

20기 정규세션

ToBig's 19기 강연자
이영아

EDA & GIT

Contents

EDA

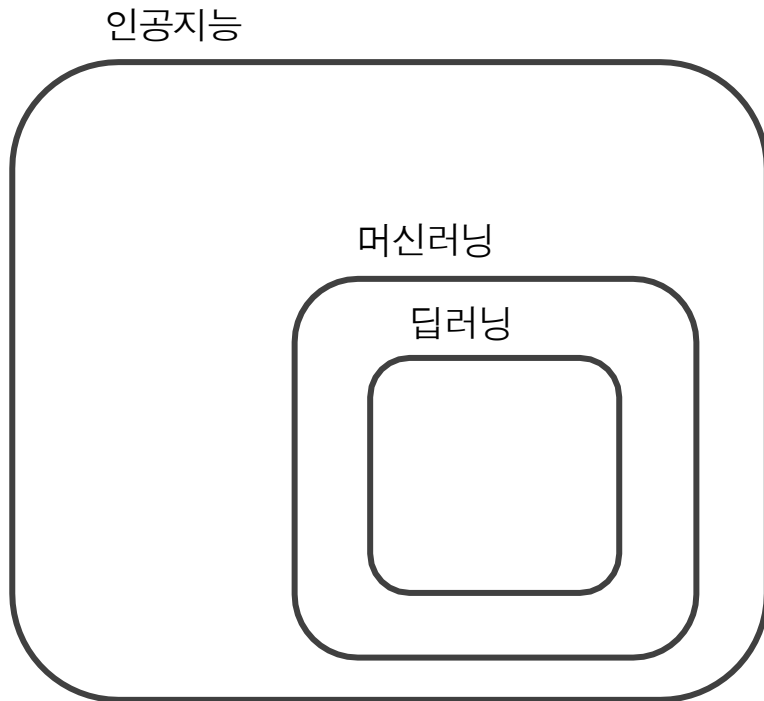
Unit 01 | Introduction

Unit 02 | 전처리 및 EDA 소개

Unit 03 | 실습 및 과제

Unit 01 | Introduction

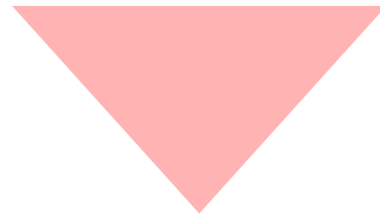
✓ 인공지능, 머신러닝, 딥러닝



딥러닝 : **인공신경망(Neural Network) 알고리즘**을 이용한
머신러닝의 한 분야

Unit 01 | Introduction

모든 과정은 데이터에 대한
충분한 이해가 수반되어야 함



전처리, EDA

Unit 02 | 전처리 및 EDA 소개

✓ 전처리 (Data pre-processing)

1. 결측치

- (1) 제거
- (2) 평균 대체
- (3) 중앙값 대체
- (4) 최빈값 대체
- (5) 선형보간법(interpolate)

Unit 02 | 전처리 및 EDA 소개

✓ 전처리 (Data pre-processing)

2. 이상치

- (1) IQR 방식 ($75\% \text{ percentile} + 1.5 * \text{IQR}$ 이상이거나 $25\% \text{ percentile} - 1.5 * \text{IQR}$ 이하인 경우
이상치로 제거 혹은 상하한값으로 대체)
- (2) 표준점수로 변환 후 -3 이하 및 +3 제거
- (3) 수치형 변수 범주화
- (4) 도메인 지식 이용

이 외에도 다양한 결측치, 이상치 처리 방식 존재

Unit 02 | 전처리 및 EDA 소개

✓ EDA란?

Exploratory Data Analysis

탐색적 데이터 분석

데이터를 분석하고 결과를 내는 과정에 있어서
지속적으로 해당 데이터에 대한 탐색과 이해를 기본적으로 가져야 한다는 의미

Unit 02 | 전처리 및 EDA 소개

데이터에 대한 개괄적 이해 & 도메인 자료조사

데이터에 대한 가설 설정

가설에 따라 여러 feature로 필터링하고 값을 출력 표와

그래프로 시각화 하며 인사이트 도출(사전 검증)



Valuable information 창출

Unit 03 | 실습 및 과제

1) 데이터 출처와 주제에 대한 이해

Uncover the factors that lead to employee attrition and explore important questions such as ‘show me a breakdown of distance from home by job role and attrition’ or ‘compare average monthly income by education and attrition’. This is a fictional data set created by IBM data scientists.

- IBM 데이터 과학자들이 만든 가상의 HR 데이터셋
- 1,470명에 대한 35개의 변수가 있으며, 종속변수는 Attrition, 즉 0 또는 1의 퇴사여부

Unit 03 | 실습 및 과제

2) 변수 종류 확인 및 분석 목적 설정

Attributes

Tid	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

Objects

데이터 셋의 구성

- Object
- Attribute : data object의 특징 혹은 피쳐(feature)

Unit 03 | 실습 및 과제

2) 변수 종류 확인 및 분석 목적 설정

종속변수

- Numerical 변수 → Regression
- Categorical 변수 → Nomial, Binary, Ordinal → Classification

Unit 03 | 실습 및 과제

2) 변수 종류 확인 및 분석 목적 설정

Numerical 변수 → Regression

- Interval Scaled : 연속형 변수 중에서 차이가 유의미하며, 순서가 존재하는 변수
- No true zero point
- Ex) 섭씨, 화씨, 날짜

- Ratio Scaled : 연속형 변수 중에서 차이와 비율이 유의미한 변수
- True zero point
- Ex) 돈, 길이, 무게, 개수

Unit 03 | 실습 및 과제

2) 변수 종류 확인 및 분석 목적 설정

Categorical 변수 → Nomial, Binary, Ordinal

- Nomial : 이름 혹은 상태와 같은 범주형 변수(순서X)
- Ex) 우편번호, 국적, 색깔, 직업
- Binary : Nomial 변수 중에서 0과 1로 2가지 상태만 존재
- Ex) 양성/음성, 남/여
- Ordinal : 변수들 간의 유의미한 순서 有, 연속적인 값 간의 차이 명확하지 X
- Ex) 성적, 계급, 사이즈

Unit 03 | 실습 및 과제

3) 데이터 살펴보기

- 분석의 목적과 해결해야 할 문제에 집중하며 데이터가 전체적으로 어떤 모양인지 살펴보기
- 주로 head(), tail(), info(), describe() 등을 통해 모든 기본 정보 체크

```
data.head()
```

	Age	Attrition	BusinessTravel	DailyRate	Department	DistanceFromHome	Education	EducationI
0	41	Yes	Travel_Rarely	1102	Sales	1	2	Life Sc
1	49	No	Travel_Frequently	279	Research & Development	8	1	Life Sc
2	37	Yes	Travel_Rarely	1373	Research & Development	2	2	
3	33	No	Travel_Frequently	1392	Research & Development	3	4	Life Sc

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>  
RangeIndex: 1470 entries, 0 to 1469  
Data columns (total 35 columns):  
#   Column                Non-Null Count  Dtype  
---  -  
0   Age                   1470 non-null  int64  
1   Attrition             1470 non-null  object  
2   BusinessTravel        1470 non-null  object  
3   DailyRate              1470 non-null  int64  
4   Department             1470 non-null  object  
5   DistanceFromHome      1470 non-null  int64  
~   ~~~~~
```

Unit 03 | 실습 및 과제

4) 결측치 및 이상치 탐색

- isnull()
- isna()

결측치처럼 보이지 않아도 결측의 의미를 지니는 변수 주의
메타데이터를 보면 [00 : 응답없음, 99 : 알수없음]
등의 결측치를 코드화하여 표시한 것들이 있음.

코드상으로는 결측치가 없다고 하지만, 이는 의미가 없는 데이터이므로 주의.

Unit 03 | 실습 및 과제

4) 결측치 및 이상치 탐색

- 평균과 표준편차 수치
- 데이터 표준화
- 박스플롯

다양한 방법을 사용하여 이상치가 있는지 확인하고 처리.

데이터의 의미를 생각하며 이상치 고려할 것.

Ex. ' 판매가 ' 라는 column에 (-) 음수가 측정되었다면 이상치일까?

Ex. ' 개인 소득 ' 이라는 column에 경계값에서 벗어난 높은 값이 있다면 제거해야 할까?

Unit 03 | 실습 및 과제

5) 데이터 시각화

범주형 자료
(Categorical Variable)

명목형
(Nominal)

Ex. 성별, 종교

순서형
(Ordinal)

Ex. 학년, 장애등급

수치형 자료
(Numeric Variable)

연속형
(Continuous)

Ex. 키, 몸무게

이산형
(Discrete)

Ex. 사고 건수

Unit 03 | 실습 및 과제

5) 데이터 시각화

명목형 자료 (Nominal Variable)

- 최빈값
- 범주별 측정값 개수
- 구성비율
- 바차트, 파이차트

순서형 자료 (Ordinal Variable)

- 최빈값, 평균, 중앙값
- 사분위수
- 범주별 측정값 개수
- 구성비율
- 바차트, 파이차트

수치형 자료 (Numerical Variable)

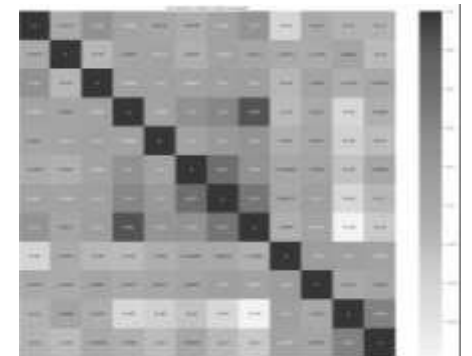
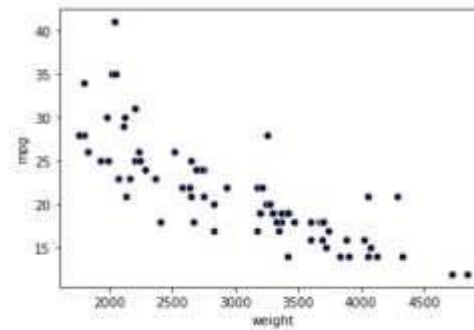
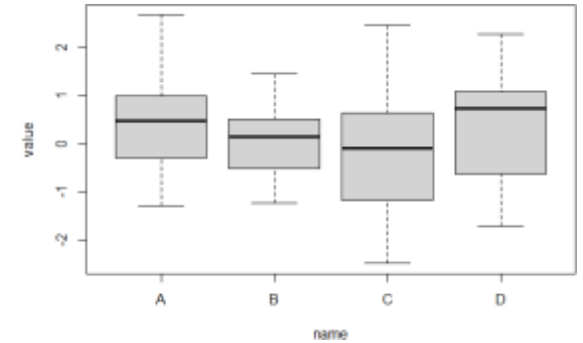
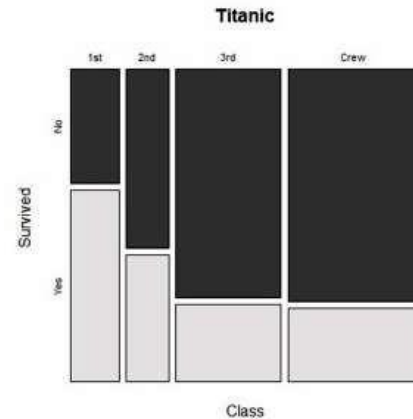
- 평균, 중앙값, 최빈값
- 분산, 표준편차
- 사분위수, 백분위수
- 히스토그램, 박스플롯

Unit 03 | 실습 및 과제

5) 데이터 시각화

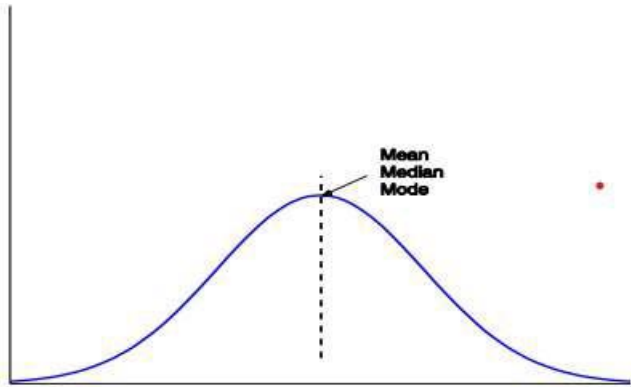
변수간 관계 살펴 보기

- 범주형 - 범주형
: 모자이크 플롯 등
- 수치형 - 범주형
: Side-by-side box plot 등
- 수치형 - 수치형
: 산점도, corr plot 등



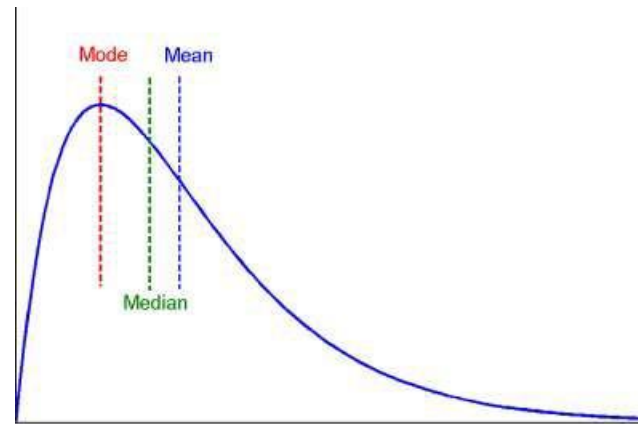
Unit 03 | 실습 및 과제

5) 데이터 시각화



Symmetric Data

- 이상적인 형태의 데이터 분포
- 정규분포에 유사한 형태



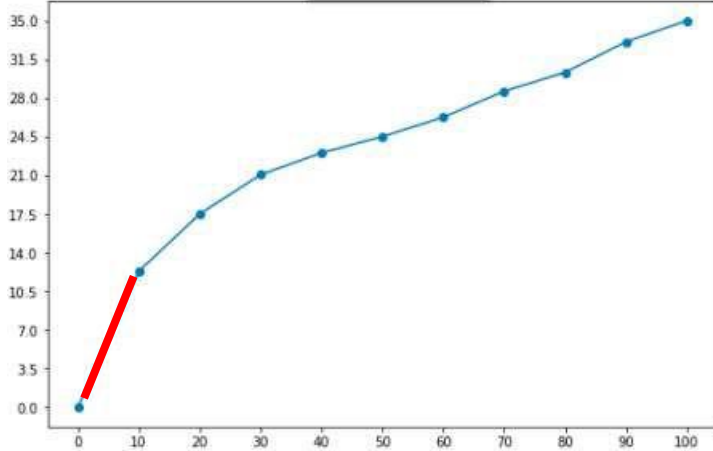
Skewed Data

- 왜도 Skew 값 3을 기준으로 일반적으로 판단
- 왜도 값이 높으면 간단한 전처리 진행(log1p)

Unit 03 | 실습 및 과제

5) 데이터 시각화

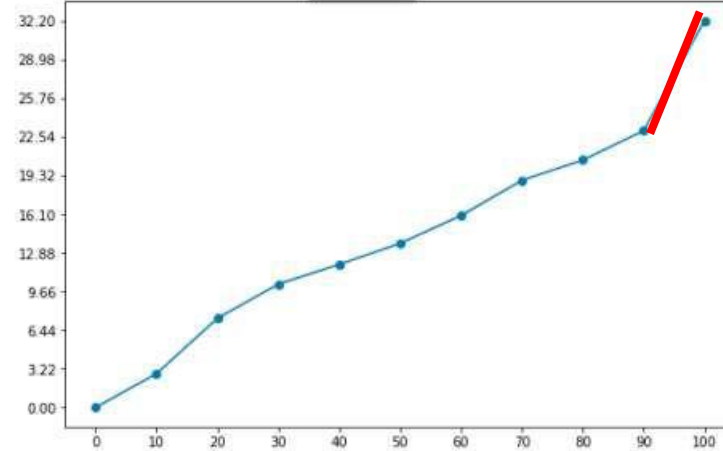
midterm



- 급격한 증가가 나타나는 하위 구간
- Variance가 주지 못하는 정보

Percentile Plot

final



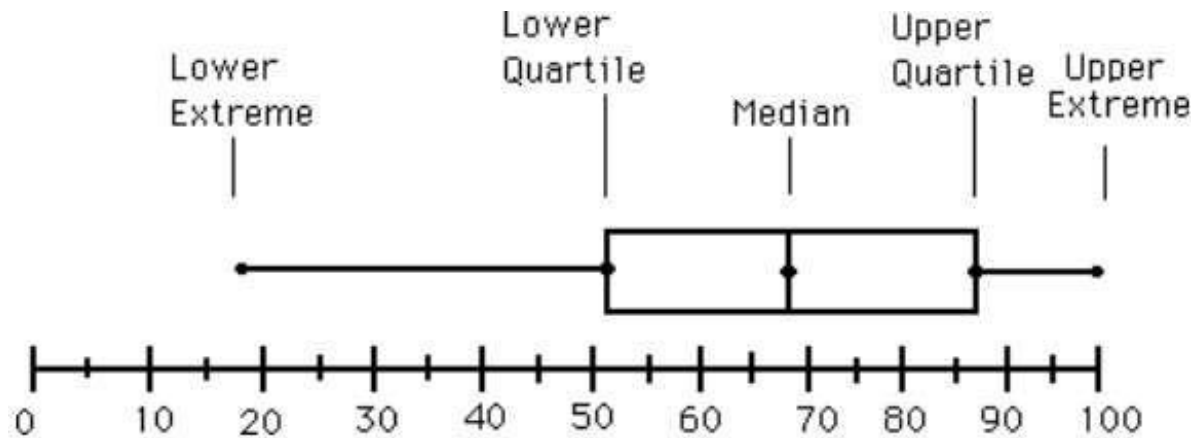
- 소수의 학생들만의 높은 점수
- 나머지 구간에서는 Linear

- P-th percentile : X_p
- X_p 보다 작은 값은 P%이다.
- 크기가 있는 값들로 이뤄진 자료들을 순서대로 나열했을 때, 백분율로 나타낸 특정 위치의 값

Unit 03 | 실습 및 과제

5) 데이터 시각화

Box Plot



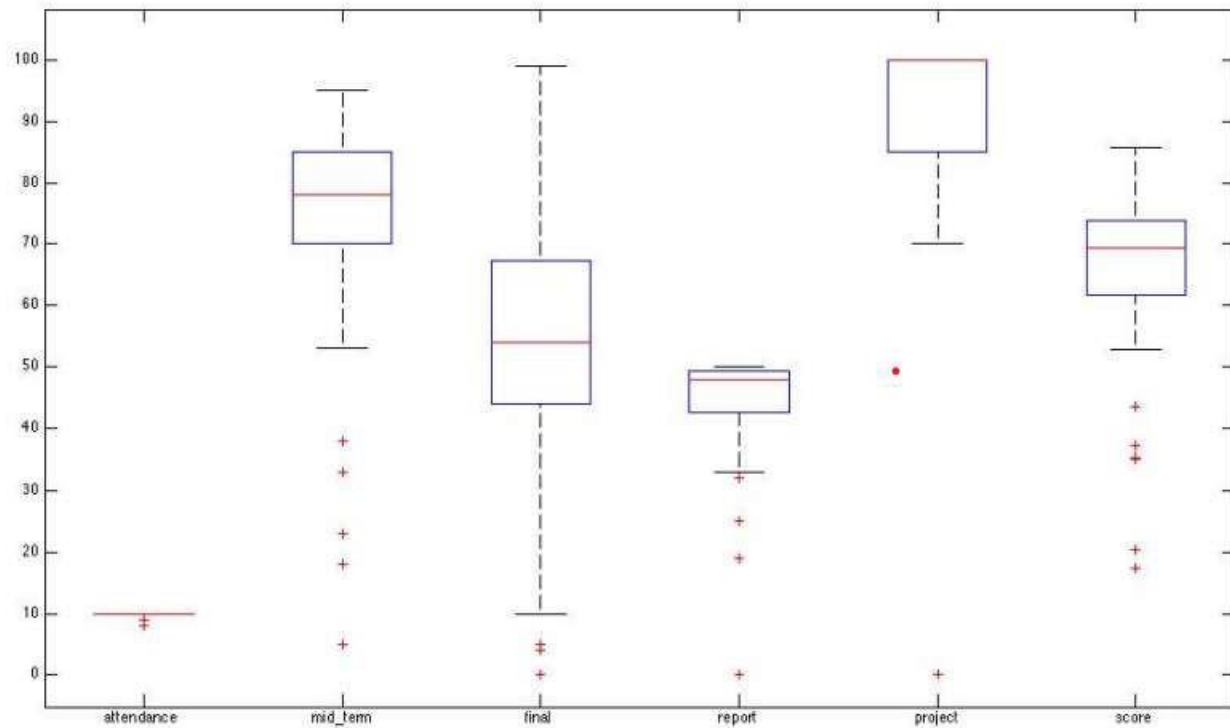
- Box 형태로 데이터를 표현
- Box의 끝점은 Q1과 Q3를 의미
- 중앙값(median)이 박스 중앙에 위치
- 이상치 판단 기준
- 데이터 분포를 파악하는데 용이

$$X > Q3 + 1.5 * IQR \text{ or } X < Q1 - 1.5 * IQR$$

Unit 03 | 실습 및 과제

5) 데이터 시각화

Box Plot

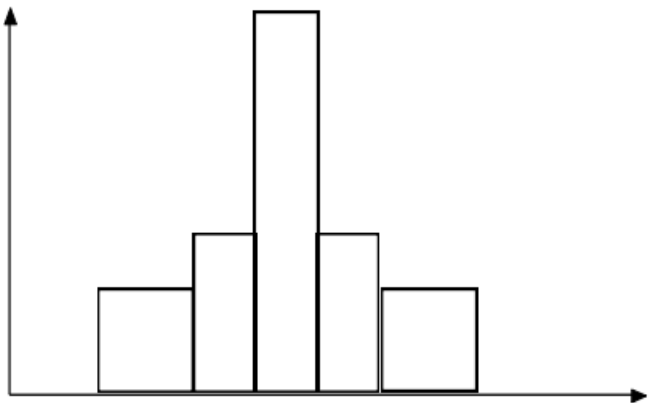
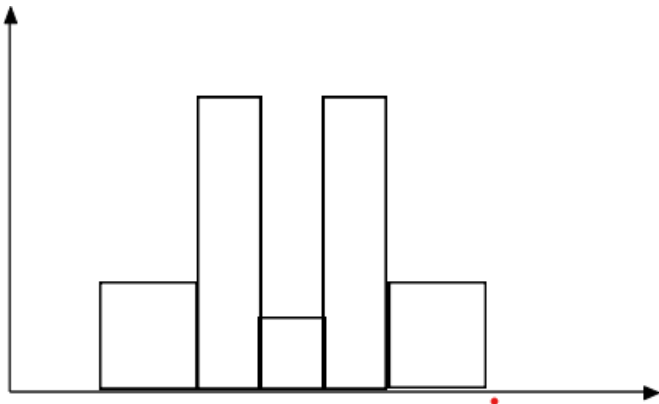


- 2번째(mid term)과 3번째(final) 간의 비교
- 2번째 Box의 크기가 더 작음 → 분산이 작음
- 3번째 Box의 크기가 더 큼 → 분산이 큼
- 3번째 Box의 경우 Extreme 간의 값 차이가 큼
- 5번째 Box의 경우 이상치가 적게 나타남

Unit 03 | 실습 및 과제

5) 데이터 시각화

Histogram

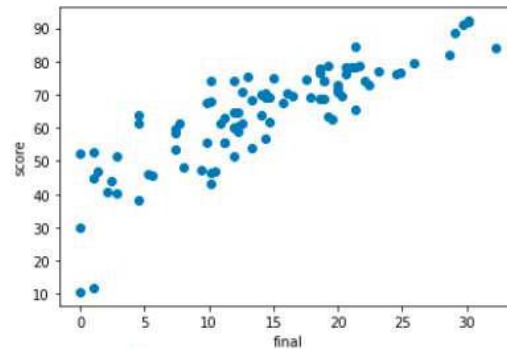
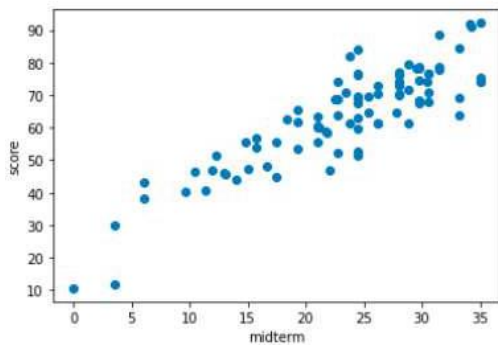
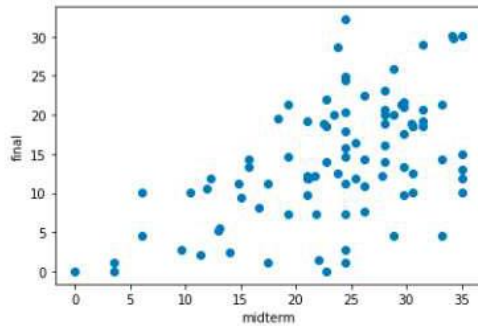


- 표로 되어있는 도수 분포를 그림으로 나타낸 것
- 변수의 구간에 따라 잘라서 나눠준 것
- Box Plot이 해석하지 못하는 정보 제공
- 좌측 두 히스토그램은 lower, Q1, median, Q3, upper 값이 같음
- 그러나, 데이터 분포가 다름
- **히스토그램을 통해 데이터 분포 반드시 확인!**

Unit 03 | 실습 및 과제

5) 데이터 시각화

Scatter Plot & Correlation

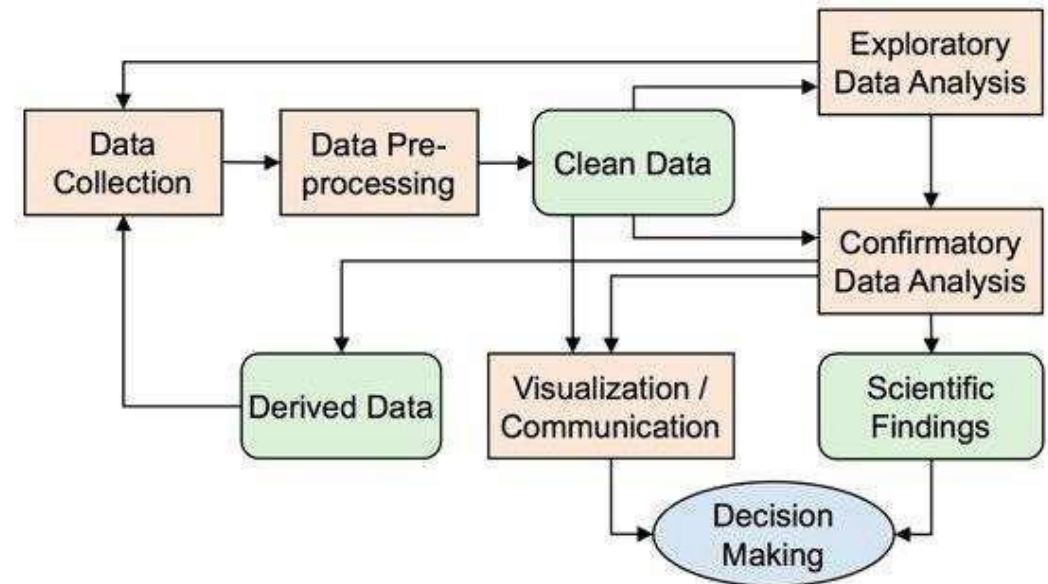


- 산점도(Scatter Plot)을 통해 두 변수 간의 선형 상관관계 파악
- 선형 상관관계? A 변수가 증가하면, B 변수가 증가하는지 혹은 감소하는지
- 단순한 선형적인 관계를 파악하는 것
- 대략적인 데이터 분포 및 관계 파악에 용이

Unit 03 | 실습 및 과제

6) EDA 그 이후

- EDA에서 얻은 인사이트로 적절한 분석방법 모색
- EDA를 통해 다시 데이터 전처리를 할 것은 없는지 검토
- 추후 분석에서 조심해야 할 부분, 주의 점 등 기록



Unit 03 | 실습 및 과제

✓ 과제

- 데이터 : IBM HR Attrition Data

<https://www.kaggle.com/datasets/pavansubhasht/ibm-hr-analytics-attrition-dataset>

혹은 자신이 원하는 데이터셋 사용 가능. 단, 다른 데이터 셋을 사용할 시에는 Data description을 볼 수 있는 원본 링크 참고 필수

- 파이썬을 이용하여 전처리 및 EDA를 진행해주세요.
 - 결측치, 이상치 검토
 - 유의미한 시각화 5개 이상
 - 수치형 변수 간 상관관계 파악
 - 파생변수 생성

Unit 03 | 실습 및 과제

Data Set Description

1. Age : 해당 직원의 나이
2. Attrition : 퇴직 여부 Target값 (종속변수)
3. BusinessTravel : 출장의 빈도
4. DailyRate : 일 대비 급여의 수준
5. Department : 업무분야
6. DistanceFromHome : 집과의 거리
7. Education : 교육의 정도
8. EducationField : 전공
9. EmployeeCount : 직원 숫자
10. EmployeeNumber : 직원 ID
11. EnvironmentSatisfaction : 업무 환경에 대한 만족도
12. Gender : 성별
13. HourlyRate : 시간 대비 급여의 수준
14. JobInvolvement : 업무 참여도
15. JobLevel : 업무의 수준
16. JobRole : 업무 종류
17. JobSatisfaction : 업무 만족도
18. MaritalStatus : 결혼 여부
19. MonthlyIncome : 월 소득
20. MonthlyRate : 월 대비 급여 수준
21. NumCompaniesWorked : 일한 회사의 수
22. Over18 : 18세 이상
23. OverTime : 규정외 노동시간
24. PercentSalaryHike : 급여의 증가분 백분율
25. PerformanceRating : 업무 성과
26. RelationshipSatisfaction : 대인관계 만족도
27. StandardHours : 표준 시간
28. StockOptionLevel : 스톡옵션 정도
29. TotalWorkingYears : 경력 기간
30. TrainingTimesLastYear : 교육 시간
31. WorkLifeBalance : 일과 생활의 균형 정도
32. YearsAtCompany : 근속 연수
33. YearsInCurrentRole : 현재 역할의 년수
34. YearsSinceLastPromotion : 마지막 프로모션
35. YearsWithCurrManager : 현재 관리자와 함께 보낸 시간

Contents

GIT

Unit 01 | GIT 기초 – 로컬 저장소

Unit 02 | Git 협업 – 원격 저장소

Unit 03 | 과제 설명

Unit 01 | GIT 기초 - 로컬 저장소



깃(Git /git/)은 컴퓨터 파일의 변경사항을 추적하고
여러 명의 사용자들 간에 해당 파일들의 작업을 조율하기 위한 **분산 버전 관리 시스템**

Unit 01 | GIT 기초 - 로컬 저장소

버전 관리 시스템 (VCS)



수많은 버전의 파일 중...

Unit 01 | GIT 기초 - 로컬 저장소

버전 관리 시스템 (VCS)

과
거



컨퍼모델.py



컨퍼모델_최종.py



컨퍼모델_최종최종.py



컨퍼모델_진짜최종.py



컨퍼모델_라스트.py



컨퍼최종.py



현
재

최종제출.py

헛갈리지 않도록 최종 파일만 보여줍니다

Unit 01 | GIT 기초 - 로컬 저장소

버전 관리 시스템 (VCS)

과
거



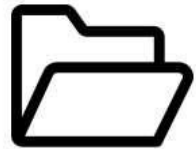
컨퍼모델.py



컨퍼모델_최종.py



컨퍼모델_최종최종.py



컨퍼모델_진짜최종.py



컨퍼모델_라스트.py



컨퍼최종.py



최종제출.py

현
재

원하는 버전으로 다시 돌아갈 수도 있어요!

Unit 01 | GIT 기초 - 로컬 저장소

Git을 GUI로 사용할 수 있는
다양한 응용프로그램들이 있지만

- GitHub Desktop
- SourceTree
- GitKraken
- Tortoise Git

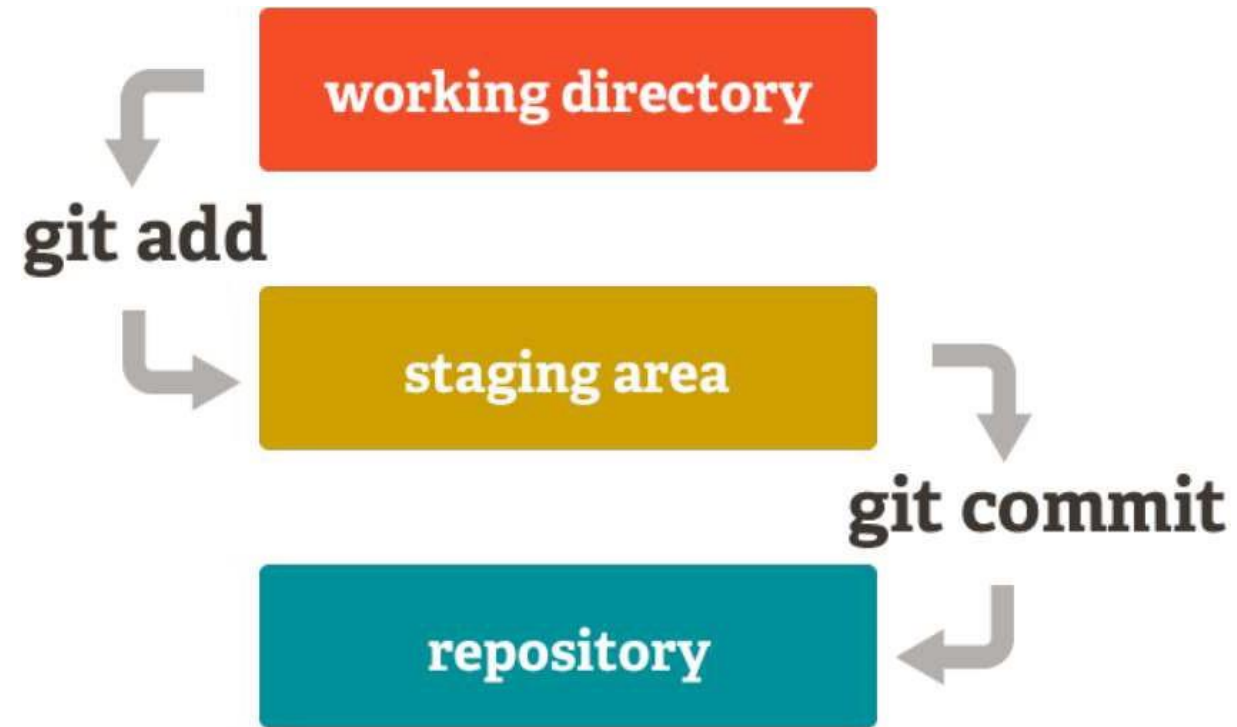
...

GUI가 더 단순하기 때문에
CLI를 사용할 줄 알면 GUI도
사용 가능

-> 각 환경에 맞게
명령 프롬프트(cmd) 창 사용하기!

깃 설치에 대한 내용은 강의에 포함되어 있지 않으니,
설치 돼 있지 않다면 설치 & github 계정도 만들기!

Unit 01 | GIT 기초 - 로컬 저장소



로컬 저장소(local git repository) 만들고 파일 올리며 관리하기

Unit 01 | GIT 기초 - 로컬 저장소

로컬 저장소와 원격 저장소

- 로컬 저장소(Local Repository)

: 내 PC에 파일이 저장되는 개인 전용 저장소입니다.

- 원격 저장소(Remote Repository)

: 파일이 원격 저장소 전용 서버에서 관리되며 여러 사람이 함께 공유하기 위한 저장소입니다.

⇒ 로컬에 파일이 올라가도 다른 사람은 아직 볼 수 없습니다

Unit 01 | GIT 기초 - 로컬 저장소

로컬 저장소에 파일 업로드 하는 프로세스



Unit 01 | GIT 기초 - 로컬 저장소

• 로컬 저장소 만들기

1) 저장소 새로 만들기

git init

저장소 만들 디렉토리 위치에서 실행

2) 원격 저장소를 로컬로 복사해오기

git clone [원격 저장소 주소]

실행한 디렉토리에 원격 저장소 폴더가 복사됨

```
tobigs_test — Jeongmin@Jeongminui-MacBookAir — ..p/tobigs_test — zsh — 90x38
Last login: Tue Jul 19 23:03:00 on ttys000
/Users/Jeongmin/.zshrc:source:102: no such file or directory: /Users/Jeongmin/zsh-syntax-h
ighlighting/zsh-syntax-highlighting.zsh
(base) ❶ cd desktop
(base) ❷ cd tobigs_test
(base) ❸ git init
hint: Using 'master' as the name for the initial branch. This default branch name
hint: is subject to change. To configure the initial branch name to use in all
hint: of your new repositories, which will suppress this warning, call:
hint:
hint:   git config --global init.defaultBranch <name>
hint:
hint: Names commonly chosen instead of 'master' are 'main', 'trunk' and
hint: 'development'. The just-created branch can be renamed via this command:
hint:
hint:   git branch -m <name>
Initialized empty Git repository in /Users/Jeongmin/Desktop/tobigs_test/.git/
(base) ❹ master ❺ cd ..
(base) ❻ git clone https://github.com/positivejmk/open-data-analysis-basic.git
Cloning into 'open-data-analysis-basic'...
remote: Enumerating objects: 260, done.
remote: Counting objects: 100% (39/39), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 260 (delta 37), reused 36 (delta 36), pack-reused 221
Receiving objects: 100% (260/260), 42.67 MiB | 411.00 KiB/s, done.
Resolving deltas: 100% (151/151), done.
```

Unit 01 | GIT 기초 - 로컬 저장소

• 깃 상태 확인 git status

- On branch master
: 현재 master 브랜치에 있다는 뜻
- Nothing to commit
: 커밋할 파일이 없다.

* 변경사항 있을 경우 확인 가능

```
(base) ~/desktop/tobigs_test [?] master [?] git status
On branch master

No commits yet

nothing to commit (create/copy files and use "git add" to track)
```

Unit 01 | GIT 기초 - 로컬 저장소

• Working directory → staging area

처음 코드 작성 시 working directory에 위치하는 파일들, 로컬에 올리기 전 대기시키기(staging)



- `git add [파일명]` 실행 시 staging area로 이동함. 원하는 파일들만 인덱스에 등록해 커밋 가능
- `git add .` 사용하면 모든 파일 전체 이동

Unit 01 | GIT 기초 - 로컬 저장소

• Staging area → local repository

로컬 저장소(local repository)에 올리기, 기록하기



- `git commit -m "[커밋 메시지]"` 실행 시 로컬 저장소로 커밋
- 이전 커밋 상태부터 현재 상태까지의 변경 이력이 시간순으로 기록된 커밋(혹은 리비전)이 만들어집니다.
- 커밋은 이력을 남기는 중요한 작업, 커밋메시지는 필수!
- 명료하고 이해하기 쉽게 남겨야 본인 뿐만 아니라 다른 사람이 커밋 이력을 확인하기 쉽습니다.

Unit 01 | GIT 기초 - 로컬 저장소

• 만들어진 커밋들 기록 확인

• git log

실행 시 만들어진 커밋들 기록 확인 가능
(커밋마다 고유 이름이 붙습니다)

• git log --oneline

보다 간략하게 확인할 수 있습니다.
(이름이 보다 간단해집니다)

```
(base) ~/desktop/tobigs_test ? ? master ? git status
On branch master

No commits yet

nothing to commit (create/copy files and use "git add" to track)
```

```
(base) ~/desktop/tobigs_test ? ? master ? git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
  hello.rtf
```

```
(base) * ? ? desktop/tobigs_test ? ? master ? git add .
(base) ~/desktop/tobigs_test ? ? master + ? git commit -m "add: hello message"
[master (root-commit) 27e6269] add: hello message
1 file changed, 8 insertions(+)
create mode 100644 hello.rtf
```

```
((base) ~/desktop/tobigs_test ? ? master ? git log
((base) ~/desktop/tobigs_test ? ? master ? git log --oneline

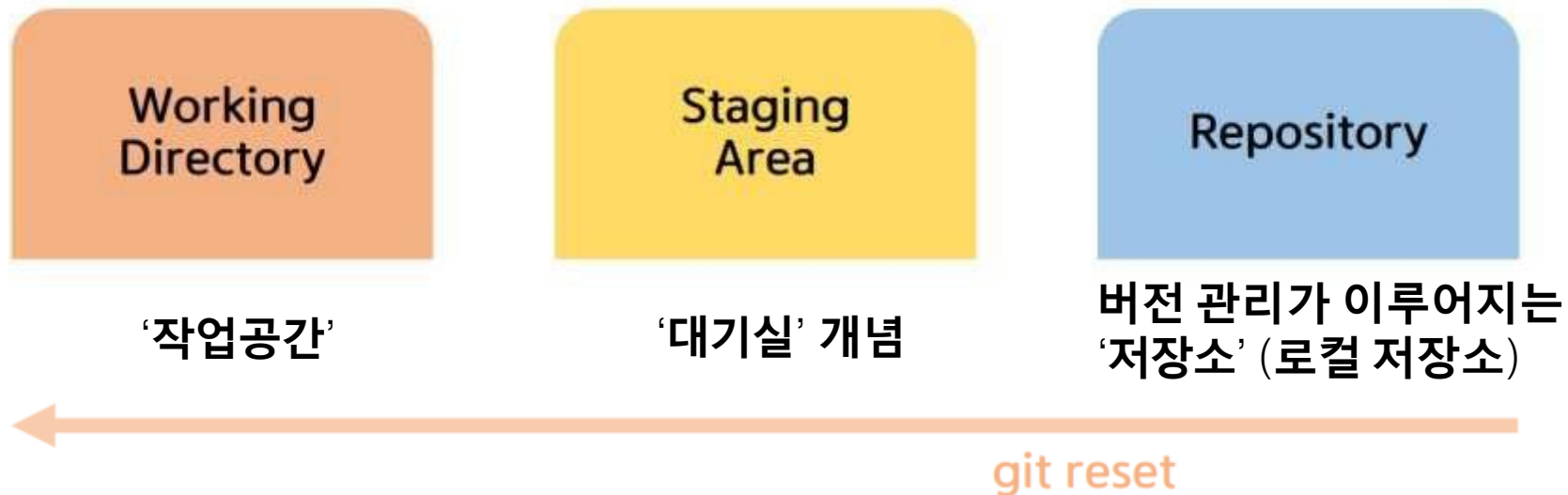
commit 27e6269f9e9fb2a9a99bf443963a9e5273c7ccc (HEAD -> master)
Author: positivejmk <positivejmk@gmail.com>
Date: Tue Jul 19 23:16:13 2022 +0900

    add: hello message

27e6269 (HEAD -> master) add: hello message
(END)
```

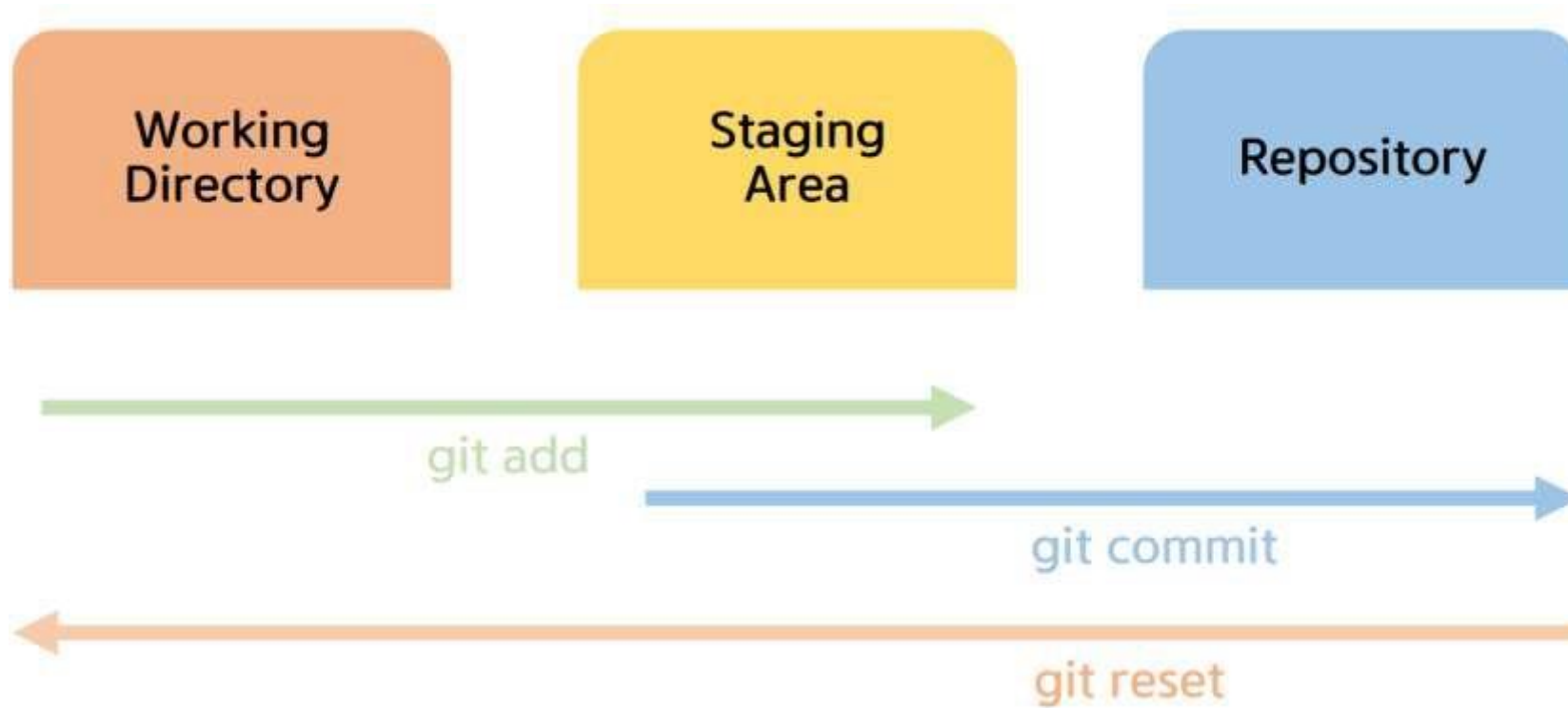
Unit 01 | GIT 기초 - 로컬 저장소

- local repository → working directory



- **git reset** 실행 시 과거 버전으로 되돌아감
- **git reset --hard** 보다 강제로 되돌리기
- **git reset [고유이름]** or **git reset --hard [고유이름]**
git log에서 확인한 커밋의 고유 이름으로, 특정 버전을 지정하여 이동할 수 있음

Unit 01 | GIT 기초 - 로컬 저장소



Unit 02 | GIT 협업 - 원격 저장소



깃허브를 이용해 원격 저장소를 관리해봅시다!

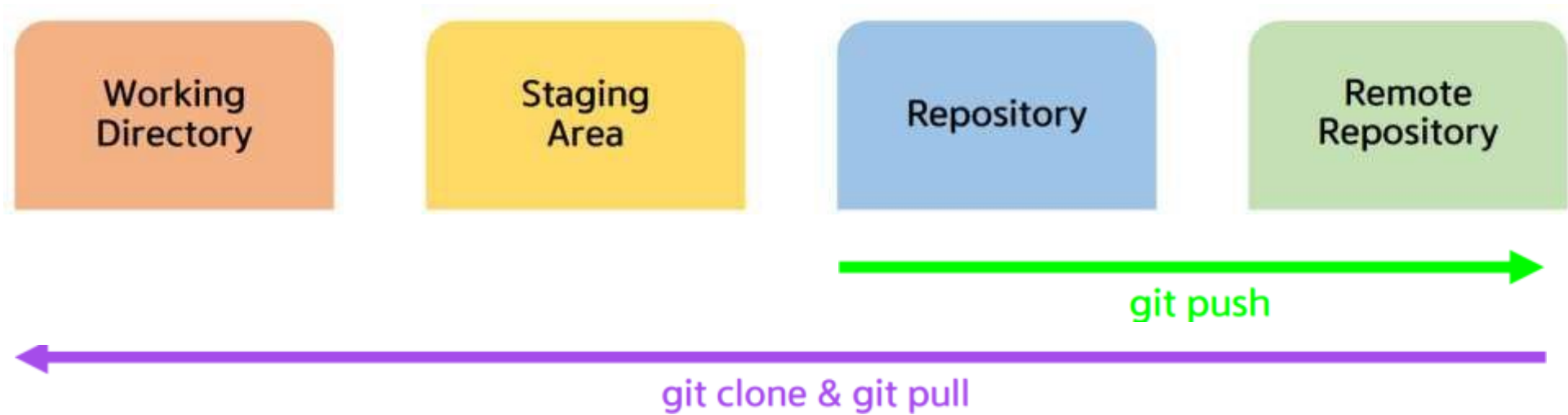
깃허브 (GitHub)는
깃(Git)을 사용하는 프로젝트를 지원하는 웹호스팅 서비스
다시 말해 Git 저장소 웹사이트이며
개발자들의 버전 제어 및 공동 작업을 위한 플랫폼

Bitbucket, Gitlab 등 다른 플랫폼도 존재

Unit 02 | GIT 협업 - 원격 저장소

• 원격 저장소에 로컬 파일 올리기, 내려받기, 충돌 해결하기

- local repository → remote repository
- remote repository → working directory



Unit 02 | GIT 협업 - 원격 저장소

• 내 작업 환경에 원격저장소 등록하기



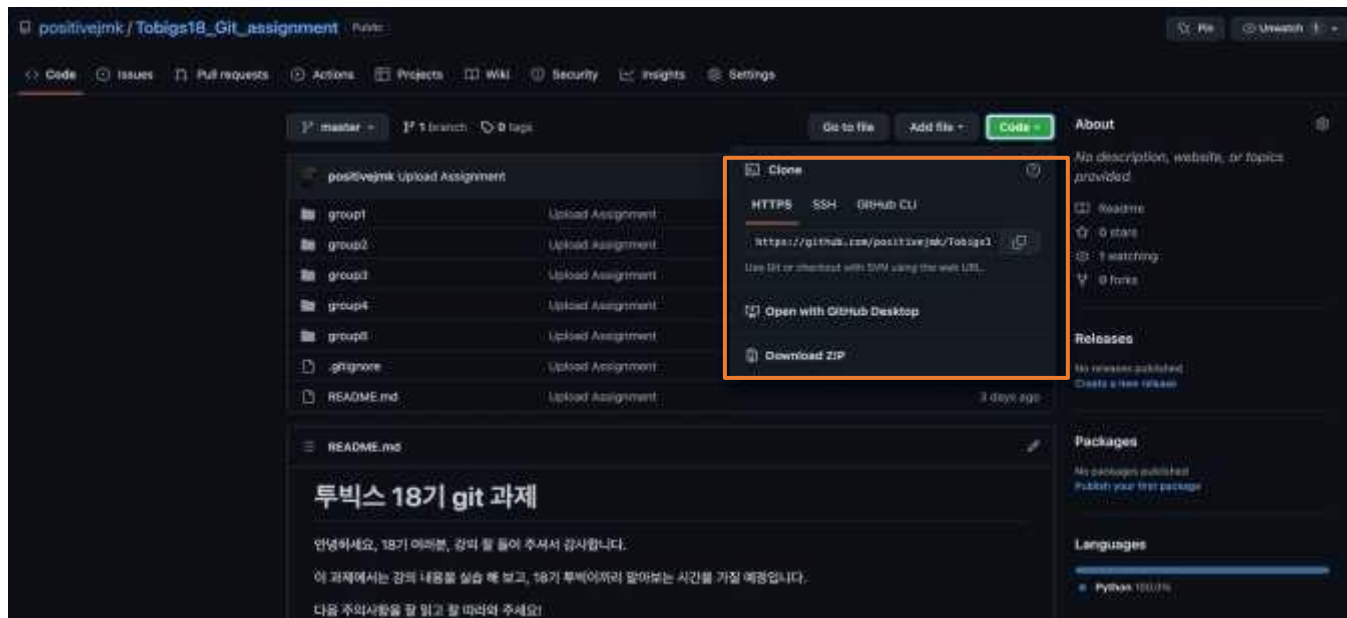
Repository 탭에 있는 저장소 목록에서 새로 만든 저장소로 들어가기

‘code’ 클릭해서 나오는 Git 주소 복사

CMD 창에서 working directory로 이동한 뒤,

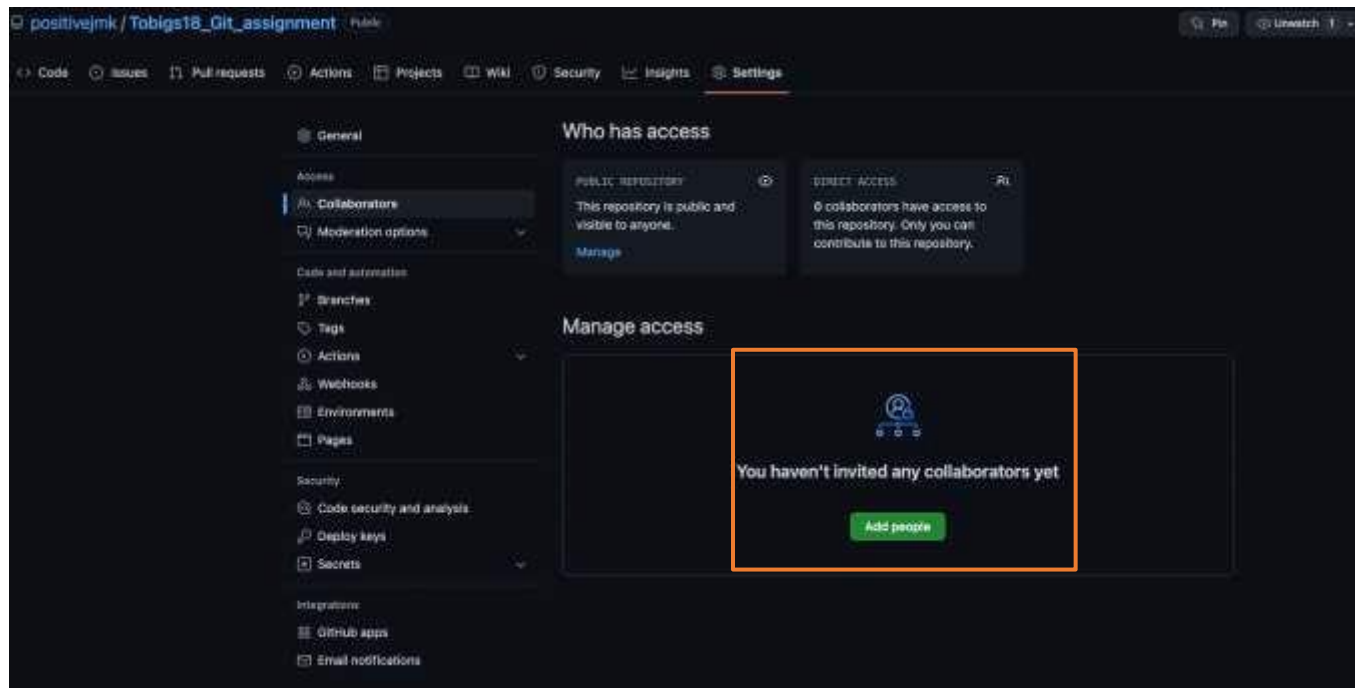
git remote add origin [원격 저장소 주소]

실행해서 내 작업 환경에 github 주소 등록



Unit 02 | GIT 협업 - 원격 저장소

- 해당 저장소에 같이 사용할 collaborator 등록



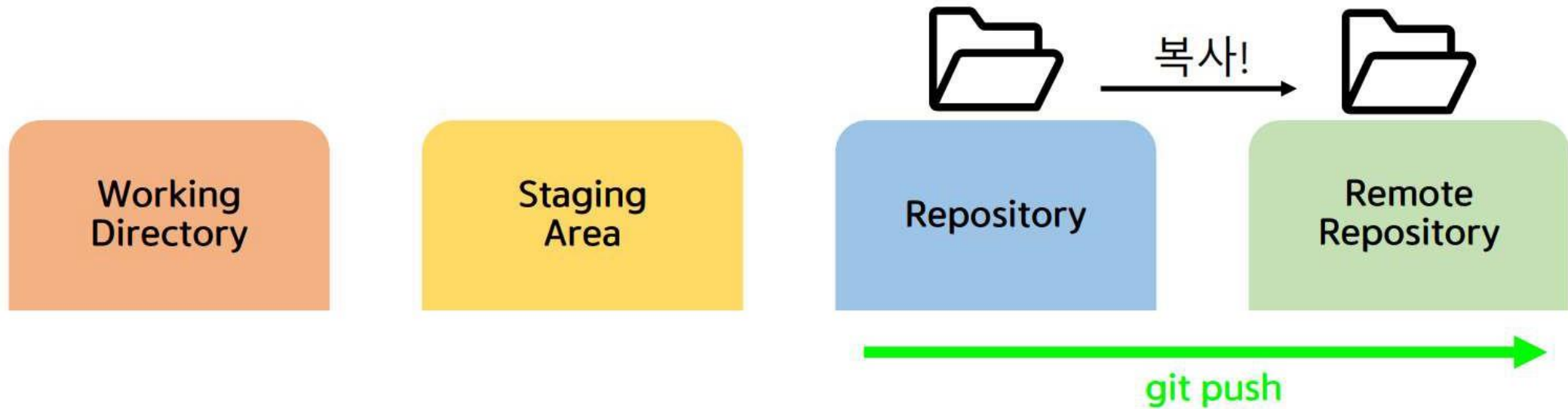
Setting의 collaborators에서
해당 원격저장소를 같이 사용할 사람 등록 가능!

‘Add people’ 버튼 클릭하고
Username, fullname, email 검색해서 등록

Unit 02 | GIT 협업 - 원격 저장소

• local repository → remote repository

내 로컬 저장소에 있는 내용을 원격 저장소 서버로 올리기

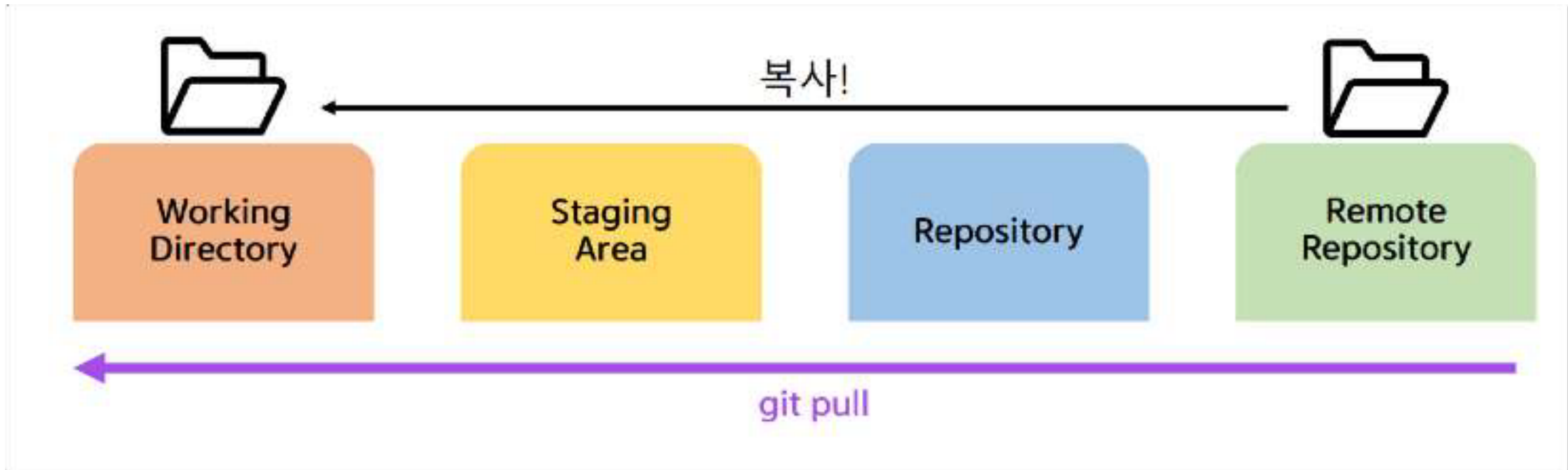


- git push [Remote명] [Branch명] 실행 시 원격 저장소로 푸시
- git push origin main 가 기본

Unit 02 | GIT 협업 - 원격 저장소

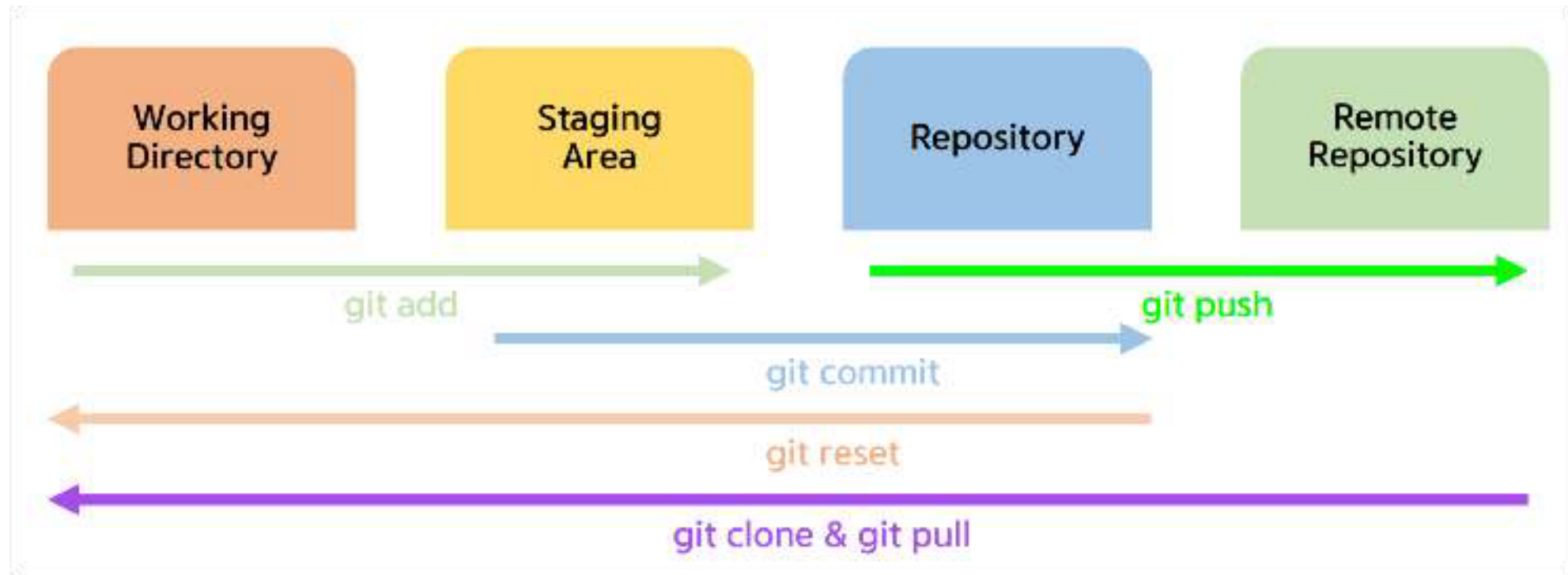
- remote repository → working directory

원격저장소에서 최신 내용 가져와서 내 wd 업데이트



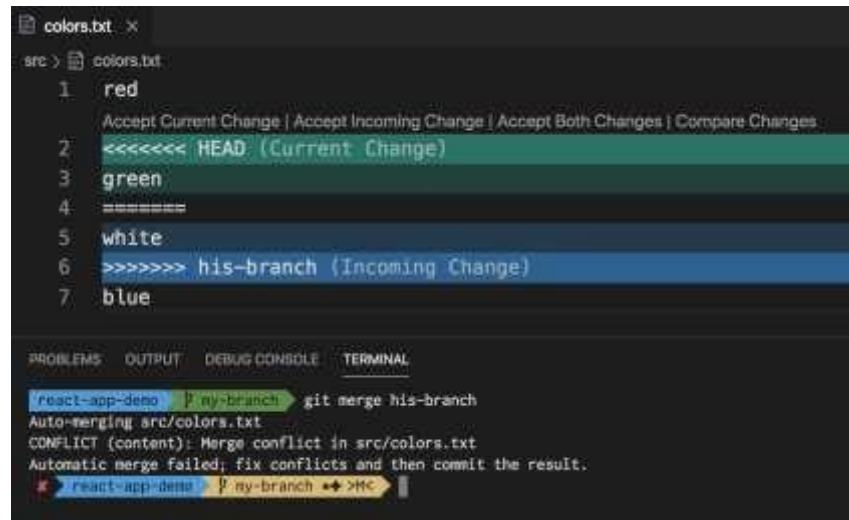
- `git pull [Remote명] [Branch명]` 실행 시 원격 저장소에서 최신 내용 받아오기
- `git pull origin main` 이 기본

Unit 02 | GIT 협업 - 원격 저장소



Unit 02 | GIT 협업 - 원격 저장소

- merge conflict
- commit & push 하려고 하는데, 내가 수정한 부분이 원격 저장소에서도 동시에 수정됐을 때 발생하는 오류.
- 해당 원격 저장소를 다시 pull 해서 업데이트 해보면,
- 충돌이 발생한 파일에 세부적인 내용이 표시됨.
- 겹치는 부분 중 어느 부분 남길지 선택,
(내 수정본/새로운 수정본/둘 다...).
- 수동으로 수정하여 원하는 부분만 남긴 후 다시 commit & push!



```
colors.txt x
src > colors.txt
1 red
   Accept Current Change | Accept Incoming Change | Accept Both Changes | Compare Changes
2 <<<<<< HEAD (Current Change)
3 green
4 =====
5 white
6 >>>>>> his-branch (Incoming Change)
7 blue

PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL
/react-app-demo P ny-branch git merge his-branch
Auto-merging src/colors.txt
CONFLICT (content): Merge conflict in src/colors.txt
Automatic merge failed; fix conflicts and then commit the result.
* > react-app-demo P ny-branch <<>> HMC
```

Unit 02 | GIT 협업 - 원격 저장소

+ 심화 : Collaborator 이외의 원격 저장소에 대한 협업

- 협업하고 싶은 원격저장소를 Fork 해서 복제한 후 작업

나와 내 collaborators가 작업한 내용을 해당 저장소에 반영시킬 수 있습니다.

- 합쳐 달라고(merge해달라고) 요청하면 (pull request하면)

- 해당 저장소의 주인이 Collaborators
Conflict확인 후 merge시켜 해당 저장소에 반영됩니다.

Unit 03 | 실습 및 과제

앞으로 정규세션 대부분의 과제는 깃허브 링크를 통해 제출합니다.

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere?
[Import a repository.](#)

Owner * / Repository name *

yxbxn / Tobigs19 ✓

Great repository names are short and memorable. Need inspiration? How about [fuzzy-guacamole?](#)

Description (optional)

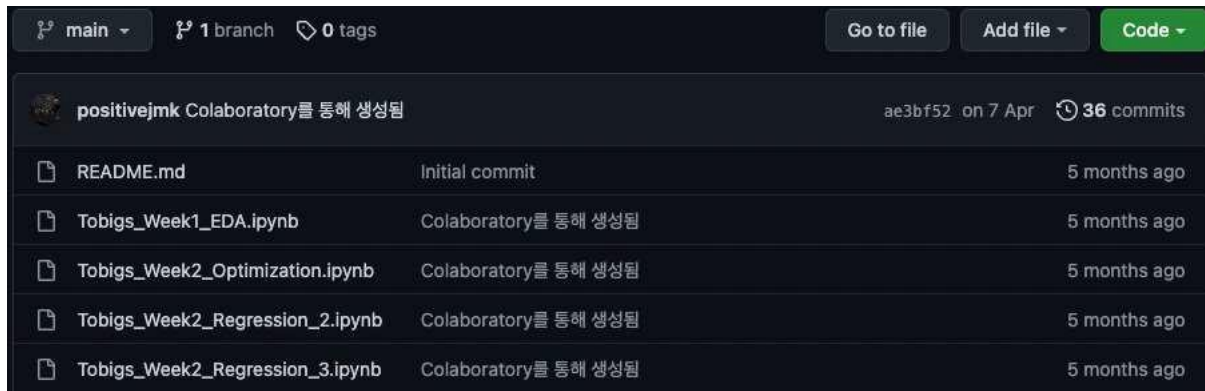
☒ Public
Anyone on the internet can see this repository. You choose who can commit.

☐ Private
You choose who can see and commit to this repository.

투빅스 과제제출 전용 레포를 하나 만들어주세요.

collaborators에 'tobigs-datamarket'
투빅스 공식 계정을 등록해주세요 (Private 생성시 필수)

Unit 03 | 실습 및 과제



과제를 push!



push 한 과제 링크 복사

Unit 03 | 실습 및 과제

트위크스 빅데이터강의 커뮤니티 세미나 **멤버게시판** Facebook 검색 | 내정보 | 로그아웃

공지사항
강의자료
과제게시판
스터디 게시판

본문화장 놀이속소 놀이화장

멤버게시판 Home > 멤버게시판 > 과제게시판

공지사항
강의자료
과제게시판
스터디 게시판
+ sns스터디

번호	제목	글쓴이	날짜	조회 수
3931	이정우(18기) - 10주차 과제 [GAN_adv]	이정우	2022.10.08	1
3930	김성훈(18기) - 10주차 과제 [GAN_adv]	tamsasu	2022.10.05	1
3929	손유진(18기) - 10주차 과제 [GAN_adv]	손유진	2022.10.04	2
3928	이기정(18기) - 10주차 과제 [GAN_adv]	이기정	2022.10.04	1
3927	김정우(18기) - 10주차 과제 [GAN_adv]	김정우	2022.10.04	1
3926	장호준(18기) - 10주차 과제 [GAN_adv]	장호준	2022.10.03	2
3925	고나경(18기) - 10주차 과제 [GAN_adv]	고나경	2022.10.03	0
3924	이선민(18기) - 10주차 과제 [GAN_adv]	sun	2022.10.03	2
3923	지은형(18기) - 10주차 과제 [GAN_adv]	지은형	2022.10.03	1
3922	배유진(18기) - 10주차 과제 [GAN_adv]	배유진	2022.10.03	1
3921	권소영(18기) - 10주차 과제 [GAN_adv]	여기갑자함	2022.10.03	1
3920	김희경(18기) - 10주차 과제 [GAN_adv]	김희경	2022.10.02	1
3919	손유진(18기) - 10주차 과제 [RS_adv]	손유진	2022.09.29	2
3918	김희경(18기) - 10주차 과제 [RS_adv]	김희경	2022.09.29	0
3917	이기정(18기) - 10주차 과제 [RS_adv]	이기정	2022.09.29	0
3916	김정우(18기) - 10주차 과제 [RS_adv]	김정우	2022.09.28	1

데이터 마켓 로그인 후
멤버게시판 → 과제게시판

Unit 03 | 실습 및 과제

본문확장 높이축소 넓이확장

멤버 게시판

공지사항

강의자료

과제게시판

스터디 게시판

▶ sas스터디

Home > 멤버 게시판 > 과제게시판

김유빈(18기) - n주차 과제 [강의명]

블러오기

← → 형식 글꼴 12px 줄간격 B U I S C Xⁿ X_n [List Icons] URL ※

확장 컴포넌트 HTML 편집기

https://github.com/yxbxn/tobigs18/blob/4190ca83f22eb4aaefc0e53aa2a83d8388c60e2a/Week1/Week1_EDA.ipynb

Unit 03 | 실습 및 과제

깃 과제 설명

https://github.com/dlduddk/Tobigs_20_Git_assignment

팀끼리 깃허브 버전관리를 익혀보자!

1. 팀 확인
2. 조장은 해당 레포를 fork + collaborators에 조원 초대
3. 조장의 fork한 레포를 모두 clone
4. 자신이 해당하는 조의 폴더에서 지시 내용에 따라 add, commit, push를 통해 과제를 수행
5. 팀장은 모든 조원이 과제를 수행하면 확인하여 마무리

이번 주차 EDA & GIT 과제 – 2가지

EDA

- 파이썬을 이용하여 전처리 및 EDA를 진행해주세요.
 - 결측치, 이상치 검토
 - 유의미한 시각화 5개 이상
 - 수치형 변수 간 상관관계 파악
 - 파생변수 생성

GIT

- 팀끼리 깃허브 버전관리를 익혀보자!
 1. 팀 확인
 2. 조장은 해당 레포를 fork + collaborators에 조원 초대
 3. 조장의 fork한 레포를 모두 clone
 4. 자신이 해당하는 조의 폴더에서 지시 내용에 따라 add, commit, push를 통해 과제를 수행
 5. 팀장은 모든 조원이 과제를 수행하면 확인하여 마무리

EDA 과제는 앞에서 말씀드린 것처럼 데이터 마켓에 올려주시면 됩니다.
이번 주 과제하시면서 어려움이 있으시면 19기 이영아에게 연락주세요. 20기 화이팅!

참고자료

투빅스 18기 김유빈님 1주차 EDA & GIT 강의자료

투빅스 17기 김정민님 1주차 EDA & GIT 강의자료

투빅스 16기 정수연님 1주차 EDA 강의자료

투빅스 14기 이해린님 1주차 EDA 강의자료 강명욱 외 7인, '통계학입문' 도서

<https://wikidocs.net/47193> <https://jjeongil.tistory.com/352>

<https://webnautes.tistory.com/1670>

[IBM HR 데이터 관련 포스팅](#)

투빅스 16기 김송민님 2주차 GIT & Framework 강의자료

(참고) 생활코딩 깃 강의 <https://opentutorials.org/course/3837> → 깃을 처음 접하신 분들이라면 이 강의 추천드려요!

(참고) 깃 브랜치 관리 <https://velog.io/@kw2577/Git-branch-%EC%A0%84%EB%9E%B5>

Q & A

들어주셔서 감사합니다.