Assignment - Data Breach Analysis

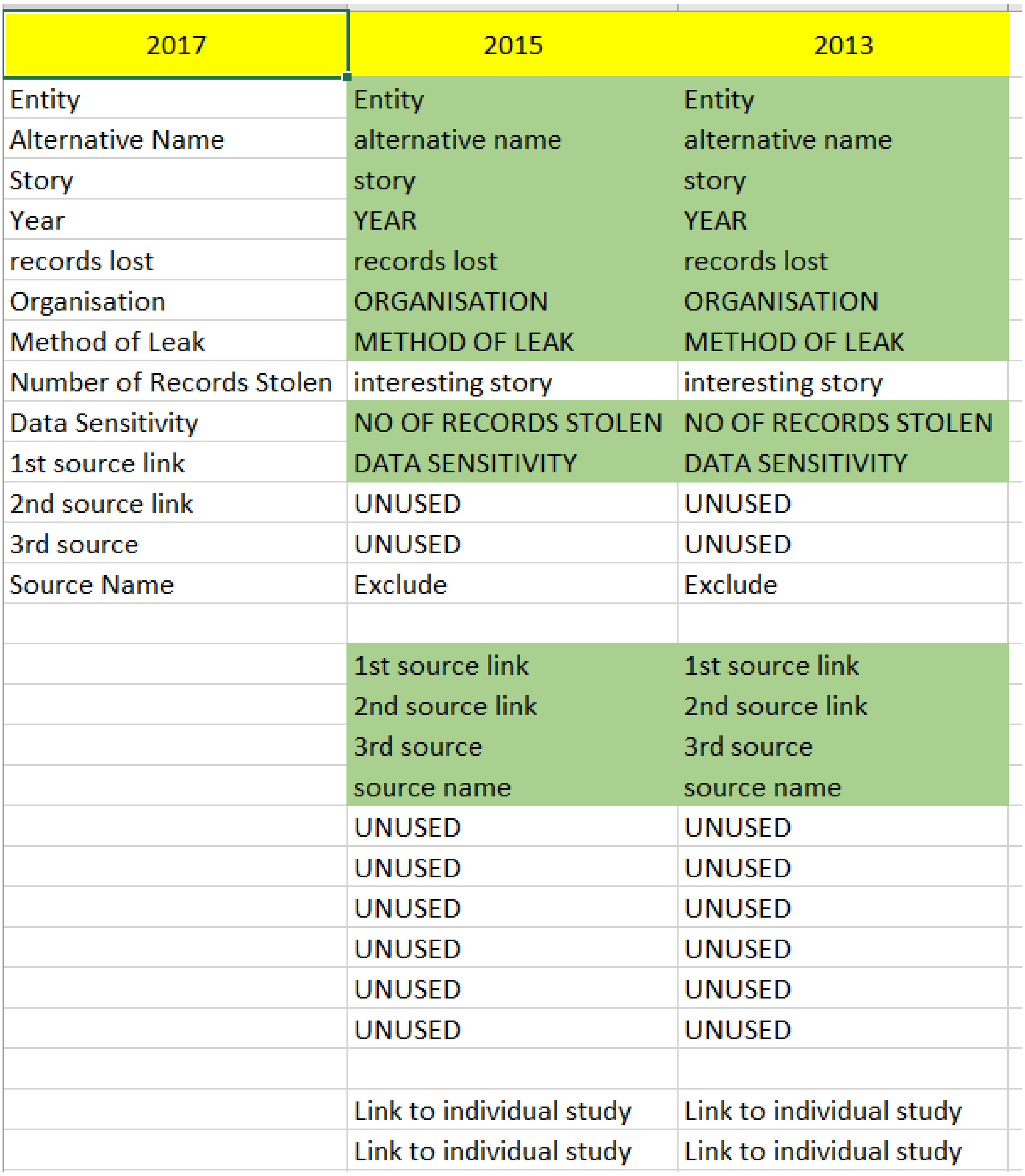
Problem statement: To analyze the data, clean for visualization and share insights using the dashboard.

Tools Used:

I. Google Colab – Python Code
II. Tableau Desktop
III. Word / PDF

Data Cleaning

- The input data source contained 3 tabs of which 2017 updated tab has the most updated data which is used for further analysis.
- 2013 / 2015 tab's have unwanted columns and redundant data.
- I have used Google **Colab** (**python code** to analyze and clean data)
- Please refer screenshot below:



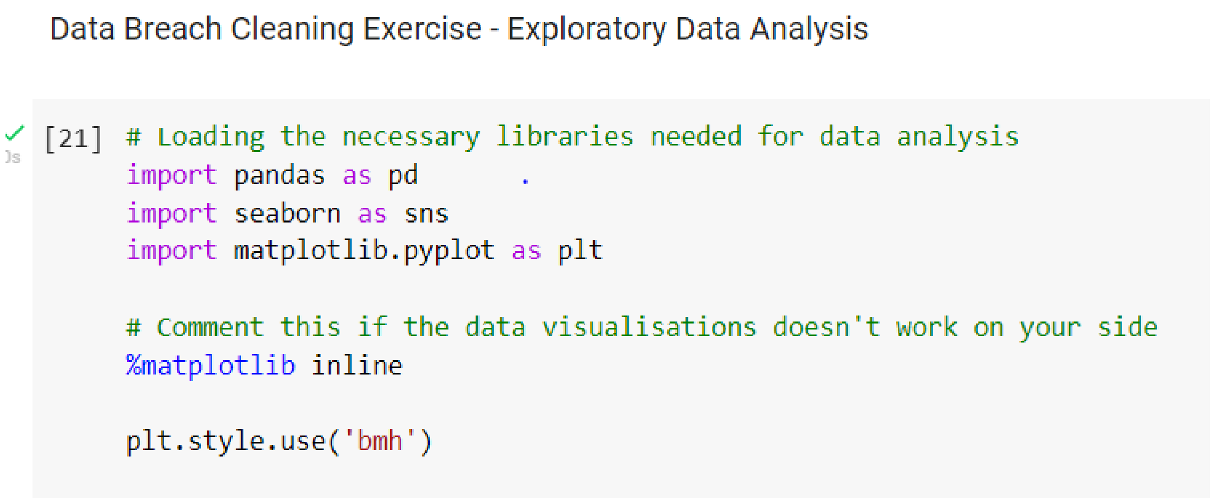
2017	2015	2013
Entity	Entity	Entity
Alternative Name	alternative name	alternative name
Story	story	story
Year	YEAR	YEAR
records lost	records lost	records lost
Organisation	ORGANISATION	ORGANISATION
Method of Leak	METHOD OF LEAK	METHOD OF LEAK
Number of Records Stolen	interesting story	interesting story
Data Sensitivity	NO OF RECORDS STOLEN	NO OF RECORDS STOLEN
1st source link	DATA SENSITIVITY	DATA SENSITIVITY
2nd source link	UNUSED	UNUSED
3rd source	UNUSED	UNUSED
Source Name	Exclude	Exclude
	1st source link	1st source link
	2nd source link	2nd source link
	3rd source	3rd source
	source name	source name
	UNUSED	UNUSED
	UNUSED	UNUSED
	UNUSED	UNUSED
	UNUSED	UNUSED
	UNUSED	UNUSED
	UNUSED	UNUSED
	Link to individual study	Link to individual study
	Link to individual study	Link to individual study

- Colab, or "Colaboratory", allows you to write and execute Python in your browser, with

  I. Zero configuration required
  II. Access to GPUs free of charge
  III. Easy sharing

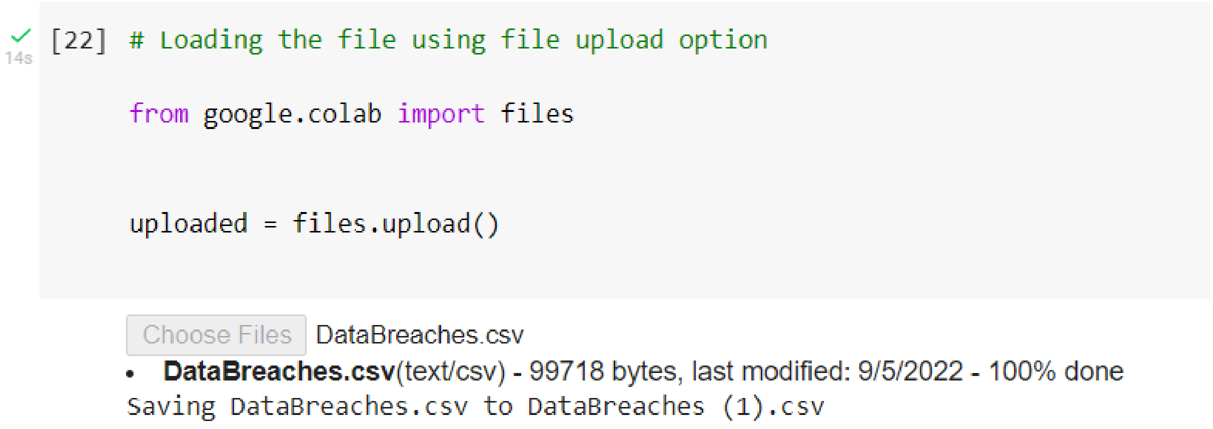
- Since the excel file is not huge in this scenario, I have converted the excel file into CSV file before loading in the data repository

- Loaded the necessary libraries and input file for data cleaning process



Data Breach Cleaning Exercise - Exploratory Data Analysis

```
# Loading the necessary libraries needed for data analysis
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt

# Comment this if the data visualisations doesn't work on your side
%matplotlib inline

plt.style.use('bmh')
```



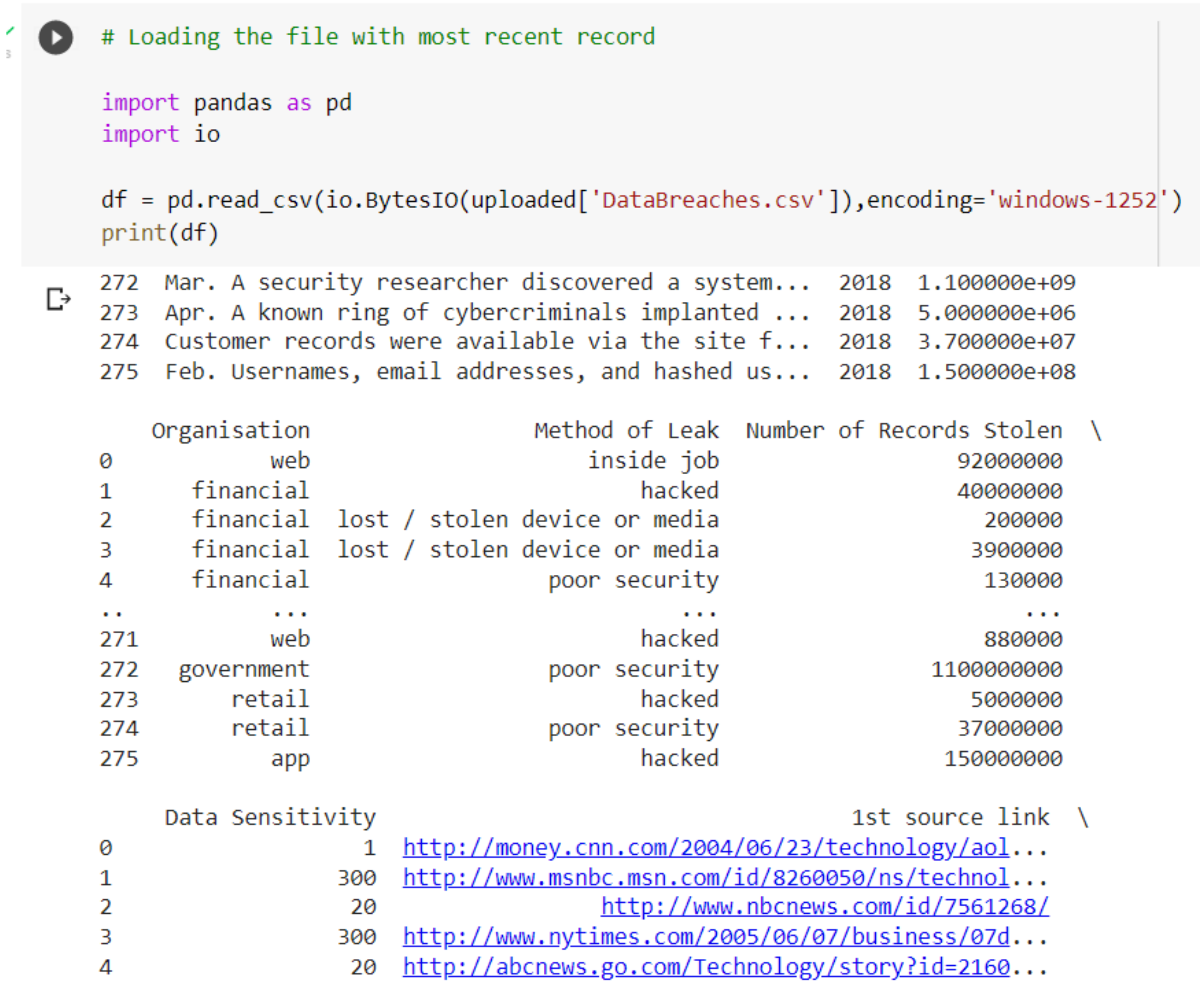
```
# Loading the file using file upload option

from google.colab import files


uploaded = files.upload()
```

DataBreaches.csv

- **DataBreaches.csv**(text/csv) - 99718 bytes, last modified: 9/5/2022 - 100% done

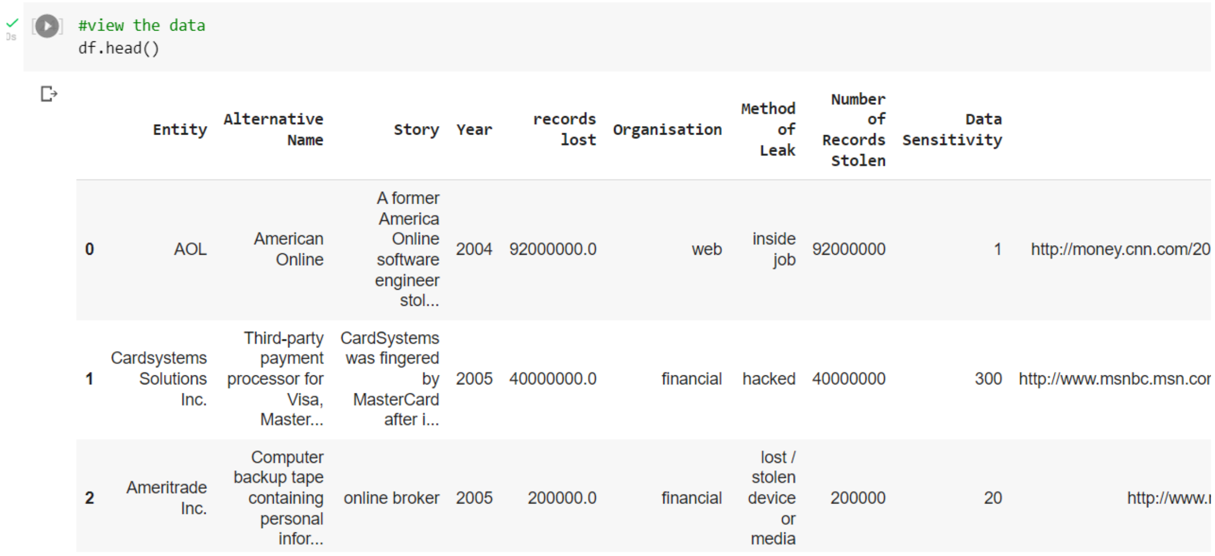
```
Saving DataBreaches.csv to DataBreaches (1).csv
```

- Checked whether data is completely loaded and dimensions of the data frame



```python
# Loading the file with most recent record

import pandas as pd
import io

df = pd.read_csv(io.BytesIO(uploaded['DataBreaches.csv']),encoding='windows-1252')
print(df)
```

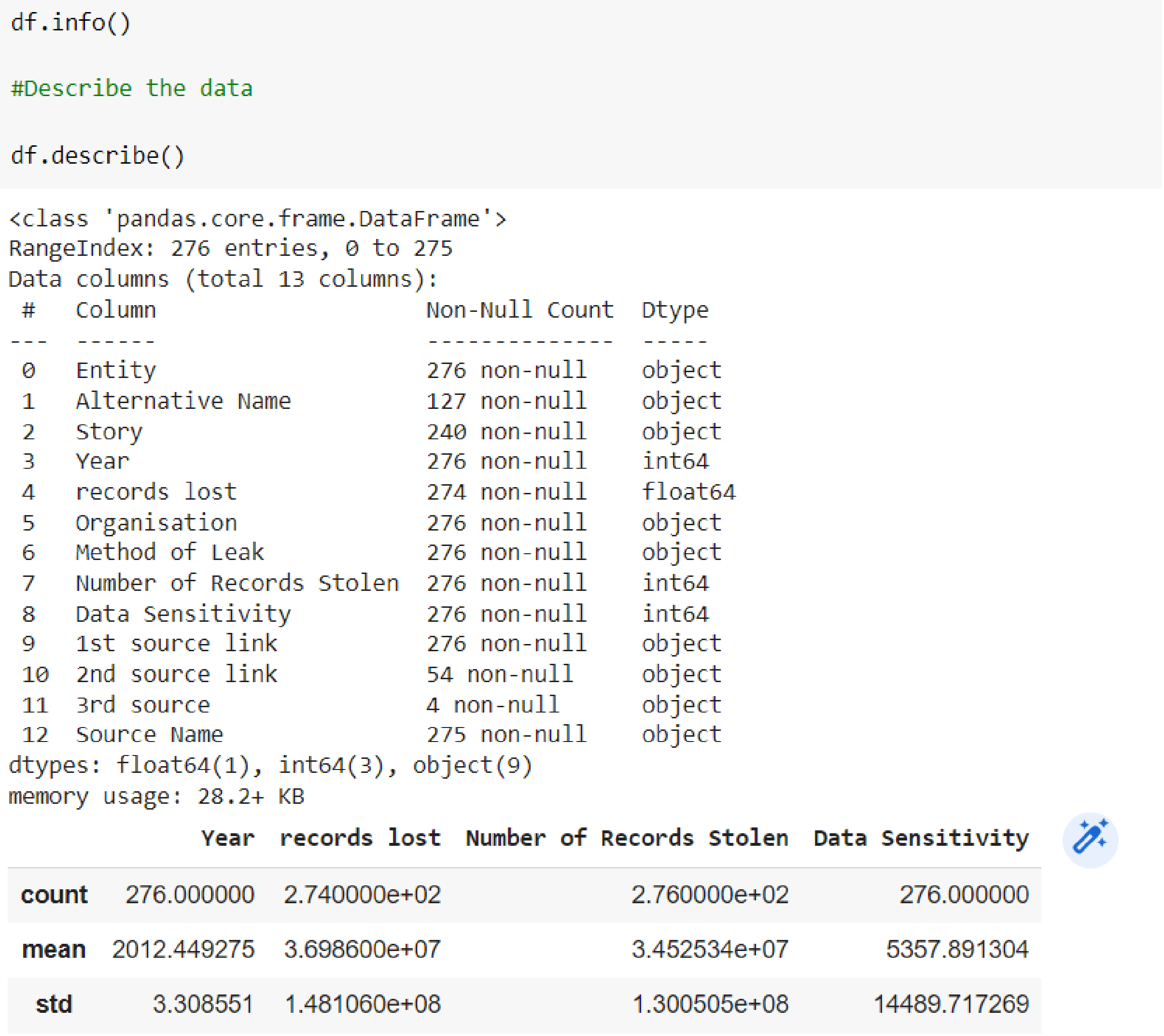
```
272  Mar. A security researcher discovered a system...  2018  1.100000e+09
273  Apr. A known ring of cybercriminals implanted ...  2018  5.000000e+06
274  Customer records were available via the site f...  2018  3.700000e+07
275  Feb. Usernames, email addresses, and hashed us...  2018  1.500000e+08

    Organisation                Method of Leak  Number of Records Stolen  \
0            web                    inside job                  92000000
1      financial                        hacked                  40000000
2      financial  lost / stolen device or media                   200000
3      financial  lost / stolen device or media                  3900000
4      financial                 poor security                    130000
..           ...                           ...                       ...
271          web                        hacked                    880000
272   government                 poor security                1100000000
273       retail                        hacked                   5000000
274       retail                 poor security                  37000000
275          app                        hacked                 150000000

     Data Sensitivity                                     1st source link  \
0                   1  http://money.cnn.com/2004/06/23/technology/aol...
1                 300  http://www.msnbc.msn.com/id/8260050/ns/technol...
2                  20                http://www.nbcnews.com/id/7561268/
3                 300  http://www.nytimes.com/2005/06/07/business/07d...
4                  20  http://abcnews.go.com/Technology/story?id=2160...
```



```python
#view the data
df.head()
```

	Entity	Alternative Name	Story	Year	records lost	Organisation	Method of Leak	Number of Records Stolen	Data Sensitivity	
0	AOL	American Online	A former America Online software engineer stol...	2004	92000000.0	web	inside job	92000000	1	http://money.cnn.com/20
1	Cardsystems Solutions Inc.	Third-party payment processor for Visa, Master...	CardSystems was fingered by MasterCard after i...	2005	40000000.0	financial	hacked	40000000	300	http://www.msnbc.msn.cor
2	Ameritrade Inc.	Computer backup tape containing personal infor...	online broker	2005	200000.0	financial	lost / stolen device or media	200000	20	http://www.i

- Checking for not null values on respective columns

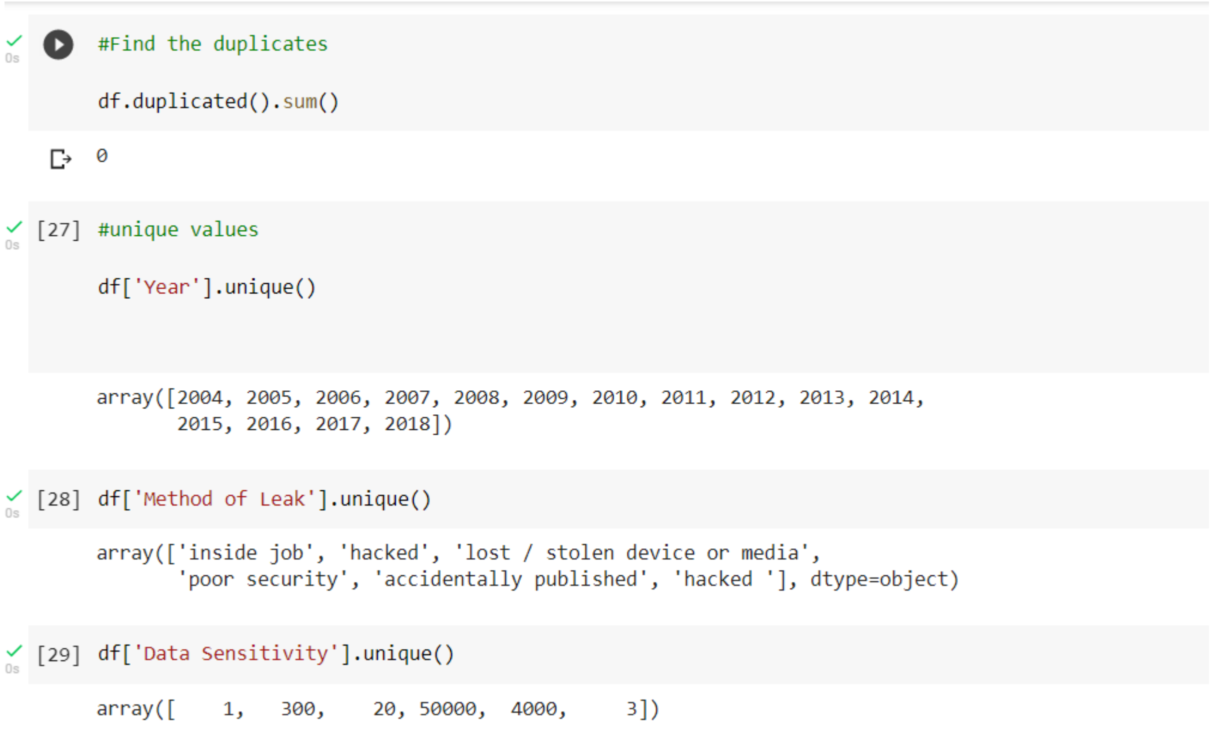


```
df.info()

#Describe the data

df.describe()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 276 entries, 0 to 275
Data columns (total 13 columns):
 #   Column                   Non-Null Count  Dtype  
---  ------                   --------------  -----  
 0   Entity                   276 non-null    object 
 1   Alternative Name         127 non-null    object 
 2   Story                    240 non-null    object 
 3   Year                     276 non-null    int64  
 4   records lost             274 non-null    float64
 5   Organisation             276 non-null    object 
 6   Method of Leak           276 non-null    object 
 7   Number of Records Stolen 276 non-null    int64  
 8   Data Sensitivity         276 non-null    int64  
 9   1st source link          276 non-null    object 
 10  2nd source link          54 non-null     object 
 11  3rd source               4 non-null      object 
 12  Source Name              275 non-null    object 
dtypes: float64(1), int64(3), object(9)
memory usage: 28.2+ KB
```

	Year	records lost	Number of Records Stolen	Data Sensitivity
count	276.000000	2.740000e+02	2.760000e+02	276.000000
mean	2012.449275	3.698600e+07	3.452534e+07	5357.891304
std	3.308551	1.481060e+08	1.300505e+08	14489.717269

- Checked for the unique values



```
#Find the duplicates

df.duplicated().sum()
```

```
0
```

```
[27] #unique values

     df['Year'].unique()
```

```
array([2004, 2005, 2006, 2007, 2008, 2009, 2010, 2011, 2012, 2013, 2014,
       2015, 2016, 2017, 2018])
```

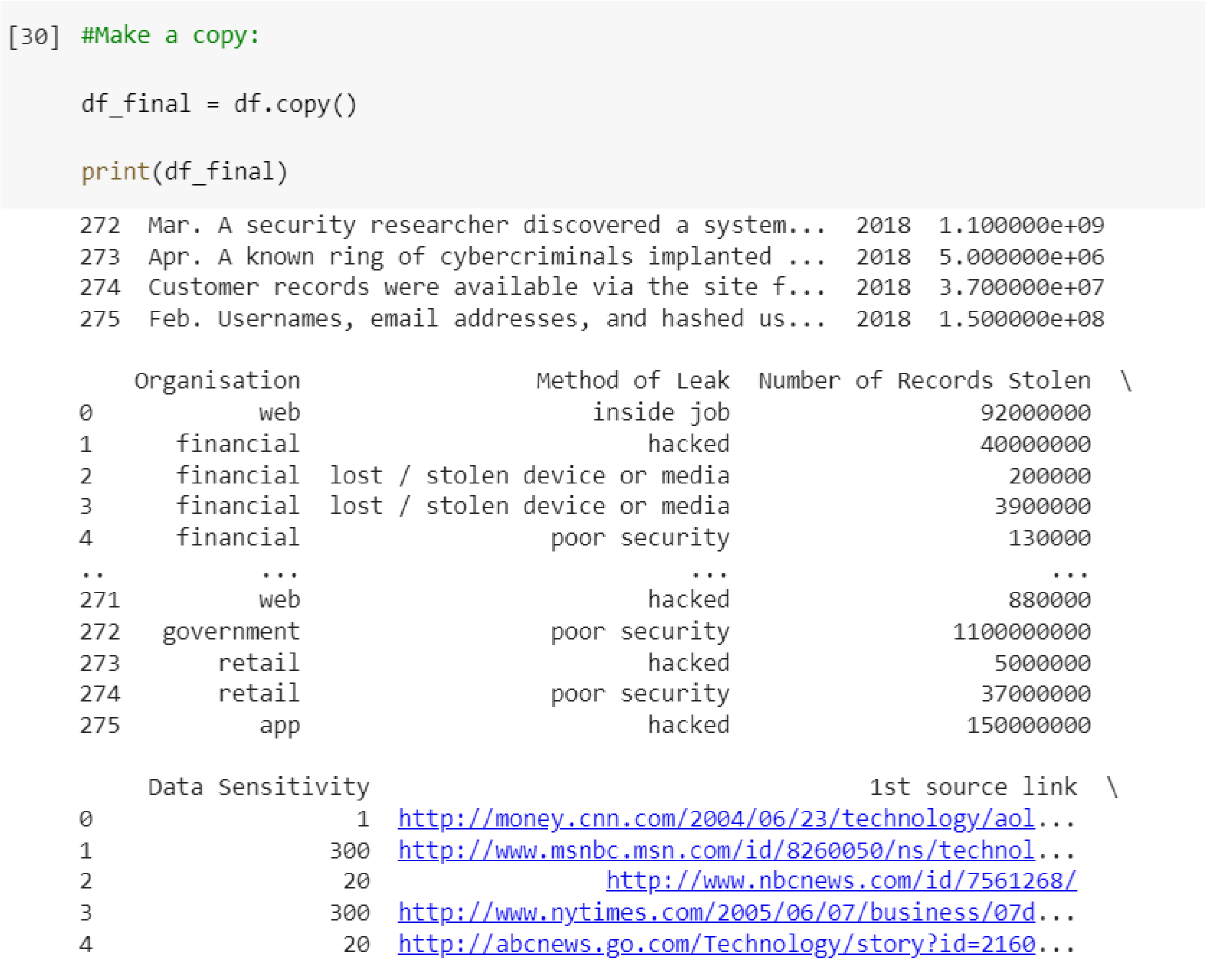
```
[28] df['Method of Leak'].unique()
```

```
array(['inside job', 'hacked', 'lost / stolen device or media',
       'poor security', 'accidentally published', 'hacked '], dtype=object)
```

```
[29] df['Data Sensitivity'].unique()
```

```
array([    1,   300,    20, 50000,  4000,     3])
```

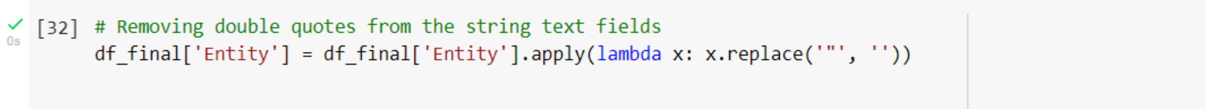
- Taking a backup of the data frame before starting cleaning process



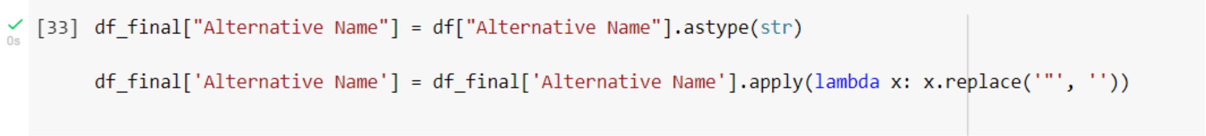
```
[30] #Make a copy:

     df_final = df.copy()

     print(df_final)
```

```
272  Mar. A security researcher discovered a system...  2018  1.100000e+09
273  Apr. A known ring of cybercriminals implanted ...  2018  5.000000e+06
274  Customer records were available via the site f...  2018  3.700000e+07
275  Feb. Usernames, email addresses, and hashed us...  2018  1.500000e+08

    Organisation                 Method of Leak  Number of Records Stolen  \
0            web                     inside job                  92000000
1      financial                         hacked                  40000000
2      financial  lost / stolen device or media                    200000
3      financial  lost / stolen device or media                   3900000
4      financial                  poor security                    130000
..           ...                            ...                       ...
271          web                         hacked                    880000
272   government                  poor security                1100000000
273       retail                         hacked                   5000000
274       retail                  poor security                  37000000
275          app                         hacked                 150000000

     Data Sensitivity                                   1st source link  \
0                   1  http://money.cnn.com/2004/06/23/technology/aol...
1                 300  http://www.msnbc.msn.com/id/8260050/ns/technol...
2                  20                http://www.nbcnews.com/id/7561268/
3                 300  http://www.nytimes.com/2005/06/07/business/07d...
4                  20  http://abcnews.go.com/Technology/story?id=2160...
```

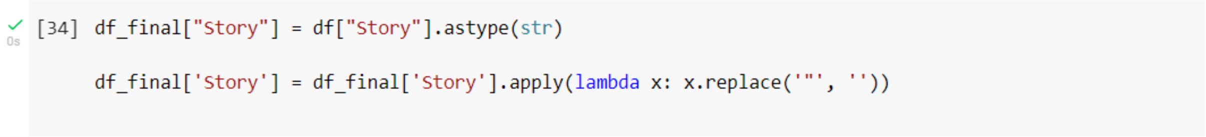
- Removed special characters “” – double quotes on the free text columns



```
[32] # Removing double quotes from the string text fields
     df_final['Entity'] = df_final['Entity'].apply(lambda x: x.replace('"', ''))
```

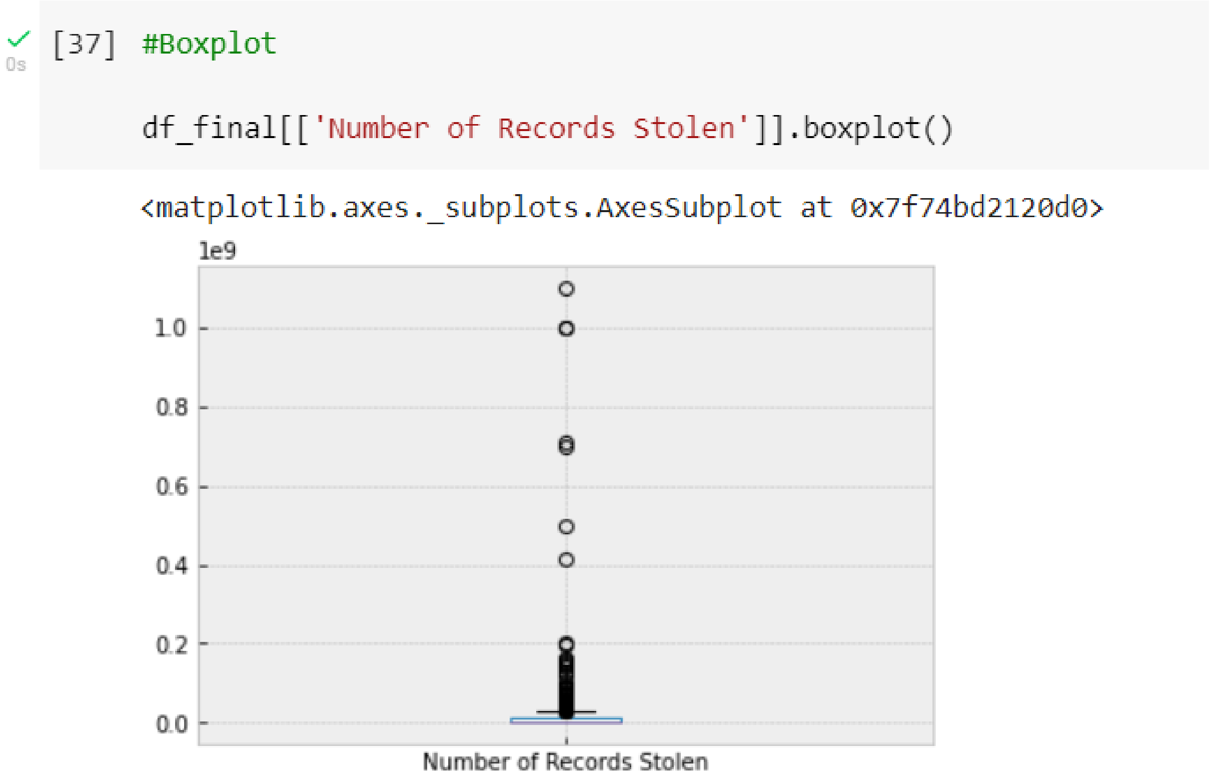


```
[33] df_final["Alternative Name"] = df["Alternative Name"].astype(str)

     df_final['Alternative Name'] = df_final['Alternative Name'].apply(lambda x: x.replace('"', ''))
```



```
[34] df_final["Story"] = df["Story"].astype(str)

     df_final['Story'] = df_final['Story'].apply(lambda x: x.replace('"', ''))
```

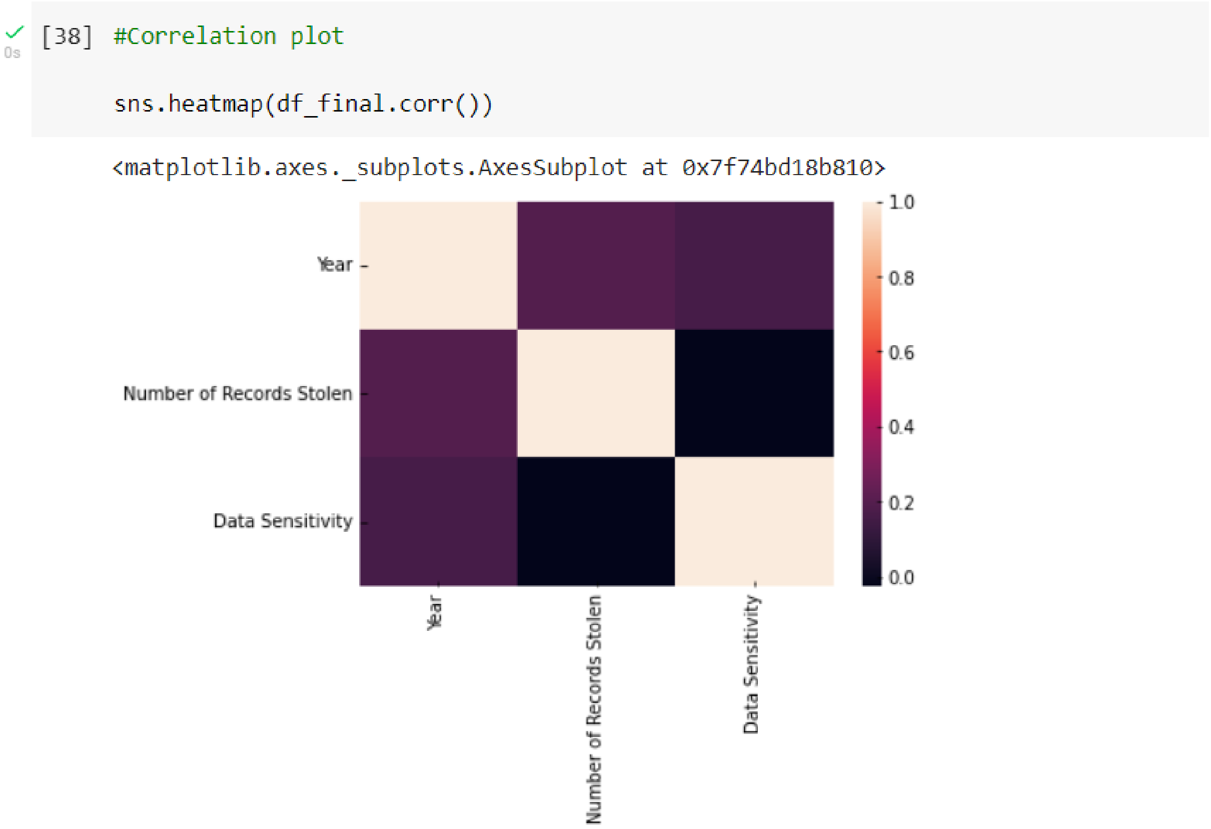
- Creating a box plot for numeric columns



```
[37] #Boxplot

     df_final[['Number of Records Stolen']].boxplot()
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f74bd2120d0>
```

- Created a corelation plot diagram



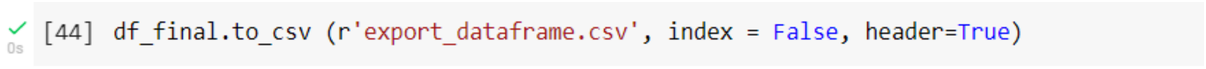
```
[38] #Correlation plot

     sns.heatmap(df_final.corr())

<matplotlib.axes._subplots.AxesSubplot at 0x7f74bd18b810>
```

- Converting Data Sensitivity Numeric column into Text details.

From	To
1	Just email address/Online information
20	SSN/Personal details
300	Credit card information
4000	Email password/Health records
50000	Full bank account details

```
#Converting Data Sensitivity Numeric column into Text details

df_final['Data Sensitivity'] = df_final['Data Sensitivity'].map({
1: "Email address/Online Info", 20: "SSN/Personal details", 300:
"Credit Card Info", 4000: "Email password/Health records", 50000:
 "Full bank account details"})
```

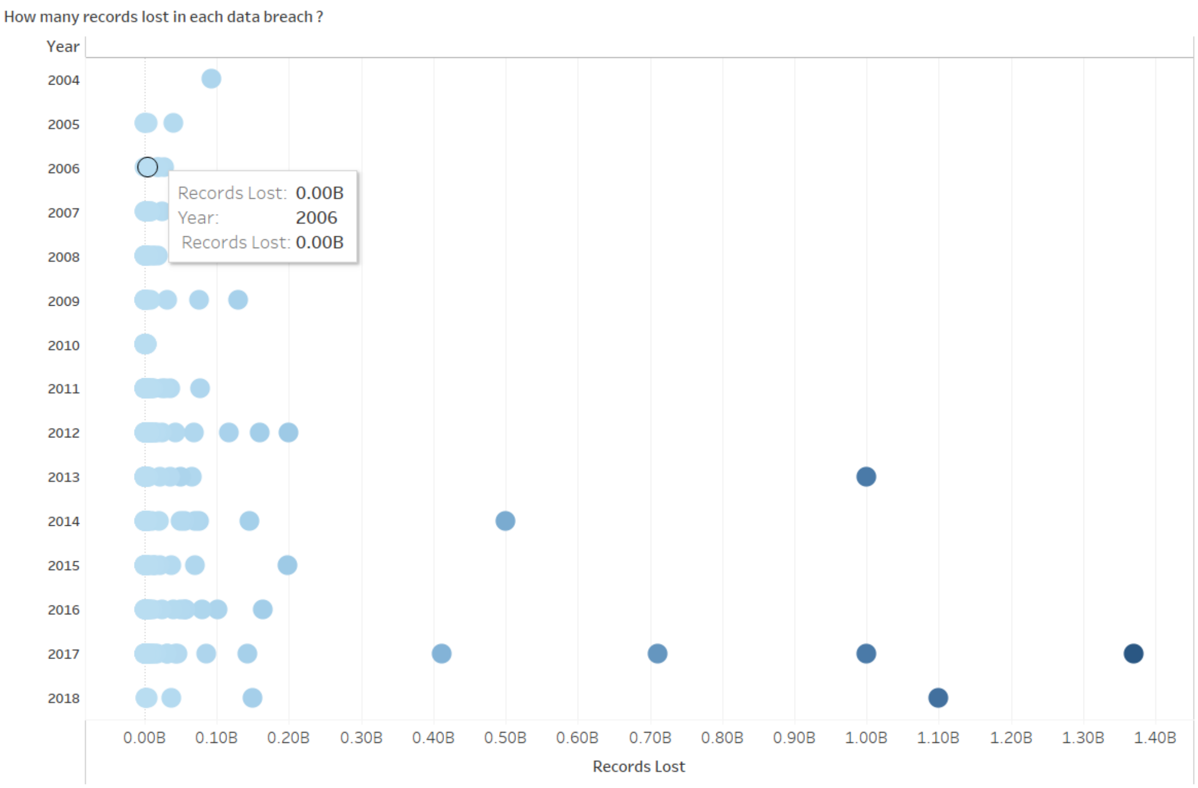
- Exporting the data frame to be loaded into Tableau.

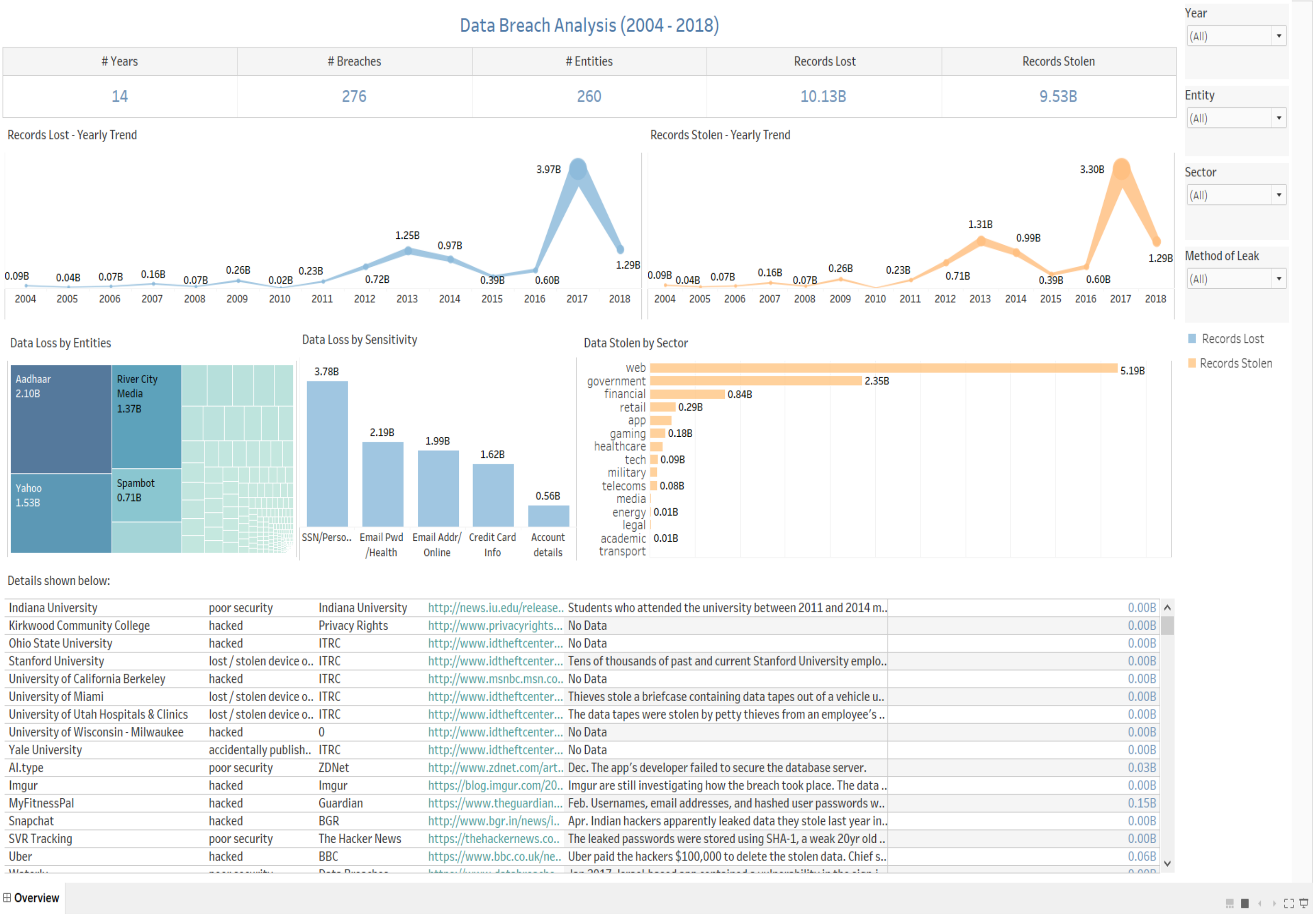


```
[44] df_final.to_csv (r'export_dataframe.csv', index = False, header=True)
```

Tell a Story / Visualizations

- For Visualization, I have used Tableau desktop application.
- Dashboard shows:
    I. Overview / Summary of the data
    II. Yearly Trend – Records Lost
    III. Yearly Trend – Records Stolen
    IV. Data Loss by Entities
    V. Data Loss by Sensitivity
    VI. Data Stolen by Sector
    VII. Detailed view for granular information
- Changed alias names to make data more readable
- Grouping done on Organization field (new field – Sector)
- Added Actions to make the dashboard interactive
- Added filters to get drill down information
- Key highlights are mentioned below
    a) Significant increase in the number of data breaches – data loss and data stolen over a span of years
    b) There were 276 data breaches and 260 companies affected.
    c) The total data loss was 10.13 B and data stolen was 9.53B.
    d) There was a major hike for data breaches on year 2017.
    e) Aadhar and Yahoo were amongst the top two companies affected.
    f) Majority of the data stolen was from Web sector
    g) In terms of data sensitivity, SSN / Personal details were lost / stolen from the companies database. There were couple of other factors too.



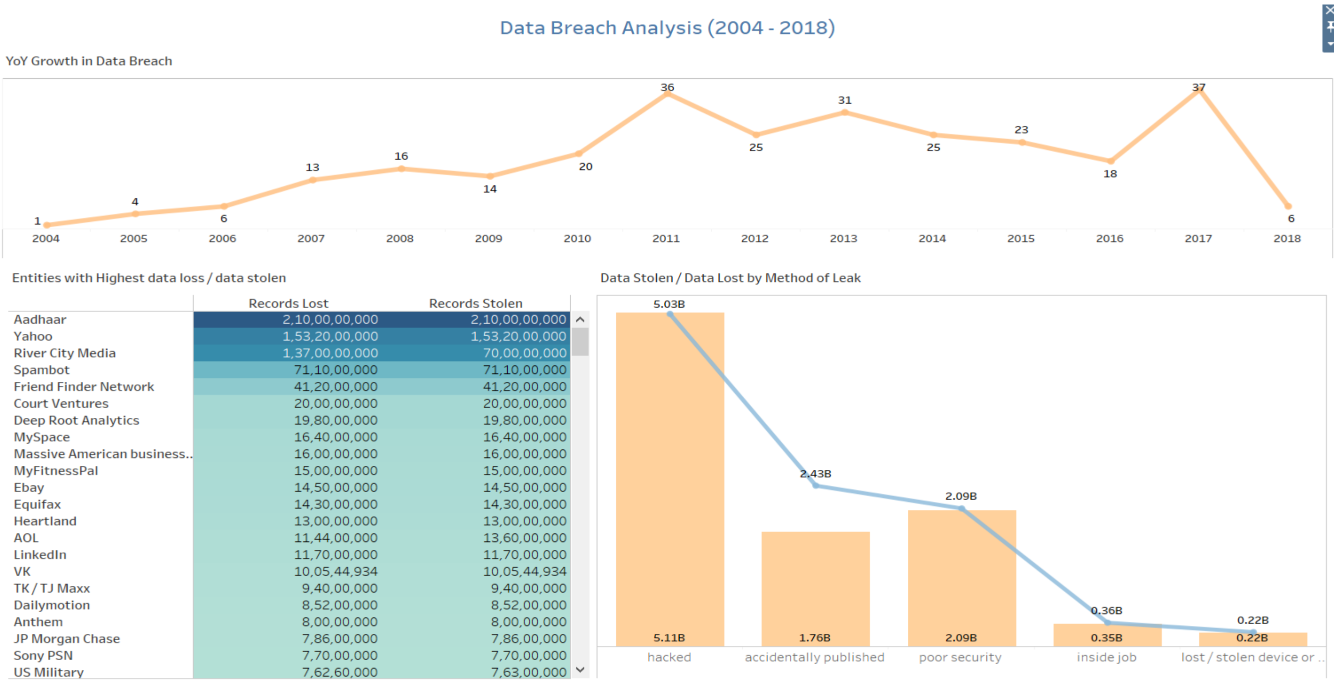


Data Breach Analysis (2004 - 2018)

# Years	# Breaches	# Entities	Records Lost	Records Stolen
14	276	260	10.13B	9.53B

Details shown below:

Indiana University	poor security	Indiana University	http://news.iu.edu/release..	Students who attended the university between 2011 and 2014 m..	0.00B
Kirkwood Community College	hacked	Privacy Rights	http://www.privacyrights...	No Data	0.00B
Ohio State University	hacked	ITRC	http://www.idtheftcenter...	No Data	0.00B
Stanford University	lost / stolen device o..	ITRC	http://www.idtheftcenter...	Tens of thousands of past and current Stanford University emplo..	0.00B
University of California Berkeley	hacked	ITRC	http://www.msnbc.msn.co..	No Data	0.00B
University of Miami	lost / stolen device o..	ITRC	http://www.idtheftcenter...	Thieves stole a briefcase containing data tapes out of a vehicle u..	0.00B
University of Utah Hospitals & Clinics	lost / stolen device o..	ITRC	http://www.idtheftcenter...	The data tapes were stolen by petty thieves from an employee's ..	0.00B
University of Wisconsin - Milwaukee	hacked	0	http://www.idtheftcenter...	No Data	0.00B
Yale University	accidentally publish..	ITRC	http://www.idtheftcenter...	No Data	0.00B
Al.type	poor security	ZDNet	http://www.zdnet.com/art..	Dec. The app's developer failed to secure the database server.	0.03B
Imgur	hacked	Imgur	https://blog.imgur.com/20..	Imgur are still investigating how the breach took place. The data ..	0.00B
MyFitnessPal	hacked	Guardian	https://www.theguardian...	Feb. Usernames, email addresses, and hashed user passwords w..	0.15B
Snapchat	hacked	BGR	http://www.bgr.in/news/i..	Apr. Indian hackers apparently leaked data they stole last year in..	0.00B
SVR Tracking	poor security	The Hacker News	https://thehackernews.co..	The leaked passwords were stored using SHA-1, a weak 20yr old ..	0.00B
Uber	hacked	BBC	https://www.bbc.co.uk/ne..	Uber paid the hackers $100,000 to delete the stolen data. Chief s..	0.06B



Data Breach Analysis (2004 - 2018)

YoY Growth in Data Breach

Entities with Highest data loss / data stolen

	Records Lost	Records Stolen
Aadhaar	2,10,00,00,000	2,10,00,00,000
Yahoo	1,53,20,00,000	1,53,20,00,000
River City Media	1,37,00,00,000	70,00,00,000
Spambot	71,10,00,000	71,10,00,000
Friend Finder Network	41,20,00,000	41,20,00,000
Court Ventures	20,00,00,000	20,00,00,000
Deep Root Analytics	19,80,00,000	19,80,00,000
MySpace	16,40,00,000	16,40,00,000
Massive American business..	16,00,00,000	16,00,00,000
MyFitnessPal	15,00,00,000	15,00,00,000
Ebay	14,50,00,000	14,50,00,000
Equifax	14,30,00,000	14,30,00,000
Heartland	13,00,00,000	13,00,00,000
AOL	11,44,00,000	13,60,00,000
LinkedIn	11,70,00,000	11,70,00,000
VK	10,05,44,934	10,05,44,934
TK / TJ Maxx	9,40,00,000	9,40,00,000
Dailymotion	8,52,00,000	8,52,00,000
Anthem	8,00,00,000	8,00,00,000
JP Morgan Chase	7,86,00,000	7,86,00,000
Sony PSN	7,70,00,000	7,70,00,000
US Military	7,62,60,000	7,63,00,000

Data Stolen / Data Lost by Method of Leak

Next Steps

- Doing deep dive analysis for small set of case types to understand more on data breaches and its significance.
- Enhancing capabilities to make dashboard more user friendly and improved experience.
- Taking Feedback / Inputs from stakeholders and give best optimal solution.
- Creating conversion factor to showcase numbers in millions, billions and whole numbers.