



판다스 데이터프레임

인공지능 플랫폼 설계 - 11주차

학습 내용

1. 데이터프레임 다루기 2
2. 데이터베이스 사용



Python 프로그래밍

_Python 데이터프레임

행 중심 입력

```
col_names = ['과목번호', '과목명', '강의실', '시간수']
list1 = [['C1', '인공지능개론', 'R1', 3],
          ['C2', '웃음치료', 'R2', 2],
          ['C3', '경영학', 'R3', 3],
          ['C4', '3D디자인', 'R4', 4],
          ['C5', '스포츠경영', 'R2', 2],
          ['C6', '예술의 세계', 'R3', 1]
])

df = pd.DataFrame(list1, columns=col_names)
```

| 과목번호 | 과목명 | 강의실 | 시간수 |
|------|--------|-----|-----|
| C1 | 인공지능개론 | R1 | 3 |
| C2 | 웃음치료 | R2 | 2 |
| C3 | 경영학 | R3 | 3 |
| C4 | 3D디자인 | R4 | 4 |
| C5 | 스포츠경영 | R2 | 2 |
| C6 | 예술의 세계 | R3 | 1 |

열 중심 입력

```
data = {
    '과목번호' : ['C1', 'C2', 'C3', 'C4', 'C5', 'C6'],
    '과목명' : ['인공지능개론', '웃음치료', '경영학', '3D디자인', '스포츠경영', '예술의 세계'],
    '강의실' : ['R1', 'R2', 'R3', 'R4', 'R5', 'R6'],
    '시간수' : [3, 2, 3, 4, 2, 1]
}

df = pd.DataFrame(data)
```



_데이터 프레임 추가/삭제

- 열 추가

- `df[새 열이름] = [값1, 값2 ...]`

- 행추가

- `df.loc[new인덱스] = ([값1, 값2 ...])`

- 1열/행(시리즈) 삭제

- 열 삭제: `df.drop([열이름], axis=1)`

- 행 삭제: `df.drop([인덱스이름], axis=0)`

- * axis의 1은 열, 0은 행을 의미

- 여러 열 또는 행을 삭제

- 열 삭제: `df.drop([열이름1, 열이름2], axis=1)`

- 행 삭제: `df.drop([인덱스1, 인덱스2])`

| | 과목번호 | 과목명 | 강의실 | 시간수 | |
|---|------|--------|-----|-----|--|
| 0 | C1 | 인공지능개론 | R1 | 3 | |
| 1 | C2 | 웃음치료 | R2 | 2 | |
| 2 | C3 | 경영학 | R3 | 3 | |
| 3 | C4 | 3D디자인 | R4 | 4 | |
| 4 | C5 | 스포츠경영 | R5 | 2 | |
| 5 | C6 | 예술의세계 | R6 | 1 | |
| | | | | | |

```
df['담당교수'] = ['홍길동', '김철수', '이영희', '박영수',  
                 '최영희', '김영수']
```

```
df.loc[6] = ['C7', '통계학', 'R7', 3, '이철수']
```

```
df1 = df.drop(['강의실'], axis=1)
```

```
df2 = df.drop([5], axis=0)
```



_데이터 필터 (filtering)

- 1개의 열, 1개의 행, 1개의 셀

- 1개의 열 : `df[열이름]`

- 1개의 행 : `df.loc[인덱스이름]`

- 1개의 셀 : `df.loc[인덱스이름, 열이름]`

- 범위로 지정 → 필터링 방법

- . **[시작번호 : 끝번호]** → ex) `[0:3]`

- `df.loc[0:2]`

- . **[['이름1', '이름2' ...]]** → ex) `[['이름', '나이']]`

- `df[['과목명', '담당교수']]`

- 검색으로 지정하는 방법 → **`df.loc[행조건, 열이름]`**

- . 행 찾기 : `df.loc[df['이름'] == '은지']`

- . 셀 찾기 : `df.loc[df['이름'] == '은지', '생일']`

- `df.loc[df['이름'] == '은지', '생일'].values[0]`

| | 과목번호 | 과목명 | 강의실 | 시간수 | 담당교수 |
|---|------|--------|-----|-----|------|
| 0 | C1 | 인공지능개론 | R1 | 3 | 홍길동 |
| 1 | C2 | 웃음치료 | R2 | 2 | 김철수 |
| 2 | C3 | 경영학 | R3 | 3 | 이영희 |
| 3 | C4 | 3D디자인 | R4 | 4 | 박영수 |
| 4 | C5 | 스포츠경영 | R5 | 2 | 최영희 |
| 5 | C6 | 예술의세계 | R6 | 1 | 김영수 |
| 6 | C7 | 통계학 | R7 | 3 | 이철수 |

`df.loc[0:2]`

`df[['과목명', '담당교수']]`

`df['과목명'] == '경영학'`

`df.loc[df['과목명'] == '경영학']`

`df.loc[df['시간수'] > 2]`

`df.loc[df['과목명'] == '경영학']['담당교수']`

`df.loc[df['과목명'] == '경영학']['담당교수'].values[0]`

`df.loc[df['과목명'] == '경영학', '담당교수']`

`df.loc[df['과목명'] == '경영학', '담당교수'].values[0]`



_데이터 프레임 - 셀데이터 수정

- 셀 수정 : 셀에 대해 대입연산자 사용
`df.loc[인덱스이름, 컬럼이름] = 새 값`
* 형식에 주의

| | 과목번호 | 과목명 | 강의실 | 시간수 | 담당교수 |
|---|------|--------|-----|-----|------|
| 0 | C1 | 인공지능개론 | R1 | 3 | 홍길동 |
| 1 | C2 | 웃음치료 | R2 | 2 | 김철수 |
| 2 | C3 | 경영학 | R3 | 3 | 이영희 |
| 3 | C4 | 3D디자인 | R4 | 4 | 박영수 |
| 4 | C5 | 스포츠경영 | R5 | 2 | 최영희 |
| 5 | C6 | 예술의세계 | R6 | 1 | 김영수 |
| 6 | C7 | 통계학 | R7 | 3 | 이철수 |

- 검색으로 셀을 찾아서 수정

`df.loc[df['과목명'] == '경영학', '담당교수'] = '이경영'`

```
df.loc[3, '담당교수'] = '이경영'  
df.loc[3]['담당교수'] = '이경영'  
print(df, end='\n\n')
```

```
df.loc[df['과목명'] == '경영학', '담당교수'] = '이경영'  
print(df, end='\n\n')
```



_Python 실습

```
data = {'이름' : ['Kim', 'Park', 'Lee', 'Ho'],  
        '국어' : [90, 58, 88, 100],  
        '영어' : [100, 60, 80, 70],  
        '수학' : [55, 65, 76, 88] }
```

| | 이름 | 국어 | 영어 | 수학 |
|---|------|-----|-----|----|
| 0 | Kim | 90 | 100 | 55 |
| 1 | Park | 58 | 60 | 65 |
| 2 | Lee | 88 | 80 | 76 |
| 3 | Ho | 100 | 70 | 88 |

1. data를 데이터 프레임으로 만드시오
2. 1에서 만든 데이터 프레임을 출력하시오
3. 학생 이름만 추출해서 출력하시오 (열 추출)
4. 'Park' 성적만 출력하시오
5. 'Ho' 학생의 수학점수를 90점으로 수정하시오
6. 'Oh' 학생의 국어(100), 영어(70), 수학(80) 성적을 새로 추가하시오
7. 'Lee' 학생의 성적을 삭제하시오.



IT 플랫폼 실습

_MariaDB 실행

- 최초 접속

mysql -u 자기계정 -p

Enter password:

[비번 입력] → !ai123

- 데이터베이스 선택

show databases ;

use edudb ;

- 테이블 리스트 보기, 쿼리 실행

show tables ;

select * from city ;

quit ;



_기본 쿼리 실습

- SELECT는 테이블의 데이터를 읽어 출력
- **SELECT** 필드목록
FROM 테이블
WHERE 조건
[ORDER BY 정렬기준]
- SELECT와 FROM 사이의 필드 목록에 출력할 필드의 이름을 지정하되 * 기호는 모든 필드 출력
 - SELECT * FROM city ;
 - SELECT Name, Population FROM city ;
- WHERE 절은 읽을 레코드의 조건을 지정한다.
 - 필드와 특정값을 비교하는 조건문 형식으로 작성한다.
 - SELECT * FROM city WHERE Population > 700000;
 - 문자열이나 날짜는 작은 따옴표로 감싼다.
- **AND**는 두 조건이 모두 참인 레코드를 검색하며 **OR**는 두 조건 중 하나라도 참인 레코드를 검사
 - SELECT * FROM city WHERE Population >= 100000 AND Population >= 700000;



_기본 쿼리 실습

이 쿼리는 어떤 데이터를 얻기 위한 쿼리인가?

```
SELECT * FROM city WHERE Population > 700000;
```

```
SELECT Name, Population FROM city WHERE Population > 700000;
```

```
SELECT * FROM city WHERE Population >= 100000 AND Population <= 700000 ;
```

```
SELECT * FROM city WHERE name LIKE 'Over%';
```

```
SELECT * FROM city ORDER BY population DESC LIMIT 4;
```



_기본 쿼리 실습

쿼리 문제 : 다음 조건을 검색할 수 있는 쿼리를 만드시오

2. 'countrylanguage' 테이블에서 언어(Language 필드)를 'Dutch'로 쓰는 나라의 국가코드(CountryCode 필드)를 검색하시오.
3. 'country' 테이블에서 'Australia'(Name 필드)의 GNP(GNP 필드)가 얼마인지 찾으시오.
4. 'city' 테이블에서 'Seoul' ('Name' 필드)의 인구(Population 필드)가 얼마인지 검색하시오.



_기본 쿼리 실습

쿼리 문제 : 다음 조건을 검색할 수 있는 쿼리를 만드시오

2. 'countrylanguage' 테이블에서 언어(Language 필드)를 'Dutch'로 쓰는 나라의 국가코드(CountryCode 필드)를 검색하시오.

→ `SELECT CountryCode FROM countrylanguage
WHERE Language='Dutch' ;`

3. 'country' 테이블에서 'Australia'(Name 필드)의 GNP(GNP 필드)가 얼마인지 찾으시오.

→ `SELECT GNP FROM country
WHERE Name='Australia' ;`

4. 'city' 테이블에서 'Seoul' ('Name' 필드)의 인구(Population 필드)가 얼마인지 검색하시오.

→ `SELECT Population FROM city
WHERE Name='Seoul' ;`






_HeidiSQL 설치


- HeidiSQL 설치


https://www.heidisql.com/download.php#google_vignette

Download HeidiSQL 12.6, released on 05 Nov 2023

 **Installer, 32/64 bit combined** 

 Portable version (zipped): 32 bit ≈ 64 bit ≈

 [Sourcecode](#)

 [Previous releases](#)



Compatibility notes

- HeidiSQL runs fine on Windows 10 and 11 (and on Windows 7 + 8 with some minor issues)
- Running HeidiSQL on Wine can have various issues, depending on the version of Wine and HeidiSQL.

Nightly Builds of heidisql.exe

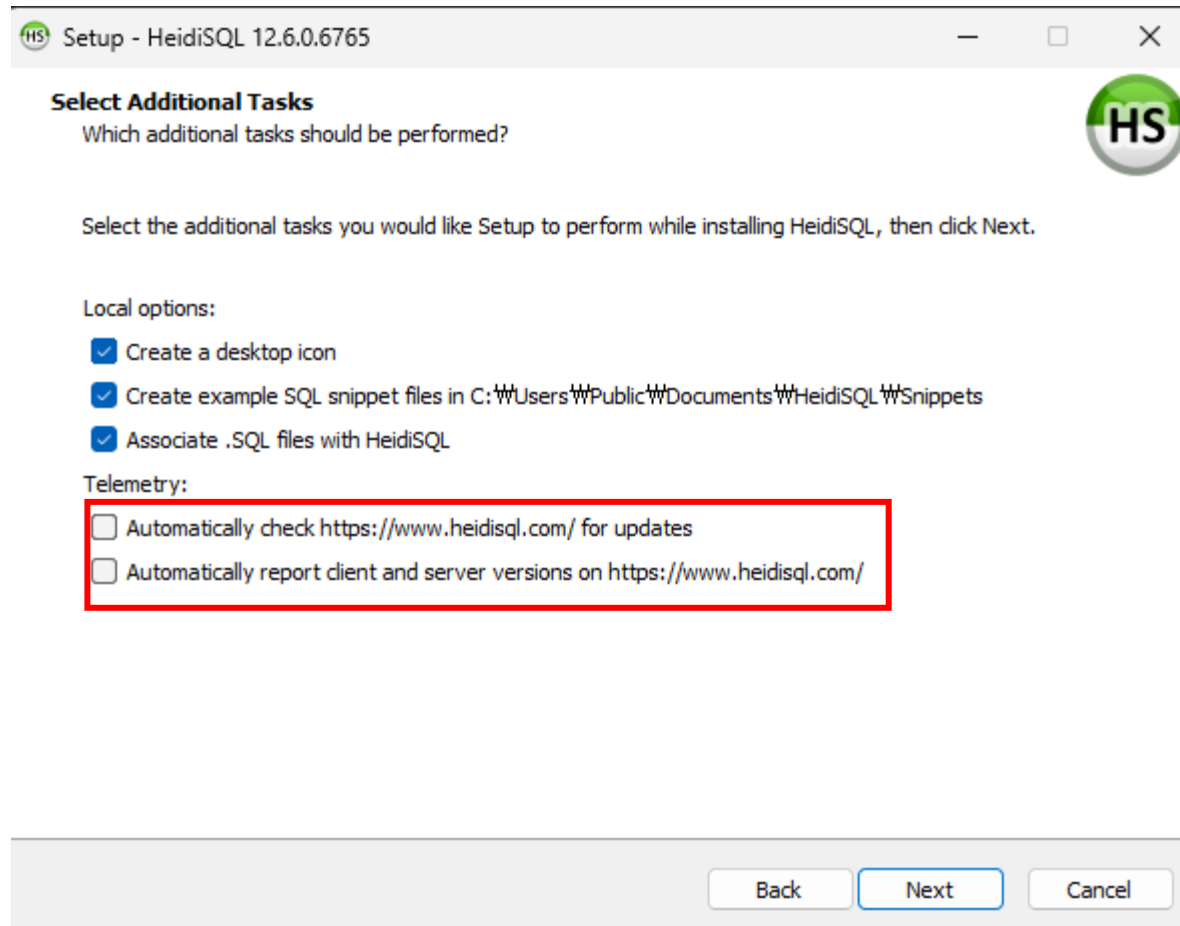
Latest automatically compiled main executable and installers for HeidiSQL. Just download and overwrite your existing `C:\Program Files\HeidiSQL\heidisql.exe`. Be aware that these builds are not official releases and therefore probably have more bugs, possibly serious ones.

| | | |
|-------------|---------------------------------------------------------------------------------------------------------------------|-----------------------------|
| Revision | <i>Hopefully fix slow loading tabs. Related to commit 1d1d1a493892eba9152a868bc303a11c1826af74.</i> | |
| 12.6.0.6813 | committed: 13 Mar 2024 15:37 | compiled: 13 Mar 2024 16:25 |
| | 64bit build 22.6 MB ≈ | |

| | | |
|-------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------|
| Revision | <i>Host tab: use session name as tab caption, and show host name as mouse-over hint. See https://www.heidisql.com/forum.php?t=41399</i> | |
| 12.6.0.6812 | committed: 09 Mar 2024 16:48 | compiled: 09 Mar 2024 17:25 |
| | 64bit build 22.6 MB ≈ | |

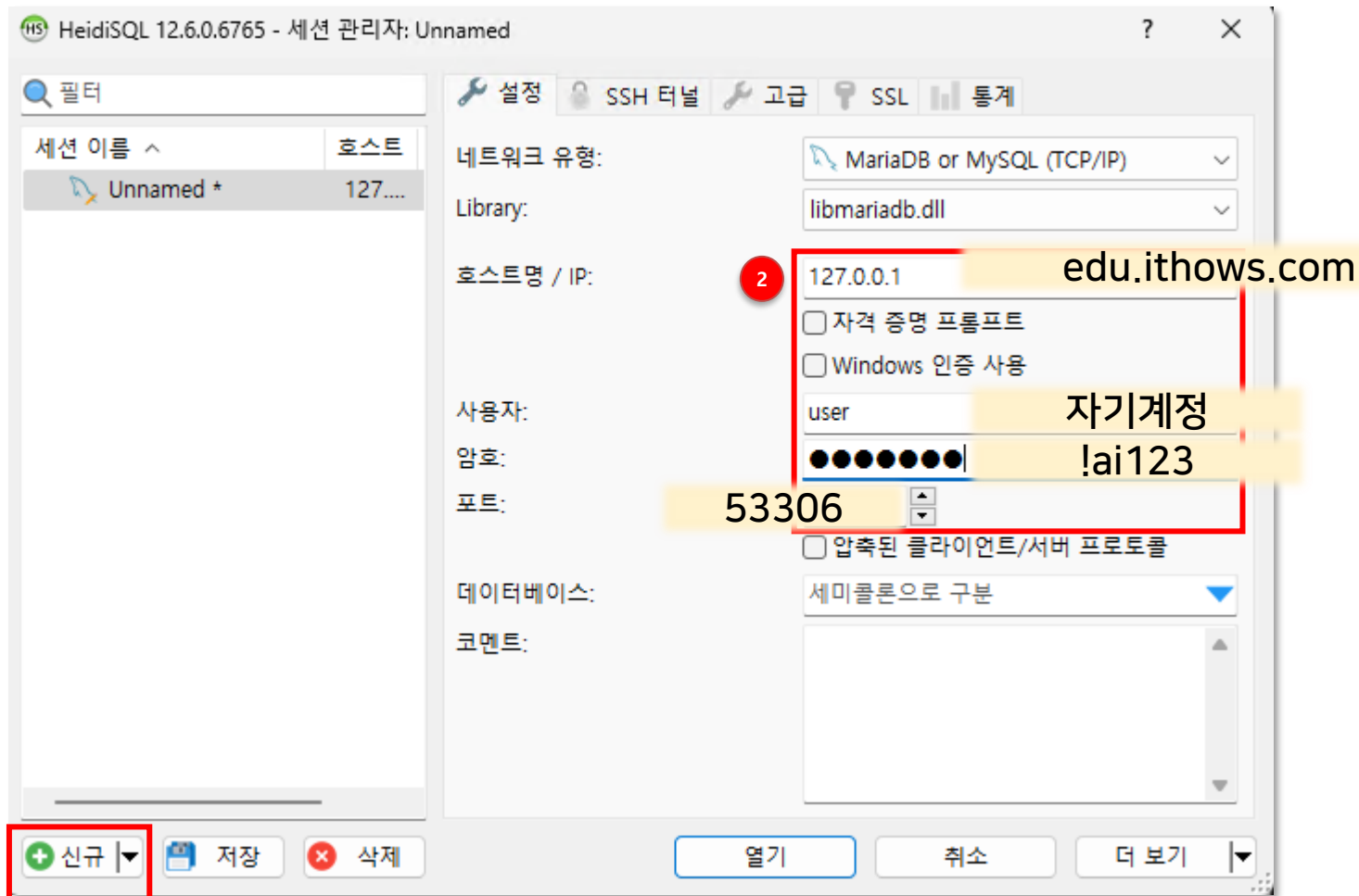


_HeidiSQL 설치



_HeidiSQL 설치

접속 확인



_HeidiSQL 설치

edu\edudb\ - HeidiSQL 12.6.0.6765

파일 편집 검색 쿼리 도구 이동 도움말

데이터베이스 필터 테이블 필터

호스트: edu.ithows.com 데이터베이스: edudb 쿼리*

edu

- edudb 784.0 KIB
 - city 528.0 KIB
 - country 96.0 KIB
 - countrylanguage 160.0 KIB
- information_schema
- mysql
- performance_schema

```
1 SELECT *
2 FROM city;
3
```

city (4,080r x 5c)

| # | ID | Name | CountryCode | District |
|----|----|----------------|-------------|---------------|
| 1 | 1 | Kabul | AFG | Kabul |
| 2 | 2 | Qandahar | AFG | Qandahar |
| 3 | 3 | Herat | AFG | Herat |
| 4 | 4 | Mazar-e-Sharif | AFG | Balkh |
| 5 | 5 | Amsterdam | NLD | Noord-Holland |
| 6 | 6 | Rotterdam | NLD | Zuid-Holland |
| 7 | 7 | Haag | NLD | Zuid-Holland |
| 8 | 8 | Utrecht | NLD | Utrecht |
| 9 | 9 | Eindhoven | NLD | Noord-Brabant |
| 10 | 10 | Tilburg | NLD | Noord-Brabant |
| 11 | 11 | Groningen | NLD | Groningen |
| 12 | 12 | Breda | NLD | Noord-Brabant |
| 13 | 13 | Apeldoorn | NLD | Gelderland |
| 14 | 14 | Nijmegen | NLD | Gelderland |
| 15 | 15 | Enschede | NLD | Overijssel |
| 16 | 16 | Haarlem | NLD | Noord-Holland |



_MariaDB 활용

edu\edudb\ - HeidiSQL 12.6.0.6765

파일 편집 검색 쿼리 도구 이동 도움말

데이터베이스 필터 테이블 필터

호스트: edu.ithows.com 데이터베이스: edudb 쿼리+

edu

- edudb 784.0 KiB
 - city 528.0 KiB
 - country 96.0 KiB
 - countrylanguage 160.0 KiB
- information_schema
- mysql
- performance_schema

```
SELECT *  
FROM city;
```

city (4,080r × 5c)

| # | ID | Name | CountryCode | District |
|----|----|----------------|-------------|---------------|
| 1 | 1 | Kabul | AFG | Kabul |
| 2 | 2 | Qandahar | AFG | Qandahar |
| 3 | 3 | Herat | AFG | Herat |
| 4 | 4 | Mazar-e-Sharif | AFG | Balkh |
| 5 | 5 | Amsterdam | NLD | Noord-Holland |
| 6 | 6 | Rotterdam | NLD | Zuid-Holland |
| 7 | 7 | Haag | NLD | Zuid-Holland |
| 8 | 8 | Utrecht | NLD | Utrecht |
| 9 | 9 | Eindhoven | NLD | Noord-Brabant |
| 10 | 10 | Tilburg | NLD | Noord-Brabant |
| 11 | 11 | Groningen | NLD | Groningen |
| 12 | 12 | Breda | NLD | Noord-Brabant |
| 13 | 13 | Apeldoorn | NLD | Gelderland |
| 14 | 14 | Nijmegen | NLD | Gelderland |
| 15 | 15 | Enschede | NLD | Overijssel |
| 16 | 16 | Haarlem | NLD | Noord-Holland |

