In [1]:

```
import pandas as pd
```

In [2]:

```
pip install pandas
```

```
Requirement already satisfied: pandas in /opt/conda/lib/python3.10/site-package
s (1.5.2)
Requirement already satisfied: python-dateutil>=2.8.1 in /opt/conda/lib/python
3.10/site-packages (from pandas) (2.8.2)
Requirement already satisfied: pytz>=2020.1 in /opt/conda/lib/python3.10/site-p
ackages (from pandas) (2022.6)
Requirement already satisfied: numpy>=1.21.0 in /opt/conda/lib/python3.10/site-
packages (from pandas) (1.23.5)
Requirement already satisfied: six>=1.5 in /opt/conda/lib/python3.10/site-packa
ges (from python-dateutil>=2.8.1->pandas) (1.16.0)
Note: you may need to restart the kernel to use updated packages.
```

In [3]:

```
# to read a file in pandas csv file
df = pd.read_csv("services.csv")
```

In [4]:

```
# in pandas head is give 1st 5 data bidiffault
df.head()
```

Out[4]:

| | id | location_id | program_id | accepted_payments | alternate_name | application_process | audi |
|---|---|---|---|---|---|---|---|
| **0** | 1 | 1 | NaN | NaN | NaN | Walk in or apply by phone. | ( adult: 55 or e minori |
| **1** | 2 | 2 | NaN | NaN | NaN | Apply by phone for an appointment. | Resic o M Cc age |
| **2** | 3 | 3 | NaN | NaN | NaN | Phone for information (403-4300 Ext. 4322). | ( adult: 55 or wh benef |
| **3** | 4 | 4 | NaN | NaN | NaN | Apply by phone. | Pa chil far prob |
| **4** | 5 | 5 | NaN | NaN | NaN | Phone for information. | in wo far chi t |

5 rows × 22 columns

```
In [5]: # to read only 3 data
        df.head(3)
```

Out[5]:

| | id | location_id | program_id | accepted_payments | alternate_name | application_process | audi |
|---|---|---|---|---|---|---|---|
| **0** | 1 | 1 | NaN | NaN | NaN | Walk in or apply by phone. | (<br>adult<br>55 or<br>e<br>minori |
| **1** | 2 | 2 | NaN | NaN | NaN | Apply by phone for an appointment. | Resi<br>o<br>M<br>Co<br>age |
| **2** | 3 | 3 | NaN | NaN | NaN | Phone for information (403-4300 Ext. 4322). | (<br>adult<br>55 or<br>wh<br>benef |

3 rows × 22 columns

In [6]:

```
# to read last 5 data bidifault
df.tail()
```

Out[6]:

| | id | location_id | program_id | accepted_payments | alternate_name | application_process | au |
|---|---|---|---|---|---|---|---|
| **18** | 19 | 19 | NaN | NaN | NaN | Call for screening appointment (650-347-3648). | |
| **19** | 20 | 20 | NaN | NaN | NaN | Walk in. | |
| **20** | 21 | 21 | NaN | NaN | NaN | By phone during business hours. | |
| **21** | 22 | 22 | NaN | Cash, Check, Credit Card | Fotos para pasaportes | Walk in or apply by phone or mail | Pro<br>no<br>busi<br>the |
| **22** | 23 | 22 | NaN | NaN | NaN | Walk in or apply by phone or mail | S<br>s<br>no<br>busi |

5 rows × 22 columns

In [7]:

```
# to read last 3 data
df.tail(3)
```

Out[7]:

| | id | location_id | program_id | accepted_payments | alternate_name | application_process | au |
|---|---|---|---|---|---|---|---|
| **20** | 21 | 21 | NaN | NaN | NaN | By phone during business hours. | |
| **21** | 22 | 22 | NaN | Cash, Check, Credit Card | Fotos para pasaportes | Walk in or apply by phone or mail | Pro no busi the |
| **22** | 23 | 22 | NaN | NaN | NaN | Walk in or apply by phone or mail | S s no busi |

3 rows × 22 columns

In [8]:
```
type(df)
```

Out[8]:
```
pandas.core.frame.DataFrame
```

In [9]:
```
# To see all the columns name in a list
list(df.columns)
```

Out[9]:
```
['id',
 'location_id',
 'program_id',
 'accepted_payments',
 'alternate_name',
 'application_process',
 'audience',
 'description',
 'eligibility',
 'email',
 'fees',
 'funding_sources',
 'interpretation_services',
 'keywords',
 'languages',
 'name',
 'required_documents',
 'service_areas',
 'status',
 'wait_time',
 'website',
 'taxonomy_ids']
```

In [10]:
```
# to get one column data
df['service_areas']
```

Out[10]:

```
0                                     Colma
1                          San Mateo County
2                          San Mateo County
3                          San Mateo County
4                          San Mateo County
5                          San Mateo County
6       Belmont, Burlingame, East Palo Alto
7                   Belmont, East Palo Alto
8                          San Mateo County
9                          San Mateo County
10                         San Mateo County
11                                Daly City
12                         San Mateo County
13      Belmont, Burlingame, East Palo Alto
14         Alameda County, San Mateo County
15                                      NaN
16    Colma, Daly City, South San Francisco
17                           East Palo Alto
18                      Belmont, Burlingame
19                                      NaN
20                         San Mateo County
21         Alameda County, San Mateo County
22         San Mateo County, Alameda County
Name: service_areas, dtype: object
```

In [11]:

```
type(df['service_areas'])
```

Out[11]:

```
pandas.core.series.Series
```

In [12]:

```
list(df['service_areas'])
```

Out[12]:

```
['Colma',
 'San Mateo County',
 'San Mateo County',
 'San Mateo County',
 'San Mateo County',
 'San Mateo County',
 'Belmont, Burlingame, East Palo Alto',
 'Belmont, East Palo Alto',
 'San Mateo County',
 'San Mateo County',
 'San Mateo County',
 'Daly City',
 'San Mateo County',
 'Belmont, Burlingame, East Palo Alto',
 'Alameda County, San Mateo County',
 nan,
 'Colma, Daly City, South San Francisco',
 'East Palo Alto',
 'Belmont, Burlingame',
 nan,
 'San Mateo County',
 'Alameda County, San Mateo County',
 'San Mateo County, Alameda County']
```

In [13]:

```
df[['service_areas']]
```

Out[13]:

| | service_areas |
|---|---|
| **0** | Colma |
| **1** | San Mateo County |
| **2** | San Mateo County |
| **3** | San Mateo County |
| **4** | San Mateo County |
| **5** | San Mateo County |
| **6** | Belmont, Burlingame, East Palo Alto |
| **7** | Belmont, East Palo Alto |
| **8** | San Mateo County |
| **9** | San Mateo County |
| **10** | San Mateo County |
| **11** | Daly City |
| **12** | San Mateo County |
| **13** | Belmont, Burlingame, East Palo Alto |
| **14** | Alameda County, San Mateo County |
| **15** | NaN |
| **16** | Colma, Daly City, South San Francisco |
| **17** | East Palo Alto |
| **18** | Belmont, Burlingame |
| **19** | NaN |
| **20** | San Mateo County |
| **21** | Alameda County, San Mateo County |
| **22** | San Mateo County, Alameda County |

In [14]:

```
type(df[['service_areas']])
```

Out[14]:

```
pandas.core.frame.DataFrame
```

In [15]:

```
# a column is simply passing then it's type is series
# if a column is passing inside a list then it's type is dataframe
```

In [16]:

```
df.columns
```

Out[16]:

```
Index(['id', 'location_id', 'program_id', 'accepted_payments',
       'alternate_name', 'application_process', 'audience', 'description',
       'eligibility', 'email', 'fees', 'funding_sources',
       'interpretation_services', 'keywords', 'languages', 'name',
       'required_documents', 'service_areas', 'status', 'wait_time', 'website',
       'taxonomy_ids'],
      dtype='object')
```

In [17]:
```
# to get data from multiple column then all the columns name is passing through
df[['email','keywords','name']]
```

Out[17]:

| | email | keywords | name |
|---|---|---|---|
| **0** | NaN | ADULT PROTECTION AND CARE SERVICES, Meal Sites... | Fair Oaks Adult Activity Center |
| **1** | NaN | EMPLOYMENT/TRAINING SERVICES, Job Development,... | Second Career Employment Program |
| **2** | NaN | Geriatric Counseling, Older Adults, Gay, Lesbi... | Senior Peer Counseling |
| **3** | NaN | INDIVIDUAL AND FAMILY DEVELOPMENT SERVICES, Gr... | Family Visitation Center |
| **4** | NaN | COMMUNITY SERVICES, Speakers, Automobile Loans | Economic Self-Sufficiency Program |
| **5** | NaN | ADULT PROTECTION AND CARE SERVICES, In-Home Su... | Little House Recreational Activities |
| **6** | NaN | ADULT PROTECTION AND CARE SERVICES, Adult Day ... | Rosener House Adult Day Services |
| **7** | NaN | ADULT PROTECTION AND CARE SERVICES, Meal Sites... | Meals on Wheels - South County |
| **8** | NaN | EDUCATION SERVICES, Library, Libraries, Public... | Fair Oaks Branch |
| **9** | NaN | EDUCATION SERVICES, Library, Libraries, Public... | Main Library |
| **10** | NaN | EDUCATION SERVICES, Library, Libraries, Public... | Schaberg Branch |
| **11** | NaN | EDUCATION SERVICES, Adult, Alternative, Litera... | Project Read |
| **12** | NaN | EDUCATION SERVICES, Library, Libraries, Public... | Redwood Shores Branch |
| **13** | NaN | COMMUNITY SERVICES, Interpretation/Translation... | Redwood City Corps |
| **14** | NaN | ALCOHOLISM SERVICES, Residential Care, DRUG AB... | Adult Rehabilitation Center |
| **15** | NaN | COMMODITY SERVICES, Clothing/Personal Items, C... | Sunnyvale Corps |
| **16** | NaN | COMMODITY SERVICES, Clothing/Personal Items, C... | South San Francisco Citadel Corps |
| **17** | NaN | HEALTH SERVICES, Outpatient Care, Community Cl... | Project Smile |
| **18** | NaN | HEALTH SERVICES, Outpatient Care, Community Cl... | San Mateo Free Medical Clinic |
| **19** | NaN | NaN | Service with blank fields |
| **20** | NaN | NaN | Service for Admin Test Location |
| **21** | passports@example.org | Salud, Medicina | Passport Photos |
| **22** | NaN | Ruby on Rails/Postgres/Redis, testing, wic | Example Service Name |

In [18]:

```python
# to see what type of data present in each columns in datafreams
# here object means string

df.dtypes
```

Out[18]:

```
id                           int64
location_id                  int64
program_id                 float64
accepted_payments           object
alternate_name              object
application_process         object
audience                    object
description                 object
eligibility                 object
email                       object
fees                        object
funding_sources             object
interpretation_services     object
keywords                    object
languages                   object
name                        object
required_documents          object
service_areas               object
status                      object
wait_time                   object
website                     object
taxonomy_ids                object
dtype: object
```

In [19]:

```python
# read data from excel file
df1 = pd.read_excel("LUSID Excel - Setting up your market data.xlsx")
```

In [20]:

```python
type(df1)
```

Out[20]:

```
pandas.core.frame.DataFrame
```

In [21]:

```python
# to see what type of data present in each columns in datafreams
# here object means string

df1.dtypes
```

Out[21]:

```
Unnamed: 0    float64
Unnamed: 1    float64
Unnamed: 2    float64
Unnamed: 3     object
Unnamed: 4     object
Unnamed: 5     object
Unnamed: 6    float64
Unnamed: 7     object
Unnamed: 8     object
Unnamed: 9     object
dtype: object
```

In [22]:

```python
df1.columns
```

Out[22]:

```
Index(['Unnamed: 0', 'Unnamed: 1', 'Unnamed: 2', 'Unnamed: 3', 'Unnamed: 4',
       'Unnamed: 5', 'Unnamed: 6', 'Unnamed: 7', 'Unnamed: 8', 'Unnamed: 9'],
      dtype='object')
```

In [23]:
```
df1[['Unnamed: 6','Unnamed: 7']]
```

Out[23]:

| | Unnamed: 6 | Unnamed: 7 |
|---|---|---|
| **0** | NaN | NaN |
| **1** | NaN | NaN |
| **2** | NaN | NaN |
| **3** | NaN | NaN |
| **4** | NaN | NaN |
| **5** | NaN | NaN |
| **6** | NaN | NaN |
| **7** | NaN | NaN |
| **8** | NaN | NaN |
| **9** | NaN | NaN |
| **10** | NaN | NaN |
| **11** | NaN | LUSID also accepts UTC, UTS offsets and cutlabels |
| **12** | NaN | NaN |
| **13** | NaN | Additional formats |
| **14** | NaN | NaN |
| **15** | NaN | Standard UTC timestamp: |
| **16** | NaN | NaN |
| **17** | NaN | Convert date into a string |
| **18** | NaN | NaN |
| **19** | NaN | UTC offset (not recognised as an excel date) |
| **20** | NaN | NaN |
| **21** | NaN | Cutlabel |
| **22** | NaN | NaN |
| **23** | NaN | NaN |
| **24** | NaN | NaN |
| **25** | NaN | NaN |
| **26** | NaN | NaN |
| **27** | NaN | NaN |

In [24]:
```
# to read the data from web link
df2 = pd.read_csv("https://raw.githubusercontent.com/datasciencedojo/datasets/ma
```

In [25]:
```
df2.head(3)
```

Out[25]:

| | PassengerId | Survived | Pclass | Name | Sex | Age | SibSp | Parch | Ticket | Fare | Ca |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 1 | 0 | 3 | Braund, Mr. Owen Harris | male | 22.0 | 1 | 0 | A/5 21171 | 7.2500 | I |
| **1** | 2 | 1 | 1 | Cumings, Mrs. John Bradley (Florence Briggs Th... | female | 38.0 | 1 | 0 | PC 17599 | 71.2833 | |
| **2** | 3 | 1 | 3 | Heikkinen, Miss. Laina | female | 26.0 | 0 | 0 | STON/O2. 3101282 | 7.9250 | I |

In [26]:
```
df2.columns
```

Out[26]:
```
Index(['PassengerId', 'Survived', 'Pclass', 'Name', 'Sex', 'Age', 'SibSp',
       'Parch', 'Ticket', 'Fare', 'Cabin', 'Embarked'],
      dtype='object')
```

In [27]:
```
type(df2)
```

Out[27]:
```
pandas.core.frame.DataFrame
```

In [28]:
```
df2['Survived']
```

Out[28]:
```
0      0
1      1
2      1
3      1
4      0
      ..
886    0
887    1
888    0
889    1
890    0
Name: Survived, Length: 891, dtype: int64
```

In [29]:
```
type(df2['Survived'])
```

Out[29]:
```
pandas.core.series.Series
```

In [30]:
```
df2[['Survived', 'Pclass', 'Name']]
```

Out[30]:

| | Survived | Pclass | Name |
|---|---|---|---|
| **0** | 0 | 3 | Braund, Mr. Owen Harris |
| **1** | 1 | 1 | Cumings, Mrs. John Bradley (Florence Briggs Th... |
| **2** | 1 | 3 | Heikkinen, Miss. Laina |
| **3** | 1 | 1 | Futrelle, Mrs. Jacques Heath (Lily May Peel) |
| **4** | 0 | 3 | Allen, Mr. William Henry |
| **...** | ... | ... | ... |
| **886** | 0 | 2 | Montvila, Rev. Juozas |
| **887** | 1 | 1 | Graham, Miss. Margaret Edith |
| **888** | 0 | 3 | Johnston, Miss. Catherine Helen "Carrie" |
| **889** | 1 | 1 | Behr, Mr. Karl Howell |
| **890** | 0 | 3 | Dooley, Mr. Patrick |

891 rows × 3 columns

In [31]:

```
df2.tail(3)
```

Out[31]:

| | PassengerId | Survived | Pclass | Name | Sex | Age | SibSp | Parch | Ticket | Fare | Cabi |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **888** | 889 | 0 | 3 | Johnston, Miss. Catherine Helen "Carrie" | female | NaN | 1 | 2 | W./C. 6607 | 23.45 | Nal |
| **889** | 890 | 1 | 1 | Behr, Mr. Karl Howell | male | 26.0 | 0 | 0 | 111369 | 30.00 | C14 |
| **890** | 891 | 0 | 3 | Dooley, Mr. Patrick | male | 32.0 | 0 | 0 | 370376 | 7.75 | Nal |

In [32]:

```
# to get the data from html
import lxml
url_data = pd.read_html("https://www.basketball-reference.com/leagues/NBA_2015_t
```

In [33]:

```
pip install lxml
```

```
Requirement already satisfied: lxml in /opt/conda/lib/python3.10/site-packages
(4.9.2)
Note: you may need to restart the kernel to use updated packages.
```

In [34]:

```
type(url_data)
```

Out[34]:

```
list
```

In [38]:

```
# how many table is present
len(url_data)
```

```
Out[38]: 1
```

```
In [36]: df3 = url_data[0]
```

```
In [37]: df3
```

Out[37]:

| | Rk | Player | Pos | Age | Tm | G | GS | MP | FG | FGA | ... | FT% | ORB | DRB | TRB | AS |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 1 | Quincy Acy | PF | 24 | NYK | 68 | 22 | 1287 | 152 | 331 | ... | .784 | 79 | 222 | 301 | 6 |
| **1** | 2 | Jordan Adams | SG | 20 | MEM | 30 | 0 | 248 | 35 | 86 | ... | .609 | 9 | 19 | 28 | 1 |
| **2** | 3 | Steven Adams | C | 21 | OKC | 70 | 67 | 1771 | 217 | 399 | ... | .502 | 199 | 324 | 523 | 6 |
| **3** | 4 | Jeff Adrien | PF | 28 | MIN | 17 | 0 | 215 | 19 | 44 | ... | .579 | 23 | 54 | 77 | 1 |
| **4** | 5 | Arron Afflalo | SG | 29 | TOT | 78 | 72 | 2502 | 375 | 884 | ... | .843 | 27 | 220 | 247 | 12 |
| **...** | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | |
| **670** | 490 | Thaddeus Young | PF | 26 | TOT | 76 | 68 | 2434 | 451 | 968 | ... | .655 | 127 | 284 | 411 | 17 |
| **671** | 490 | Thaddeus Young | PF | 26 | MIN | 48 | 48 | 1605 | 289 | 641 | ... | .682 | 75 | 170 | 245 | 13 |
| **672** | 490 | Thaddeus Young | PF | 26 | BRK | 28 | 20 | 829 | 162 | 327 | ... | .606 | 52 | 114 | 166 | 3 |
| **673** | 491 | Cody Zeller | C | 22 | CHO | 62 | 45 | 1487 | 172 | 373 | ... | .774 | 97 | 265 | 362 | 10 |
| **674** | 492 | Tyler Zeller | C | 25 | BOS | 82 | 59 | 1731 | 340 | 619 | ... | .823 | 146 | 319 | 465 | 11 |

675 rows × 30 columns

```
In [39]: df3.head()
```

Out[39]:

| | Rk | Player | Pos | Age | Tm | G | GS | MP | FG | FGA | ... | FT% | ORB | DRB | TRB | AST | ST |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 1 | Quincy Acy | PF | 24 | NYK | 68 | 22 | 1287 | 152 | 331 | ... | .784 | 79 | 222 | 301 | 68 | 2 |
| **1** | 2 | Jordan Adams | SG | 20 | MEM | 30 | 0 | 248 | 35 | 86 | ... | .609 | 9 | 19 | 28 | 16 | 1 |
| **2** | 3 | Steven Adams | C | 21 | OKC | 70 | 67 | 1771 | 217 | 399 | ... | .502 | 199 | 324 | 523 | 66 | 3 |
| **3** | 4 | Jeff Adrien | PF | 28 | MIN | 17 | 0 | 215 | 19 | 44 | ... | .579 | 23 | 54 | 77 | 15 | |
| **4** | 5 | Arron Afflalo | SG | 29 | TOT | 78 | 72 | 2502 | 375 | 884 | ... | .843 | 27 | 220 | 247 | 129 | 4 |

5 rows × 30 columns

In [40]:
```
df3.tail(3)
```

Out[40]:

| | Rk | Player | Pos | Age | Tm | G | GS | MP | FG | FGA | ... | FT% | ORB | DRB | TRB | AS |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **672** | 490 | Thaddeus Young | PF | 26 | BRK | 28 | 20 | 829 | 162 | 327 | ... | .606 | 52 | 114 | 166 | 3 |
| **673** | 491 | Cody Zeller | C | 22 | CHO | 62 | 45 | 1487 | 172 | 373 | ... | .774 | 97 | 265 | 362 | 10 |
| **674** | 492 | Tyler Zeller | C | 25 | BOS | 82 | 59 | 1731 | 340 | 619 | ... | .823 | 146 | 319 | 465 | 11 |

3 rows × 30 columns

In [41]:
```
df3.columns
```

Out[41]:
```
Index(['Rk', 'Player', 'Pos', 'Age', 'Tm', 'G', 'GS', 'MP', 'FG', 'FGA', 'FG%',
       '3P', '3PA', '3P%', '2P', '2PA', '2P%', 'eFG%', 'FT', 'FTA', 'FT%',
       'ORB', 'DRB', 'TRB', 'AST', 'STL', 'BLK', 'TOV', 'PF', 'PTS'],
      dtype='object')
```

In [42]:
```
df3.dtypes
```

Out[42]:

```
Rk        object
Player    object
Pos       object
Age       object
Tm        object
G         object
GS        object
MP        object
FG        object
FGA       object
FG%       object
3P        object
3PA       object
3P%       object
2P        object
2PA       object
2P%       object
eFG%      object
FT        object
FTA       object
FT%       object
ORB       object
DRB       object
TRB       object
AST       object
STL       object
BLK       object
TOV       object
PF        object
PTS       object
dtype: object
```

In [43]:

```
df3[['Pos', 'Age', 'Tm']]
```

Out[43]:

| | Pos | Age | Tm |
|---|---|---|---|
| **0** | PF | 24 | NYK |
| **1** | SG | 20 | MEM |
| **2** | C | 21 | OKC |
| **3** | PF | 28 | MIN |
| **4** | SG | 29 | TOT |
| **...** | ... | ... | ... |
| **670** | PF | 26 | TOT |
| **671** | PF | 26 | MIN |
| **672** | PF | 26 | BRK |
| **673** | C | 22 | CHO |
| **674** | C | 25 | BOS |

675 rows × 3 columns

In [45]:

```
# to store the data in my local system or in a file
# here index = False use for not store the index value. bidifault it is True
```

```
df3.to_csv("players_data.csv",index = False)
```

In [ ]:

```
# DataFrem is a tabular Structure
```