Звіт до лабораторної роботи №3

з предмету "Моделі інтелектуальних систем"

Дровольського Ярослава, ІПС-42

Варіант №2

Install pandas for Jupyter Notebook environment

In [2]:
```
!pip install pandas
```

```
Collecting pandas
  Downloading pandas-2.2.0-cp311-cp311-win_amd64.whl.metadata (19 kB)
Requirement already satisfied: numpy<2,>=1.23.2 in d:\programfiles\anaconda3\envs
\notebook\lib\site-packages (from pandas) (1.26.3)
Requirement already satisfied: python-dateutil>=2.8.2 in c:\users\lenovo\appdata
\roaming\python\python311\site-packages (from pandas) (2.8.2)
Requirement already satisfied: pytz>=2020.1 in d:\programfiles\anaconda3\envs\not
ebook\lib\site-packages (from pandas) (2023.3.post1)
Collecting tzdata>=2022.7 (from pandas)
  Downloading tzdata-2023.4-py2.py3-none-any.whl.metadata (1.4 kB)
Requirement already satisfied: six>=1.5 in c:\users\lenovo\appdata\roaming\python
\python311\site-packages (from python-dateutil>=2.8.2->pandas) (1.16.0)
Downloading pandas-2.2.0-cp311-cp311-win_amd64.whl (11.6 MB)
   ---------------------------------------- 0.0/11.6 MB ? eta -:--:--
   ---------------------------------------- 0.0/11.6 MB 660.6 kB/s eta 0:00:18
   ---------------------------------------- 0.1/11.6 MB 1.7 MB/s eta 0:00:07
   - -------------------------------------- 0.4/11.6 MB 3.1 MB/s eta 0:00:04
   -- ------------------------------------- 0.7/11.6 MB 4.2 MB/s eta 0:00:03
   -- ------------------------------------- 0.9/11.6 MB 3.9 MB/s eta 0:00:03
   ---- ----------------------------------- 1.2/11.6 MB 4.4 MB/s eta 0:00:03
   ----- ---------------------------------- 1.5/11.6 MB 4.9 MB/s eta 0:00:03
   ------ --------------------------------- 1.8/11.6 MB 5.0 MB/s eta 0:00:02
   ------- -------------------------------- 2.1/11.6 MB 5.0 MB/s eta 0:00:02
   -------- ------------------------------- 2.4/11.6 MB 5.3 MB/s eta 0:00:02
   --------- ------------------------------ 2.8/11.6 MB 5.3 MB/s eta 0:00:02
   ---------- ----------------------------- 3.1/11.6 MB 5.5 MB/s eta 0:00:02
   ----------- ---------------------------- 3.4/11.6 MB 5.7 MB/s eta 0:00:02
   ------------ --------------------------- 3.7/11.6 MB 5.7 MB/s eta 0:00:02
   -------------- ------------------------- 4.1/11.6 MB 5.8 MB/s eta 0:00:02
   --------------- ------------------------ 4.4/11.6 MB 6.0 MB/s eta 0:00:02
   ---------------- ----------------------- 4.7/11.6 MB 5.9 MB/s eta 0:00:02
   ----------------- ---------------------- 5.0/11.6 MB 6.0 MB/s eta 0:00:02
   ------------------ --------------------- 5.4/11.6 MB 6.0 MB/s eta 0:00:02
   ------------------- -------------------- 5.7/11.6 MB 6.1 MB/s eta 0:00:01
   -------------------- ------------------- 6.0/11.6 MB 6.1 MB/s eta 0:00:01
   ---------------------- ----------------- 6.4/11.6 MB 6.2 MB/s eta 0:00:01
   ----------------------- ---------------- 6.7/11.6 MB 6.2 MB/s eta 0:00:01
   ------------------------ --------------- 7.1/11.6 MB 6.3 MB/s eta 0:00:01
   ------------------------- -------------- 7.4/11.6 MB 6.3 MB/s eta 0:00:01
   -------------------------- ------------- 7.7/11.6 MB 6.3 MB/s eta 0:00:01
   --------------------------- ------------ 8.1/11.6 MB 6.4 MB/s eta 0:00:01
   ---------------------------- ----------- 8.4/11.6 MB 6.4 MB/s eta 0:00:01
   ----------------------------- ---------- 8.7/11.6 MB 6.4 MB/s eta 0:00:01
   ------------------------------ --------- 9.1/11.6 MB 6.4 MB/s eta 0:00:01
   ------------------------------- -------- 9.4/11.6 MB 6.5 MB/s eta 0:00:01
   -------------------------------- ------- 9.7/11.6 MB 6.4 MB/s eta 0:00:01
   -------------------------------- ------- 9.9/11.6 MB 6.4 MB/s eta 0:00:01
   --------------------------------- ------ 10.3/11.6 MB 6.5 MB/s eta 0:00:01
   ---------------------------------- ----- 10.6/11.6 MB 6.7 MB/s eta 0:00:01
   ---------------------------------- ----- 10.9/11.6 MB 6.6 MB/s eta 0:00:01
   ---------------------------------------  11.3/11.6 MB 6.9 MB/s eta 0:00:01
   ---------------------------------------  11.6/11.6 MB 6.8 MB/s eta 0:00:01
   ---------------------------------------- 11.6/11.6 MB 6.7 MB/s eta 0:00:00
Downloading tzdata-2023.4-py2.py3-none-any.whl (346 kB)
   ---------------------------------------- 0.0/346.6 kB ? eta -:--:--
   ---------------------------------------  337.9/346.6 kB 10.6 MB/s eta 0:00:01
   ---------------------------------------- 346.6/346.6 kB 10.8 MB/s eta 0:00:00
Installing collected packages: tzdata, pandas
Successfully installed pandas-2.2.0 tzdata-2023.4
```

Install matplotlib for Jupyter Notebook environment to make plots

In [171…

```
!pip install matplotlib
```

```
Collecting matplotlib
  Downloading matplotlib-3.8.2-cp311-cp311-win_amd64.whl.metadata (5.9 kB)
Collecting contourpy>=1.0.1 (from matplotlib)
  Downloading contourpy-1.2.0-cp311-cp311-win_amd64.whl.metadata (5.8 kB)
Collecting cycler>=0.10 (from matplotlib)
  Downloading cycler-0.12.1-py3-none-any.whl.metadata (3.8 kB)
Collecting fonttools>=4.22.0 (from matplotlib)
  Downloading fonttools-4.48.1-cp311-cp311-win_amd64.whl.metadata (162 kB)
     ---------------------------------------- 0.0/162.2 kB ? eta -:--:--
     -- ------------------------------------- 10.2/162.2 kB ? eta -:--:--
     ------------- -------------------------- 61.4/162.2 kB 656.4 kB/s eta 0:00:01
     ---------------------------------------- 162.2/162.2 kB 1.4 MB/s eta 0:00:00
Collecting kiwisolver>=1.3.1 (from matplotlib)
  Downloading kiwisolver-1.4.5-cp311-cp311-win_amd64.whl.metadata (6.5 kB)
Requirement already satisfied: numpy<2,>=1.21 in d:\programfiles\anaconda3\envs\n
otebook\lib\site-packages (from matplotlib) (1.26.3)
Requirement already satisfied: packaging>=20.0 in d:\programfiles\anaconda3\envs
\notebook\lib\site-packages (from matplotlib) (23.1)
Collecting pillow>=8 (from matplotlib)
  Downloading pillow-10.2.0-cp311-cp311-win_amd64.whl.metadata (9.9 kB)
Collecting pyparsing>=2.3.1 (from matplotlib)
  Downloading pyparsing-3.1.1-py3-none-any.whl.metadata (5.1 kB)
Requirement already satisfied: python-dateutil>=2.7 in c:\users\lenovo\appdata\ro
aming\python\python311\site-packages (from matplotlib) (2.8.2)
Requirement already satisfied: six>=1.5 in c:\users\lenovo\appdata\roaming\python
\python311\site-packages (from python-dateutil>=2.7->matplotlib) (1.16.0)
Downloading matplotlib-3.8.2-cp311-cp311-win_amd64.whl (7.6 MB)
   ---------------------------------------- 0.0/7.6 MB ? eta -:--:--
   - -------------------------------------- 0.2/7.6 MB 5.8 MB/s eta 0:00:02
   -- ------------------------------------- 0.5/7.6 MB 5.6 MB/s eta 0:00:02
   --- ------------------------------------ 0.7/7.6 MB 4.8 MB/s eta 0:00:02
   ----- ---------------------------------- 1.0/7.6 MB 5.7 MB/s eta 0:00:02
   ------- -------------------------------- 1.3/7.6 MB 5.7 MB/s eta 0:00:02
   -------- ------------------------------- 1.6/7.6 MB 5.9 MB/s eta 0:00:02
   --------- ------------------------------ 1.8/7.6 MB 5.6 MB/s eta 0:00:02
   ---------- ----------------------------- 2.0/7.6 MB 5.3 MB/s eta 0:00:02
   ----------- ---------------------------- 2.2/7.6 MB 5.2 MB/s eta 0:00:02
   ------------ --------------------------- 2.4/7.6 MB 5.0 MB/s eta 0:00:02
   ------------- -------------------------- 2.5/7.6 MB 5.0 MB/s eta 0:00:02
   -------------- ------------------------- 2.7/7.6 MB 4.8 MB/s eta 0:00:02
   -------------- ------------------------- 2.8/7.6 MB 4.6 MB/s eta 0:00:02
   --------------- ------------------------ 2.9/7.6 MB 4.4 MB/s eta 0:00:02
   --------------- ------------------------ 3.0/7.6 MB 4.3 MB/s eta 0:00:02
   ---------------- ----------------------- 3.1/7.6 MB 4.1 MB/s eta 0:00:02
   ---------------- ----------------------- 3.2/7.6 MB 3.9 MB/s eta 0:00:02
   ----------------- ---------------------- 3.3/7.6 MB 3.9 MB/s eta 0:00:02
   ----------------- ---------------------- 3.4/7.6 MB 3.8 MB/s eta 0:00:02
   ----------------- ---------------------- 3.4/7.6 MB 3.7 MB/s eta 0:00:02
   ------------------ --------------------- 3.5/7.6 MB 3.6 MB/s eta 0:00:02
   ------------------ --------------------- 3.6/7.6 MB 3.5 MB/s eta 0:00:02
   ------------------- -------------------- 3.7/7.6 MB 3.4 MB/s eta 0:00:02
   ------------------- -------------------- 3.8/7.6 MB 3.3 MB/s eta 0:00:02
   -------------------- ------------------- 3.9/7.6 MB 3.2 MB/s eta 0:00:02
   -------------------- ------------------- 3.9/7.6 MB 3.2 MB/s eta 0:00:02
   --------------------- ------------------ 4.0/7.6 MB 3.1 MB/s eta 0:00:02
   --------------------- ------------------ 4.1/7.6 MB 3.1 MB/s eta 0:00:02
   --------------------- ------------------ 4.1/7.6 MB 3.0 MB/s eta 0:00:02
   ---------------------- ----------------- 4.2/7.6 MB 3.0 MB/s eta 0:00:02
   ---------------------- ----------------- 4.3/7.6 MB 2.9 MB/s eta 0:00:02
   ---------------------- ----------------- 4.3/7.6 MB 2.8 MB/s eta 0:00:02
```

```
---------------------- ----------------- 4.4/7.6 MB 2.8 MB/s eta 0:00:02
---------------------- ----------------- 4.4/7.6 MB 2.7 MB/s eta 0:00:02
---------------------- ----------------- 4.5/7.6 MB 2.7 MB/s eta 0:00:02
---------------------- ----------------- 4.5/7.6 MB 2.6 MB/s eta 0:00:02
---------------------- ----------------- 4.5/7.6 MB 2.6 MB/s eta 0:00:02
---------------------- ----------------- 4.6/7.6 MB 2.5 MB/s eta 0:00:02
----------------------- ---------------- 4.6/7.6 MB 2.5 MB/s eta 0:00:02
----------------------- ---------------- 4.6/7.6 MB 2.4 MB/s eta 0:00:02
----------------------- ---------------- 4.7/7.6 MB 2.4 MB/s eta 0:00:02
----------------------- ---------------- 4.7/7.6 MB 2.4 MB/s eta 0:00:02
----------------------- ---------------- 4.7/7.6 MB 2.3 MB/s eta 0:00:02
----------------------- ---------------- 4.8/7.6 MB 2.3 MB/s eta 0:00:02
------------------------ --------------- 4.8/7.6 MB 2.3 MB/s eta 0:00:02
------------------------ --------------- 4.9/7.6 MB 2.2 MB/s eta 0:00:02
------------------------ --------------- 4.9/7.6 MB 2.2 MB/s eta 0:00:02
------------------------ --------------- 4.9/7.6 MB 2.2 MB/s eta 0:00:02
------------------------- -------------- 5.0/7.6 MB 2.1 MB/s eta 0:00:02
------------------------- -------------- 5.0/7.6 MB 2.1 MB/s eta 0:00:02
------------------------- -------------- 5.1/7.6 MB 2.1 MB/s eta 0:00:02
------------------------- -------------- 5.1/7.6 MB 2.1 MB/s eta 0:00:02
------------------------- -------------- 5.1/7.6 MB 2.1 MB/s eta 0:00:02
-------------------------- ------------- 5.2/7.6 MB 2.0 MB/s eta 0:00:02
-------------------------- ------------- 5.2/7.6 MB 2.0 MB/s eta 0:00:02
-------------------------- ------------- 5.3/7.6 MB 2.0 MB/s eta 0:00:02
-------------------------- ------------- 5.3/7.6 MB 2.0 MB/s eta 0:00:02
-------------------------- ------------- 5.3/7.6 MB 1.9 MB/s eta 0:00:02
--------------------------- ------------ 5.4/7.6 MB 1.9 MB/s eta 0:00:02
--------------------------- ------------ 5.4/7.6 MB 1.9 MB/s eta 0:00:02
--------------------------- ------------ 5.5/7.6 MB 1.9 MB/s eta 0:00:02
--------------------------- ------------ 5.5/7.6 MB 1.9 MB/s eta 0:00:02
--------------------------- ------------ 5.5/7.6 MB 1.9 MB/s eta 0:00:02
---------------------------- ----------- 5.6/7.6 MB 1.8 MB/s eta 0:00:02
---------------------------- ----------- 5.6/7.6 MB 1.8 MB/s eta 0:00:02
---------------------------- ----------- 5.7/7.6 MB 1.8 MB/s eta 0:00:02
---------------------------- ----------- 5.7/7.6 MB 1.8 MB/s eta 0:00:02
---------------------------- ----------- 5.7/7.6 MB 1.8 MB/s eta 0:00:02
----------------------------- ---------- 5.8/7.6 MB 1.8 MB/s eta 0:00:02
----------------------------- ---------- 5.8/7.6 MB 1.7 MB/s eta 0:00:02
----------------------------- ---------- 5.8/7.6 MB 1.7 MB/s eta 0:00:02
----------------------------- ---------- 5.9/7.6 MB 1.7 MB/s eta 0:00:02
------------------------------ --------- 5.9/7.6 MB 1.7 MB/s eta 0:00:01
------------------------------ --------- 6.0/7.6 MB 1.7 MB/s eta 0:00:01
------------------------------ --------- 6.0/7.6 MB 1.7 MB/s eta 0:00:01
------------------------------ --------- 6.1/7.6 MB 1.7 MB/s eta 0:00:01
------------------------------- -------- 6.1/7.6 MB 1.7 MB/s eta 0:00:01
------------------------------- -------- 6.2/7.6 MB 1.7 MB/s eta 0:00:01
------------------------------- -------- 6.2/7.6 MB 1.7 MB/s eta 0:00:01
-------------------------------- ------- 6.3/7.6 MB 1.7 MB/s eta 0:00:01
-------------------------------- ------- 6.4/7.6 MB 1.7 MB/s eta 0:00:01
-------------------------------- ------- 6.4/7.6 MB 1.6 MB/s eta 0:00:01
-------------------------------- ------- 6.5/7.6 MB 1.6 MB/s eta 0:00:01
--------------------------------- ------ 6.6/7.6 MB 1.6 MB/s eta 0:00:01
--------------------------------- ------ 6.6/7.6 MB 1.6 MB/s eta 0:00:01
--------------------------------- ------ 6.7/7.6 MB 1.6 MB/s eta 0:00:01
---------------------------------- ----- 6.8/7.6 MB 1.6 MB/s eta 0:00:01
---------------------------------- ----- 6.8/7.6 MB 1.6 MB/s eta 0:00:01
---------------------------------- ----- 6.9/7.6 MB 1.6 MB/s eta 0:00:01
---------------------------------- ----- 7.0/7.6 MB 1.6 MB/s eta 0:00:01
---------------------------------- ----- 7.0/7.6 MB 1.6 MB/s eta 0:00:01
---------------------------------- ----- 7.1/7.6 MB 1.6 MB/s eta 0:00:01
```

```
   ---------------------------------------- -- 7.2/7.6 MB 1.6 MB/s eta 0:00:01
   ---------------------------------------- - 7.3/7.6 MB 1.6 MB/s eta 0:00:01
   ---------------------------------------- - 7.4/7.6 MB 1.6 MB/s eta 0:00:01
   ---------------------------------------  7.5/7.6 MB 1.6 MB/s eta 0:00:01
   ---------------------------------------  7.5/7.6 MB 1.6 MB/s eta 0:00:01
   ---------------------------------------  7.6/7.6 MB 1.6 MB/s eta 0:00:01
   ---------------------------------------- 7.6/7.6 MB 1.6 MB/s eta 0:00:00
Downloading contourpy-1.2.0-cp311-cp311-win_amd64.whl (187 kB)
   ---------------------------------------- 0.0/187.6 kB ? eta -:--:--
   ---------------- ---------------------- 81.9/187.6 kB 2.2 MB/s eta 0:00:01
   ---------------------------------------- -- 174.1/187.6 kB 1.7 MB/s eta 0:00:01
   ---------------------------------------- 187.6/187.6 kB 1.6 MB/s eta 0:00:00
Downloading cycler-0.12.1-py3-none-any.whl (8.3 kB)
Downloading fonttools-4.48.1-cp311-cp311-win_amd64.whl (2.2 MB)
   ---------------------------------------- 0.0/2.2 MB ? eta -:--:--
   - -------------------------------------- 0.1/2.2 MB 1.7 MB/s eta 0:00:02
   --- ------------------------------------ 0.2/2.2 MB 1.9 MB/s eta 0:00:02
   ----- ---------------------------------- 0.3/2.2 MB 2.1 MB/s eta 0:00:01
   ------ --------------------------------- 0.4/2.2 MB 2.0 MB/s eta 0:00:01
   ------- -------------------------------- 0.4/2.2 MB 1.9 MB/s eta 0:00:01
   -------- ------------------------------- 0.5/2.2 MB 1.8 MB/s eta 0:00:01
   --------- ------------------------------ 0.5/2.2 MB 1.7 MB/s eta 0:00:01
   ---------- ----------------------------- 0.6/2.2 MB 1.6 MB/s eta 0:00:02
   ----------- ---------------------------- 0.6/2.2 MB 1.5 MB/s eta 0:00:02
   ------------ --------------------------- 0.7/2.2 MB 1.5 MB/s eta 0:00:02
   ------------- -------------------------- 0.7/2.2 MB 1.4 MB/s eta 0:00:02
   -------------- ------------------------- 0.8/2.2 MB 1.4 MB/s eta 0:00:02
   --------------- ------------------------ 0.9/2.2 MB 1.4 MB/s eta 0:00:01
   ---------------- ----------------------- 0.9/2.2 MB 1.4 MB/s eta 0:00:01
   ----------------- ---------------------- 1.0/2.2 MB 1.4 MB/s eta 0:00:01
   ------------------ --------------------- 1.0/2.2 MB 1.4 MB/s eta 0:00:01
   ------------------- -------------------- 1.1/2.2 MB 1.3 MB/s eta 0:00:01
   -------------------- ------------------- 1.1/2.2 MB 1.3 MB/s eta 0:00:01
   --------------------- ------------------ 1.2/2.2 MB 1.3 MB/s eta 0:00:01
   ---------------------- ----------------- 1.2/2.2 MB 1.3 MB/s eta 0:00:01
   ----------------------- ---------------- 1.3/2.2 MB 1.2 MB/s eta 0:00:01
   ------------------------ --------------- 1.3/2.2 MB 1.2 MB/s eta 0:00:01
   ------------------------ --------------- 1.3/2.2 MB 1.2 MB/s eta 0:00:01
   ------------------------- -------------- 1.4/2.2 MB 1.2 MB/s eta 0:00:01
   -------------------------- ------------- 1.5/2.2 MB 1.2 MB/s eta 0:00:01
   --------------------------- ------------ 1.5/2.2 MB 1.2 MB/s eta 0:00:01
   ---------------------------- ----------- 1.6/2.2 MB 1.2 MB/s eta 0:00:01
   ---------------------------- ----------- 1.6/2.2 MB 1.2 MB/s eta 0:00:01
   ----------------------------- ---------- 1.7/2.2 MB 1.2 MB/s eta 0:00:01
   ------------------------------ --------- 1.7/2.2 MB 1.2 MB/s eta 0:00:01
   ------------------------------- -------- 1.8/2.2 MB 1.2 MB/s eta 0:00:01
   -------------------------------- ------- 1.9/2.2 MB 1.2 MB/s eta 0:00:01
   -------------------------------- ------- 1.9/2.2 MB 1.2 MB/s eta 0:00:01
   --------------------------------- ------ 2.0/2.2 MB 1.2 MB/s eta 0:00:01
   ---------------------------------- ----- 2.1/2.2 MB 1.2 MB/s eta 0:00:01
   ---------------------------------- ----- 2.1/2.2 MB 1.2 MB/s eta 0:00:01
   ---------------------------------------  2.2/2.2 MB 1.2 MB/s eta 0:00:01
   ---------------------------------------- 2.2/2.2 MB 1.2 MB/s eta 0:00:00
Downloading kiwisolver-1.4.5-cp311-cp311-win_amd64.whl (56 kB)
   ---------------------------------------- 0.0/56.1 kB ? eta -:--:--
   -------------------- ------------------- 30.7/56.1 kB 1.3 MB/s eta 0:00:01
   ---------------------------- ---------- 41.0/56.1 kB 991.0 kB/s eta 0:00:01
   ---------------------------------------- 56.1/56.1 kB 588.2 kB/s eta 0:00:00
Downloading pillow-10.2.0-cp311-cp311-win_amd64.whl (2.6 MB)
   ---------------------------------------- 0.0/2.6 MB ? eta -:--:--
```

```
   ---------------------------------------- 0.0/2.6 MB 1.4 MB/s eta 0:00:02
    --------------------------------------- 0.1/2.6 MB 825.8 kB/s eta 0:00:04
   - -------------------------------------- 0.1/2.6 MB 751.6 kB/s eta 0:00:04
   -- ------------------------------------- 0.1/2.6 MB 774.0 kB/s eta 0:00:04
   -- ------------------------------------- 0.2/2.6 MB 807.1 kB/s eta 0:00:04
   --- ------------------------------------ 0.2/2.6 MB 811.5 kB/s eta 0:00:03
   --- ------------------------------------ 0.3/2.6 MB 827.5 kB/s eta 0:00:03
   ---- ----------------------------------- 0.3/2.6 MB 863.3 kB/s eta 0:00:03
   ----- ---------------------------------- 0.4/2.6 MB 857.5 kB/s eta 0:00:03
   ------ --------------------------------- 0.4/2.6 MB 859.0 kB/s eta 0:00:03
   ------ --------------------------------- 0.5/2.6 MB 880.6 kB/s eta 0:00:03
   ------- -------------------------------- 0.5/2.6 MB 874.9 kB/s eta 0:00:03
   -------- ------------------------------- 0.5/2.6 MB 879.2 kB/s eta 0:00:03
   -------- ------------------------------- 0.6/2.6 MB 884.7 kB/s eta 0:00:03
   --------- ------------------------------ 0.6/2.6 MB 879.4 kB/s eta 0:00:03
   --------- ------------------------------ 0.6/2.6 MB 865.0 kB/s eta 0:00:03
   ---------- ----------------------------- 0.7/2.6 MB 852.2 kB/s eta 0:00:03
   ---------- ----------------------------- 0.7/2.6 MB 857.5 kB/s eta 0:00:03
   ----------- ---------------------------- 0.7/2.6 MB 830.9 kB/s eta 0:00:03
   ----------- ---------------------------- 0.8/2.6 MB 833.0 kB/s eta 0:00:03
   ------------ --------------------------- 0.8/2.6 MB 824.3 kB/s eta 0:00:03
   ------------ --------------------------- 0.8/2.6 MB 817.2 kB/s eta 0:00:03
   ------------- -------------------------- 0.9/2.6 MB 809.9 kB/s eta 0:00:03
   ------------- -------------------------- 0.9/2.6 MB 803.3 kB/s eta 0:00:03
   -------------- ------------------------- 1.0/2.6 MB 793.7 kB/s eta 0:00:03
   -------------- ------------------------- 1.0/2.6 MB 798.0 kB/s eta 0:00:03
   -------------- ------------------------- 1.0/2.6 MB 799.4 kB/s eta 0:00:02
   --------------- ------------------------ 1.1/2.6 MB 803.9 kB/s eta 0:00:02
   ---------------- ----------------------- 1.1/2.6 MB 813.8 kB/s eta 0:00:02
   ---------------- ----------------------- 1.1/2.6 MB 817.7 kB/s eta 0:00:02
   ----------------- ---------------------- 1.2/2.6 MB 817.5 kB/s eta 0:00:02
   ----------------- ---------------------- 1.2/2.6 MB 819.2 kB/s eta 0:00:02
   ------------------ --------------------- 1.3/2.6 MB 827.4 kB/s eta 0:00:02
   ------------------ --------------------- 1.3/2.6 MB 835.1 kB/s eta 0:00:02
   ------------------- -------------------- 1.4/2.6 MB 841.0 kB/s eta 0:00:02
   -------------------- ------------------- 1.5/2.6 MB 847.8 kB/s eta 0:00:02
   -------------------- ------------------- 1.5/2.6 MB 860.2 kB/s eta 0:00:02
   --------------------- ------------------ 1.6/2.6 MB 873.9 kB/s eta 0:00:02
   --------------------- ------------------ 1.6/2.6 MB 885.1 kB/s eta 0:00:02
   ---------------------- ----------------- 1.7/2.6 MB 894.0 kB/s eta 0:00:02
   ---------------------- ----------------- 1.8/2.6 MB 898.7 kB/s eta 0:00:01
   ----------------------- ---------------- 1.8/2.6 MB 913.5 kB/s eta 0:00:01
   ----------------------- ---------------- 1.9/2.6 MB 922.7 kB/s eta 0:00:01
   ------------------------ --------------- 1.9/2.6 MB 925.5 kB/s eta 0:00:01
   ------------------------ --------------- 2.0/2.6 MB 934.9 kB/s eta 0:00:01
   ------------------------- -------------- 2.1/2.6 MB 936.4 kB/s eta 0:00:01
   ------------------------- -------------- 2.1/2.6 MB 939.7 kB/s eta 0:00:01
   -------------------------- ------------- 2.2/2.6 MB 942.7 kB/s eta 0:00:01
   -------------------------- ------------- 2.2/2.6 MB 952.2 kB/s eta 0:00:01
   --------------------------- ------------ 2.3/2.6 MB 944.2 kB/s eta 0:00:01
   --------------------------- ------------ 2.3/2.6 MB 955.5 kB/s eta 0:00:01
   ---------------------------- ----------- 2.4/2.6 MB 958.1 kB/s eta 0:00:01
   ---------------------------- ----------- 2.4/2.6 MB 960.9 kB/s eta 0:00:01
   ----------------------------- ---------- 2.5/2.6 MB 971.2 kB/s eta 0:00:01
   ----------------------------- ----------  2.6/2.6 MB 971.3 kB/s eta 0:00:01
   ----------------------------- ----------  2.6/2.6 MB 979.3 kB/s eta 0:00:01
   ---------------------------------------- 2.6/2.6 MB 977.2 kB/s eta 0:00:00
Downloading pyparsing-3.1.1-py3-none-any.whl (103 kB)
   ---------------------------------------- 0.0/103.1 kB ? eta -:--:--
   -------------- ------------------------ 41.0/103.1 kB 1.9 MB/s eta 0:00:01
```

```
   ---------------------------------------  102.4/103.1 kB 1.5 MB/s eta 0:00:01
   ---------------------------------------- 103.1/103.1 kB 988.3 kB/s eta 0:00:00
Installing collected packages: pyparsing, pillow, kiwisolver, fonttools, cycler,
contourpy, matplotlib
Successfully installed contourpy-1.2.0 cycler-0.12.1 fonttools-4.48.1 kiwisolver-
1.4.5 matplotlib-3.8.2 pillow-10.2.0 pyparsing-3.1.1
```

Import pandas

In [2]:
```
import pandas as pd
```

Обрахунок номеру варіанта:

In [1]:
```
n = 2
print((n + 4) % 5 + 1)
```

```
2
```

Далі було завантажено набір даних про дитячі імена в США за посиланням

Читаємо CSV-файл у DataFrame:

In [3]:
```
df = pd.read_csv('NationalNames.csv')
```

Подивимося на загальний вигляд таблиці. Бачимо, що в ньому вказана популярність імен в кожен рік з 1880 по 2014.

In [4]:
```
df
```

Out[4]:

	Id	Name	Year	Gender	Count
0	1	Mary	1880	F	7065
1	2	Anna	1880	F	2604
2	3	Emma	1880	F	2003
3	4	Elizabeth	1880	F	1939
4	5	Minnie	1880	F	1746
...	...	...	...	...	...
1825428	1825429	Zykeem	2014	M	5
1825429	1825430	Zymeer	2014	M	5
1825430	1825431	Zymiere	2014	M	5
1825431	1825432	Zyran	2014	M	5
1825432	1825433	Zyrin	2014	M	5

1825433 rows × 5 columns

Вправа 3

Отримайте імена стовпців набору даних.

In [18]:
```
df.columns
```

Out[18]:
```
Index(['Id', 'Name', 'Year', 'Gender', 'Count'], dtype='object')
```

Вправа 4

Отримайте загальну інформацію про дані у наборі даних.

In [14]:
```
df.describe()
```

Out[14]:

	Id	Year	Count
count	1.825433e+06	1.825433e+06	1.825433e+06
mean	9.127170e+05	1.972620e+03	1.846879e+02
std	5.269573e+05	3.352891e+01	1.566711e+03
min	1.000000e+00	1.880000e+03	5.000000e+00
25%	4.563590e+05	1.949000e+03	7.000000e+00
50%	9.127170e+05	1.982000e+03	1.200000e+01
75%	1.369075e+06	2.001000e+03	3.200000e+01
max	1.825433e+06	2.014000e+03	9.968000e+04

Вправа 5

Знайдіть кількість унікальних імен у наборі даних

In [29]:
```
df['Name'].nunique()
```

Out[29]:
```
93889
```

Вправа 8

Знайдіть найпопулярніше ім’я за результатами одного року (ім’я, для якого Count максимальне)

In [80]:
```
def task_8(year):
    # print selected part of dataframe
    df_for_year = df[df.Year == year]
    print(df_for_year)

    row_index = df_for_year['Count'].idxmax() # .idxmax() returns label index of
    row = df.loc[[row_index]]
    name = df.at[row_index, 'Name']
    print("\n\nLabel-based index of row: " + str(row_index))
    print("\nRow: \n" + str(row))
#    print("\nThe name: \'" + str(name) + "\' in ")
    print(f"\nThe name is '{name}' in {year}")

task_8(1947)
```

```
            Id      Name  Year Gender  Count
431052  431053     Linda  1947      F  99680
431053  431054      Mary  1947      F  71684
431054  431055  Patricia  1947      F  51273
431055  431056   Barbara  1947      F  48793
431056  431057    Sandra  1947      F  34770
...        ...       ...   ...    ...    ...
441419  441420  Zacarias  1947      M      5
441420  441421       Zan  1947      M      5
441421  441422    Zannie  1947      M      5
441422  441423    Zebbie  1947      M      5
441423  441424      Zell  1947      M      5

[10372 rows x 5 columns]


Label-based index of row: 431052

Row:
            Id   Name  Year Gender  Count
431052  431053  Linda  1947      F  99680

The name is 'Linda' in 1947
```

Вправа 9

Підрахуйте кількість записів, для яких Count - мінімальне у наборі.

In [92]:
```
def task_9():
    count_column = df['Count']
    count_min = count_column.min()
    result = count_column[count_column == count_min].size
    print(result)

task_9()
```

```
254615
```

Вправа 11

Знайдіть рік із найбільшою кількістю унікальних імен.

In [116…
```
def get_names_for_year(year):
    return df[df.Year == year]['Name']

def task_11():
    years = df['Year'].unique() # get array of years

    # create dataframe with two columns: {Year}, {Number of unique names in year
    df_unique_names = pd.DataFrame(data = [{'Year': year, 'Name': get_names_for_
                                   index = years)

    row_index = df_unique_names['Name'].idxmax()
    print(df_unique_names.loc[[row_index]])


task_11()
```

```
      Year   Name
2008  2008  32488
```

Вправа 12

Знайдіть найпопулярніше ім'я в році з найбільшою кількістю унікальних імен (тобто у 2008 році)

In [118…
```
def task_12(year):
    row_index = df[df.Year == year]['Count'].idxmax()
    name = df.at[row_index, 'Name']
    print(f"'{name}'")

task_12(2008)
```

```
'Jacob'
```

Вправа 13

Знайдіть рік, коли ім'я "Jacob" було найпопулярнішим серед жіночих імен

Зважаючи на очікуваний результат (у методичці), умову можна переформулювати так: "Знайдіть рік, коли жіноче ім'я "Jacob" мало максимальну популярність".

In [132…
```
def task_13(name, gender):
    row_index = df[(df.Name == name) & (df.Gender == gender)]['Count'].idxmax()
    print(df.loc[[row_index]])

task_13('Jacob', 'F')
```

```
             Id    Name  Year Gender  Count
1455556  1455557  Jacob  2004      F    171
```

Вправа 14

Знайти рік із найбільшою кількістю гендерно нейтральних імен (однакові чоловічі та жіночі імена)

In [157…
```
import numpy as np

def get_all_gender_unique_names(gender):
    return df[df.Gender == gender]['Name'].unique() # returns NumPy array

def get_unique_names_for_year(year):
    return df[df.Year == year]['Name'].unique()

def get_gender_neutral_names_for_year(year):
    male_names = df[(df.Year == year) & (df.Gender == 'M')]['Name'].to_numpy()
    female_names = df[(df.Year == year) & (df.Gender == 'F')]['Name'].to_numpy()
    return np.intersect1d(male_names, female_names)


def task_14():
    years = df['Year'].unique() # get array with years

    # create dataframe with two columns: {year}, {number of gender neutral names
```

```
    df_gender_neutral_names = pd.DataFrame(data = [{'Year': year,
                                                    'Gender_neutral_names': get_
                                    index = years)

    row_index = df_gender_neutral_names['Gender_neutral_names'].idxmax()

    print(df_gender_neutral_names.loc[[row_index]])


task_14()
```

```
      Year  Gender_neutral_names
2008  2008                  2557
```

Вправа 16

Знайдіть рік, коли народилося найбільше дітей

In [161…

```
def get_number_of_people_for_year(year):
    return df[df.Year == year]['Count'].sum()

def task_16():
    years = df['Year'].unique() # get array with years

    # create dataframe with two columns: {year}, {number of people born in this
    df_count = pd.DataFrame(data = [{'Year': year, 'Count': get_number_of_people
                                    index = years)

    row_index = df_count['Count'].idxmax()

#    print(df_count.loc[[row_index]])
    print(df_count.at[row_index, 'Year'])

task_16()
```

```
1957
```

Вправа 17

Знайдіть кількість дівчаток та хлопчиків, які народились кожного року

In [4]:

```
def get_number_of_gender_for_year(year, gender):
    return df[(df.Year == year) & (df.Gender == gender)]['Count'].sum()

def task_17():
    years = df['Year'].unique() # get array with years

    # create dataframe with three columns: {year}, {number of Female}, {number o
    df_genders = pd.DataFrame(data = [{'Year': year, 'F': get_number_of_gender_f
                                      'M': get_number_of_gender_for_year(year, 'M
                                    index = years)

    return df_genders

df_genders = task_17()
df_genders
```

Out[4]:

	Year	F	M
1880	1880	90993	110491
1881	1881	91954	100745
1882	1882	107850	113688
1883	1883	112321	104629
1884	1884	129022	114445
...	...	...	...
2010	2010	1772738	1913851
2011	2011	1753500	1893230
2012	2012	1753922	1889414
2013	2013	1745339	1881463
2014	2014	1768775	1901376

135 rows × 3 columns

Вправа 18

Підрахуйте кількість років, коли дівчаток народжувалось більше, ніж хлопчиків.

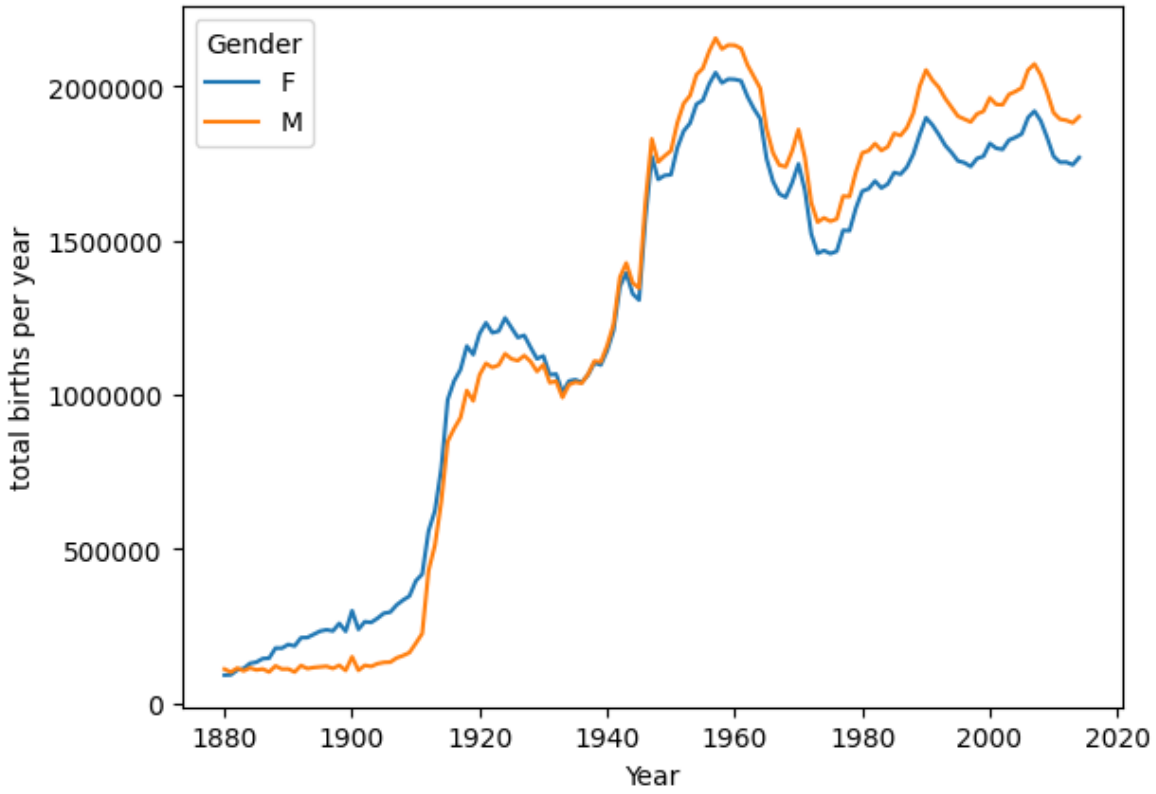
In [169...
```
df_female_more_male = df_genders[df_genders.F > df_genders.M]
len(df_female_more_male.index)
```

Out[169... 54

Вправа 19

Накресліть графік загальної кількості народжень хлопчиків та дівчаток на рік.

In [16]:
```
# for successful work, task_17 should be executed first

import matplotlib.pyplot as plt
import matplotlib.ticker as ticker

g = df_genders.plot(y=['F', 'M'], xlabel='Year', ylabel='total births per year')

# add legent title
g.legend(title='Gender')

# format ticks on Y axis
formatter = ticker.ScalarFormatter()
formatter.set_scientific(False)
g.yaxis.set_major_formatter(formatter)

plt.show()
```



Вправа 20

Підрахуйте кількість гендерно-нейтральних імен (однакових для дівчат та хлопців)

In [21]:
```
import numpy as np

def get_gender_names(gender):
    return df[df.Gender == gender]['Name'].unique()

def task_20():
    male_names = get_gender_names('M')
    female_names = get_gender_names('F')
    return np.intersect1d(male_names, female_names).size

task_20()
```

Out[21]: 10221

Вправа 22

Підрахуйте скільки років проводилось спостереження

In [22]:
```
def task_22():
    n_years = df['Year'].nunique()
    print(f"'Спостереження проводилось {n_years} років'")

task_22()
```

'Спостереження проводилось 135 років'

Вправа 23

Знати найпопулярніші гендерно-нейтральні імена (ті, що присутні кожного року)

In [47]:
```
def get_gender_names(gender):
    return df[df.Gender == gender]['Name'].unique()

def get_gender_neutral_names():
    male_names = get_gender_names('M')
    female_names = get_gender_names('F')
    return np.intersect1d(male_names, female_names)

def task_23():
    neutral_names = get_gender_neutral_names()
    number_of_years = df['Year'].nunique()

    # dataframe for neutral names
    df_neutral_names = df[df.Name.isin(neutral_names)]

    # dataframe: for each name it is number of occurrences (years) wher it is pr
    # (Note that if the same name is present in same year twice (because of M an
    df_neutral_names_count_years = df_neutral_names.groupby("Name")['Year'].coun

    # Get names that we need by task. Name must be occured twice (for M and F ge
    result_names = df_neutral_names_count_years[df_neutral_names_count_years ==

    # calculate number of overall count for each result name
    df_result_names = df[df.Name.isin(result_names)]
    result_names_counts = df_result_names.groupby("Name")['Count'].sum()

    print(result_names_counts.sort_values(ascending=False))


task_23()
```

```
Name
James      5129096
John       5106590
Robert     4816785
William    4071368
Joseph     2580687
Jean        480901
Jesse       421406
Leslie      376587
Francis     312147
Lee         291691
Jessie      274931
Marion      259549
Johnnie     149953
Sidney      105185
Ollie        56482
Tommie       51315
Name: Count, dtype: int64
```

Вправа 24

Знайти найпопулярніше серед непопулярних імен (непопулярне ім'я, яким називали дітей найбільшу кількість разі

Знайти ім'я, що мають найбільший Count серед імен, що використані у найменшій кількості років)

In [70]:
```
def task_24():
    # create series with number of years specific name was used for each name
    df_names_years_count = df.groupby("Name")['Year'].count()
    min_year_count = df_names_years_count.min()

    # list of names that used in min number of years
    unpopular_names = df_names_years_count[df_names_years_count == min_year_coun

    # get Count for each unpopular name
    df_unpopular_names = df[df.Name.isin(unpopular_names)]
    popular_unpopular_names = df_unpopular_names.groupby("Name")['Count'].sum()

    # get the most popular from unpopular names
    max_count = popular_unpopular_names.max()
    most_popular_unpopular_names = popular_unpopular_names[popular_unpopular_nam

    name = most_popular_unpopular_names.index[0]
    count = most_popular_unpopular_names.values[0]
    print(f"The most popular name from unpopular ones is '{name}'. It were used

task_24()
```

```
The most popular name from unpopular ones is 'Christop'. It were used 1082 times
```

Вправа 27

Знайти найпопулярніші імена в кожному році.

In [87]:
```
def find_popular_name_in_year(year):
    df_year = df[df.Year == year]
    index_max = df_year['Count'].idxmax()
    return [df_year.at[index_max, 'Name'], df_year.at[index_max, 'Count']]

def task_27():
    years = df['Year'].unique() # get array with years

    # create dataframe with three columns: {year}, {the most popular name in thi
    df_popular_names_in_year = pd.DataFrame(data = [{'Year': year, 'Name': find_
                                        'Count': find_popular_name_in_year(year)[1]
                                    index = years)

    return df_popular_names_in_year

task_27()
```

Out[87]:

	Year	Name	Count
1880	1880	John	9655
1881	1881	John	8769
1882	1882	John	9557
1883	1883	John	8894
1884	1884	John	9388
...	...	...	...
2010	2010	Isabella	22883
2011	2011	Sophia	21816
2012	2012	Sophia	22267
2013	2013	Sophia	21147
2014	2014	Emma	20799

135 rows × 3 columns