

- 데이터 받기
- 지하철 유무임별 이용현황 데이터 정제하기
- 유임 승차 비율이 가장 높은 역알아보기
- 유무임 승하차 인원이 가장 많은 역
- 모든 역의 유무임 승하차 비율 알아보기

- 엑셀 첨부파일 다운로드

대중교통통계자료 상세보기 - 이용안내 - T-money - Chrome

https://www.t-money.co.kr/ncs/pct/ugd/ReadTrcrStstDtl.dev?useYm=201901&rgtDtm=20190203103140

T 티머니 홈 money 이용안내 이용안내 홈

교통카드 통계자료

월간 교통카드 통계자료를 확인할 수 있습니다.

제목	2019년 01월 교통카드 통계자료		
첨부파일	 2019년 01월 교통카드 통계자료.xls	첨부파일 다운로드 >	등록일
			2019.02

이용안내



그 동안 행정기관 및 연구기관이 공공의 목적으로 이용하는 경우에만 제한적으로 제공되던 교통카드 자료를 서울시의 '열린시정 2.0'에 따라 공개하고자 합니다.

누구든지 서울시 홈페이지(<http://data.seoul.go.kr/index.jsp>)와 티머니홈페이지(<http://www.t-money.co.kr>)를 통해 쉽고 편리하게 이용할 수 있습니다.

제공 자료는 '지하철 노선별/역별 승하차인원'과 '버스 노선별/정류장별 승하차인원'으로 매월 업데이트 할 예정입니다.

데이터 받기

- 데이터 출력해보기

```
import csv  
f = open('subwayfee.csv')  
data = csv.reader(f)  
  
for row in data :  
    print(row)
```

```
['사용월', '호선명', '역ID', '지하철역', '유임 승차', '유임하차', '무임 승차', '무임하차']  
['2019-01', '1호선', '0150', '서울역', '1516452', '1400464', '221180', '211764']  
['2019-01', '1호선', '0151', '시청', '676932', '676266', '102388', '101033']  
['2019-01', '1호선', '0152', '종각', '1240810', '1177643', '162410', '152062']  
['2019-01', '1호선', '0153', '종로3가', '703110', '674148', '387062', '362735']  
['2019-01', '1호선', '0154', '종로5가', '571824', '583955', '275447', '263687']  
(생략)
```

데이터 정제하기

- 인원 숫자 데이터 정수로 바꾸기

```
next(data)
(생략)
for i in range(4,8) :
    row[i] = int(row[i])
print(row)
```

['2019-01', '1호선', '0150', '서울역', 1516452, 1400464, 221180, 211764]

['2019-01', '1호선', '0151', '시청', 676932, 676266, 102388, 101033]

['2019-01', '1호선', '0152', '종각', 1240810, 1177643, 162410, 152062]

['2019-01', '1호선', '0153', '종로3가', 703110, 674148, 387062, 362735]

['2019-01', '1호선', '0154', '종로5가', 571824, 583955, 275447, 263687]

(생략)

유임 승차 비율이 가장 높은 역알아보기

- 알고리즘 생각하기

- 1 | 데이터를 읽어온다.
- 2 | 모든 역의 데이터를 바탕으로 각 역의 비율(rate)을 계산한다.
- 3 | 비율이 가장 높은 역을 찾는다.
- 4 | 비율이 가장 높은 역이 어디인지, 그 비율이 얼마인지 출력한다.

* 각 역별 승차인원이 가장 많은 역 찾기

유임 승차 비율이 가장 높은 역알아보기

- 유임승차 / 무임승차 비율의 최댓값 찾기

```
import csv
f = open('subwayfee.csv')
data = csv.reader(f)
next(data)
mx = 0
rate = 0
for row in data :
    for i in range(4,8) :
        row[i] = int(row[i])
    rate = row[4] / row[6]
    if rate > mx :
        mx = rate
print(mx)
```

- 무임승차와 비교하여 유임승차의 비율 구하기
- $\text{rate} = \text{유임승차인원} / \text{무임승차인원}$

유임 승차 비율이 가장 높은 역알아보기

- 오류 메시지 읽고 원인 생각해보기

```
ZeroDivisionError                                Traceback (most recent call last)
<ipython-input-6-852b60f527e2> in <module>()
      8     for i in range(4,8):
      9         row[i] = int(row[i])
———> 10     rate = row[4]/ row[6]
     11     if rate > mx :
     12         mx = rate

ZeroDivisionError:division by zero
```

>> 에러를 찾기위해
row[6]가 0인 역을 출력해 봅시다.
if row[6]==0:
 print(row)

유임 승차 비율이 가장 높은 역알아보기

- row[6]이 0인 역 제외하고 출력하기

```
if row[6] != 0 :                # 만약 row[6](유임 승차 인원) 값이 0이 아니라면
    rate = row[4] / row[6]
    if rate > mx :                # 만약 rate 값이 mx 값보다 크다면
        mx = rate                # 만약 mx 값을 rate 값으로 업데이트하기
    print(row, round(rate, 2))    # 업데이트된 값 출력하기
```

```
['2019-01', '1호선', '0150', '서울역', 1516452, 1400464, 221180, 211764] 6.86
['2019-01', '1호선', '0152', '종각', 1240810, 1177643, 162410, 152062] 7.64
['2019-01', '2호선', '0201', '시청', 706097, 679503, 61058, 55387] 11.56
['2019-01', '2호선', '0202', '을지로입구', 1415414, 1414980, 120425, 111592] 11.75
['2019-01', '2호선', '0209', '한양대', 233785, 256079, 15380, 16652] 15.2
['2019-01', '2호선', '0222', '강남', 3153418, 3210437, 186486, 167666] 16.91
['2019-01', '2호선', '0239', '홍대입구', 2351935, 2507561, 114832, 111488] 20.48
['2019-01', '3호선', '0321', '충무로', 55, 0, 2, 0] 27.5
```

환승역이라 유임하차, 무임하차 모두 0이다.

유임 승차 비율이 가장 높은 역알아보기

- 새로운 기준으로 최댓값 찾기

```
if row[6] !=0 and (row[4]+row[6]) > 100000 :  
    rate = row[4] / (row[4]+row[6])  
(생략)
```

유임 승차 비율을 무임승차 인원과 비교하는
방법이 아닌, 전체 인원중 유임 승차 인원을
구하는 방법으로 데이터를 처리하자

유임 승차 비율이 가장 높은 역알아보기

```
import csv

f = open('subwayfee.csv')
data = csv.reader(f)
next(data)

mx = 0
rate = 0
mx_station = ''

for row in data :
    for i in range(4,8) :
        row[i] = int(row[i])
    if row[6] != 0 and (row[4]+row[6]) > 100000 :
        rate = row[4] / (row[4]+row[6])
        if rate > mx :
            mx = rate
            mx_station = row[3] + ' ' + row[1]

print(mx_station, round(mx*100,2))
```

유임 승차 비율이 가장 높은 역은?

유무임 승하차 인원이 가장 많은 역

- 알고리즘 설계하기
 - 1 | 데이터를 읽어온다.
 - 2 | 모든 역의 데이터를 바탕으로 유임 승차, 유임 하차, 무임 승차, 무임 하차 인원이 가장 많은 역을 찾는다.
 - 3 | 각각의 인원이 가장 많은 역을 출력한다.

유무임 승하차 인원이 가장 많은 역

- 데이터 저장할 리스트 초기화하기

```
import csv
f = open('subwayfee.csv')
data = csv.reader(f)
next(data)
mx = [0] * 4
mx_station = [''] * 4
```

유무임 승하차 인원이 가장 많은 역

- 유무임 승하차 인원이 가장 많은 역 찾기

```
import csv
f = open('subwayfee.csv')
data = csv.reader(f)
next(data)
mx = [0] * 4
mx_station = [''] * 4
label = ['유임승차', '유임하차', '무임승차', '무임하차']
for row in data :
    for i in range(4,8) :
        row[i] = int(row[i])
        if row[i] > mx[i-4] :
            mx[i-4] = row[i]
            mx_station[i-4] = row[3] + ' ' + row[1]
for i in range(4) :
    print(label[i]+' : '+mx_station[i], mx[i])
```

모든역의 유무임 승하차 비율을 파이차트로 나타내기

- 유무임 승하차 비율 파이 차트로 표현하기

```
import csv
import matplotlib.pyplot as plt

f = open('subwayfee.csv')
data = csv.reader(f)
next(data)
label = ['유임 승차', '유임하차', '무임 승차', '무임하차']
for row in data :
    for i in range(4,8) :
        row[i] = int(row[i])
    plt.pie(row[4:8])
    plt.axis('equal')
    plt.show()
```

차트 수정하기

1. 컬러를 4가지 지정
2. 폰트지정
3. label 지정
4. 비율값표시
5. 이미지 파일로 저장하기

모든역의 유무임 승하차 비율을 파이차트로 나타내기

```
1  import matplotlib.pyplot as plt
2  color = ['#14ccc0', '#389993', '#ff1c6a', '#00ff00']
3
4  plt.rc('font', family = 'Malgun Gothic')
5
6  for row in data:
7      for i in range(4,8):
8          row[i] = int(row[i])
9
10 plt.figure(dpi=300)
11 plt.title(row[3] + ' ' + row[1])
12 plt.pie(row[4:8], labels = label, colors=color, autopct='%1.1f%%')
13 plt.axis('equal')
14 plt.savefig(row[3] + ' ' + row[1] + '.png')
15 plt.show()
```