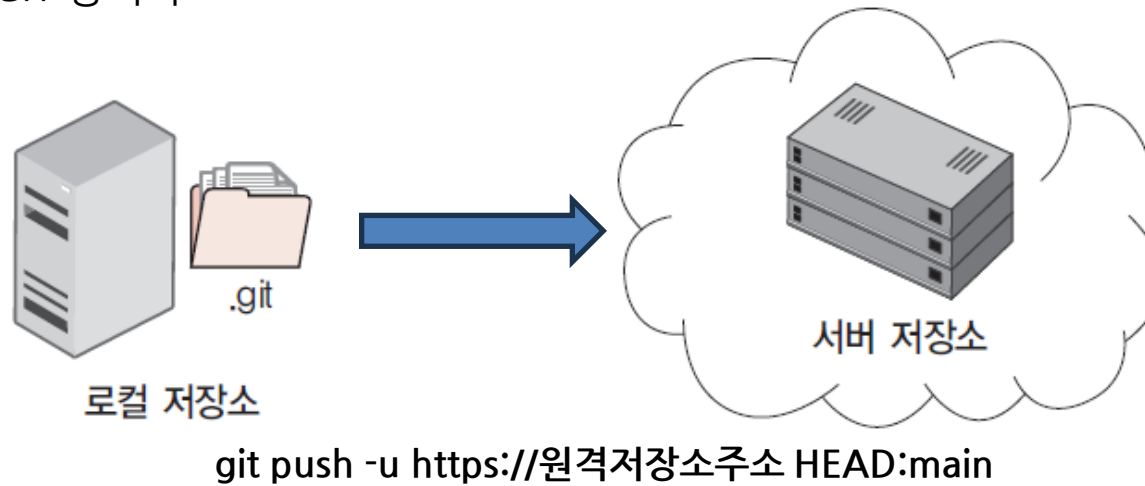
`cd` 저장소이름

`git log`



_Git : 원격과 연결하기

- 로컬 GIT과 원격 GIT 동기화



_Git : 원격과 연결하기 실습

```
git clone https://github.com/topmentor/edutest.git edutest
```

```
cd edutest
```

```
git log
```

*** gituse_학번.txt 파일 편집**

이름 : 홍길동

학번 : 2000

```
git add .
```

```
git commit -m '[2000] add gituse.txt '
```

```
git log --stat
```



_Git : 원격과 연결하기 실습 2

금일 실습 내용을 git으로 업로드(push) 하시오.

- clone 받은 edutest 폴더에 금일 실습한 파이썬 소스 복사
 - . 자기학번으로 폴더를 만드시오
 - . 자기학번 폴더 내에 파이썬 파일을 복사하시오

git add .

git commit -m "[20000] upload edu files"

git pull https://github.com/topmentor/edutest.git


git push https://github.com/topmentor/edutest.git HEAD:main

```
PS C:\gittest\edutest> git pull https://github.com/topmentor/edutest.git
remote: Enumerating objects: 4, done.
remote: Counting objects: 100% (4/4), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), 967 bytes | 96.00 KiB/s, done.
From https://github.com/topmentor/edutest
 * branch                HEAD       -> FETCH_HEAD
Updating 674b1eb..f65ba41
Fast-forward
 readinfo.txt | 1 +
 1 file changed, 1 insertion(+)
 create mode 100644 readinfo.txt
```



_Git 기본 실습

git push https://github.com/topmentor/edutest.git HEAD:main




Sign in to GitHub
to continue to Git Credential
Manager

Username or email address




Password [Forgot password?](#)



Sign in

Or

 Sign in with a passkey

```
PS C:\gittest\edutest> git push https://github.com/topmentor/edutest.git HEAD:main
info: please complete authentication in your browser...
Enumerating objects: 8, done.
Counting objects: 100% (8/8), done.
Delta compression using up to 4 threads
Compressing objects: 100% (5/5), done.
Writing objects: 100% (6/6), 813 bytes | 813.00 KiB/s, done.
Total 6 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/1), done.
To https://github.com/topmentor/edutest.git
f65ba41..5f8a8ea HEAD -> main
```

 13week_200000	[20000] edu files upload	7 minutes ago
 README.md	Initial commit	47 minutes ago
 readinfo.txt	Create readinfo.txt	30 minutes ago

 README 

edutest

GIT 테스트 repo

