



Overview

Comma Separate Value

최성철 교수  
Director of TEAMLAB

---

# Comma Separate Value

- CSV, 필드를 쉼표(,)로 구분한 텍스트 파일
- 엑셀 양식의 데이터를 프로그램에 상관없이 쓰기 위한 데이터 형식이라고 생각하면 쉬움
- 탭(TSV), 빈칸(SSV) 등으로 구분해서 만들기도 함
- 통칭하여 character-separated values (CSV) 부름
- 엑셀에서는 “다름 이름 저장” 기능으로 사용 가능

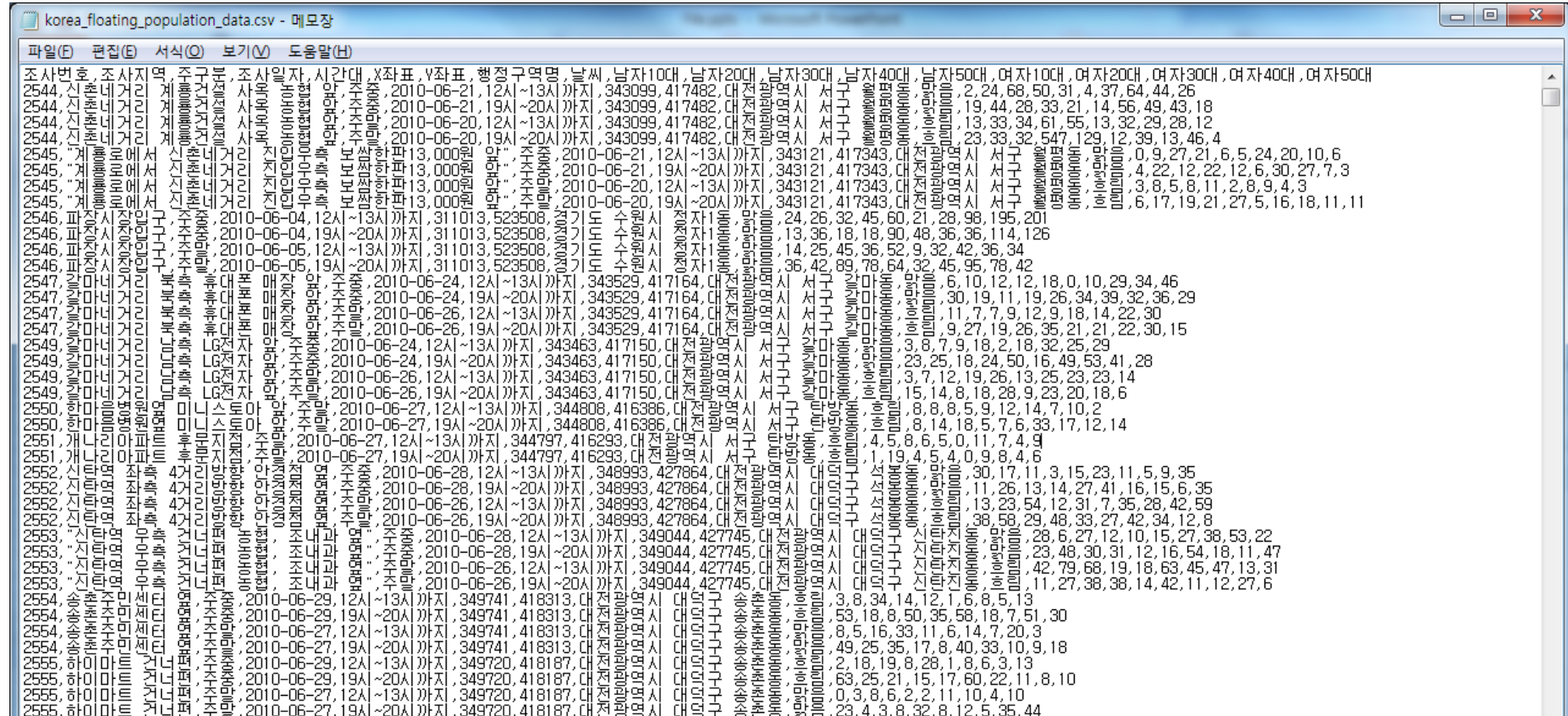
---

# 엑셀로 CSV 파일 만들기

- 1) 파일 다운로드 - <https://goo.gl/IQvsQF>
- 2) 파일 열기
- 3) 파일 → 다른 이름으로 저장  
→ CSV(쉼표로 분리) 선택 후 → 파일명 입력
- 4) 엑셀 종료 후 Notepad로 파일 열어보기

# 엑셀로 CSV 파일 만들기

데이터가 “,” 로 나뉘져 있음



조사항번	소재지	연월일	시간대	X좌표	Y좌표	행정구역명	날씨	남자10대	남자20대	남자30대	남자40대	남자50대	여자10대	여자20대	여자30대	여자40대	여자50대
2544	충청북도청주시	2010-06-21	12시~13시	343099	417482	대전광역시 서구	맑음	2,24	68,50	31,4	37,64	44,26					
2544	충청북도청주시	2010-06-21	19시~20시	343099	417482	대전광역시 서구	맑음	19,44	28,33	21,14	56,49	43,18					
2544	충청북도청주시	2010-06-20	12시~13시	343099	417482	대전광역시 서구	맑음	13,33	34,61	55,13	32,29	28,12					
2544	충청북도청주시	2010-06-20	19시~20시	343099	417482	대전광역시 서구	맑음	23,33	32,54	7,12	39,13	46,4					
2545	충청북도청주시	2010-06-21	12시~13시	343121	417343	대전광역시 서구	맑음	0,9	27,21	6,5	24,20	10,6					
2545	충청북도청주시	2010-06-21	19시~20시	343121	417343	대전광역시 서구	맑음	4,22	12,22	12,6	30,27	7,3					
2545	충청북도청주시	2010-06-20	12시~13시	343121	417343	대전광역시 서구	맑음	3,8	5,8	11,2	8,9	4,3					
2545	충청북도청주시	2010-06-20	19시~20시	343121	417343	대전광역시 서구	맑음	6,17	19,21	27,5	16,18	11,11					
2546	충청북도청주시	2010-06-04	12시~13시	311013	523508	충청북도청주시	맑음	24,26	32,45	60,21	28,98	195,201					
2546	충청북도청주시	2010-06-04	19시~20시	311013	523508	충청북도청주시	맑음	13,36	18,18	90,48	36,36	114,126					
2546	충청북도청주시	2010-06-05	12시~13시	311013	523508	충청북도청주시	맑음	14,25	45,36	52,9	32,42	36,34					
2546	충청북도청주시	2010-06-05	19시~20시	311013	523508	충청북도청주시	맑음	36,42	89,78	64,32	45,95	78,42					
2547	충청북도청주시	2010-06-24	12시~13시	343529	417164	대전광역시 서구	맑음	6,10	12,12	18,0	10,29	34,46					
2547	충청북도청주시	2010-06-24	19시~20시	343529	417164	대전광역시 서구	맑음	30,19	11,19	26,34	39,32	36,29					
2547	충청북도청주시	2010-06-26	12시~13시	343529	417164	대전광역시 서구	맑음	11,7	7,9	12,9	18,14	22,30					
2547	충청북도청주시	2010-06-26	19시~20시	343529	417164	대전광역시 서구	맑음	9,27	19,26	35,21	21,22	30,15					
2549	충청북도청주시	2010-06-24	12시~13시	343463	417150	대전광역시 서구	맑음	3,8	7,9	18,2	18,32	25,29					
2549	충청북도청주시	2010-06-24	19시~20시	343463	417150	대전광역시 서구	맑음	23,25	18,24	50,16	49,53	41,28					
2549	충청북도청주시	2010-06-26	12시~13시	343463	417150	대전광역시 서구	맑음	3,7	12,19	26,13	25,23	23,14					
2549	충청북도청주시	2010-06-26	19시~20시	343463	417150	대전광역시 서구	맑음	15,14	8,18	28,9	23,20	18,6					
2550	충청북도청주시	2010-06-27	12시~13시	344808	416386	대전광역시 서구	맑음	8,8	8,5	9,12	14,7	10,2					
2550	충청북도청주시	2010-06-27	19시~20시	344808	416386	대전광역시 서구	맑음	8,14	18,5	7,6	33,17	12,14					
2551	충청북도청주시	2010-06-27	12시~13시	344797	416293	대전광역시 서구	맑음	4,5	8,6	5,0	11,7	4,9					
2551	충청북도청주시	2010-06-27	19시~20시	344797	416293	대전광역시 서구	맑음	1,19	4,5	4,0	9,8	4,6					
2552	충청북도청주시	2010-06-28	12시~13시	348993	427864	대전광역시 서구	맑음	30,17	11,3	15,23	11,5	9,35					
2552	충청북도청주시	2010-06-28	19시~20시	348993	427864	대전광역시 서구	맑음	11,26	13,14	27,41	16,15	6,35					
2552	충청북도청주시	2010-06-26	12시~13시	348993	427864	대전광역시 서구	맑음	13,23	54,12	31,7	35,28	42,59					
2552	충청북도청주시	2010-06-26	19시~20시	348993	427864	대전광역시 서구	맑음	38,58	29,48	33,27	42,34	12,8					
2553	충청북도청주시	2010-06-28	12시~13시	349044	427745	대전광역시 서구	맑음	28,6	27,12	10,15	27,38	53,22					
2553	충청북도청주시	2010-06-28	19시~20시	349044	427745	대전광역시 서구	맑음	23,48	30,31	12,16	54,18	11,47					
2553	충청북도청주시	2010-06-26	12시~13시	349044	427745	대전광역시 서구	맑음	42,79	68,19	18,63	45,47	13,31					
2553	충청북도청주시	2010-06-26	19시~20시	349044	427745	대전광역시 서구	맑음	11,27	38,38	14,42	11,12	27,6					
2554	충청북도청주시	2010-06-29	12시~13시	349741	418313	대전광역시 서구	맑음	3,8	34,14	12,1	6,8	5,13					
2554	충청북도청주시	2010-06-29	19시~20시	349741	418313	대전광역시 서구	맑음	53,18	8,50	35,58	18,7	51,30					
2554	충청북도청주시	2010-06-27	12시~13시	349741	418313	대전광역시 서구	맑음	8,5	16,33	11,6	14,7	20,3					
2554	충청북도청주시	2010-06-27	19시~20시	349741	418313	대전광역시 서구	맑음	49,25	35,17	8,40	33,10	9,18					
2555	충청북도청주시	2010-06-29	12시~13시	349720	418187	대전광역시 서구	맑음	2,18	19,8	28,1	8,6	3,13					
2555	충청북도청주시	2010-06-29	19시~20시	349720	418187	대전광역시 서구	맑음	63,25	21,15	17,60	22,11	8,10					
2555	충청북도청주시	2010-06-27	12시~13시	349720	418187	대전광역시 서구	맑음	0,3	8,6	2,2	11,10	4,10					
2555	충청북도청주시	2010-06-27	19시~20시	349720	418187	대전광역시 서구	맑음	23,4	3,8	32,8	12,5	35,44					

# CSV 파일 다루기

---

# CSV 파일 읽기/쓰기

- Text 파일 형태로 데이터 처리 예제
- 예제 데이터: customer.csv (<https://goo.gl/FIsMqB>)
- 일반적 **textfile**을 처리하듯 파일을 읽어온 후, **한 줄 한 줄씩** 데이터를 처리함

<http://goo.gl/1JdKst>

# CSV 파일 예제

customers
customerNumber INT (11)
customerName VARCHAR (50)
contactLastName VARCHAR (50)
contactFirstName VARCHAR (50)
phone VARCHAR (50)
addressLine1 VARCHAR (50)
addressLine2 VARCHAR (50)
city VARCHAR (50)
state VARCHAR (50)
postalCode VARCHAR (15)
country VARCHAR (50)
salesRepEmployeeNumber INT (11)
creditLimit DOUBLE
Indexes

해당데이터는 고객의  
FullName, 성, 이름, 전화번호, 국가 등 기록  
" " 로 분리

customers.csv - 메모장
customerNumber,customerName,contactLastName,contactFirstName,phone,addressLine1,addressLine2,city,state,postalCode,country,salesRepEmployeeNumber,creditLimit
103,"Atelier graphique",Schmitt,"Carine",40.32.2555,"54, rue Royale",NULL,Nantes,France,44000,1370,21000
112,"Signal Gift Stores",King,Jean,7025551838,"8489 Strong St.",NULL,"Las Vegas",NV,83030,USA,1166,71800
114,"Australian Collectors, Co.",Ferguson,Peter,03 9520 4555,"636 St Kilda Road",Level 3,Melbourne,Victoria,3004,Australia,1611,117300
119,"Rochelle Gifts",Lafune,Janine,40.67.6555,"57, rue des Cinqvante Otages",NULL,Nantes,France,44000,1370,118200
121,"Baane Mini Imports",Bergulfs,Jonas,07 99 9555,"Erling Skakkes gate 78",NULL,Stavern,Norway,1504,81700
124,"Mini Gifts Distributors Ltd.",Nelson,Susan,4155551450,"5677 Strong St.",NULL,"San Rafael",CA,97562,USA,1165,210500
125,"Havel & Zbyszek Co.",Piestrzeniewicz,Zbyszek,"(26) 642-7555",ul. Filtrowa 68,NULL,Warszawa,Poland,01-012,12800
128,"Blauer See Auto, Co.",Keitel,Roland,49 69 66 90 2555,"Lyonerstr. 34",NULL,Frankfurt,Germany,60528,1504,59700
129,"Mini Wheels Co.",Murphy,Julie,650555787,"5557 North Pendale Street",NULL,"San Francisco",CA,94217,USA,1165,64600
131,"Land of Nod, Inc.",Lee,Kwai,2125557413,"4032 Furth Circle",Suite 400,NYC,NY,10022,USA,1286,138500
141,"Euro Shopping Channel",Freyre,Diego,"(91) 555 94 44",C/ Moralzarzal, 86,NULL,Madrid,Spain,28034,1370,227600
144,"Volvo Model Replicas, Co.",Berglund,Christina,"0921-12 3555",Bersuuvägen 8,NULL,Luleå,Sweden,1504,53100
145,"Danish Wholesale Imports",Petersen,Jytte,"31 12 3555",Vindbæltet 34,NULL,København,Denmark,1401,83400
146,"Saveley & Henriot, Co.",Saveley,Mary,"78 32 5555",2, rue du Commerce,NULL,Lyon,France,69004,1337,123900
148,"Dragon Souvenirs, Ltd.",Natividad,Eric,"+65 221 7555",Bronz Sok.,Bronz Apt,3/6 Tesviklye,Singapore,1621,103800
151,"Muscle Machine",Jeff,Jeff,2125557413,"4032 Furth Circle",Suite 400,NYC,NY,10022,USA,1286,138500
157,"Diecast Classics Inc.",Leong,Kelvin,2155551555,"7586 Pompton St.",NULL,Allentown,PA,70267,USA,1216,100600
161,"Technics Stores Inc.",Hashimoto,Juri,6505556809,"9408 Furth Circle",NULL,Burlingame,CA,94217,USA,1165,84600
166,"Handji Gifts & Co",Victorino,Wendy,"+65 224 1555",106 Linden Road Sandown,"2nd Floor",Singapore,1612,97900
167,"Herkuu Gifts",Oeztan,Veyzel,"+47 2267 3215",Brehøien St. 121,PR 334 Sentrum,Bergen,Norway,N 5804,1504,96800
168,"American Souvenirs Inc",Francis,Kelth,2035557845,"149 Spinnaker Dr.",Suite 101,New Haven,CT,97823,USA,1286,0
169,"Porto Imports Co.",de Castro,Isabel,"(1) 356-5555",Estrada da saude n. 58,NULL,Lisboa,Portugal,1756,76400
171,"Daedalus Designs Imports",Rancé,Martine,"20.16.1555",184, chaussée de Tournai,NULL,Lille,France,59000,1370,82900
172,"La Corne D'abondance, Co.",Bertrand,Marie,"(1) 42.34.2555",265, boulevard Charonne,NULL,Paris,France,75012,1337,84300
173,"Cambridge Collectables Co.",Tseng,Jerry,6175555555,"4658 Baden Av.",NULL,Cambridge,MA,51247,USA,1188,43400
175,"Gift Depot Inc.",King,Julie,2035552570,"25593 South Bay Ln.",NULL,Bridgewater,CT,97562,USA,1323,84300
177,"Osaka Souvenirs Co.",Kentary,Mary,"+61 06 6942 5555",1-6-20 Dojima,NULL,Kita-ku,Osaka,530-0003,Japan,1621,81200
181,"Vitechrome Inc.",Frick,Michael,2125551500,"2578 Kingston Rd.",Suite 101,NYC,NY,10022,USA,1286,76400
186,"Toys of Finland, Co.",Karttunen,Matti,"90-224 8555",Keskuskatu 45,NULL,Helsinki,Finland,1501,96500
187,"AV Stores, Co.",Ashworth,Rachel,"(171) 555-1555",Fauntleroy Circus,NULL,Manchester,UK,EC2 5NT,1501,136800
189,"Clover Collections, Co.",Cassidy,Dean,"+353 1862 1555",25 Maiden Lane,"Floor No. 4",Dublin,Ireland,1504,69400
198,"Auto-Moto Classics Inc.",Taylor,Leslie,6175558428,"16780 Pompton St.",NULL,Brickhaven,MA,58339,USA,1188,67700
201,"UK Collectables, Ltd.",Devon,Elizabeth,"(171) 555-2232",12 Berkeley Gardens,Blyton,Norfolk,England,UK,1501,92700
202,"Canadian Gift Exchange Network",Tamuri,Yoshi,"(604) 555-3392",1900 Oak St.,NULL,Vancouver,BC,V3F 2K1,Canada,1323,90300
204,"Online Mini Collectables",Barajas,Miguel,6175557555,"7635 Spinnaker Dr.",NULL,Brickhaven,MA,58339,USA,1188,67700
205,"Toys4GrownUps.com",Young,Julie,6265557265,"78934 Hillside Dr.",NULL,Pasadena,CA,90003,USA,1166,90700
206,"Asian Shopping Network, Co.",Walker,Brydev,"+612 9411 1555",Suntec Tower Three,"8 Temasek",Singapore,038988,1612,97900
209,"Mini Caravay",Citeaux,Fredérique,"88.60.1555",24, place Kleber,NULL,Strasbourg,France,67000,1370,53800
211,"King Kong Collectables, Co.",Gao,Mike,"+852 2251 1555",Bank of China Tower,"1 Garden Road",Central Hong Kong,NULL,Norway,N 5804,1504,96800
216,"Enaco Distributors",Saavedra,Eduardo,"(93) 203 4555",Rambla de Catalunya, 23,NULL,Barcelona,Spain,08022,1370,82900
219,"Boards & Toys Co.",Young,Mary,3105552373,"4097 Douglas Av.",NULL,Glendale,CA,91217,USA,1166,11000
223,"Natürlich Autos",Kloss,Horst,"0372-555188",Taucherstrasse 10,NULL,Cunewalde,Germany,01307,12800
227,"Heintze Collectables",Ibsen,Palle,"86 21 3555",Sveagloset 45,NULL,Århus,Denmark,8200,1401,120800
233,"Quebec Home Shopping Network",Fresniere,Jean,"(514) 555-8054",43 rue St. Laurent,NULL,Montréal,Quebec,H1J 1C3,Canada,1286,48700
237,"ANG Resellers, Casino",Alejandra,"(91) 745 5555",1, rue de la Paix,NULL,Paris,France,75001,1370,82900
239,"Collectable Mini Designs Co.",Thompson,Valarie,7605558146,"361 Furth Circle",NULL,"San Diego",CA,91217,USA,1166,105000
240,giftsbymail.co.uk,Bennett,Helen,"(198) 555-8888",Garden House,"Crowther Way 23",Cowes,Isle of Wight,"PO31 7PJ",UK,1501,93900
242,"Alpha Cognac",Roulet,Annette,"61 77 6555",1 rue Alsace-Lorraine,NULL,Toulouse,France,31000,1370,61100
247,"Messner Shopping Network",Messner,Renate,"069-0959984",Magazinweg 7,NULL,Frankfurt,Germany,60528,1504,59700
249,"Mica Models & Co.",Accorti,Paolo,"011-4988555",Via Monte Bianco 34,NULL,Torino,Italy,10100,113000
250,"Lyon Souvenirs",da Silva,Daniel,"+33 1 46 82 7555",27 rue du Colonel Pierre Avia,NULL,Paris,France,75508,1370,68100
256,"Auto Associes & Cie.",Tonini,Daniel,"30.59.8555",67, avenue de l'Europe,NULL,Versailles,France,78000,1370,77900
259,"Toms Spezialitäten, Ltd",Pfalzheim,Henriette,"0221-5554327",Mehreimerstr. 369,NULL,Köln,Germany,50739,1504,120400
260,"Royal Canadian Collectables, Ltd.",Lincoln,Elizabeth,"(604) 555-4555",23 Tsawassen Blvd.,NULL,Tsawassen,BC,T2F 8M4,Canada,1323,89600
273,"Franken Gifts, Co",Franken,Peter,"089-0877555",Berliner Platz 43,NULL,München,Germany,80805,1504,120400
276,"Anna's Decorations, Ltd",O'Hara,Anna,"02 9536 8555",201 Miller Street,Level 15,North Sydney,NSW,2060,Australia,1611,107800

# CSV 파일 읽기 예제

```
line_counter = 0    #파일의 총 줄수를 세는 변수
data_header = []    #data의 필드값을 저장하는 list
customer_list = []  #customer 개별 List를 저장하는 List
```

```
with open ("customers.csv") as customer_data: #customer.csv 파일을 customer_data 객체에 저장
```

```
while 1:
```

```
    data = customer_data.readline() #customer.csv에 한줄씩 data 변수에 저장
```

```
    if not data: break #데이터가 없을 때, Loop 종료
```

```
    if line_counter==0:    #첫번째 데이터는 데이터의 필드
```

```
        data_header = data.split(",") #데이터의 필드는 data_header List에 저장, 데이터 저장시 “,” 로 분리
```

```
    else:
```

```
        customer_list.append(data.split(",")) #일반 데이터는 customer_list 객체에 저장, 데이터 저장시 “,” 로 분리
```

```
    line_counter += 1
```

```
print("Header :Wt", data_header)    #데이터 필드 값 출력
```

```
for i in range(0,10):    #데이터 출력 (샘플 10개만)
```

```
    print ("Data",i,":WtWt",customer_list[i])
```

```
print (len(customer_list))    #전체 데이터 크기 출력
```



# CSV 파일 쓰기 예제

```
line_counter = 0
data_header = []
employee = []
customer_USA_only_list = []
customer = None
```

```
with open ("customers.csv", "r") as customer_data:
```

```
    while 1:
```

```
        data = customer_data.readline()
```

```
        if not data:
```

```
            break
```

```
        if line_counter==0:
```

```
            data_header = data.split(",")
```

```
        else:
```

```
            customer = data.split(",")
```

```
            if customer[10].upper() == "USA": #customer 데이터의 offset 10번째 값
```

```
                customer_USA_only_list.append(customer) #즉 country 필드가 "USA" 것만
```

```
            line_counter+=1 #sutomer_USA_only_list에 저장
```

```
print ("Header :Wt", data_header)
```

```
for i in range(0,10):
```

```
    print ("Data :WtWt",customer_USA_only_list[i])
```

```
print (len(customer_USA_only_list))
```

```
with open ("customers_USA_only.csv", "w") as customer_USA_only_csv:
```

```
    for customer in customer_USA_only_list:
```

```
        customer_USA_only_csv.write(",".join(customer).strip('Wn')+ "Wn")
```

```
        #cutomer_USA_only_list 객체에 있는 데이터를 customers_USA_only.csv 파일에 쓰기
```

# CSV 객체 사용하기

# CSV 객체로 CSV 처리

- Text 파일 형태로 데이터 처리시 문장 내에 들어가 있는 “,” 등에 대해  
전처리 과정이 필요
- 파이썬에서는 간단히 CSV파일을 처리하기 위해 **csv 객체**를 제공함
- 예제 데이터: korea\_floating\_population\_data.csv (from <http://www.data.go.kr>)
- 예제 데이터는 국내 주요 상권의 유동인구 현황 정보
- 한글로 되어 있어 한글 처리가 필요

# CSV 객체로 CSV 처리

- 시간대/조사일자/행정구역/날씨 등을 기준으로  
연령별/성별 유동인가 해당 지역에 몇 명인가 표시

<https://goo.gl/ujHuXQ>

[illegible]

# CSV 객체 활용

```
import csv
reader = csv.reader(f,
                    delimiter=',', quotechar='"',
                    quoting=csv.QUOTE_ALL)
```

Attribute	Default	Meaning
delimiter	,	글자를 나누는 기준
lineterminator	\r\n	줄 바꿈 기준
quotechar	"	문자열을 둘러싸는 신호 문자
quoting	QUOTE_MINIMAL	데이터 나누는 기준이 quotechar에 의해 둘러싸인 레벨

# CSV 객체 활용

## 유동 인구 데이터 중 성남 데이터만 수집

```
import csv
seoung_nam_data = []
header = []
rownum = 0

with open("korea_floating_population_data.csv", "r", encoding="cp949") as p_file:
    csv_data = csv.reader(p_file) #csv 객체를 이용해서 csv_data 읽기
    for row in csv_data: #읽어온 데이터를 한 줄씩 처리
        if rownum == 0:
            header = row #첫 번째 줄은 데이터 필드로 따로 저장
            location = row[7]
            # "행정구역" 필드 데이터 추출, 한글 처리로 유니코드 데이터를 cp949로 변환
            if location.find(u"성남시") != -1:
                # "행정구역" 데이터에 성남시가 들어가 있으면 seoung_nam_data List에 추가
                seoung_nam_data.append(row)
            rownum += 1

with open("seoung_nam_floating_population_data.csv", "w", encoding="utf8") as s_p_file:
    writer = csv.writer(s_p_file, delimiter='\\t', quotechar='"', quoting=csv.QUOTE_ALL)
    # csv.writer를 사용해서 csv 파일 만들기 delimiter 필드 구분자
    # quotechar는 필드 각 데이터는 묶는 문자, quoting는 묶는 범위
    writer.writerow(header) #제목 필드 파일에 쓰기
    for row in seoung_nam_data:
        writer.writerow(row) #seoung_nam_data에 있는 정보 list에 쓰기
```



**Human knowledge belongs to the world.**

# Logging Overview

## Logging

최성철 교수  
Director of TEAMLAB

01100  
00110



게임을 만들었는데  
HACK쓰는 애들 때문에  
망했어요...

**어떻게 잡을 수 있을까?**

**일단은 기록부터!!**

---

# 로그 남기기 - Logging

- 프로그램이 실행되는 동안 일어나는 정보를 기록을 남기기
- 유저의 접근, 프로그램의 Exception, 특정 함수의 사용
- Console 화면에 출력, 파일에 남기기, DB에 남기기 등등
- 기록된 로그를 분석하여 의미있는 결과를 도출 할 수 있음
- 실행시점에서 남겨야 하는 기록, 개발시점에서 남겨야하는 기록

---

# print vs logging

- 기록을 print로 남기는 것도 가능함
- 그러나 Console 창에만 남기는 기록은 분석시 사용불가
- 때로는 레벨별(개발, 운영)로 기록을 남길 필요도 있음
- 모듈별로 별도의 logging을 남길 필요도 있음
- 이러한 기능을 체계적으로 지원하는 모듈이 필요함

# logging 모듈

## - Python의 기본 Log 관리 모듈

```
import logging
```

```
logging.debug("틀렸잖아!")
```

```
logging.info("확인해")
```

```
logging.warning("조심해!")
```

```
logging.error("에러났어!!!")
```

```
logging.critical ("망했다...")
```

---

# logging level

- 프로그램 진행 상황에 따라 다른 Level의 Log를 출력함
- 개발 시점, 운영 시점 마다 다른 Log가 남을 수 있도록 지원함
- **DEBUG > INFO > WARNING > ERROR > Critical**
- Log 관리시 가장 기본이 되는 설정 정보

# logging level

<https://stackoverflow.com/questions/2031163/when-to-use-the-different-log-levels>

Level	개요	예시
debug	개발시 처리 기록을 남겨야하는 로그 정보를 남김	<ul style="list-style-type: none"><li>- 다음 함수로 A 를 호출함</li><li>- 변수 A 를 무엇으로 변경함</li></ul>
info	처리가 진행되는 동안의 정보를 알림	<ul style="list-style-type: none"><li>- 서버가 시작되었음</li><li>- 서버가 종료됨</li><li>- 사용자 A가 프로그램에 접속함</li></ul>
warning	사용자가 잘못 입력한 정보나 처리는 가능하나 원래 개발시 의도치 않는 정보가 들어왔을 때 알림	<ul style="list-style-type: none"><li>- Str입력을 기대했으나, Int가 입력됨 ➔ Str casting으로 처리함</li><li>- 함수에 argument로 이차원 리스트를 기대했으나 ➔ 일차원 리스트가 들어옴, 이차원으로 변환후 처리</li></ul>
error	잘못된 처리로 인해 에러가 났으나, 프로그램은 동작할 수 있음을 알림	<ul style="list-style-type: none"><li>- 파일에 기록을 해야하는데 파일이 없음 --&gt; Exception 처리후 사용자에게 알림</li><li>- 외부서비스와 연결 불가</li></ul>
critical	잘못된 처리로 데이터 손실이나 더이상 프로그램이 동작할 수 없음을 알림	<ul style="list-style-type: none"><li>- 잘못된 접근으로 해당 파일이 삭제됨</li><li>- 사용자의 의한 강제 종료</li></ul>



# logging level

```
import logging
```

```
logger = logging.getLogger("main")
```

Logger 선언

```
stream_handler = logging.StreamHandler()
```

Logger의 output 방법 선언

```
logger.addHandler(stream_handler)
```

Logger의 output 등록

```
logger.setLevel(logging.DEBUG)
```

```
logger.debug("틀렸잖아!")
```

```
logger.info("확인해")
```

```
logger.warning("조심해!")
```

```
logger.error("에러났어!!!")
```

```
logger.critical("망했다...")
```

```
logger.setLevel(logging.CRITICAL)
```

```
logger.debug("틀렸잖아!")
```

```
logger.info("확인해")
```

```
logger.warning("조심해!")
```

```
logger.error("에러났어!!!")
```

```
logger.critical("망했다...")
```



**Human knowledge belongs to the world.**

# Configparser & Argparser

## Logging

최성철 교수  
Director of TEAMLAB

01100  
00110

**실제 프로그램을 실행할 때  
여러 설정이 필요**

**데이터 파일 위치**

**파일 저장 장소**

**Operation Type 등**

**이러한 정보를  
설정해줄 방법이 필요**

**1) Configparser - 파일에**

**2) Argparser - 실행시점에**

# Configparser



---

# Configparser

- 프로그램의 실행 설정을 file에 저장함
- Section, Key, Value 값의 형태로 설정된 설정 파일을 사용
- 설정파일을 Dict Type으로 호출후 사용

# Config file

[SectionOne]  
Status: Single  
Name: Derek  
Value: Yes  
Age: 30  
Single: True

Section- 대괄호

속성 - Key : Value

[SectionTwo]  
FavoriteColor = Green

[SectionThree]  
FamilyName: Johnson

# Configparser file

```
import configparser
config = configparser.ConfigParser()
config.sections()

config.read('example.cfg')
config.sections()

for key in config['SectionOne']:
    print(key)

config['SectionOne']['status']
```

`'example.cfg'`

**[SectionOne]**  
**Status: Single**  
**Name: Derek**  
**Value: Yes**  
**Age: 30**  
**Single: True**

**[SectionTwo]**  
**FavoriteColor = Green**

**[SectionThree]**  
**FamilyName: Johnson**

# Argparser

---

# Argparser

- Console 창에서 프로그램 실행시 Setting 정보를 저장함
- 거의 모든 Console 기반 Python 프로그램 기본으로 제공
- 특수 모듈도 많이 존재하지만(TF), 일반적으로 argparse를 사용
- Command-Line Option 이라고 부름

C:\WINDOWS\system32\cmd.exe

```
F:\workspace\introduction_to_pythoy_TEAMLAB_MOOC\code>python --v
Unknown option: --
usage: python [option] ... [-c cmd | -m mod | file | -] [arg] ...
Try `python -h' for more information.
```

```
F:\workspace\introduction_to_pythoy_TEAMLAB_MOOC\code>python -h
usage: python [option] ... [-c cmd | -m mod | file | -] [arg] ...
Options and arguments (and corresponding environment variables):
-b          : issue warnings about str(bytes_instance), str(bytearray_instance)
              and comparing bytes/bytearray with str. (-bb: issue errors)
-B          : don't write .py[co] files on import; also PYTHONDONTWRITEBYTECODE=x
-c cmd      : program passed in as string (terminates option list)
-d          : debug output from parser; also PYTHONDEBUG=x
-E          : ignore PYTHON* environment variables (such as PYTHONPATH)
-h          : print this help message and exit (also --help)
-i          : inspect interactively after running script; forces a prompt even
              if stdin does not appear to be a terminal; also PYTHONINSPECT=x
-l          : isolate Python from the user's environment (implies -E and -s)
-m mod      : run library module as a script (terminates option list)
-O          : optimize generated bytecode slightly; also PYTHONOPTIMIZE=x
-OO         : remove doc-strings in addition to the -O optimizations
-q          : don't print version and copyright messages on interactive startup
-s          : don't add user site directory to sys.path; also PYTHONNOUSERSITE
-S          : don't imply 'import site' on initialization
-u          : unbuffered binary stdout and stderr, stdin always buffered;
              also PYTHONUNBUFFERED=x
              see man page for details on internal buffering relating to '-u'
-v          : verbose (trace import statements); also PYTHONVERBOSE=x
              can be supplied multiple times to increase verbosity
```

# Argparser

```
import argparse
```

```
parser = argparse.ArgumentParser(description='Sum two integers.')
```

	짧은 이름	긴 이름	표시명	Help 설명	Argument Type	
parser.add_argument('	-a'		--a_value"	dest="A_value"	help="A integers"	type=int)
parser.add_argument('	-b'		--b_value"	dest="B_value"	help="B integers"	type=int)

```
args = parser.parse_args()
print(args)
print(args.a)
print(args.b)
print(args.a + args.b)
```



**Human knowledge belongs to the world.**

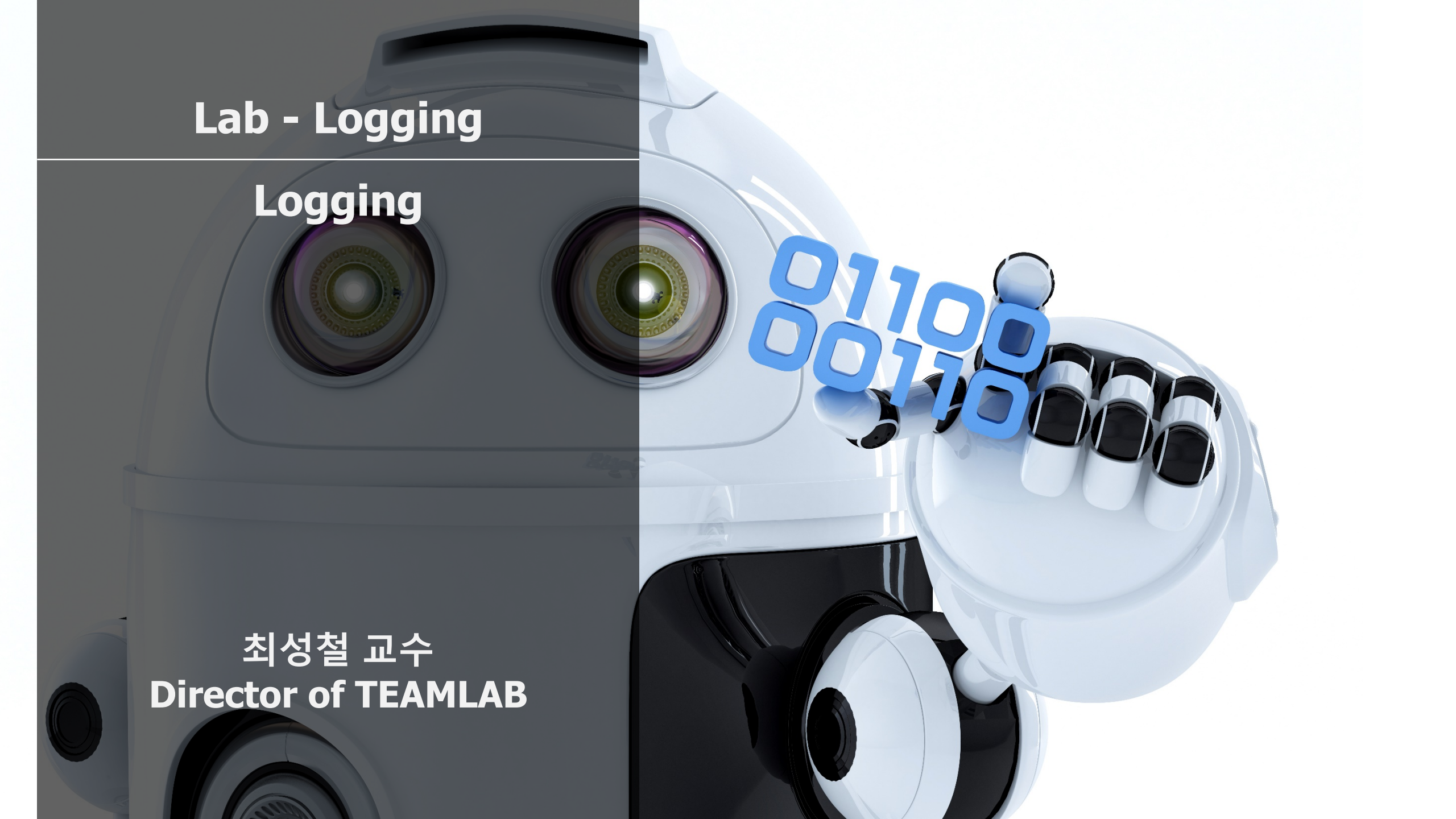


**Lab - Logging**

**Logging**

**최성철 교수**  
**Director of TEAMLAB**

01100  
00110



---

# Logging formmater

- Log의 결과값의 format을 지정해줄 수 있음

```
formatter =  
logging.Formatter('%(asctime)s %(levelname)s %(process)d %(message)s')
```

```
2018-01-18 22:47:04,385 ERROR 4410 ERROR occurred  
2018-01-18 22:47:22,458 ERROR 4439 ERROR occurred  
2018-01-18 22:47:22,458 INFO 4439 HERE WE ARE  
2018-01-18 22:47:24,680 ERROR 4443 ERROR occurred  
2018-01-18 22:47:24,681 INFO 4443 HERE WE ARE  
2018-01-18 22:47:24,970 ERROR 4445 ERROR occurred  
2018-01-18 22:47:24,970 INFO 4445 HERE WE ARE
```

# Log config file

```
logging.config.fileConfig('logging.conf')  
logger = logging.getLogger()
```

```
[loggers]
```

```
keys=root
```

```
[handlers]
```

```
keys=consoleHandler
```

```
[formatters]
```

```
keys=simpleFormatter
```

```
[logger_root]
```

```
level=DEBUG
```

```
handlers=consoleHandler
```

```
[handler_consoleHandler]
```

```
class=StreamHandler
```

```
level=DEBUG
```

```
formatter=simpleFormatter
```

```
args=(sys.stdout,)
```

# Logging examples

```
logger.info('Open file {0}'.format("customers.csv",))
try:
    with open("customers.csv", "r") as customer_data:
        customer_reader = csv.reader(customer_data, delimiter=',', quotechar='"')
        for customer in customer_reader:
            if customer[10].upper() == "USA": #customer 데이터의 offset 10번째 값
                logger.info('ID {0} added'.format(customer[0],))
                customer_USA_only_list.append(customer) #즉 country 필드가 "USA" 것만
except FileNotFoundError as e:
    logger.error('File NOT found {0}'.format(e,))
```



**Human knowledge belongs to the world.**