Collect Data from Files

We may have stored data in multiple types of files, such as text, csv, excel, xml, html, etc. We can load them into dataframes.

```
import pandas as pd
```

CSV

We have done this when we learned pandas. You can get the path of your csv file, and feed the path to the function `read_csv`.

Default setting

A lot cases, default setting will do the job.

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

```
df.head()
```

	Unnamed: 0	work_year	experience_level	employment_type	job_title	salary	s
0	0	2020	MI	FT	Data Scientist	70000	
1	1	2020	SE	FT	Machine Learning Scientist	260000	
2	2	2020	SE	FT	Big Data Engineer	85000	
3	3	2020	MI	FT	Product Data Analyst	20000	
4	4	2020	SE	FT	Machine Learning Engineer	150000	

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 607 entries, 0 to 606
Data columns (total 12 columns):
 #   Column              Non-Null Count  Dtype 
---  ------              --------------  ----- 
 0   Unnamed: 0          607 non-null    int64 
 1   work_year           607 non-null    int64 
 2   experience_level    607 non-null    object
 3   employment_type     607 non-null    object
 4   job_title           607 non-null    object
 5   salary              607 non-null    int64 
 6   salary_currency     607 non-null    object
 7   salary_in_usd       607 non-null    int64 
 8   employee_residence  607 non-null    object
 9   remote_ratio        607 non-null    int64 
 10  company_location    607 non-null    object
 11  company_size        607 non-null    object
dtypes: int64(5), object(7)
memory usage: 57.0+ KB
```

Customize setting

You can manipulate arguments for your specific csv file

```
df = pd.read_csv('/content/ds_salaries.csv', header = None)
df.head()
```

	0	1	2	3	4	5	6	
0	NaN	work_year	experience_level	employment_type	job_title	salary	salary_currency	sa
1	0.0	2020	MI	FT	Data Scientist	70000	EUR	
2	1.0	2020	SE	FT	Machine Learning Scientist	260000	USD	
3	2.0	2020	SE	FT	Big Data Engineer	85000	GBP	
4	3.0	2020	MI	FT	Product Data Analyst	20000	USD	

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 608 entries, 0 to 607
Data columns (total 12 columns):
 #   Column  Non-Null Count  Dtype  
---  ------  --------------  -----  
 0   0       607 non-null    float64
 1   1       608 non-null    object 
 2   2       608 non-null    object 
 3   3       608 non-null    object 
 4   4       608 non-null    object 
 5   5       608 non-null    object 
 6   6       608 non-null    object 
 7   7       608 non-null    object 
 8   8       608 non-null    object 
 9   9       608 non-null    object 
 10  10      608 non-null    object 
 11  11      608 non-null    object 
dtypes: float64(1), object(11)
memory usage: 57.1+ KB
```

```
df = pd.read_csv('/content/ds_salaries.csv', header = None, skiprows=1)
df.head()
```

	0	1	2	3	4	5	6	7	8	9	10	11
0	0	2020	MI	FT	Data Scientist	70000	EUR	79833	DE	0	DE	L
1	1	2020	SE	FT	Machine Learning Scientist	260000	USD	260000	JP	0	JP	S
2	2	2020	SE	FT	Big Data Engineer	85000	GBP	109024	GB	50	GB	M
3	3	2020	MI	FT	Product Data Analyst	20000	USD	20000	HN	0	HN	S
4	4	2020	SE	FT	Machine Learning Engineer	150000	USD	150000	US	50	US	L

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 607 entries, 0 to 606
Data columns (total 12 columns):
 #   Column  Non-Null Count  Dtype 
---  ------  --------------  ----- 
 0   0       607 non-null    int64 
 1   1       607 non-null    int64 
 2   2       607 non-null    object
 3   3       607 non-null    object
 4   4       607 non-null    object
 5   5       607 non-null    int64 
 6   6       607 non-null    object
 7   7       607 non-null    int64 
 8   8       607 non-null    object
 9   9       607 non-null    int64 
 10  10      607 non-null    object
 11  11      607 non-null    object
dtypes: int64(5), object(7)
memory usage: 57.0+ KB
```

```
df = pd.read_csv('/content/ds_salaries.csv', header = None, skiprows=1, skipfooter=
df.head()
```

```
<ipython-input-21-7818cbc15790>:1: ParserWarning: Falling back to the 'python'
  df = pd.read_csv('/content/ds_salaries.csv', header = None, skiprows=1, skip
```

	0	1	2	3	4	5	6	7	8	9	10	11
0	0	2020	MI	FT	Data Scientist	70000	EUR	79833	DE	0	DE	L
1	1	2020	SE	FT	Machine Learning Scientist	260000	USD	260000	JP	0	JP	S
2	2	2020	SE	FT	Big Data Engineer	85000	GBP	109024	GB	50	GB	M
3	3	2020	MI	FT	Product Data Analyst	20000	USD	20000	HN	0	HN	S
4	4	2020	SE	FT	Machine Learning Engineer	150000	USD	150000	US	50	US	L

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 307 entries, 0 to 306
Data columns (total 12 columns):
 #   Column  Non-Null Count  Dtype 
---  ------  --------------  ----- 
 0   0       307 non-null    int64 
 1   1       307 non-null    int64 
 2   2       307 non-null    object
 3   3       307 non-null    object
 4   4       307 non-null    object
 5   5       307 non-null    int64 
 6   6       307 non-null    object
 7   7       307 non-null    int64 
 8   8       307 non-null    object
 9   9       307 non-null    int64 
 10  10      307 non-null    object
 11  11      307 non-null    object
dtypes: int64(5), object(7)
memory usage: 28.9+ KB
```

TXT

> If the txt follows csv format, then it can be read as a csv file

```
df = pd.read_csv('/content/ds_salaries.txt')
df
```

	Unnamed: 0	work_year	experience_level	employment_type	job_title	salary
0	0	2020	MI	FT	Data Scientist	70000
1	1	2020	SE	FT	Machine Learning Scientist	260000
2	2	2020	SE	FT	Big Data Engineer	85000
3	3	2020	MI	FT	Product Data Analyst	20000
4	4	2020	SE	FT	Machine Learning Engineer	150000
...	...	...	...	...	...	...
602	602	2022	SE	FT	Data Engineer	154000
603	603	2022	SE	FT	Data Engineer	126000
604	604	2022	SE	FT	Data Analyst	129000
605	605	2022	SE	FT	Data Analyst	150000
606	606	2022	MI	FT	AI Scientist	200000

607 rows × 12 columns

Excel

```
df = pd.read_excel('/content/ds_salaries.xlsx')
```

```
df.head()
```

	Unnamed: 0	work_year	experience_level	employment_type	job_title	salary	s
0	0	2020	MI	FT	Data Scientist	70000	
1	1	2020	SE	FT	Machine Learning Scientist	260000	
2	2	2020	SE	FT	Big Data Engineer	85000	
3	3	2020	MI	FT	Product Data Analyst	20000	
4	4	2020	SE	FT	Machine Learning Engineer	150000	

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 607 entries, 0 to 606
Data columns (total 12 columns):
 #   Column              Non-Null Count  Dtype 
---  ------              --------------  ----- 
 0   Unnamed: 0          607 non-null    int64 
 1   work_year           607 non-null    int64 
 2   experience_level    607 non-null    object
 3   employment_type     607 non-null    object
 4   job_title           607 non-null    object
 5   salary              607 non-null    int64 
 6   salary_currency     607 non-null    object
 7   salary_in_usd       607 non-null    int64 
 8   employee_residence  607 non-null    object
 9   remote_ratio        607 non-null    int64 
 10  company_location    607 non-null    object
 11  company_size        607 non-null    object
dtypes: int64(5), object(7)
memory usage: 57.0+ KB
```

▾ json

```
df = pd.read_json('/content/ds_salaries.json')
df.head()
```

	FIELD1	work_year	experience_level	employment_type	job_title	salary	sal
0	0	2020	MI	FT	Data Scientist	70000	
1	1	2020	SE	FT	Machine Learning Scientist	260000	
2	2	2020	SE	FT	Big Data Engineer	85000	
3	3	2020	MI	FT	Product Data Analyst	20000	
4	4	2020	SE	FT	Machine Learning Engineer	150000	

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 607 entries, 0 to 606
Data columns (total 12 columns):
 #   Column              Non-Null Count  Dtype 
---  ------              --------------  ----- 
 0   FIELD1              607 non-null    int64 
 1   work_year           607 non-null    int64 
 2   experience_level    607 non-null    object
 3   employment_type     607 non-null    object
 4   job_title           607 non-null    object
 5   salary              607 non-null    int64 
 6   salary_currency     607 non-null    object
 7   salary_in_usd       607 non-null    int64 
 8   employee_residence  607 non-null    object
 9   remote_ratio        607 non-null    int64 
 10  company_location    607 non-null    object
 11  company_size        607 non-null    object
dtypes: int64(5), object(7)
memory usage: 57.0+ KB
```

XML

```
df = pd.read_xml('/content/ds_salaries.xml')
df.head()
```

	FIELD1	work_year	experience_level	employment_type	job_title	salary	sal
0	0	2020	MI	FT	Data Scientist	70000	
1	1	2020	SE	FT	Machine Learning Scientist	260000	
2	2	2020	SE	FT	Big Data Engineer	85000	
3	3	2020	MI	FT	Product Data Analyst	20000	
4	4	2020	SE	FT	Machine Learning Engineer	150000	

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 607 entries, 0 to 606
Data columns (total 12 columns):
 #   Column              Non-Null Count  Dtype 
---  ------              --------------  ----- 
 0   FIELD1              607 non-null    int64 
 1   work_year           607 non-null    int64 
 2   experience_level    607 non-null    object
 3   employment_type     607 non-null    object
 4   job_title           607 non-null    object
 5   salary              607 non-null    int64 
 6   salary_currency     607 non-null    object
 7   salary_in_usd       607 non-null    int64 
 8   employee_residence  607 non-null    object
 9   remote_ratio        607 non-null    int64 
 10  company_location    607 non-null    object
 11  company_size        607 non-null    object
dtypes: int64(5), object(7)
memory usage: 57.0+ KB
```

HTM

```
df = pd.read_html('/content/ds_salaries.htm')[0]
df.head()
```

	FIELD1	work_year	experience_level	employment_type	job_title	salary	sal
0	0	2020	MI	FT	Data Scientist	70000	
1	1	2020	SE	FT	Machine Learning Scientist	260000	
2	2	2020	SE	FT	Big Data Engineer	85000	
3	3	2020	MI	FT	Product Data Analyst	20000	
4	4	2020	SE	FT	Machine Learning Engineer	150000	

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 607 entries, 0 to 606
Data columns (total 12 columns):
 #   Column              Non-Null Count  Dtype 
---  ------              --------------  ----- 
 0   FIELD1              607 non-null    int64 
 1   work_year           607 non-null    int64 
 2   experience_level    607 non-null    object
 3   employment_type     607 non-null    object
 4   job_title           607 non-null    object
 5   salary              607 non-null    int64 
 6   salary_currency     607 non-null    object
 7   salary_in_usd       607 non-null    int64 
 8   employee_residence  607 non-null    object
 9   remote_ratio        607 non-null    int64 
 10  company_location    607 non-null    object
 11  company_size        607 non-null    object
dtypes: int64(5), object(7)
memory usage: 57.0+ KB
```

Documentation

It is always good to have a reference of the read files functions in pandas. You can find it via https://pandas.pydata.org/docs/reference/io.html