Exploring Pandas for Data Analysis

```
import pandas as pd
```

```
employee_df=pd.read_csv("/content/employee_information.csv")
employee_df
```

	First Name	Last Name	Salary	Years with Company	Postal Code	Email
0	Mike	Moe	5000.00	3	N94 3M0	bird@gmail.com
1	Noah	Ryan	10000.00	8	N8S 14K	nsmall@hotmail.com
2	Nina	Keller	9072.02	17	S1T 4E6	azikez@gahew.mr
3	Chanel	Steve	11072.02	12	N7T 3E6	chanel@gmail.com
4	Kate	Noor	5000.00	23	K8N 5H6	kate@hotmail.com
5	Samer	Mo	100000.00	13	J7H 3HY	samer@gmail.com
6	Heba	Ismail	50000.00	7	K8Y 3M8	heba.ismail@hotmail.com
7	Laila	Aly	20000.00	5	J8Y 3M0	Laila.a@hotmail.com
8	Joseph	Patton	2629.13	2	M6U 5U7	daafeja@boh.jm
9	Noah	Moran	8626.96	11	K2D 4M9	guutodi@bigwoc.kw

```
house_prices_df=pd.read_html('https://www.livingin-canada.com/house-prices-canada.html')
house_prices_df[0]
```

	City	Average House Price	12 Month Change
0	Vancouver, BC	$1,036,000	+ 2.63 %
1	Toronto, Ont	$870,000	+10.2 %
2	Ottawa, Ont	$479,000	+ 15.4 %
3	Calgary, Alb	$410,000	− 1.5 %
4	Montreal, Que	$435,000	+ 9.3 %
5	Halifax, NS	$331,000	+ 3.6 %
6	Regina, Sask	$254,000	− 3.9 %
7	Fredericton, NB	$198,000	− 4.3 %
8	(adsbygoogle = window.adsbygoogle || []).push(...	(adsbygoogle = window.adsbygoogle || []).push(...	(adsbygoogle = window.adsbygoogle || []).push(...

```
house_prices_df[1]
```

	Province	Average House Price	12 Month Change
0	British Columbia	$736,000	+ 7.6 %
1	Ontario	$594,000	− 3.2 %
2	Alberta	$353,000	− 7.5 %
3	Quebec	$340,000	+ 7.6 %
4	Manitoba	$295,000	− 1.4 %
5	Saskatchewan	$271,000	− 3.8 %
6	Nova Scotia	$266,000	+ 3.5 %
7	Prince Edward Island	$243,000	+ 3.0 %
8	Newfoundland / Labrador	$236,000	− 1.6 %
9	New Brunswick	$183,000	− 2.2 %
10	Canadian Average	$488,000	− 1.3 %

```
employee_df.head()
```

	First Name	Last Name	Salary	Years with Company	Postal Code	Email
0	Mike	Moe	5000.00	3	N94 3M0	bird@gmail.com
1	Noah	Ryan	10000.00	8	N8S 14K	nsmall@hotmail.com
2	Nina	Keller	9072.02	17	S1T 4E6	azikez@gahew.mr
3	Chanel	Steve	11072.02	12	N7T 3E6	chanel@gmail.com

```
employee_df.columns
```

```
Index(['First Name', 'Last Name', 'Salary', 'Years with Company',
       'Postal Code', 'Email'],
      dtype='object')
```

```
email_series=employee_df['Email']
email_series
```

	Email
0	bird@gmail.com
1	nsmall@hotmail.com
2	azikez@gahew.mr
3	chanel@gmail.com
4	kate@hotmail.com
5	samer@gmail.com
6	heba.ismail@hotmail.com
7	Laila.a@hotmail.com
8	daafeja@boh.jm
9	guutodi@bigwoc.kw

dtype: object

```
employee_df['Years with Company']
```

	Years with Company
0	3
1	8
2	17
3	12
4	23
5	13
6	7
7	5
8	2
9	11

dtype: int64

```
Name_Salary_df=employee_df[['First Name','Salary']]
Name_Salary_df
```

	First Name	Salary
0	Mike	5000.00
1	Noah	10000.00
2	Nina	9072.02
3	Chanel	11072.02
4	Kate	5000.00
5	Samer	100000.00
6	Heba	50000.00
7	Laila	20000.00
8	Joseph	2629.13
9	Noah	8626.96

```
employee_df['Age']=[25,30,34,35,56,44,32,35,28,29]
employee_df
```

	First Name	Last Name	Salary	Years with Company	Postal Code	Email	Age
0	Mike	Moe	5000.00	3	N94 3M0	bird@gmail.com	25
1	Noah	Ryan	10000.00	8	N8S 14K	nsmall@hotmail.com	30
2	Nina	Keller	9072.02	17	S1T 4E6	azikez@gahew.mr	34
3	Chanel	Steve	11072.02	12	N7T 3E6	chanel@gmail.com	35
4	Kate	Noor	5000.00	23	K8N 5H6	kate@hotmail.com	56
5	Samer	Mo	100000.00	13	J7H 3HY	samer@gmail.com	44
6	Heba	Ismail	50000.00	7	K8Y 3M8	heba.ismail@hotmail.com	32
7	Laila	Aly	20000.00	5	J8Y 3M0	Laila.a@hotmail.com	35
8	Joseph	Patton	2629.13	2	M6U 5U7	daafeja@boh.jm	28

```
employee_df.loc[employee_df['First Name'] == 'Noah']
```

	First Name	Last Name	Salary	Years with Company	Postal Code	Email	Age
1	Noah	Ryan	10000.00	8	N8S 14K	nsmall@hotmail.com	30

```
employee_df.iloc[0]
```

	0
First Name	Mike
Last Name	Moe
Salary	5000.0
Years with Company	3
Postal Code	N94 3M0
Email	bird@gmail.com
Age	25

dtype: object

```
employee_df.iloc[2:5]
```

	First Name	Last Name	Salary	Years with Company	Postal Code	Email	Age
2	Nina	Keller	9072.02	17	S1T 4E6	azikez@gahew.mr	34
3	Chanel	Steve	11072.02	12	N7T 3E6	chanel@gmail.com	35

```
employee_df.iloc[1,0:4]
```

	1
First Name	Noah
Last Name	Ryan
Salary	10000.0
Years with Company	8

dtype: object

```
employee_df.sample(n=3,axis=0)
```

	First Name	Last Name	Salary	Years with Company	Postal Code	Email	Age
5	Samer	Mo	100000.00	13	J7H 3HY	samer@gmail.com	44
1	Noah	Ryan	10000.00	8	N8S 14K	nsmall@hotmail.com	30

```
employee_df['Salary'].describe()
```

	Salary
count	10.000000
mean	22140.013000
std	30613.044026
min	2629.130000
25%	5906.740000
50%	9536.010000
75%	17768.005000
max	100000.000000

dtype: float64

```
employee_df['Salary'] = employee_df['Salary'] +1000
```

```
employee_df['Salary'].describe()
```

	Salary
count	10.000000
mean	23140.013000
std	30613.044026
min	3629.130000
25%	6906.740000
50%	10536.010000
75%	18768.005000
max	101000.000000

dtype: float64

```
employee_df.head()
```

	First Name	Last Name	Salary	Years with Company	Postal Code	Email	Age
0	Mike	Moe	6000.00	3	N94 3M0	bird@gmail.com	25
1	Noah	Ryan	11000.00	8	N8S 14K	nsmall@hotmail.com	30
2	Nina	Keller	10072.02	17	S1T 4E6	azikez@gahew.mr	34
3	Chanel	Steve	12072.02	12	N7T 3E6	chanel@gmail.com	35

```
employee_df.sort_values(by='Years with Company',ascending=False, inplace = True)
employee_df
```

	First Name	Last Name	Salary	Years with Company	Postal Code	Email	Age
4	Kate	Noor	6000.00	23	K8N 5H6	kate@hotmail.com	56
2	Nina	Keller	10072.02	17	S1T 4E6	azikez@gahew.mr	34
5	Samer	Mo	101000.00	13	J7H 3HY	samer@gmail.com	44
3	Chanel	Steve	12072.02	12	N7T 3E6	chanel@gmail.com	35
9	Noah	Moran	9626.96	11	K2D 4M9	guutodi@bigwoc.kw	29
1	Noah	Ryan	11000.00	8	N8S 14K	nsmall@hotmail.com	30
6	Heba	Ismail	51000.00	7	K8Y 3M8	heba.ismail@hotmail.com	32
7	Laila	Aly	21000.00	5	J8Y 3M0	Laila.a@hotmail.com	35
0	Mike	Moe	6000.00	3	N94 3M0	bird@gmail.com	25

```
#reset index
employee_df.reset_index(drop=True, inplace=True)
employee_df
```

	First Name	Last Name	Salary	Years with Company	Postal Code	Email	Age
0	Kate	Noor	6000.00	23	K8N 5H6	kate@hotmail.com	56
1	Nina	Keller	10072.02	17	S1T 4E6	azikez@gahew.mr	34
2	Samer	Mo	101000.00	13	J7H 3HY	samer@gmail.com	44
3	Chanel	Steve	12072.02	12	N7T 3E6	chanel@gmail.com	35
4	Noah	Moran	9626.96	11	K2D 4M9	guutodi@bigwoc.kw	29
5	Noah	Ryan	11000.00	8	N8S 14K	nsmall@hotmail.com	30
6	Heba	Ismail	51000.00	7	K8Y 3M8	heba.ismail@hotmail.com	32
7	Laila	Aly	21000.00	5	J8Y 3M0	Laila.a@hotmail.com	35
8	Mike	Moe	6000.00	3	N94 3M0	bird@gmail.com	25

```
#Define a function to increase salary by 10% for all employees
def salary_update(salary):
  return salary*1.1
```

```
employee_df['Salary']=employee_df['Salary'].apply(salary_update)
employee_df
```

	First Name	Last Name	Salary	Years with Company	Postal Code	Email	Age
0	Kate	Noor	6600.000	23	K8N 5H6	kate@hotmail.com	56
1	Nina	Keller	11079.222	17	S1T 4E6	azikez@gahew.mr	34
2	Samer	Mo	111100.000	13	J7H 3HY	samer@gmail.com	44
3	Chanel	Steve	13279.222	12	N7T 3E6	chanel@gmail.com	35
4	Noah	Moran	10589.656	11	K2D 4M9	guutodi@bigwoc.kw	29
5	Noah	Ryan	12100.000	8	N8S 14K	nsmall@hotmail.com	30
6	Heba	Ismail	56100.000	7	K8Y 3M8	heba.ismail@hotmail.com	32
7	Laila	Aly	23100.000	5	J8Y 3M0	Laila.a@hotmail.com	35
8	Mike	Moe	6600.000	3	N94 3M0	bird@gmail.com	25

```
df_loyalty=employee_df[employee_df['Years with Company']>10]
df_loyalty
```

	First Name	Last Name	Salary	Years with Company	Postal Code	Email	Age
0	Kate	Noor	6600.000	23	K8N 5H6	kate@hotmail.com	56
1	Nina	Keller	11079.222	17	S1T 4E6	azikez@gahew.mr	34
2	Samer	Mo	111100.000	13	J7H 3HY	samer@gmail.com	44
3	Chanel	Steve	13279.222	12	N7T 3E6	chanel@gmail.com	35

```
mask_1=employee_df['Years with Company']>10
mask_2=employee_df['Age']>30
df_loyalty=employee_df[mask_1 & mask_2]
df_loyalty
```

	First Name	Last Name	Salary	Years with Company	Postal Code	Email	Age
0	Kate	Noor	6600.000	23	K8N 5H6	kate@hotmail.com	56
1	Nina	Keller	11079.222	17	S1T 4E6	azikez@gahew.mr	34
2	Samer	Mo	111100.000	13	J7H 3HY	samer@gmail.com	44

```
df_high_salary=employee_df[employee_df['Salary']>60000]
df_high_salary
```

	First Name	Last Name	Salary	Years with Company	Postal Code	Email	Age

```
employee_df
```

```
  First Name Last Name      Salary  Years with Company  Postal Code                   Email  Age
0       Kate      Noor    6600.000                  23      K8N 5H6         kate@hotmail.com   56
1       Nina    Keller   11079.222                  17      S1T 4E6          azikez@gahew.mr   34
2      Samer        Mo  111100.000                  13      J7H 3HY         samer@gmail.com   44
3     Chanel     Steve   13279.222                  12      N7T 3E6        chanel@gmail.com   35
4       Noah     Moran   10589.656                  11      K2D 4M9        guutodi@bigwoc.kw   29
5       Noah      Ryan   12100.000                   8      N8S 14K      nsmall@hotmail.com   30
6       Heba    Ismail   56100.000                   7      K8Y 3M8  heba.ismail@hotmail.com  32
7      Laila       Aly   23100.000                   5      J8Y 3M0     Laila.a@hotmail.com   35
8       Mike       Moe    6600.000                   3      N94 3M0          bird@gmail.com   25
```

```
df_filtered = employee_df[employee_df['First Name']== 'Mike']
df_filtered
```

```
  First Name Last Name Salary Years with Company Postal Code     Email Age
```

```
mask = employee_df['Last Name'].isin(['Moe','Ryan'])
df_filtered = employee_df[mask]
df_filtered
```

```
  First Name Last Name  Salary Years with Company Postal Code              Email  Age
5       Noah      Ryan 12100.0                  8     N8S 14K nsmall@hotmail.com   30
```

```
employee_df[employee_df['Salary'].between(5000,9000)]
```

```
  First Name Last Name Salary Years with Company Postal Code            Email  Age
0       Kate      Noor 6600.0                 23     K8N 5H6 kate@hotmail.com   56
```

```
employee_df['First Name'].duplicated(keep=False)
```

```
  First Name
0      False
1      False
2      False
3      False
4       True
5       True
6      False
7      False
8      False
9      False

dtype: bool
```

```
#reversing
mask = ~employee_df['First Name'].duplicated(keep=False)
```

```
mask
```

	First Name
0	True
1	True
2	True
3	True
4	False
5	False
6	True
7	True
8	True
9	True

dtype: bool

```
employee_df[mask]
```

	First Name	Last Name	Salary	Years with Company	Postal Code	Email	Age
0	Kate	Noor	6600.000	23	K8N 5H6	kate@hotmail.com	56
1	Nina	Keller	11079.222	17	S1T 4E6	azikez@gahew.mr	34
2	Samer	Mo	111100.000	13	J7H 3HY	samer@gmail.com	44
3	Chanel	Steve	13279.222	12	N7T 3E6	chanel@gmail.com	35
6	Heba	Ismail	56100.000	7	K8Y 3M8	heba.ismail@hotmail.com	32
7	Laila	Aly	23100.000	5	J8Y 3M0	Laila.a@hotmail.com	35
8	Mike	Moe	6600.000	3	N94 3M0	bird@gmail.com	25

```
sales_df=pd.read_csv('/content/ecommerce_sales.csv', encoding='unicode_escape')
sales_df
```

	InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	CustomerID	Country
0	536365	85123A	WHITE HANGING HEART T-LIGHT HOLDER	6	12/1/2010 8:26	2.55	17850.0	United Kingdom
1	536365	71053	WHITE METAL LANTERN	6	12/1/2010 8:26	3.39	17850.0	United Kingdom
2	536365	84406B	CREAM CUPID HEARTS COAT HANGER	8	12/1/2010 8:26	2.75	17850.0	United Kingdom
3	536365	84029G	KNITTED UNION FLAG HOT WATER BOTTLE	6	12/1/2010 8:26	3.39	17850.0	United Kingdom
4	536365	84029E	RED WOOLLY HOTTIE WHITE HEART.	6	12/1/2010 8:26	3.39	17850.0	United Kingdom
...	...	...	...	...	...	...	...	...
541904	581587	22613	PACK OF 20 SPACEBOY NAPKINS	12	12/9/2011 12:50	0.85	12680.0	France
541905	581587	22899	CHILDREN'S APRON DOLLY GIRL	6	12/9/2011 12:50	2.10	12680.0	France
541906	581587	23254	CHILDRENS CUTLERY DOLLY GIRL	4	12/9/2011 12:50	4.15	12680.0	France
541907	581587	23255	CHILDRENS CUTLERY CIRCUS PARADE	4	12/9/2011 12:50	4.15	12680.0	France
541908	581587	22138	BAKING SET 9 PIECE RETROSPOT	3	12/9/2011 12:50	4.95	12680.0	France

```
sales_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 541909 entries, 0 to 541908
Data columns (total 8 columns):
 #   Column       Non-Null Count   Dtype  
---  ------       --------------   -----  
 0   InvoiceNo    541909 non-null  object 
 1   StockCode    541909 non-null  object 
 2   Description  540455 non-null  object 
 3   Quantity     541909 non-null  int64  
 4   InvoiceDate  541909 non-null  object 
 5   UnitPrice    541909 non-null  float64
 6   CustomerID   406829 non-null  float64
 7   Country      541909 non-null  object 
dtypes: float64(2), int64(1), object(5)
memory usage: 33.1+ MB
```

```
sales_df['InvoiceDate']=pd.to_datetime(sales_df['InvoiceDate'])
original_sales=sales_df.copy()
sales_df.head()
```

	InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	CustomerID	Country
0	536365	85123A	WHITE HANGING HEART T-LIGHT HOLDER	6	2010-12-01 08:26:00	2.55	17850.0	United Kingdom
1	536365	71053	WHITE METAL LANTERN	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom
2	536365	84406B	CREAM CUPID HEARTS COAT HANGER	8	2010-12-01 08:26:00	2.75	17850.0	United Kingdom
3	536365	84029G	KNITTED UNION FLAG HOT WATER BOTTLE	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom
4	536365	84029E	RED WOOLLY HOTTIE WHITE HEART.	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom

```
sales_df.isnull().sum()
```

```
                    0
InvoiceNo           0
StockCode           0
Description      1454
Quantity            0
InvoiceDate         0
UnitPrice           0
CustomerID     135080
Country             0

dtype: int64
```

```
sales_df.nunique()
```

```
                   0
InvoiceNo      25900
StockCode       4070
Description     4223
Quantity         722
InvoiceDate    23260
UnitPrice       1630
CustomerID      4372
Country           38

dtype: int64
```

```
sales_df['Country'].unique()
```

```
array(['United Kingdom', 'France', 'Australia', 'Netherlands', 'Germany',
       'Norway', 'EIRE', 'Switzerland', 'Spain', 'Poland', 'Portugal',
       'Italy', 'Belgium', 'Lithuania', 'Japan', 'Iceland',
       'Channel Islands', 'Denmark', 'Cyprus', 'Sweden', 'Austria',
       'Israel', 'Finland', 'Bahrain', 'Greece', 'Hong Kong', 'Singapore',
       'Lebanon', 'United Arab Emirates', 'Saudi Arabia',
       'Czech Republic', 'Canada', 'Unspecified', 'Brazil', 'USA',
       'European Community', 'Malta', 'RSA'], dtype=object)
```

```
sales_df.groupby('Country')['Quantity'].sum().sort_values(ascending=False)
```

	Quantity
Country	
United Kingdom	4263829
Netherlands	200128
EIRE	142637
Germany	117448
France	110480
Australia	83653
Sweden	35637
Switzerland	30325
Spain	26824
Japan	25218
Belgium	23152
Norway	19247
Portugal	16180
Finland	10666
Channel Islands	9479
Denmark	8188
Italy	7999
Cyprus	6317
Singapore	5234
Austria	4827
Hong Kong	4769
Israel	4353
Poland	3653
Unspecified	3300
Canada	2763
Iceland	2458
Greece	1556
USA	1034
United Arab Emirates	982
Malta	944

```
mean_sales = sales_df.groupby('Country')['UnitPrice'].mean()
mean_sales
```

	UnitPrice
Country	
Australia	3.220612
Austria	4.243192
Bahrain	4.556316
Belgium	3.644335
Brazil	4.456250
Canada	6.030331
Channel Islands	4.932124
Cyprus	6.302363
Czech Republic	2.938333
Denmark	3.256941
EIRE	5.911077
European Community	4.820492
Finland	5.448705
France	5.028864
Germany	3.966930
Greece	4.885548
Hong Kong	42.505208
Iceland	2.644011
Israel	3.633131
Italy	4.831121
Japan	2.276145
Lebanon	5.387556
Lithuania	2.841143
Malta	5.244173
Netherlands	2.738317
Norway	6.012026
Poland	4.170880
Portugal	8.582976
RSA	4.277586
Saudi Arabia	2.411000

```
min_sales = sales_df.groupby('Country')['UnitPrice'].min().sort_values(ascending=False)
min_sales
```

	UnitPrice
Country	
Bahrain	1.25
Lithuania	1.25
Brazil	0.85
Lebanon	0.55
European Community	0.55
USA	0.42
Saudi Arabia	0.42
United Arab Emirates	0.29
Czech Republic	0.29
Iceland	0.25
Japan	0.21
Hong Kong	0.21
Denmark	0.21
Malta	0.19
Unspecified	0.19
Poland	0.19
Sweden	0.19
Channel Islands	0.19
Singapore	0.19
Greece	0.14
Portugal	0.12
Italy	0.12
Austria	0.12
Finland	0.12
Cyprus	0.12
Belgium	0.12
Canada	0.10
Israel	0.06
Netherlands	0.00
Norway	0.00

```
sales_df.groupby(['Country','InvoiceDate'])['UnitPrice'].mean().sort_values(ascending=False)
```

Country	InvoiceDate	UnitPrice
United Kingdom	**2011-06-10 15:31:00**	38970.00
	2011-12-05 11:36:00	17836.46
	2011-01-05 09:55:00	16888.02
	2011-01-05 09:57:00	16453.71
	2010-12-07 15:49:00	13541.33
	...	...
	2011-07-28 14:58:00	0.00
	2011-03-30 17:24:00	0.00
	2011-03-29 09:36:00	0.00
	2011-08-12 14:52:00	-11062.06
	2011-08-12 14:51:00	-11062.06

23616 rows × 1 columns

dtype: float64

```
#reset index
sales_df.reset_index(drop=True, inplace=True)
sales_df
```

	InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	CustomerID	Country
0	536365	85123A	WHITE HANGING HEART T-LIGHT HOLDER	6	2010-12-01 08:26:00	2.55	17850.0	United Kingdom
1	536365	71053	WHITE METAL LANTERN	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom
2	536365	84406B	CREAM CUPID HEARTS COAT HANGER	8	2010-12-01 08:26:00	2.75	17850.0	United Kingdom
3	536365	84029G	KNITTED UNION FLAG HOT WATER BOTTLE	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom
4	536365	84029E	RED WOOLLY HOTTIE WHITE HEART.	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom
...	...	...	...	...	...	...	...	...
541904	581587	22613	PACK OF 20 SPACEBOY NAPKINS	12	2011-12-09 12:50:00	0.85	12680.0	France
541905	581587	22899	CHILDREN'S APRON DOLLY GIRL	6	2011-12-09 12:50:00	2.10	12680.0	France

```
sales_df.set_index(keys=['Country','InvoiceDate'],inplace=True)
sales_df
```

		InvoiceNo	StockCode	Description	Quantity	UnitPrice	CustomerID
Country	**InvoiceDate**						
United Kingdom	**2010-12-01 08:26:00**	536365	85123A	WHITE HANGING HEART T-LIGHT HOLDER	6	2.55	17850.0
	2010-12-01 08:26:00	536365	71053	WHITE METAL LANTERN	6	3.39	17850.0
	2010-12-01 08:26:00	536365	84406B	CREAM CUPID HEARTS COAT HANGER	8	2.75	17850.0
	2010-12-01 08:26:00	536365	84029G	KNITTED UNION FLAG HOT WATER BOTTLE	6	3.39	17850.0
	2010-12-01 08:26:00	536365	84029E	RED WOOLLY HOTTIE WHITE HEART.	6	3.39	17850.0
...	**...**	...	...	...	...	...	...
France	**2011-12-09 12:50:00**	581587	22613	PACK OF 20 SPACEBOY NAPKINS	12	0.85	12680.0
	2011-12-09 12:50:00	581587	22899	CHILDREN'S APRON DOLLY GIRL	6	2.10	12680.0
	2011-12-09 12:50:00	581587	23254	CHILDRENS CUTLERY DOLLY GIRL	4	4.15	12680.0
	2011-12-09 12:50:00	581587	23255	CHILDRENS CUTLERY CIRCUS PARADE	4	4.15	12680.0
	2011-12-09 12:50:00	581587	22138	BAKING SET 9 PIECE RETROSPOT	3	4.95	12680.0

```
sales_df.sort_index(ascending=True,inplace=True)
sales_df.head()
```

		InvoiceNo	StockCode	Description	Quantity	UnitPrice	CustomerID
Country	**InvoiceDate**						
Australia	**2010-12-01 10:03:00**	536389	22941	CHRISTMAS LIGHTS 10 REINDEER	6	8.50	12431.0
	2010-12-01 10:03:00	536389	21622	VINTAGE UNION JACK CUSHION COVER	8	4.95	12431.0
	2010-12-01 10:03:00	536389	21791	VINTAGE HEADS AND TAILS CARD GAME	12	1.25	12431.0
	2010-12-01 10:03:00	536389	35004C	SET OF 3 COLOURED FLYING DUCKS	6	5.45	12431.0
	2010-12-01 10:03:00	536389	35004G	SET OF 3 GOLD FLYING DUCKS	4	6.35	12431.0

```
sales_df.index.names
```

```
FrozenList(['Country', 'InvoiceDate'])
```

```
sales_df.index[0]
```

```
('Australia', Timestamp('2010-12-01 10:03:00'))
```

```
sales_df.index.get_level_values(0)
```

```
Index(['Australia', 'Australia', 'Australia', 'Australia', 'Australia',
       'Australia', 'Australia', 'Australia', 'Australia', 'Australia',
       ...
       'Unspecified', 'Unspecified', 'Unspecified', 'Unspecified',
       'Unspecified', 'Unspecified', 'Unspecified', 'Unspecified',
       'Unspecified', 'Unspecified'],
      dtype='object', name='Country', length=541909)
```

```
sales_df.index.get_level_values('Country')
```

```
Index(['Australia', 'Australia', 'Australia', 'Australia', 'Australia',
       'Australia', 'Australia', 'Australia', 'Australia', 'Australia',
       ...
       'Unspecified', 'Unspecified', 'Unspecified', 'Unspecified',
       'Unspecified', 'Unspecified', 'Unspecified', 'Unspecified',
       'Unspecified', 'Unspecified'],
      dtype='object', name='Country', length=541909)
```

```
sales_df.index.get_level_values(1)
```

```
DatetimeIndex(['2010-12-01 10:03:00', '2010-12-01 10:03:00',
               '2010-12-01 10:03:00', '2010-12-01 10:03:00',
               '2010-12-01 10:03:00', '2010-12-01 10:03:00',
               '2010-12-01 10:03:00', '2010-12-01 10:03:00',
               '2010-12-01 10:03:00', '2010-12-01 10:03:00',
               ...
               '2011-11-24 14:55:00', '2011-11-24 14:55:00',
               '2011-11-24 14:55:00', '2011-11-24 14:55:00',
               '2011-11-24 14:55:00', '2011-11-24 14:55:00',
               '2011-11-24 14:55:00', '2011-11-24 14:55:00',
               '2011-11-24 14:55:00', '2011-11-24 14:55:00'],
              dtype='datetime64[ns]', name='InvoiceDate', length=541909, freq=None)
```

```
sales_df.index.get_level_values('InvoiceDate')
```

```
DatetimeIndex(['2010-12-01 10:03:00', '2010-12-01 10:03:00',
               '2010-12-01 10:03:00', '2010-12-01 10:03:00',
               '2010-12-01 10:03:00', '2010-12-01 10:03:00',
               '2010-12-01 10:03:00', '2010-12-01 10:03:00',
               '2010-12-01 10:03:00', '2010-12-01 10:03:00',
               ...
               '2011-11-24 14:55:00', '2011-11-24 14:55:00',
               '2011-11-24 14:55:00', '2011-11-24 14:55:00',
               '2011-11-24 14:55:00', '2011-11-24 14:55:00',
               '2011-11-24 14:55:00', '2011-11-24 14:55:00',
               '2011-11-24 14:55:00', '2011-11-24 14:55:00'],
              dtype='datetime64[ns]', name='InvoiceDate', length=541909, freq=None)
```

```
sales_df.index.set_names(names=['Transaction Location','Transaction Date'], inplace = True)
sales_df
```

		InvoiceNo	StockCode	Description	Quantity	UnitPrice	CustomerID
Transaction Location	**Transaction Date**						
Australia	**2010-12-01 10:03:00**	536389	22941	CHRISTMAS LIGHTS 10 REINDEER	6	8.50	12431.0
	2010-12-01 10:03:00	536389	21622	VINTAGE UNION JACK CUSHION COVER	8	4.95	12431.0
	2010-12-01 10:03:00	536389	21791	VINTAGE HEADS AND TAILS CARD GAME	12	1.25	12431.0
	2010-12-01 10:03:00	536389	35004C	SET OF 3 COLOURED FLYING DUCKS	6	5.45	12431.0
	2010-12-01 10:03:00	536389	35004G	SET OF 3 GOLD FLYING DUCKS	4	6.35	12431.0
...	**...**	...	...	...	...	...	...
Unspecified	**2011-11-24 14:55:00**	578539	22560	TRADITIONAL MODELLING CLAY	24	1.25	NaN

```
sales_df.loc['Australia','1/11/2011 9:47']
```

	InvoiceNo	StockCode	Description	Quantity	UnitPrice	CustomerID
Transaction Date						
2011-01-11 09:47:00	540700	21581	SKULLS DESIGN COTTON TOTE BAG	6	2.25	12393.0
2011-01-11 09:47:00	540700	22619	SET OF 6 SOLDIER SKITTLES	8	3.75	12393.0
2011-01-11 09:47:00	540700	84997B	RED 3 PIECE RETROSPOT CUTLERY SET	6	3.75	12393.0
2011-01-11 09:47:00	540700	20727	LUNCH BAG BLACK SKULL.	20	1.65	12393.0
2011-01-11 09:47:00	540700	20726	LUNCH BAG WOODLAND	20	1.65	12393.0
2011-01-11 09:47:00	540700	22383	LUNCH BAG SUKI DESIGN	10	1.65	12393.0
2011-01-11 09:47:00	540700	21249	WOODLAND HEIGHT CHART STICKERS	6	2.95	12393.0
2011-01-11 09:47:00	540700	22378	WALL TIDY RETROSPOT	20	0.85	12393.0
2011-01-11 09:47:00	540700	22175	PINK OWL SOFT TOY	12	2.95	12393.0
2011-01-11 09:47:00	540700	22176	BLUE OWL SOFT TOY	12	2.95	12393.0
2011-01-11 09:47:00	540700	84997C	BLUE 3 PIECE POLKADOT CUTLERY SET	6	3.75	12393.0
2011-01-11 09:47:00	540700	20728	LUNCH BAG CARS BLUE	20	1.65	12393.0
2011-01-11 09:47:00	540700	22382	LUNCH BAG SPACEBOY DESIGN	20	1.65	12393.0
2011-01-11 09:47:00	540700	21915	RED HARMONICA IN BOX	12	1.25	12393.0
2011-01-11 09:47:00	540700	22549	PICTURE DOMINOES	12	1.45	12393.0
2011-01-11 09:47:00	540700	21578	WOODLAND DESIGN COTTON TOTE BAG	12	2.25	12393.0
2011-01-11 09:47:00	540700	21577	SAVE THE PLANET COTTON TOTE BAG	12	2.25	12393.0
2011-01-11 09:47:00	540700	22245	HOOK, 1 HANGER ,MAGIC GARDEN	12	0.85	12393.0
2011-01-11 09:47:00	540700	22244	3 HOOK HANGER MAGIC GARDEN	12	1.95	12393.0

```
sales_df.iloc[0]
```

	Australia
	2010-12-01 10:03:00
InvoiceNo	536389
StockCode	22941
Description	CHRISTMAS LIGHTS 10 REINDEER
Quantity	6
UnitPrice	8.5
CustomerID	12431.0

dtype: object

```
sales_df.iloc[0,2]
```

```
'CHRISTMAS LIGHTS 10 REINDEER'
```

```
Transposed_df = sales_df.transpose()
Transposed_df
```

Transaction Location	Australia										...	Unspe
Transaction Date	**2010-12-01 10:03:00**	**2010-12-01 10:03:00**	**2010-12-01 10:03:00**	**2010-12-01 10:03:00**	**2010-12-01 10:03:00**	**2010-12-01 10:03:00**	**2010-12-01 10:03:00**	**2010-12-01 10:03:00**	**2010-12-01 10:03:00**	**2010-12-01 10:03:00**	...	**2011-24 14:55**
InvoiceNo	536389	536389	536389	536389	536389	536389	536389	536389	536389	536389	...	
StockCode	22941	21622	21791	35004C	35004G	85014B	85014A	22193	22726	22727	...	
Description	CHRISTMAS LIGHTS 10 REINDEER	VINTAGE UNION JACK CUSHION COVER	VINTAGE HEADS AND TAILS CARD GAME	SET OF 3 COLOURED FLYING DUCKS	SET OF 3 GOLD FLYING DUCKS	RED RETROSPOT UMBRELLA	BLACK/BLUE POLKADOT UMBRELLA	RED DINER WALL CLOCK	ALARM CLOCK BAKELIKE GREEN	ALARM CLOCK BAKELIKE RED	...	TRAD ALP STAN
Quantity	6	8	12	6	4	6	3	2	4	4	...	
UnitPrice	8.5	4.95	1.25	5.45	6.35	5.95	5.95	8.5	3.75	3.75	...	
CustomerID	12431.0	12431.0	12431.0	12431.0	12431.0	12431.0	12431.0	12431.0	12431.0	12431.0	...	

6 rows × 541909 columns

```
sales_df.loc[('Australia','2010-12-01 10:03:00'),'UnitPrice']
```

Transaction Date	UnitPrice
2010-12-01 10:03:00	8.50
2010-12-01 10:03:00	4.95
2010-12-01 10:03:00	1.25
2010-12-01 10:03:00	5.45
2010-12-01 10:03:00	6.35
2010-12-01 10:03:00	5.95
2010-12-01 10:03:00	5.95
2010-12-01 10:03:00	8.50
2010-12-01 10:03:00	3.75
2010-12-01 10:03:00	3.75
2010-12-01 10:03:00	8.50
2010-12-01 10:03:00	8.50
2010-12-01 10:03:00	1.65
2010-12-01 10:03:00	0.85

dtype: float64

```
sales_df.loc[('Australia','2010-12-01 10:03:00'):('Belgium','2010-12-01 10:03:00'),'UnitPrice']
```

		UnitPrice
Transaction Location	**Transaction Date**	
Australia	**2010-12-01 10:03:00**	8.50
	2010-12-01 10:03:00	4.95
	2010-12-01 10:03:00	1.25
	2010-12-01 10:03:00	5.45
	2010-12-01 10:03:00	6.35
...	**...**	...
Bahrain	**2011-05-09 13:49:00**	4.25
	2011-05-19 17:47:00	9.95
	2011-05-19 17:47:00	1.45
	2011-05-19 17:47:00	9.95
	2011-05-19 17:47:00	2.95

1679 rows × 1 columns

dtype: float64

```
import datetime as dt

my_date = dt.date(2020,3,22)
my_date
```

```
datetime.date(2020, 3, 22)
```

```
str(my_date)
```

```
'2020-03-22'
```

```
my_date.day
```

```
22
```

```
my_date.month
```

```
3
```

```
my_date.year
```

```
2020
```

```
my_datetime = dt.datetime(2020,3,22,10,30,0)
my_datetime
```

```
datetime.datetime(2020, 3, 22, 10, 30)
```

```
str(my_datetime)
```

```
'2020-03-22 10:30:00'
```

```
my_datetime.hour
```

```
10
```

```
import calendar
print(calendar.month(2021,3))
```

```
     March 2021
Mo Tu We Th Fr Sa Su
 1  2  3  4  5  6  7
 8  9 10 11 12 13 14
15 16 17 18 19 20 21
22 23 24 25 26 27 28
29 30 31
```

```
dates = pd.Series(['2020/03/22','2020-10-25','February 20th,2020'])
dates
```

	0
0	2020/03/22
1	2020-10-25
2	February 20th,2020

dtype: object

```
my_dates = pd.to_datetime(dates,format='mixed')
my_dates
```

	0
0	2020-03-22
1	2020-10-25
2	2020-02-20

dtype: datetime64[ns]

```
pd.Timestamp('2020-3-22')  # Use hyphens or slashes instead of commas
```

```
Timestamp('2020-03-22 00:00:00')
```

```
day_1 = pd.Timestamp('2020-3-22')
day_2 = pd.Timestamp('2020-3-25')
day_2-day_1
```

```
Timedelta('3 days 00:00:00')
```

```
date_1=dt.date(2020,3,22)
date_2=dt.date(2020,4,22)
date_3=dt.date(2020,4,22)
dates_list =[date_1,date_2,date_3]
dates_list
```

```
[datetime.date(2020, 3, 22),
 datetime.date(2020, 4, 22),
 datetime.date(2020, 4, 22)]
```

```
dates_index=pd.DatetimeIndex(dates_list)
dates_index
```

```
DatetimeIndex(['2020-03-22', '2020-04-22', '2020-04-22'], dtype='datetime64[ns]', freq=None)
```

```
sales = pd.Series(data=[100,200,300],index=dates_index)
sales
```

	0
2020-03-22	100
2020-04-22	200
2020-04-22	300

dtype: int64

```
my_dates = pd.date_range(start='2020-3-22',end='2020-4-22',freq='D')
my_dates
```

```
DatetimeIndex(['2020-03-22', '2020-03-23', '2020-03-24', '2020-03-25',
               '2020-03-26', '2020-03-27', '2020-03-28', '2020-03-29',
               '2020-03-30', '2020-03-31', '2020-04-01', '2020-04-02',
               '2020-04-03', '2020-04-04', '2020-04-05', '2020-04-06',
               '2020-04-07', '2020-04-08', '2020-04-09', '2020-04-10',
               '2020-04-11', '2020-04-12', '2020-04-13', '2020-04-14',
               '2020-04-15', '2020-04-16', '2020-04-17', '2020-04-18',
               '2020-04-19', '2020-04-20', '2020-04-21', '2020-04-22'],
              dtype='datetime64[ns]', freq='D')
```

```
my_dates = pd.date_range(start='2020-3-22',end='2020-6-22',freq='ME')
my_dates
```

```
DatetimeIndex(['2020-03-31', '2020-04-30', '2020-05-31'], dtype='datetime64[ns]', freq='ME')
```

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

```
df=pd.read_csv('/content/text_data.csv')
df
```

	rating	date	variation	verified_reviews	feedback
0	5	31-Jul-18	Charcoal Fabric	Love my Echo!	1
1	5	31-Jul-18	Charcoal Fabric	Loved it!	1
2	4	31-Jul-18	Walnut Finish	Sometimes while playing a game, you can answer...	1
3	5	31-Jul-18	Charcoal Fabric	I have had a lot of fun with this thing. My 4 ...	1
4	5	31-Jul-18	Charcoal Fabric	Music	1
...	...	...	...	...	...
3145	5	30-Jul-18	Black Dot	Perfect for kids, adults and everyone in betwe...	1
3146	5	30-Jul-18	Black Dot	Listening to music, searching locations, check...	1
3147	5	30-Jul-18	Black Dot	I do love these things, i have them running my...	1
3148	5	30-Jul-18	White Dot	Only complaint I have is that the sound qualit...	1
3149	4	29-Jul-18	Black Dot	Good	1

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3150 entries, 0 to 3149
Data columns (total 5 columns):
 #   Column            Non-Null Count  Dtype 
---  ------            --------------  ----- 
 0   rating            3150 non-null   int64 
 1   date              3150 non-null   object
 2   variation         3150 non-null   object
 3   verified_reviews  3149 non-null   object
 4   feedback          3150 non-null   int64 
dtypes: int64(2), object(3)
memory usage: 123.2+ KB
```

```
df.describe()
```

	rating	feedback
count	3150.000000	3150.000000
mean	4.463175	0.918413
std	1.068506	0.273778
min	1.000000	0.000000
25%	4.000000	1.000000
50%	5.000000	1.000000
75%	5.000000	1.000000
max	5.000000	1.000000

```
df['variation'].nunique()
```

```
16
```

```
df['verified_reviews'].str.lower()
```

	verified_reviews
0	love my echo!
1	loved it!
2	sometimes while playing a game, you can answer...
3	i have had a lot of fun with this thing. my 4 ...
4	music
...	...
3145	perfect for kids, adults and everyone in betwe...
3146	listening to music, searching locations, check...
3147	i do love these things, i have them running my...
3148	only complaint i have is that the sound qualit...
3149	good

3150 rows × 1 columns

dtype: object

```
df.columns = df.columns.str.upper()
df
```

	RATING	DATE	VARIATION	VERIFIED_REVIEWS	FEEDBACK
0	5	31-Jul-18	Charcoal Fabric	Love my Echo!	1
1	5	31-Jul-18	Charcoal Fabric	Loved it!	1
2	4	31-Jul-18	Walnut Finish	Sometimes while playing a game, you can answer...	1
3	5	31-Jul-18	Charcoal Fabric	I have had a lot of fun with this thing. My 4 ...	1
4	5	31-Jul-18	Charcoal Fabric	Music	1
...	...	...	...	...	...
3145	5	30-Jul-18	Black Dot	Perfect for kids, adults and everyone in betwe...	1
3146	5	30-Jul-18	Black Dot	Listening to music, searching locations, check...	1
3147	5	30-Jul-18	Black Dot	I do love these things, i have them running my...	1
3148	5	30-Jul-18	White Dot	Only complaint I have is that the sound qualit...	1
3149	4	29-Jul-18	Black Dot	Good	1

```
df.columns = df.columns.str.lower()
df
```

	rating	date	variation	verified_reviews	feedback
0	5	31-Jul-18	Charcoal Fabric	Love my Echo!	1
1	5	31-Jul-18	Charcoal Fabric	Loved it!	1
2	4	31-Jul-18	Walnut Finish	Sometimes while playing a game, you can answer...	1
3	5	31-Jul-18	Charcoal Fabric	I have had a lot of fun with this thing. My 4 ...	1
4	5	31-Jul-18	Charcoal Fabric	Music	1
...	...	...	...	...	...
3145	5	30-Jul-18	Black Dot	Perfect for kids, adults and everyone in betwe...	1
3146	5	30-Jul-18	Black Dot	Listening to music, searching locations, check...	1
3147	5	30-Jul-18	Black Dot	I do love these things, i have them running my...	1
3148	5	30-Jul-18	White Dot	Only complaint I have is that the sound qualit...	1
3149	4	29-Jul-18	Black Dot	Good	1

```
negative_df = df[df['feedback'] == 0]
negative_df
```

	rating	date	variation	verified_reviews	feedback
46	2	30-Jul-18	Charcoal Fabric	It's like Siri, in fact, Siri answers more acc...	0
111	2	30-Jul-18	Charcoal Fabric	Sound is terrible if u want good music too get...	0
141	1	30-Jul-18	Charcoal Fabric	Not much features.	0
162	1	30-Jul-18	Sandstone Fabric	Stopped working after 2 weeks ,didn't follow c...	0
176	2	30-Jul-18	Heather Gray Fabric	Sad joke. Worthless.	0
...	...	...	...	...	...
3047	1	30-Jul-18	Black Dot	Echo Dot responds to us when we aren't even ta...	0
3048	1	30-Jul-18	White Dot	NOT CONNECTED TO MY PHONE PLAYLIST :(	0
3067	2	30-Jul-18	Black Dot	The only negative we have on this product is t...	0
3091	1	30-Jul-18	Black Dot	I didn’t order it	0
3096	1	30-Jul-18	White Dot	The product sounded the same as the emoji spea...	0

```
positive_df = df[df['feedback'] == 1]
positive_df
```

	rating	date	variation	verified_reviews	feedback
0	5	31-Jul-18	Charcoal Fabric	Love my Echo!	1
1	5	31-Jul-18	Charcoal Fabric	Loved it!	1
2	4	31-Jul-18	Walnut Finish	Sometimes while playing a game, you can answer...	1
3	5	31-Jul-18	Charcoal Fabric	I have had a lot of fun with this thing. My 4 ...	1
4	5	31-Jul-18	Charcoal Fabric	Music	1
...	...	...	...	...	...
3145	5	30-Jul-18	Black Dot	Perfect for kids, adults and everyone in betwe...	1
3146	5	30-Jul-18	Black Dot	Listening to music, searching locations, check...	1
3147	5	30-Jul-18	Black Dot	I do love these things, i have them running my...	1
3148	5	30-Jul-18	White Dot	Only complaint I have is that the sound qualit...	1
3149	4	29-Jul-18	Black Dot	Good	1

```
poor_rating_df=df[df['rating']==1]
poor_rating_df
```

	rating	date	variation	verified_reviews	feedback
141	1	30-Jul-18	Charcoal Fabric	Not much features.	0
162	1	30-Jul-18	Sandstone Fabric	Stopped working after 2 weeks ,didn't follow c...	0
341	1	28-Jul-18	Charcoal Fabric	Alexa hardly came on..	0
350	1	31-Jul-18	Black	Item no longer works after just 5 months of us...	0
361	1	29-Jul-18	Black	This thing barely works. You have to select 3r...	0
...	...	...	...	...	...
3024	1	30-Jul-18	Black Dot	I was really happy with my original echo so i ...	0
3047	1	30-Jul-18	Black Dot	Echo Dot responds to us when we aren't even ta...	0
3048	1	30-Jul-18	White Dot	NOT CONNECTED TO MY PHONE PLAYLIST :(	0
3091	1	30-Jul-18	Black Dot	I didn't order it	0
3096	1	30-Jul-18	White Dot	The product sounded the same as the emoji spea...	0

```
good_rating_df=df[df['rating']==5]
good_rating_df
```

	rating	date	variation	verified_reviews	feedback
0	5	31-Jul-18	Charcoal Fabric	Love my Echo!	1
1	5	31-Jul-18	Charcoal Fabric	Loved it!	1
3	5	31-Jul-18	Charcoal Fabric	I have had a lot of fun with this thing. My 4 ...	1
4	5	31-Jul-18	Charcoal Fabric	Music	1
5	5	31-Jul-18	Heather Gray Fabric	I received the echo as a gift. I needed anothe...	1
...	...	...	...	...	...
3144	5	30-Jul-18	Black Dot	love it	1
3145	5	30-Jul-18	Black Dot	Perfect for kids, adults and everyone in betwe...	1
3146	5	30-Jul-18	Black Dot	Listening to music, searching locations, check...	1
3147	5	30-Jul-18	Black Dot	I do love these things, i have them running my...	1
3148	5	30-Jul-18	White Dot	Only complaint I have is that the sound qualit...	1

```
df['reviews_length']=df['verified_reviews'].str.len()
df
```

	rating	date	variation	verified_reviews	feedback	reviews_length
0	5	31-Jul-18	Charcoal Fabric	Love my Echo!	1	13.0
1	5	31-Jul-18	Charcoal Fabric	Loved it!	1	9.0
2	4	31-Jul-18	Walnut Finish	Sometimes while playing a game, you can answer...	1	195.0
3	5	31-Jul-18	Charcoal Fabric	I have had a lot of fun with this thing. My 4 ...	1	172.0
4	5	31-Jul-18	Charcoal Fabric	Music	1	5.0
...	...	...	...	...	...	...
3145	5	30-Jul-18	Black Dot	Perfect for kids, adults and everyone in betwe...	1	50.0
3146	5	30-Jul-18	Black Dot	Listening to music, searching locations, check...	1	135.0
3147	5	30-Jul-18	Black Dot	I do love these things, i have them running my...	1	441.0
3148	5	30-Jul-18	White Dot	Only complaint I have is that the sound qualit...	1	380.0
3149	4	29-Jul-18	Black Dot	Good	1	4.0

```
min_char = df['verified_reviews'].str.len().min()
min_char
```

```
1.0
```

```
max_char = df['verified_reviews'].str.len().max()
max_char
```

```
2851.0
```

```
df[df['reviews_length'] == min_char]
```

	rating	date	variation	verified_reviews	feedback	reviews_length
60	5	30-Jul-18	Heather Gray Fabric	😍	1	1.0
85	5	30-Jul-18	Heather Gray Fabric		1	1.0
183	3	29-Jul-18	Heather Gray Fabric		1	1.0
219	5	29-Jul-18	Sandstone Fabric		1	1.0
374	1	26-Jul-18	Black		0	1.0
...	...	...	...	...	...	...
3114	3	30-Jul-18	Black Dot		1	1.0
3120	5	30-Jul-18	Black Dot		1	1.0
3123	4	30-Jul-18	Black Dot		1	1.0
3126	5	30-Jul-18	Black Dot		1	1.0
3141	3	30-Jul-18	Black Dot		1	1.0

81 rows × 6 columns

```
df[df['reviews_length'] == max_char]
```

	rating	date	variation	verified_reviews	feedback	reviews_length

```
df[df['reviews_length'] == max_char]['verified_reviews'].iloc[0]
```

```
'Incredible piece of technology.I have this right center of my living room on an island kitchen counter. The mic and speake
r goes in every direction and the quality of the sound is quite good. I connected the Echo via Bluetooth to my Sony soundba
r on my TV but find the Echo placement and 360 sound more appealing. It's no audiophile equipment but there is good range a
nd decent bass. The sound is more than adequate for any indoor entertaining and loud enough to bother neighbors in my build
ing. The knob on the top works great for adjusting volume. This is my first Echo device and I would imagine having to press
```

```
# Replace NaN values in 'verified_reviews' with an empty string before applying the mask
mask = df['verified_reviews'].fillna('').str.lower().str.endswith('love')
df_filtered = df[mask]
df_filtered
```

	rating	date	variation	verified_reviews	feedback	reviews_length
438	5	7-Jul-18	Black	Love	1	4.0

```
# Replace NaN values in 'verified_reviews' with an empty string before applying the mask
# and fill NaN values in the mask with False
mask = df['verified_reviews'].fillna('').str.lower().str.contains('bad').fillna(False)

# Now the mask should only contain True/False values, and you can use it to filter the DataFrame
df_filtered = df[mask]
df_filtered
```

	rating	date	variation	verified_reviews	feedback	reviews_length
113	5	30-Jul-18	Charcoal Fabric	Easy to set up and use. Too bad it has to be p...	1	127.0
288	5	29-Jul-18	Charcoal Fabric	Mainly use it for the music but I’m learning m...	1	99.0
567	1	11-Jun-18	White	Very bad this device, I don’t know if it’s bec...	0	205.0
601	3	5-Jun-18	Black	Not bad bad speaker for sound	1	29.0
808	5	30-Jul-18	Charcoal Fabric	Easy to set up and use. Too bad it has to be p...	1	127.0
983	5	29-Jul-18	Charcoal Fabric	Mainly use it for the music but I’m learning m...	1	99.0
1233	5	26-Jul-18	Black Spot	I purchased this on prime day mostly as a pres...	1	727.0
1338	5	19-Jul-18	White Spot	Omg where do I start, I LOVE THIS THING! None ...	1	57.0
1440	4	30-Jul-18	White Show	Love the product idea. Super easy to set up & ...	1	978.0
1448	1	30-Jul-18	White Show	Don't waste your money. I own an echo gen 1, ...	0	770.0
1786	4	30-Jul-18	Black Plus	Not that much different than the one we had. ...	1	282.0
1809	5	29-Jul-18	Black Plus	I love my echo plus. I haven't one bad thing t...	1	684.0
1967	5	25-Jul-18	White Plus	I have been hesitant to purchase anything smar...	1	882.0
1983	2	23-Jul-18	Black Plus	I'm sure the Echo is as cool as everyone says....	0	460.0
1994	5	21-Jul-18	Black Plus	I have an original Echo and really like it. I ...	1	259.0
2021	1	19-Jul-18	Black Plus	I would like to tell you that I have a reason ...	0	1124.0
2226	3	30-Jul-18	Configuration: Fire TV Stick	Works very well for Amazon and Netflix but wil...	1	626.0
2651	4	30-Jul-18	Black Dot	It isn’t bad for what it is. Have issues with ...	1	203.0
2688	2	30-Jul-18	Black Dot	Weak sound. Compared to the Google Home Mini t...	0	300.0
3002	4	30-Jul-18	Black Dot	It isn’t bad for what it is. Have issues with ...	1	203.0

```
df['verified_reviews'].str.split(' ')
```

	verified_reviews
0	[Love, my, Echo!]
1	[Loved, it!]
2	[Sometimes, while, playing, a, game,, you, can...
3	[I, have, had, a, lot, of, fun, with, this, th...
4	[Music]
...	...
3145	[Perfect, for, kids,, adults, and, everyone, i...
3146	[Listening, to, music,, searching, locations,,...
3147	[I, do, love, these, things,, i, have, them, r...
3148	[Only, complaint, I, have, is, that, the, soun...
3149	[Good]

3150 rows × 1 columns

dtype: object

```
df['verified_reviews'].str.split(' ').str.get(0)
```

	verified_reviews
0	Love
1	Loved
2	Sometimes
3	I
4	Music
...	...
3145	Perfect
3146	Listening
3147	I
3148	Only
3149	Good

3150 rows × 1 columns

dtype: object

```
import string
string.punctuation
```

```
'!"#$%&\'()*+,-./:;<=>?@[\\]^_`{|}~'
```

```
Test = '$I l@ove P$andas &Data Analytics!!!'
```

```
Test_punc_removed = [char for char in Test if char not in string.punctuation]
Test_punc_removed
```

```
['I',
 ' ',
 'l',
 'o',
 'v',
 'e',
 ' ',
 'P',
 'a',
 'n',
 'd',
 'a',
 's',
 ' ',
 'D',
 'a',
 't',
 'a',
 ' ',
 'A',
 'n',
 'a',
 'l',
 'y',
 't',
 'i',
 'c',
 's']
```

```
Test_punc_removed_join=''.join(Test_punc_removed)
Test_punc_removed_join
```

```
'I love Pandas Data Analytics'
```

```
def remove_punc(message):
  # Convert message to string first, handling NaN
  message = str(message)
  Test_punc_removed = [char for char in message if char not in string.punctuation]
  Test_punc_removed_join=''.join(Test_punc_removed)
  return Test_punc_removed_join
```

```
df['verified_reviews'].apply(remove_punc)
```

	verified_reviews
0	Love my Echo
1	Loved it
2	Sometimes while playing a game you can answer ...
3	I have had a lot of fun with this thing My 4 y...
4	Music
...	...
3145	Perfect for kids adults and everyone in between
3146	Listening to music searching locations checkin...
3147	I do love these things i have them running my ...
3148	Only complaint I have is that the sound qualit...
3149	Good

3150 rows × 1 columns

dtype: object

```
!pip install gensim
!pip install nltk
```

```
Requirement already satisfied: gensim in /usr/local/lib/python3.10/dist-packages (4.3.3)
Requirement already satisfied: numpy<2.0,>=1.18.5 in /usr/local/lib/python3.10/dist-packages (from gensim) (1.26.4)
Requirement already satisfied: scipy<1.14.0,>=1.7.0 in /usr/local/lib/python3.10/dist-packages (from gensim) (1.13.1)
Requirement already satisfied: smart-open>=1.8.1 in /usr/local/lib/python3.10/dist-packages (from gensim) (7.0.5)
Requirement already satisfied: wrapt in /usr/local/lib/python3.10/dist-packages (from smart-open>=1.8.1->gensim) (1.17.0)
Requirement already satisfied: nltk in /usr/local/lib/python3.10/dist-packages (3.9.1)
Requirement already satisfied: click in /usr/local/lib/python3.10/dist-packages (from nltk) (8.1.7)
Requirement already satisfied: joblib in /usr/local/lib/python3.10/dist-packages (from nltk) (1.4.2)
Requirement already satisfied: regex>=2021.8.3 in /usr/local/lib/python3.10/dist-packages (from nltk) (2024.9.11)
Requirement already satisfied: tqdm in /usr/local/lib/python3.10/dist-packages (from nltk) (4.66.6)
```

```
import nltk
from nltk.corpus import stopwords
nltk.download('stopwords')
import gensim
from gensim.utils import simple_preprocess
```

```
[nltk_data] Downloading package stopwords to /root/nltk_data...
[nltk_data]   Unzipping corpora/stopwords.zip.
```

```
stop_words = stopwords.words('english')
stop_words.extend(['Amazon','amazon','alexa','echo','Alexa','Device','Dot','dot'])
```

```
gensim.utils.simple_preprocess(df['verified_reviews'][0])
```

```
['love', 'my', 'echo']
```

```
def preprocess(text):
  text = str(text)
  result =[]
  for token in gensim.utils.simple_preprocess(text):
    if token not in gensim.parsing.preprocessing.STOPWORDS and len(token)>3 and token not in stop_words:
      result.append(token)
  return result
```

```
df['verified_reviews_nopunc_nostopwords']=df['verified_reviews'].apply(preprocess)
df
```

	rating	date	variation	verified_reviews	feedback	reviews_length	verified_reviews_nopunc_nostopwords
0	5	31-Jul-18	Charcoal Fabric	Love my Echo!	1	13.0	[love]
1	5	31-Jul-18	Charcoal Fabric	Loved it!	1	9.0	[loved]
2	4	31-Jul-18	Walnut Finish	Sometimes while playing a game, you can answer...	1	195.0	[playing, game, answer, question, correctly, s...
3	5	31-Jul-18	Charcoal Fabric	I have had a lot of fun with this thing. My 4 ...	1	172.0	[thing, learns, dinosaurs, control, lights, pl...
4	5	31-Jul-18	Charcoal Fabric	Music	1	5.0	[music]
...	...	...	...	...	...	...	...