

대학교 기숙사 가격에 영향을 미치는 요인 분석

*CHCOPYTHON FINAL TEAM
PROJECT*



TEAM MEMBER

김동구 김서영 이예진 이준규 정성민

2018.08.03

C O N T E N T S

- 01 Select the Subject
- 02 Set the Sample & Variable
- 03 Data Collecting
- 04 Data Processing
- 05 Data Analysis



01

CHOCHPYTHON PRJECT

Select the Subject

01 Select the Subject



02

CHOCHPYTHON PROJECT

Set the Sample & Variable

02 Set the Sample & Variable

- 데이터 표본 : 2인실 기숙사를 보유한 서울 및 수도권 지역 4년제 대학교 47개

가천대학교	덕성여자대학교	서울시립대학교	송실대학교	한국외국어대학교
가톨릭대학교	동국대학교	서울신학대학교	아주대학교	한국항공대학교
강남대학교	동덕여자대학교	서울여자대학교	연세대학교	한성대학교
건국대학교	명지대학교(자연캠퍼스)	성결대학교	을지대학교	한세대학교
경기대학교	명지대학교(인문캠퍼스)	성공회대학교	이화여자대학교	한양대학교
경희대학교	상명대학교	성균관대학교	중앙대학교	한양대학교(ERICA)
고려대학교	서강대학교	성신여자대학교	중앙대학교(안성)	홍익대학교
국민대학교	서경대학교	세종대학교	평택대학교	
단국대학교	서울과학기술대학교	수원대학교	한경대학교	
대진대학교	서울대학교	숙명여자대학교	한국산업기술대학교	

- 변수 설정 :
독립변수 x = 대학교 별 재학생 수(student) & 기숙사 수용가능 인원(accept) & 등록금(tuition) & 자취방 시세(room_total)
종속변수 y = 대학교 기숙사 가격

* 모든 가격의 단위는 1000원, 기숙사 비의 단위 1개월, 인원 단위 1명
* 자취방 시세 = (보증금 * 0.01 + 월세 값) ex) 보증금 1000, 월세 50 -> 60

- 가설 설정 : H_0 = “종속변수 y 는 독립변수 x 와 상관관계가 있다.”

03

CHOCHPYTHON PRJECT

Data Collecting

03 Data Collecting

대학 알림이에서 대학교 별 재학생 수(student), 기숙사 수용가능 인원(accept), 기숙사 비용(dormi2), 등록금(tuition) 에 대한 전국 대학교 정보 수집

2017_전국 대학교_기숙사운영 - 읽기 전용 - 호환성 모드 - Excel

파일 홈 삽입 페이지 레이아웃 수식 데이터 검토 보기 도움말 어떤 작업을 원하시나요?

자동 저장 (ON) | 클립보드 | 글꼴 | 맞춤 | 표시 형식 | 스타일 | 셀 | 편집

가운데 고딕 10 | 텍스트 줄 바꿈 | 일반 | 조건부 서식 | 표 | 셀 | 삽입 | 삭제 | 서식 | 자동 합계 | 채우기 | 지우기 | 정렬 및 필터 | 찾기 및 선택

A1 | 기숙사 수용 현황_대학]

	F	G	H	I	J	K	L	M	N	O
	학교	재학생수(A)	실수	수용가능인원(B)	기숙사수용률 (C=B/Ax100)	기숙사 지원자 수(D)	입사 경쟁률 (E=D/B)	1인실 기숙사비	2인실 기숙사비	3인실 기숙사비
4	가야대학교(고령)_제2캠퍼스	0	0	0	0	0	0	0	0	
5	가야대학교(김해)_본교	2,584	168	672	26	588	0.9	0	0	
6	가천대학교_본교	19,452	841	2,218	11.4	2,641	1.2	460	286	
7	가톨릭관동대학교_본교	8,518	758	2,970	34.9	4,843	1.6	0	132	
8	가톨릭대학교_본교	7,441	345	1,076	14.5	1,261	1.2	0	323	
9	가톨릭대학교_제2캠퍼스	1,602	41	122	7.6	179	1.5	0	229	
10	가톨릭대학교_제3캠퍼스	263	150	150	57	82	0.5	617	0	
11	감리교신학대학교_본교	1,403	150	300	21.4	382	1.3	0	114	
12	강남대학교_본교	6,931	352	934	13.5	954	1	0	298	
13	강릉원주대학교_본교	6,147	1,098	2,189	35.6	2,765	1.3	182	158	
14	강릉원주대학교_제2캠퍼스	2,116	315	628	29.7	793	1.3	361	164	
15	강원대학교_본교	15,994	1,623	3,689	23.1	6,174	1.7	125	124	

Sheet1

03 Data Collecting

온라인 부동산 서비스 시스템 '다방' 에서 표본데이터 인근 지역 자취방 데이터(room_total) 크롤링 *검색 키워드 : 대학교 명

The image displays a three-step data collection process:

- Python Script (VS Code):** A script named `chocopython_final3.py` uses Selenium to interact with a web browser. It searches for '고려대학교' (Korea University) and extracts room listing data, including price and location details.
- Web Browser (다방):** The browser shows the search results for '고려대학교' on the '다방' (Dabang) website. The search results are filtered by '이 지역 주변 방' (Rooms near this area).
- Database (HeidiSQL):** The data is stored in a database. The `chocopython.room` table is shown, containing columns: `num`, `name`, `guarantee`, and `month`. The data is sorted by `month` in descending order.

num	name	guarantee	month
1	건국대학교	500	50
2	건국대학교	1,000	80
3	건국대학교	1,000	60
4	건국대학교	1,000	50
5	건국대학교	300	30
6	건국대학교	65	60
7	건국대학교	3,000	20
8	건국대학교	1,000	36
9	건국대학교	200	37
10	건국대학교	300	42
11	건국대학교	500	50
12	건국대학교	1,000	48
13	건국대학교	300	36
14	건국대학교	2,000	30
15	건국대학교	100	27
16	건국대학교	2,000	45

04

CHOCHPYTHON PRJECT

Data Processing

04 Data processing

전국 대학교 정보 중에서 2인실 기숙사 **보유한 서울 및 수도권 지역** 4년제 종합대학교만 채택

2017_서울권 대학교_기숙사운영 - 읽기 전용 - 호환성 모드 - Excel

김 서영

파일 홈 삽입 페이지 레이아웃 수식 데이터 검토 보기 도움말 어떤 작업을 원하시나요?

자동 저장

잘라내기 붙여넣기 서식 복사 클립보드 글꼴 맞춤 표시 형식 스타일 셀 편집

A1 [기숙사 수용 현황_(대학)]

	D	E	F	G	H	I	J	K	L	M	N
	지역	상태	학교	재학생수(A)	실수	수용가능인원(B)	기숙사수용률 (C=B/Ax100)	기숙사 지원자 수(D)	입사 경쟁률 (E=D/B)	1인실 기숙사비	2인실 기숙사비
4											
5											
6	서울	기존	가톨릭대학교_제2캠퍼스	1,602	41	122	7.6	179	1.5	0	
7	서울	기존	가톨릭대학교_제3캠퍼스	263	150	150	57	82	0.5	617	
8	서울	기존	감리교신학대학교_본교	1,403	150	300	21.4	382	1.3	0	
9	서울	기존	건국대학교_본교	17,844	1,728	3,309	18.5	4,463	1.3	585	
10	서울	기존	경기대학교_제2캠퍼스	3,119	0	0	0	0	0	0	
11	서울	기존	경희대학교_본교	28,736	2,509	5,547	19.3	7,693	1.4	316	
12	서울	기존	고려대학교_본교	26,754	1,281	2,751	10.3	4,067	1.5	484	
13	서울	기존	광운대학교_본교	8,805	32	160	1.8	131	0.8	0	
14	서울	기존	국민대학교_본교	16,349	544	1,825	11.2	3,110	1.7	0	
15	서울	기존	덕성여자대학교_본교	5,913	439	779	13.2	1,258	1.6	290	
16	서울	기존	동국대학교_본교	15,764	685	1,419	9	2,566	1.8	478	
17	서울	기존	동덕여자대학교_본교	6,963	75	268	3.8	399	1.5	0	

Sheet1

04 Data processing

기숙사 보유한 서울 및 수도권 지역 4년제 종합대학교 데이터에서 기숙사 가격 & 등록금 & 수용인원에 데이터 추출

자동 저장

2017 (1) - 호환성 모드 - Excel

김 서영

파일

홈

삽입

페이지 레이아웃

수식

데이터

검토

보기

도움말

어떤 작업을 원하시나요?

잘라내기

붙여넣기

클립보드

텍스트 줄 바꿈

병합하고 가운데 맞춤

맞춤

일반

표

셀

스타일

조건부 서식

삽입

삭제

서식

Σ 자동 합계

채우기

지우기

정렬 및 찾기 및 필터

선택

L2

✕

✓

fx

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	학교	재학생수(A)	실수	수용가능인원(B)	2인실 기숙사비	등록금(D=B)								
2	건국대학교	17,844	1,728	3,309	382	8,192.3								
3	경희대학교	28,736	2,509	5,547	298	7,804.8								
4	고려대학교	26,754	1,281	2,751	383	8,256.9								
5	국민대학교	16,349	544	1,825	239	7,877.7								
6	덕성여자대학교	5,913	439	779	220	7,072.2								
7	동국대학교	15,764	685	1,419	338	7,939.2								
8	동덕여자대학교	6,963	75	268	304	7,387.3								
9	명지대학교	7,365	305	862	325	6,905.0								
10	상명대학교	7,500	215	800	167	7,652.6								
11	서강대학교	9,887	531	1,222	387	7,951.5								
12	서경대학교	6,523	104	384	186	7,812.6								
13	서울과학기술대학교	11,658	686	1,706	206	5,410.7								
14	서울대학교	27,875	3,070	5,933	170	6,011.4								

Sheet1

+

04 Data processing

2인실 기숙사를 보유한 서울 및 수도권 지역 4년제 종합대학교 엑셀데이터를 데이터베이스 형태로 가공

The screenshot illustrates a data processing workflow. At the top left, the HeidiSQL interface shows a MySQL database named 'chocopython' with a table 'dormitory'. The table contains 28 rows of data, including university names, student counts, room counts, and tuition fees. At the top right, an Excel spreadsheet displays the same data, with columns for '학교' (School), '재학생수(A)' (Students), '실수' (Rooms), '수용가능인원(B)' (Capacity), '2인실 기숙사비' (2-person dormitory fee), and '등록금(D=B)' (Tuition). At the bottom left, a Visual Studio Code window shows a Python script named 'exceltoDB.py' that uses the 'pymysql' library to connect to the database and insert data. At the bottom right, a terminal window shows the command 'python exceltoDB.py' being executed, which successfully inserts the data from the Excel spreadsheet into the 'dormitory' table.

학교	재학생수(A)	실수	수용가능인원(B)	2인실 기숙사비	등록금(D=B)
건국대학교	17,844	1,728	3,309	382	8,192.3
경희대학교	28,736	2,509	5,547	298	7,804.8
고려대학교	26,754	1,281	2,751	383	8,256.9
국민대학교	16,349	544	1,825	239	7,877.7
덕성여자대학교	5,913	439	779	220	7,072.2
동국대학교	15,764	685	1,419	338	7,939.2
동덕여자대학교	6,963	75	268	304	7,387.3
명지대학교	7,365	305	862	325	6,905.0
상명대학교	7,500	215	800	167	7,652.6
서강대학교	9,887	531	1,222	387	7,951.5
서경대학교	6,523	104	384	186	7,812.6
서울과학기술대학교	11,658	686	1,706	206	5,410.7
서울대학교	27,875	3,070	5,933	170	6,011.4
서울시립대학교	10,153	521	814	144	2,390.7
서울여자대학교	7,539	311	801	267	7,515
성공회대학교	2,258	53	149	98	7,048
성균관대학교	24,068	2,756	5,371	293	8,348.4
성신여자대학교	9,716	323	817	289	7,735.7

```
exceltoDB.py
import pymysql
import time
import math
import numpy as np
import pandas as pd

def toDB(dormi):
    result=None
    connection = None
    try:
        # DB연결
        connection = pymysql.connect(host='localhost',
                                     user='root',
                                     password='1234',
```

```
D:\junky\w4>cd D:\junky\w4\project_jk
D:\junky\w4\project_jk>python exceltoDB.py
```

04 Data processing

2인실 기숙사를 보유한 서울 및 수도권 지역 4년제 종합대학교 엑셀데이터를 데이터베이스 형태로 가공

학교	재학생수(A)	실수	수용가능인원(B)	2인실 기숙사비	등록금(D=B)
가천대학교	19,452	841	2,218	286	8,205.8
가톨릭대학교(성심교정)	7,441	345	1,076	323	6,906.1
강남대학교	6,931	352	934	298	7,358.2
건국대학교	17,844	1,728	3,309	382	8,192.3
경기대학교	11,970	1,122	2,272	330	7,263.1
경희대학교	28,736	2,509	5,547	298	7,804.8
고려대학교	26,754	1,281	2,751	383	8,256.9
국민대학교	16,349	544	1,825	239	7,877.7
단국대학교	12,569	891	2,438	310	8,001.8
대진대학교	7,799	867	1,507	154	7,157.2
덕성여자대학교	5,913	439	779	220	7,072.2
동국대학교	15,764	685	1,419	338	7,939.2
동덕여자대학교	6,963	75	268	304	7,387.3
명지대학교(인문캠퍼스)	7,365	305	862	325	6,905.0
명지대학교(자연캠퍼스)	7,146	660	2,036	228	9,114.5
상명대학교	7,500	215	800	167	7,652.6
서강대학교	9,887	531	1,222	387	7,951.5
서경대학교	6,523	104	384	186	7,812.6
서울과학기술대학교	11,658	686	1,706	206	5,410.7
서울대학교	27,875	3,070	5,933	170	6,011.4
서울시립대학교	10,153	521	814	144	2,390.7
서울신학대학교	2,976	127	403	189	6,822.7
서울여자대학교	7,539	311	801	267	7,515.0
성결대학교	5,389	106	331	143	7,183.7
성공회대학교	2,258	53	149	98	7,048.0
성균관대학교	24,068	2,756	5,371	293	8,348.4

Data 폴더의 2017.xls

성신여자대학교	9,716	323	817	289	7,735.7
세종대학교	12,718	642	1,403	232	7,835.0
수원대학교	10,025	390	902	179	7,481
숙명여자대학교	11,069	556	1,223	292	7,971.0
숭실대학교	14,134	785	1,611	330	7,919.2
아주대학교	11,288	872	2,824	266	8,231.7
연세대학교	27,064	4,323	9,570	308	9,101.6
을지대학교(성남캠퍼스)	4,270	254	502	310	8,401.5
이화여자대학교	19,475	2,426	4,284	298	8,584.5
중앙대학교	21,049	1,258	2,599	342	7,883.2
중앙대학교(안성캠퍼스)	5,698	1,012	2,030	206	8,654.7
평택대학교	4,114	350	850	201	7,438.6
한경대학교	6,061	677	1,460	145	4,062
한국산업기술대학교	6,628	620	1,436	283	9,003.7
한국외국어대학교	18,674	1,415	2,940	323	7,125.4
한국항공대학교	4,431	233	755	394	8,469.7
한성대학교	6,626	123	443	164	7,633.7
한세대학교	3,088	111	367	192	8,075.8
한양대학교	20,119	1,104	2,524	257	8,478.5
한양대학교(ERICA)	9,990	1,399	2,744	258	8,366.1
홍익대학교	13,871	680	1,575	339	8,254.0

04 Data processing

2인실 기숙사를 보유한 서울 및 수도권 지역 4년제 종합대학교 **엑셀데이터**를 **데이터베이스 형태로** 가공

exceltoDB.py

```
1 from selenium import webdriver as wd
2 # 디버 모듈 (귀리 수행)
3 import pymysql
4 import time
5 import math
6 import numpy as np
7 import pandas as pd
8
9 def toDB(dormi):
10     result=None
11     connection = None
12     try:
13         # DB연결
14         connection = pymysql.connect(host='localhost',
15                                     user='root',
16                                     password='1234',
17                                     db='chocopython',
18                                     charset='utf8mb4',
19                                     cursorclass=pymysql.cursors.DictCursor)
20         with connection.cursor() as cursor:
21             # 쿼리문 작성 및 인자 설정
22             sql = "INSERT INTO dormitory (name, student, room, accept, dormi2, tuition) VALUES (%s);" %str(list(dormi.loc[ name ])[1:-1])
23             # 쿼리 수행
24             cursor.execute(sql)
25             connection.commit()
26             # 결과 파지
27             result = cursor.fetchall()
28     except Exception as e:
29         result = None
30         print(e)
31     else:
32         print("정상 수행")
33     finally:
34         if connection: # 보일처리, connection이 0이면 넘어감
35             connection.close()
36     return result
37
38 dormi= pd.read_excel('./data/2017.xls', encoding='utf-8')
39 for name in dormi.index:
40     toDB(dormi)
41     # sql = "INSERT INTO dormitory (name, student, room, accept, dormi2, tuition) VALUES (%s);" %str(list(dormi.loc[ name ])[1:-1])
42     # print(sql)
43
44
45
46 """
47 CREATE TABLE `dormitory` (
48   `name` VARCHAR(50) NOT NULL,
49   `student` INT(11) NOT NULL,
50   `room` INT(11) NOT NULL,
51   `accept` INT(11) NOT NULL,
52   `dormi2` INT(11) NOT NULL,
53   `tuition` FLOAT NOT NULL,
54   PRIMARY KEY (`name`)
55 )
56 COLLATE='utf8_general_ci'
57 ENGINE=InnoDB
58 ;
59 """
```

04 Data processing

‘다방’에서 크롤링한 자취방 데이터를 ‘보증금*0.01+월세 값’의 공식을 사용하여 시세로 계산 & 데이터베이스 형태로 저장

```

1  from selenium import webdriver as wd
2  import pymysql
3  import time
4
5
6  def toDB(room):
7      result=None
8      connection = None
9      try:
10         # DB연결
11         connection = pymysql.connect(host='localhost',
12                                     user='root',
13                                     password='1234',
14                                     db='chocopython',
15                                     charset='utf8mb4',
16                                     cursorclass=pymysql.cursors.DictCursor)
17
18         with connection.cursor() as cursor:
19             # 커리온 작성 및 인자 세팅
20             sql = "INSERT INTO room (name, guarantee, month, total) VALUES (%s);"%str(roominfo)[1:-1]
21             # 커리 수행
22             cursor.execute(sql)
23             connection.commit()
24             # 결과 패지
25             result = cursor.fetchall()
26
27     except Exception as e:
28         result = None
29         print(e)
30
31     else:
32         print("정상 수행")
33
34     finally:
35         if connection: # 브렙저리, connection이 @이면 넘여감
36             connection.close()
37
38     return result
39
40
41 def selectSchool_list():
42     result=None
43     connection = None
44     try:
45         # DB연결
46         connection = pymysql.connect(host='localhost',
47                                     user='root',
48                                     password='1234',
49                                     db='chocopython',
50                                     charset='utf8mb4',
51                                     cursorclass=pymysql.cursors.DictCursor)
52
53         with connection.cursor() as cursor:
54             # 커리온 작성 및 인자 세팅
55             sql = "SELECT name FROM dormitory;"
56             # 커리 수행
57             cursor.execute(sql)
58             # 결과 패지
59             result = cursor.fetchall()
60
61     except Exception as e:
62         result = None
63         print(e)
64
65     else:
66         print("정상 수행")
67
68     finally:
69         if connection: # 브렙저리, connection이 @이면 넘여감
70             connection.close()
71
72     return result

```

```

try:
    # 타겟 사이트
    site_url = "https://www.dabangapp.com/search/?listId=11c30810&type=region&filters=sy_c222deposit-range%2F3A3S80%2C999999ASDQ2CK22price-range%2F3A3S80%2C999999ASDQ2CK22room-type%2F3A3S80%2C999999ASDQ2CK22position%2F7B62center%2F3A3S8126_93310928887956%2C37_46841215345735SDQ2CK22room-size%2F3A3I4770school_list-[]"
    for i in range(len(selectSchool_list)):
        school_list.append(selectSchool_list[i]['name'])
    print(school_list)
    # 브라우저 띄우기
    driver = wd.Chrome(executable_path='./chromedriver.exe')
    driver.get(site_url)
    ##### 검색하는곳 자기 서술특별시 송파구 가락동출 나중에 %s 로 판문후 나중에 데이터베이스에서 읽어오게 해서 자동검색
    for i in school_list:
        driver.find_element_by_class_name('SearchForm-input').send_keys(i)
        time.sleep(3)
        driver.find_element_by_class_name('SearchForm-btn').click()
        ##### 검색 후 이동#####
        #driver.find_element_by_id("frmNIDLogin").submit()
        #
        time.sleep(3)
        print(driver.find_element_by_css_selector('.icon-text span').text)
        print(i)
        goods=driver.find_elements_by_css_selector('.Room-list>i')

        roominfo_list=[]

        print(driver.find_element_by_css_selector('.icon-text span').text)

    for good in goods:
        try:
            Room_price=good.find_element_by_css_selector('span.RoomItem-price_type').text
            Room_summary=good.find_element_by_css_selector('span.RoomItem-summary').text
            if Room_price == '월세' and '원룸' in Room_summary:
                price_na=good.find_element_by_css_selector('span.RoomsItem-price_title').text
                price_one=price_na.split('/')[1]
                roominfo=[i,int(price_one[0])*10,i+int(price_one[1])/10 + int(price_one[1])*10]
                print(roominfo)
                toDB(roominfo)
            else:
                pass
        except Exception as e:
            print('의려 발생',e)
        print(roominfo_list,len(roominfo_list))
        driver.quit()
        driver.close()

import sys
sys.exit()

print("~성공~")
except Exception as e:
    print("~오류 발생 시발~,e)

'''

CREATE TABLE `room` (
  `num` INT(11) NOT NULL AUTO_INCREMENT,
  `name` VARCHAR(128) NOT NULL,
  `guarantee` INT(11) NOT NULL,
  `month` INT(11) NOT NULL,
  `total` INT(11) NOT NULL,
  PRIMARY KEY (`num`)
)

COLLATE='utf8_general_ci'
ENGINE=InnoDB
```

chocopython_final.
py

1. 다방에서 원룸, 보증금, 월세로 크롤링
2. 시세 = (보증금 * 0.01 + 월세 값)
3. DB 저장

```
roominfo=[i,int(price_one[0])*10,int(price_one[1])*10,int(price_one[0])/10 + int(price_one[1])*10]
```


05

CHOCHPYTHON PROJECT

Data Analysis & Data Visualization

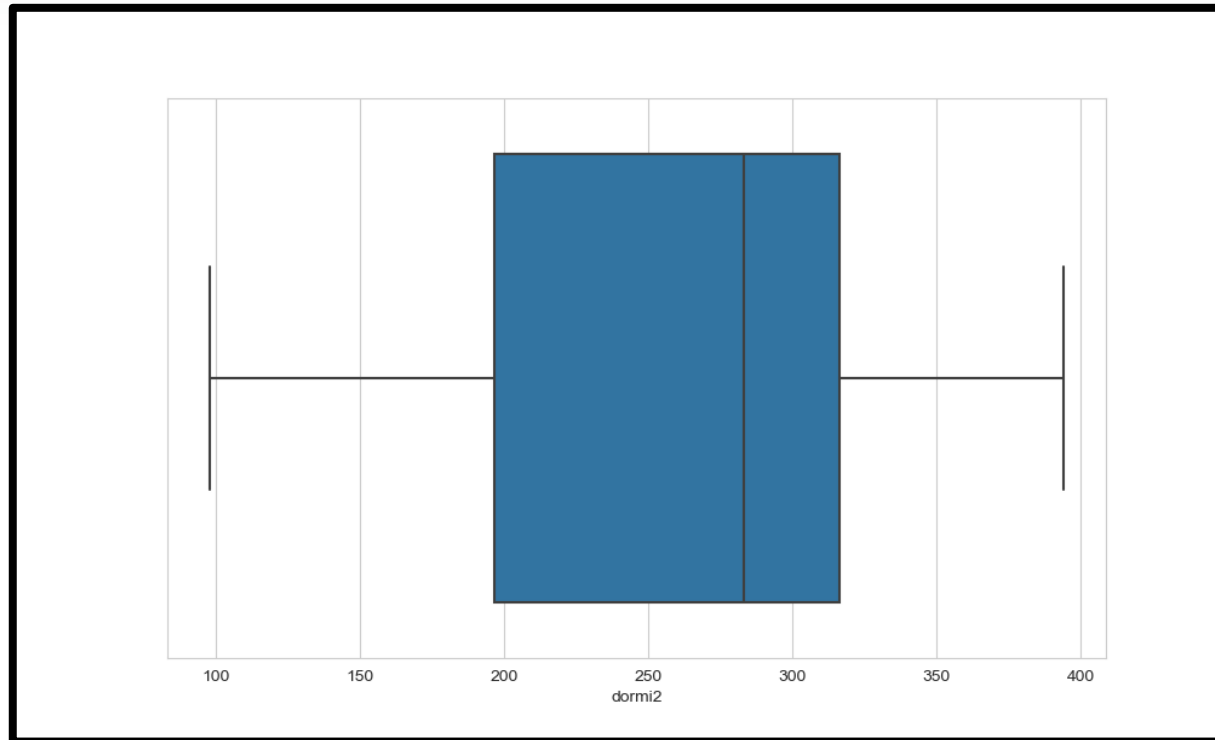
05 Data Analysis & visualization 변수 설명 import seaborn /view.py

```
1  import pymysql as sql
2  import pandas as pd
3  from sqlalchemy import create_engine
4  import pandas.io.sql as pSql
5  import numpy as np
6  from matplotlib import font_manager, rc
7  import matplotlib.pyplot as plt
8  import platform
9  import seaborn as sns
10
11 # 한글폰트 처리
12 plt.rcParams['axes.unicode_minus']=False
13 if platform.system() == 'Windows': #윈도우
14     path = 'C:\\Windows\\Fonts\\malgun.ttf'
15     font_name = font_manager.FontProperties(fname=path).get_name()
16     rc( 'font', family=font_name )
17
18 elif platform.system() == 'Darwin': #맥
19     rc( 'font', family='AppleGothic' )
20
21 else:
22     print('지원불가')
23
24 engine = create_engine("mysql+pymysql://root:1234@127.0.0.1:3306/chocopython", encoding='utf8')
25 conn = engine.connect()
26
27 dormi = pSql.read_sql("SELECT * FROM dormitory", conn)
28 room = pSql.read_sql("SELECT * FROM room", conn)
29 # name 기준으로 재정렬 평균
30 room_name= pd.pivot_table( room, index='name', aggfunc=np.mean)
31 # 테이블 합치기
32 dormi['room_total']=list(room_name['total'])
```

```
34  sns.set_style('whitegrid')
35  plt.figure(figsize=(10,6))
36  sns.boxplot(x=dormi['dormi2'])
37  plt.show()
38
39  sns.set_style('whitegrid')
40  plt.figure(figsize=(10,6))
41  sns.boxplot(x=dormi['student'])
42  plt.show()
43
44  sns.set_style('whitegrid')
45  plt.figure(figsize=(10,6))
46  sns.boxplot(x=dormi['room_total'])
47  plt.show()
48
49  sns.set_style('whitegrid')
50  plt.figure(figsize=(10,6))
51  sns.boxplot(x=dormi['tuition'])
52  plt.show()
53
54  d_s=np.corrcoef(dormi['dormi2'], dormi['student'])[0,1]
55
56  d_t=np.corrcoef(dormi['dormi2'], dormi['tuition'])[0,1]
57
58  d_r=np.corrcoef(dormi['dormi2'], dormi['room_total'])[0,1]
59  print('d_s: %s, d_t: %s, d_r: %s'%(d_s, d_t, d_r))
60
61 # 기숙사비, 등록금
62 sns.set_style('darkgrid')
63 sns.lmplot(x = 'tuition', y = 'dormi2', data = dormi, height =7, palette='Set2')
64 plt.show()
65
66 # 기숙사비, 방값
67 sns.set_style('darkgrid')
68 sns.lmplot(x = 'room_total', y = 'dormi2', data = dormi, height =7, palette='Set2')
69 plt.show()
70
71 # 기숙사비, 학생수
72 sns.set_style('darkgrid')
73 sns.lmplot(x = 'student', y = 'dormi2', data = dormi, height =7, palette='Set2')
74 plt.show()
```

05 Data Analysis & visualization 변수 설명 import seaborn

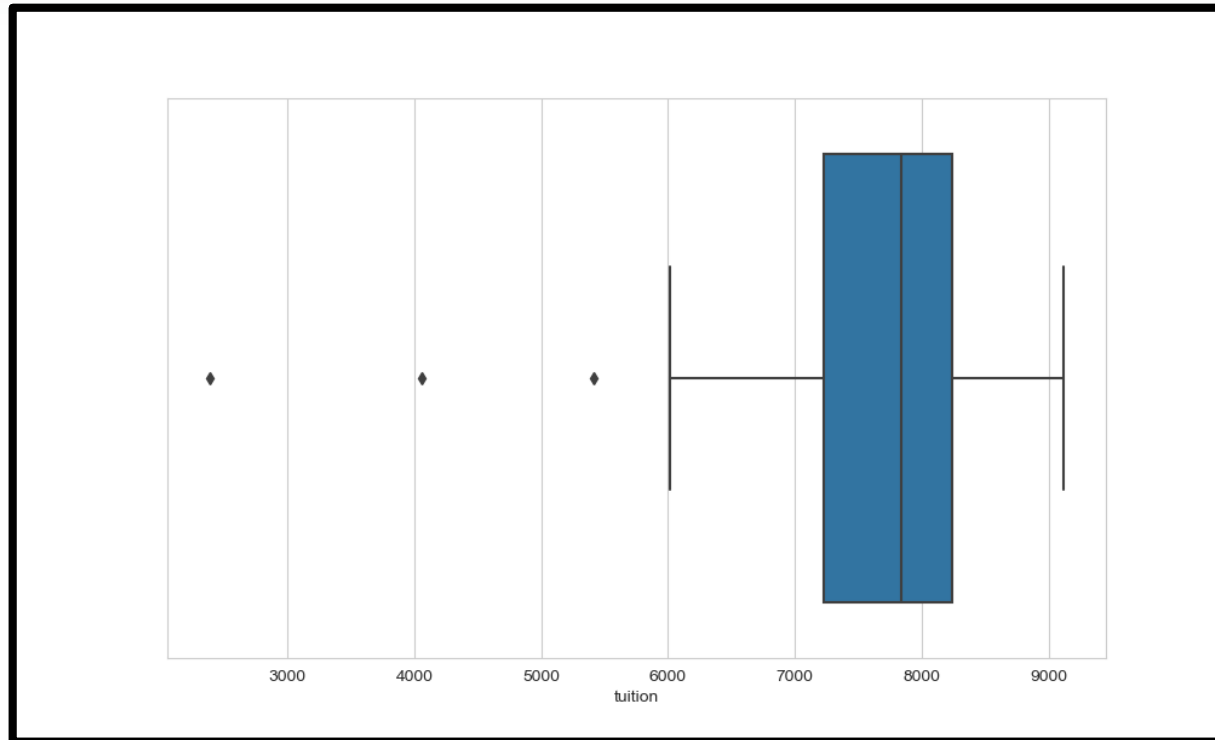
기숙사 비용(dormi2)의 boxplot



- * 단위 : 1000원
- * Q1 : 약 200
- * Q2(평균) : 약 270~280
- * Q3 : 약 310
- * Min : 약 100
- * Max : 약 400

05 Data Analysis & visualization 변수 설명 import seaborn

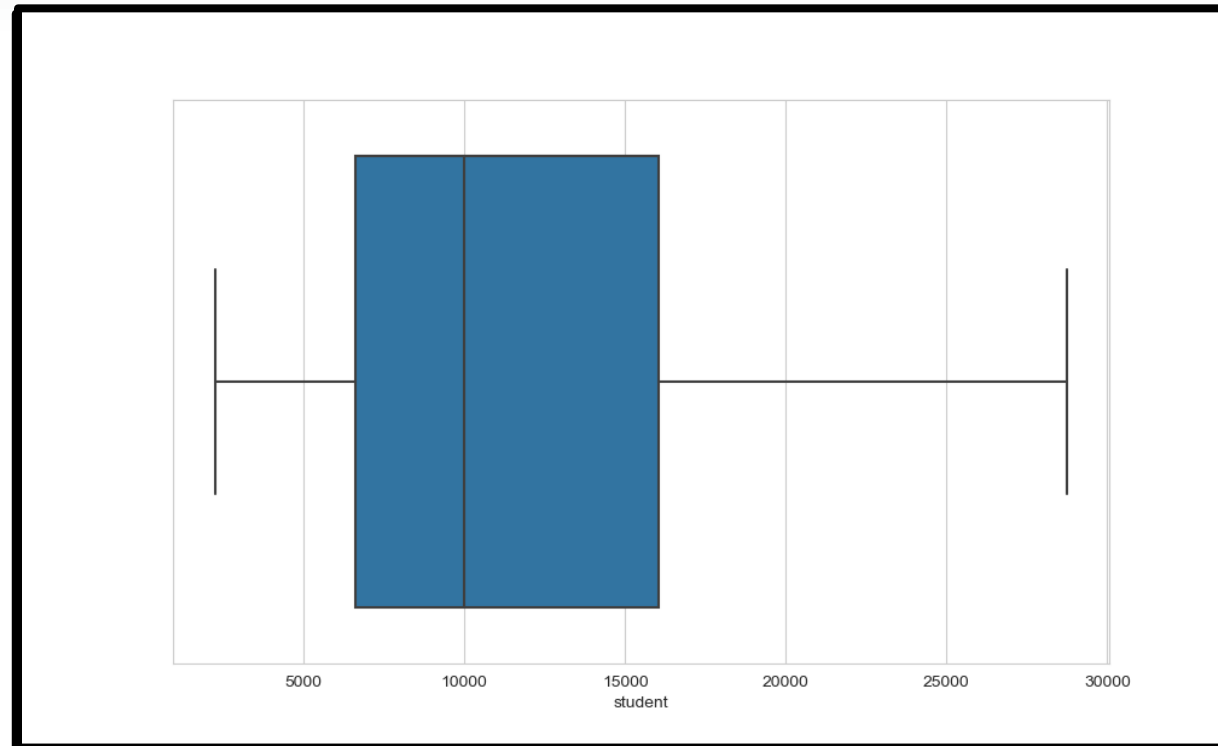
등록금(tuition)의 boxplot



- * 단위 : 1000원
- * Q1 : 약 7100
- * Q2(평균) : 약 7900
- * Q3 : 약 8100
- * Min : 약 6000
- * Max : 약 9000
- * Outlier

05 Data Analysis & visualization 변수 설명 import seaborn

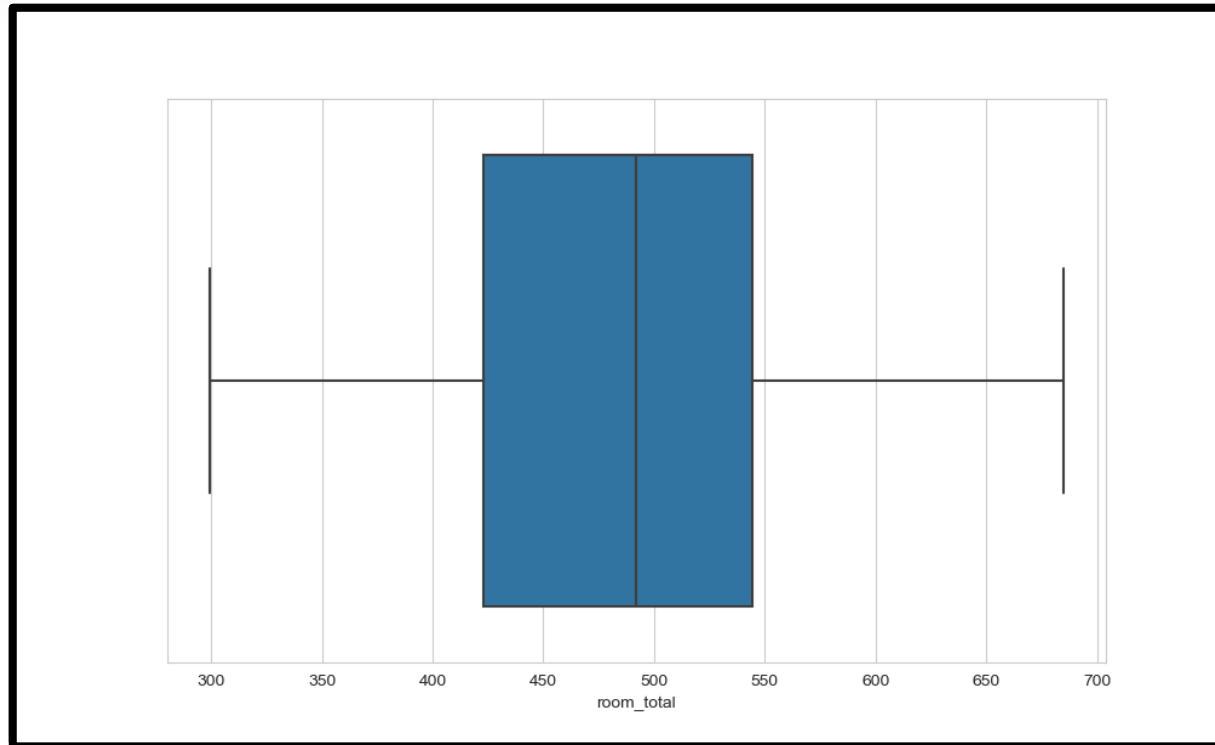
학생수(student)의 boxplot



- * 단위 : 명
- * Q1 : 약 5000
- * Q2(평균) : 약 10000
- * Q3 : 약 16000
- * Min : 약 1000
- * Max : 약 29000

05 Data Analysis & visualization 변수 설명 import seaborn

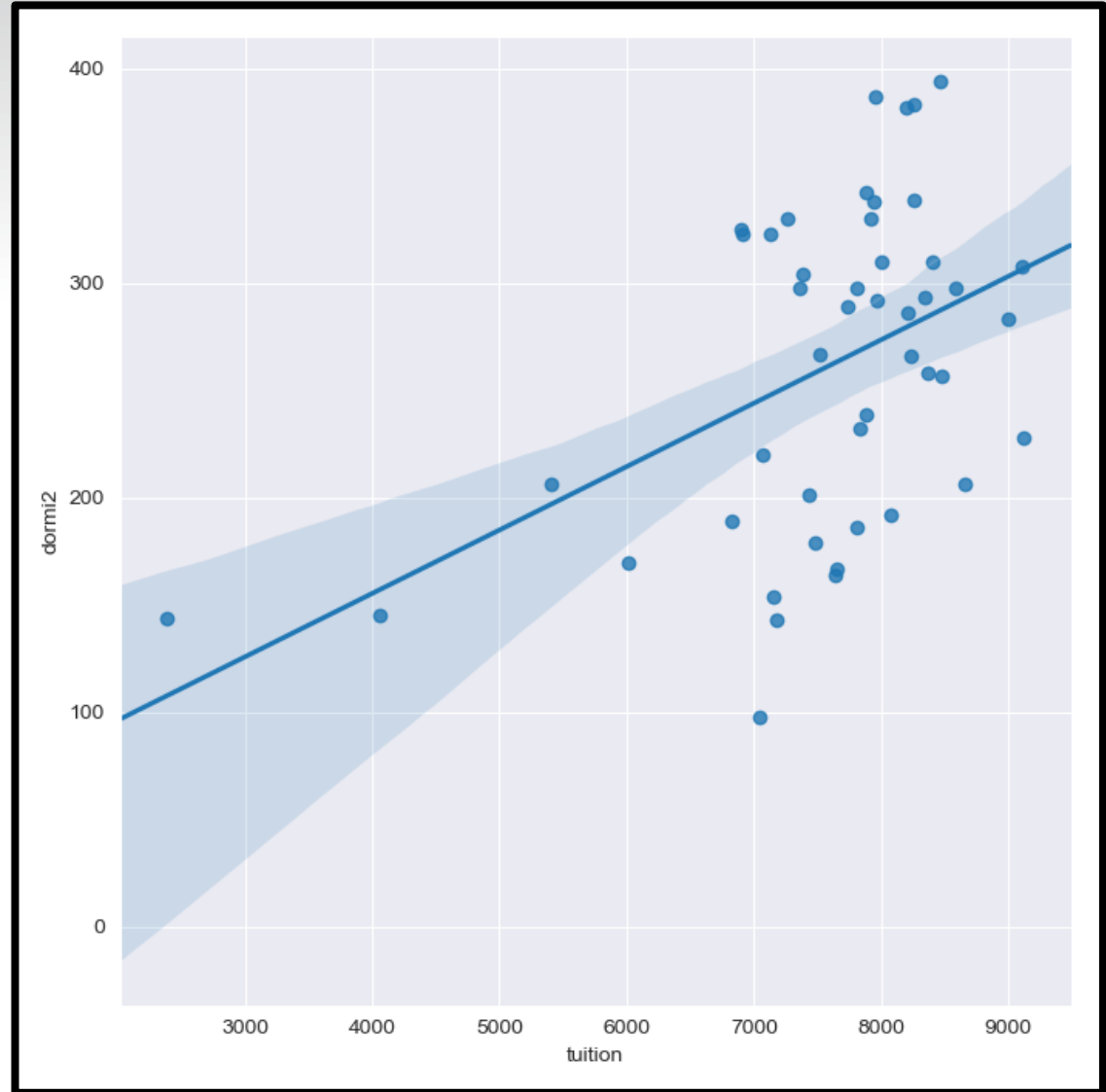
시세(room_total)의 boxplot



- * 단위 : 1000원
- * Q1 : 약 420
- * Q2(평균) : 약 490
- * Q3 : 약 550
- * Min : 약 300
- * Max : 약 690

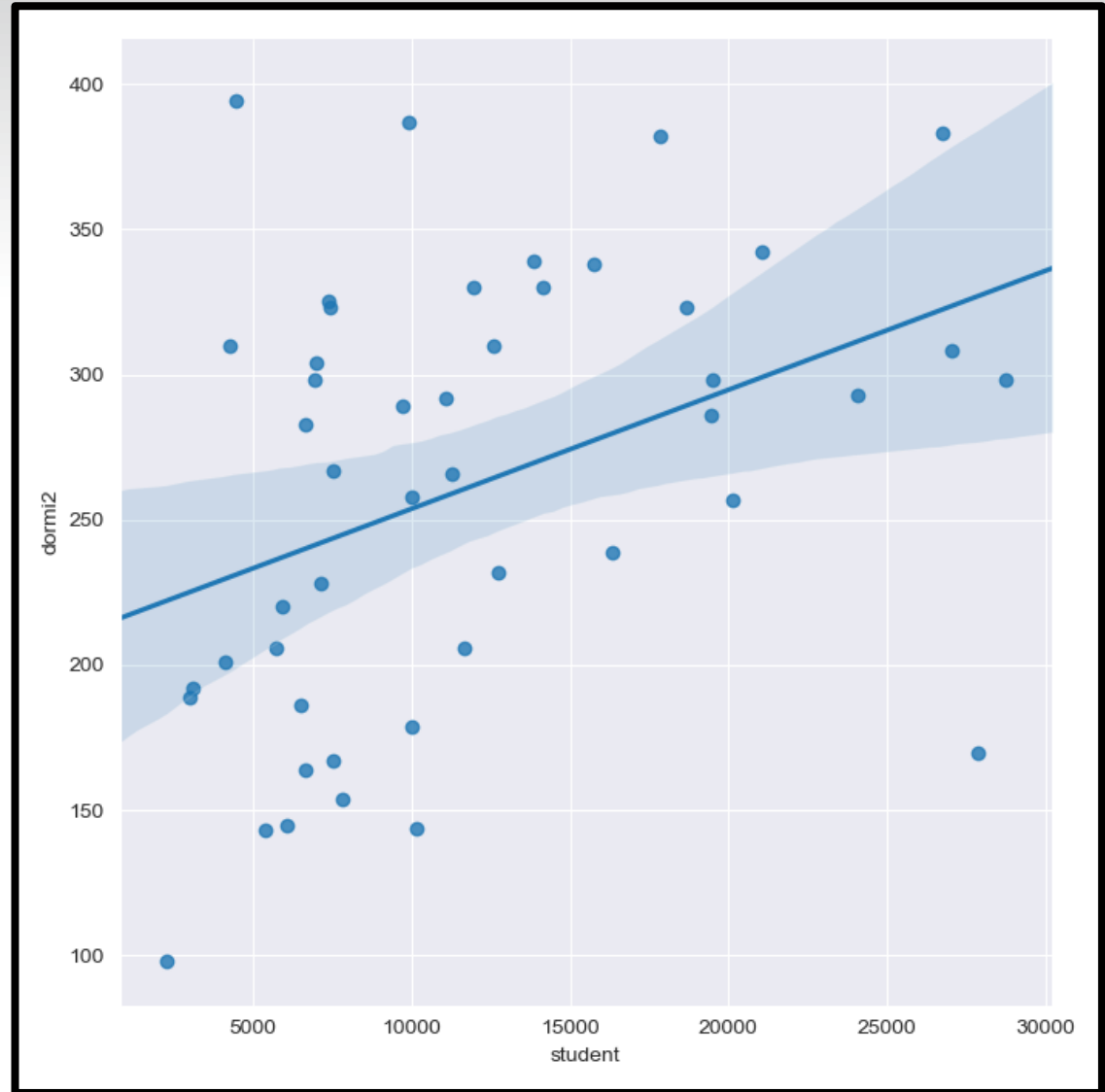
05 Data Analysis & visualization

- 기숙사 가격 & 등록금 상관성 시각화



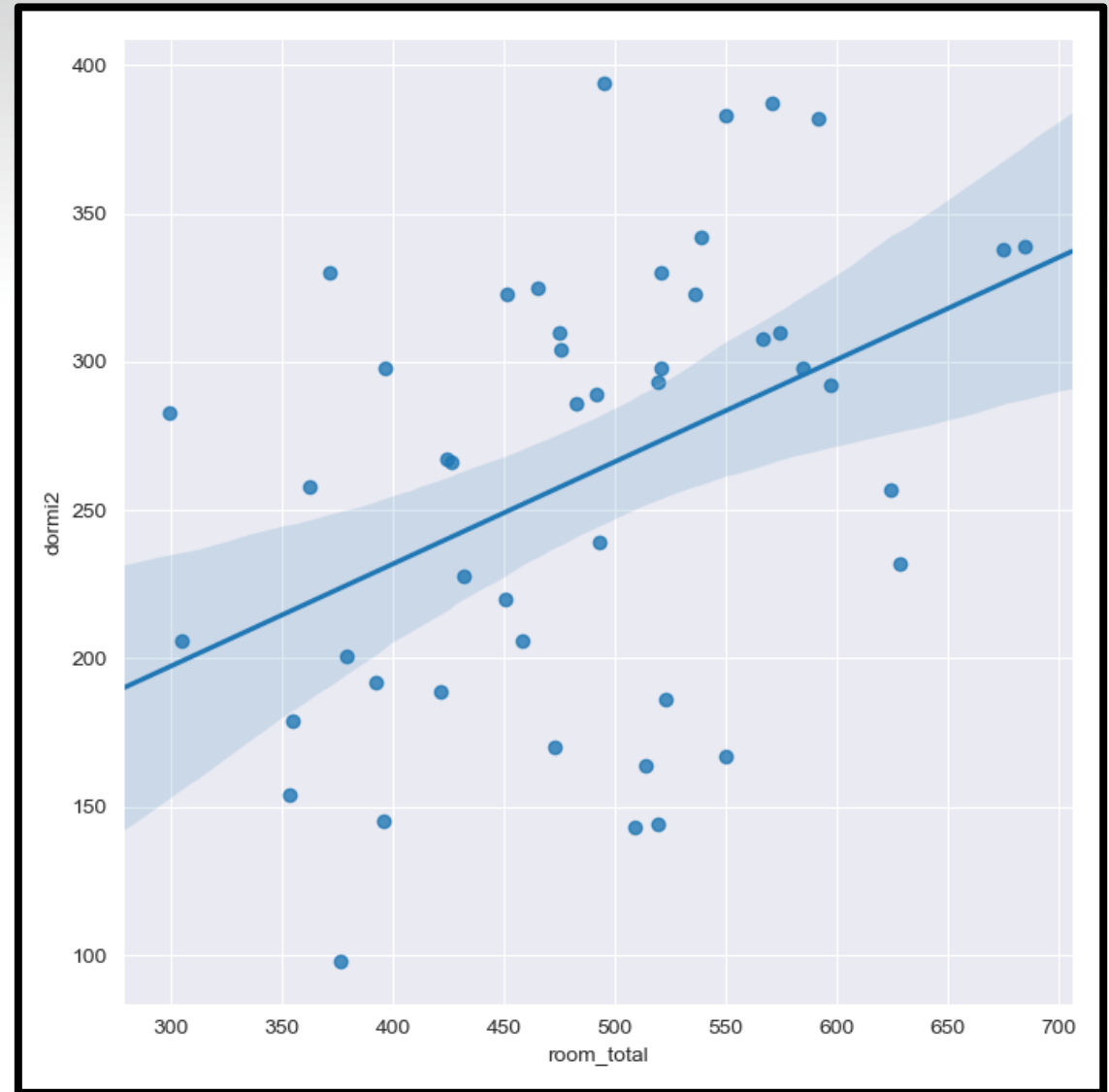
05 Data Analysis

- 기숙사 가격 & 학생수 상관성 시각화



05 Data Analysis

- 기숙사 가격 & 인근지역 시세 상관성 시각화



05 Data Analysis

상관분석을 통해 도출된 요인 별 상관계수

	student	room	accept	domi2	tuiton	room_total
domi2	0.3902674	0.2370143	0.2538953	1.0000000	0.4659546	0.4194359

Conclusion

[변수]

- 독립변수 x = 대학교 별 재학생 수(student) & 기숙사 수용가능 인원(accept) & 등록금(tuition) & 자취방 시세(room_total)
- 종속변수 y = 대학교 기숙사 가격
- 단, 모든 가격의 단위는 1000원, 기숙사 비의 단위 1개월, 인원 단위 1명 / 자취방 시세 = (보증금 * 0.01 + 월세 값) ex) 보증금 1000, 월세 50 → 60

[가설]

- H_0 : “종속변수 y 는 독립변수 x 와 상관관계가 있다.” & H_1 : “종속변수 y 는 독립변수 x 와 상관관계가 없다.”

[결론]

- 데이터 분석 결과 종속변수 y 와 독립변수 x 의 상관관계수 r 은 각각 모두 0.6 미만으로 유의미한 상관관계를 보이지 않았음.
- 아무런 관계도 없다는 뜻은 아니나, 선형관계는 존재하지 않음.
- H_0 귀무가설 기각, H_1 대립가설 채택
- 즉, 대학교 기숙사 가격은 등록금 변수와 유의미한 상관관계를 보이지만,
- 그 외의 타 변수와는 큰 상관관계가 없는 것으로 파악됨
- H_1 : “종속변수 y 는 독립변수 x 와 상관관계가 없다.”



한계점

- 대학 알림이에 공유된 대학 데이터의 양이 한정되어 있어서
분석에 쓸 수 있는 데이터의 양이 적었다는 것
- 대학교 인근지역의 시세를 월세의 평균으로 파악했으나,
이 변수가 시세의 대표성을 담보한다고 하기에는 무리가 있다는 것



Question & Answer



😊 Thank You 😊

*CHCOPYTHON FINAL TEAM
PROJECT*

