*Import pandas as pd ..*Import lib

**DataFrame:**

* data = {colname1 : [values\_of\_col1],

colname2: [values\_of\_col2]

}

* pd.DataFrame (data, index = [values\_of\_indexes])

**Save to Csv file:**

* variable.to\_csv('file\_name.csv', index=True)

**Combining two Csv:**

* Pd.concat([Variable\_One , Variable\_Two])

**Data Wrangling:**

* pd.read\_csv('path of csv file') ... call file
* df = pd.read\_csv('path of csv file') ... create dataframe
* df.shape … No’s of rows and column
* df.columns.tolist() ……. *extract* all the column names as a list
* df.head() or df.head(6) ... top lines of df
* df.tail() ...last lines of df
* df.sample() ... random line of df
* df.info() ... analyse the data type and not null values
* df.dtypes .... analyse data type
* df.isnull() .. empty cells
* df.isnull.sum()
* df.notnull() ... non empty cells

* df.count()
* df.value\_counts()
* df.mean()
* df.median()
* df.std()
* df.max()
* df[ 'Col\_Name' ].argmax() … *to identify the row index*
* df.min()
* df.describe() ... mean, median, mode, std, min, max (all in one command)
* df.rename(columns = {'column name of df' : 'updated name'}, inplace = True).. rename col name of dataframe

new\_df.rename\_axis('RowName' , axis = 'rows').rename\_axis('ColName' , axis = 'columns',inplace=True)

(to rename rows and columns name which is predefined)

* df.truncate() ... del all values
* df.drop(columns = 'col name') ... dell columns
* df.dropna() … del empty cells
* df.fillna(value\_which\_want\_to\_fill, inplace = True)…fill empty cells
* df.duplicated()…To check duplicate rows

**Indexing:**

* df.column\_name or df[’Column\_name’]…. To call selected column only
* df[‘column\_name’][n] … To call selected column nth value
* df.iloc[n] … To call all values of n row
* df.iloc[ : n] … To call all rows of dataframe till nth value
* df.loc[index\_value\_should\_be\_numeric\_or\_list , column\_name\_in\_table]

cols = ['country', 'province', 'region\_1', 'region\_2']

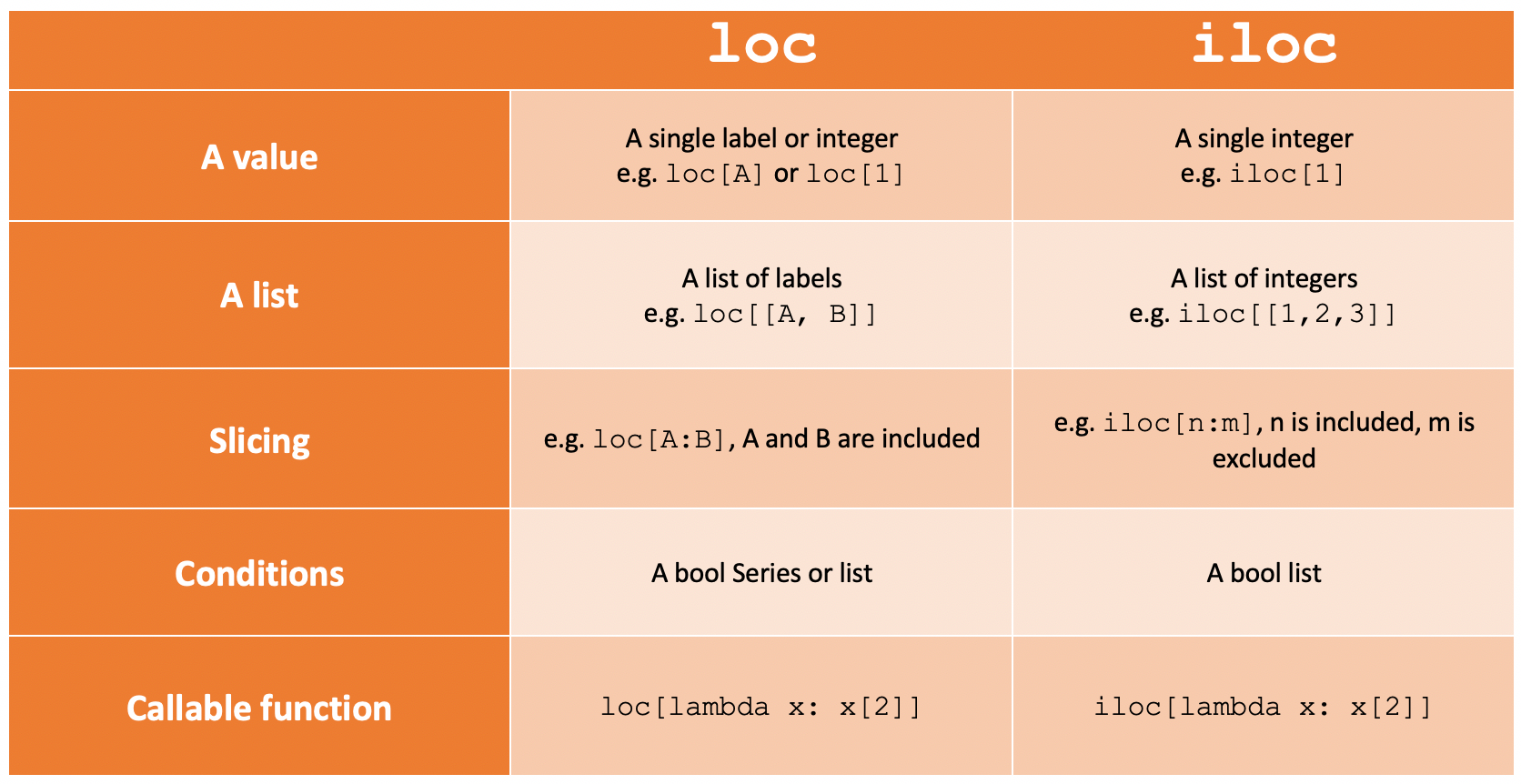
indices = [0, 1, 10, 100]

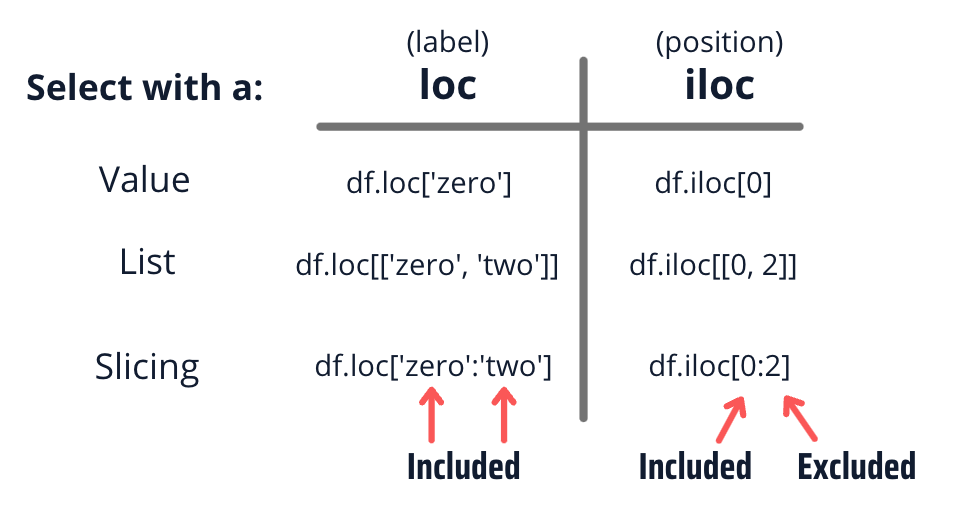
df = reviews.loc[indices,cols]

* df.loc[:nth\_col, column\_name\_in\_table\_or\_list\_columns\_name

cols = ['country', 'variety']

df = reviews.loc[:99, cols]

* sample\_list [1,3,4,8]…………….df.iloc[sample\_list] … To call selected rows
* # Conditional Indexing   
  sample\_1 = reviews[(reviews['Pclass'] >= 2) & (reviews['Gender'] == 'male') & (reviews['Age'] > 20)]  
  sample\_1
* df.iloc[[df['Col\_Name'].argmax()]] ….. to get attributes about the game, we need to use the iloc[] function



**Maping:**