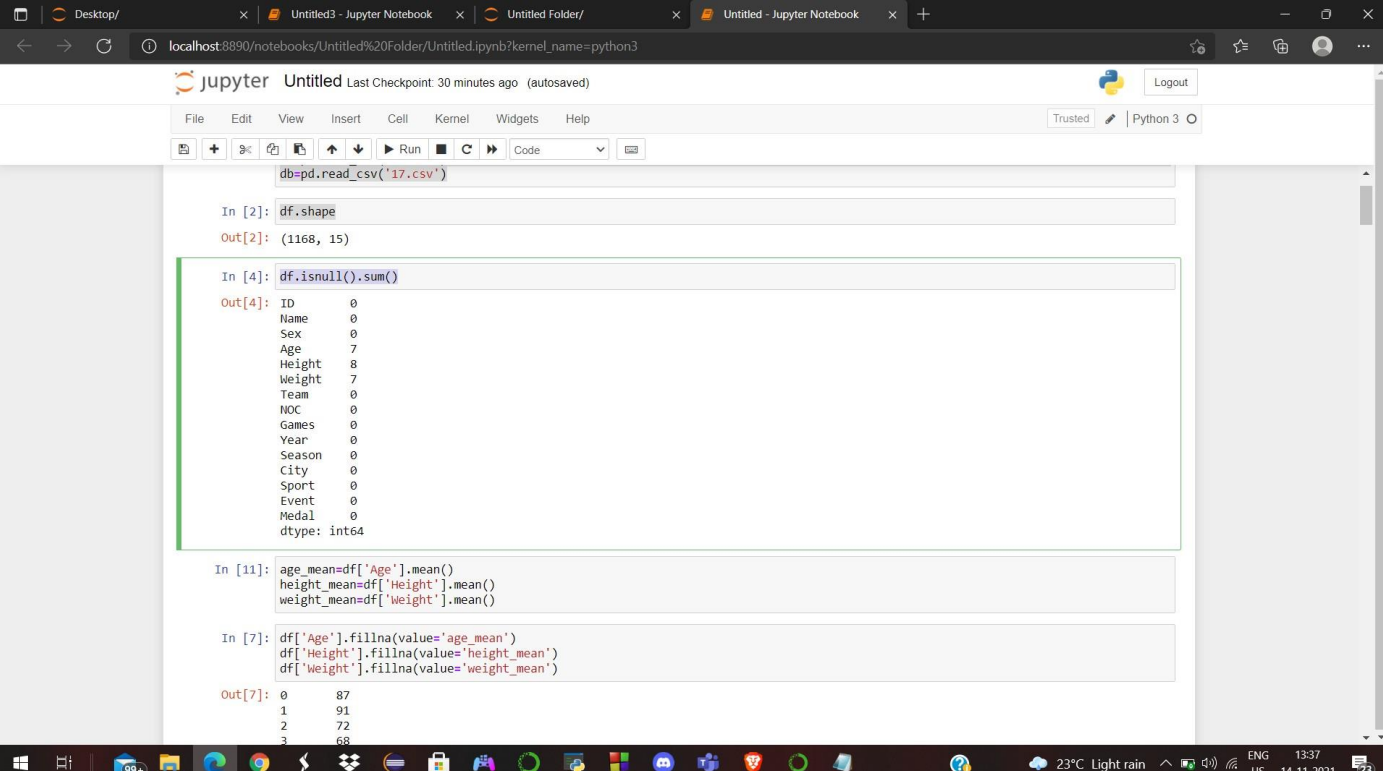


NAME: ARUN  
KUMAR RATH  
SRN: PES1UG20CS076  
CLASS: B SECTION

GIVEN DATA SET: 14.CSV

# INTRO

- import pandas as pd
- df=pd.read\_csv('14.csv')
- da=pd.read\_csv('14.csv')
- db=pd.read\_csv('14.csv')
- df.isnull().sum()



The screenshot shows a Jupyter Notebook running in a web browser. The notebook has several tabs open, including 'Desktop/', 'Untitled3 - Jupyter Notebook', 'Untitled Folder/', and 'Untitled - Jupyter Notebook'. The active notebook is titled 'Untitled' and shows the following code and output:

```
db=pd.read_csv('17.csv')
```

```
In [2]: df.shape
Out[2]: (1168, 15)
```

```
In [4]: df.isnull().sum()
Out[4]: ID      0
        Name    0
        Sex     0
        Age     7
        Height  8
        Weight  7
        Team    0
        NOC     0
        Games   0
        Year    0
        Season  0
        City    0
        Sport   0
        Event   0
        Medal   0
        dtype: int64
```

```
In [11]: age_mean=df['Age'].mean()
         height_mean=df['Height'].mean()
         weight_mean=df['Weight'].mean()
```

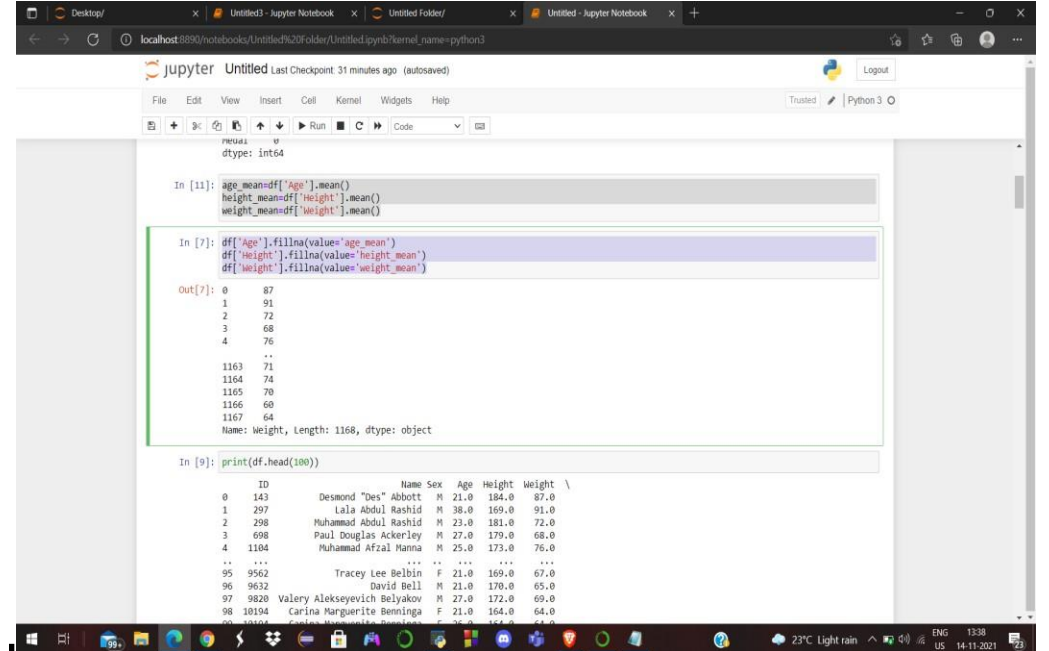
```
In [7]: df['Age'].fillna(value='age_mean')
         df['Height'].fillna(value='height_mean')
         df['Weight'].fillna(value='weight_mean')
```

```
Out[7]: 0      87
        1      91
        2      72
        3      68
```

The bottom of the screenshot shows a Windows taskbar with various icons and a system tray displaying '23°C Light rain', 'ENG US', and the date '14-11-2021'.

# TO FIND THE MEAN

- `age_mean=df['Age'].mean()`
- `height_mean=df['Height'].mean()`
- `weight_mean=df['Weight'].mean()`
- `df['Age'].fillna(value='age_mean')`
- `df['Height'].fillna(value='height_mean')`
- `df['Weight'].fillna(value='weight_mean')`



```
age_mean=df['Age'].mean()
height_mean=df['Height'].mean()
weight_mean=df['Weight'].mean()

df['Age'].fillna(value='age_mean')
df['Height'].fillna(value='height_mean')
df['Weight'].fillna(value='weight_mean')

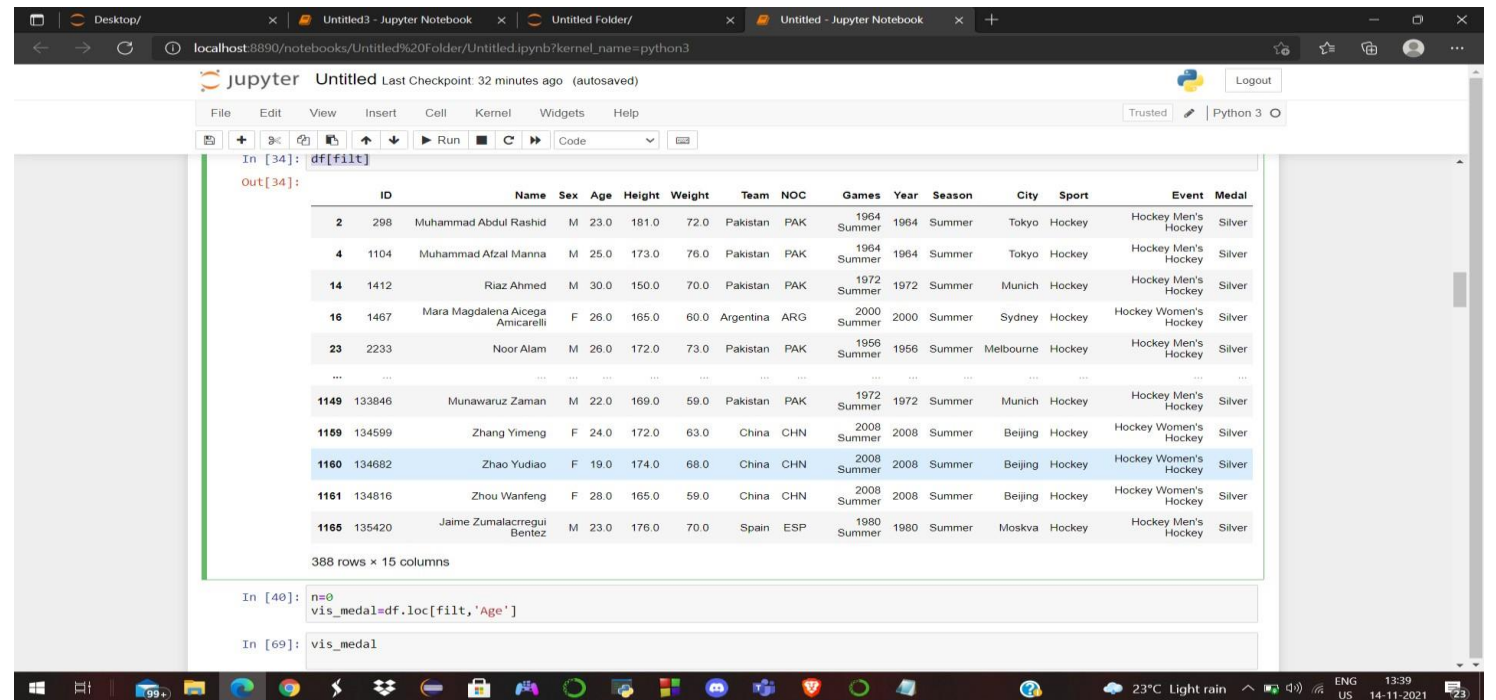
In [7]:
Out[7]:
0      87
1      91
2      72
3      68
4      76
...
1163    71
1164    74
1165    70
1166    60
1167    64
Name: Weight, Length: 1168, dtype: object

In [9]: print(df.head(100))

   ID  Name Sex  Age  Height  Weight \
0  143  Desmond "Des" Abbott  M   21.0   184.0   87.0
1  297    Lala Abdul Rashid  M   38.0   169.0   91.0
2  298  Muhammad Abdul Rashid  M   23.0   181.0   72.0
3  698    Paul Douglas Ackerley  M   27.0   179.0   68.0
4  1184  Muhammad Afzal Hanna  M   25.0   173.0   76.0
...  ...
95  9562  Tracey Lee Belbin  F   21.0   169.0   67.0
96  9632    David Bell  M   21.0   170.0   65.0
97  9820  Valery Alekseyevich Belyakov  M   27.0   172.0   69.0
98  10194  Carina Harguerite Bemminga  F   21.0   164.0   64.0
99  10194  Carina Harguerite Bemminga  F   21.0   164.0   64.0
```

# TO FILTER OUR THE SILVER MEDALS

- `print(df['Age'].isnull().tail(1000))`
- `filt=(df['Medal']=='Silver')`
- `df[filt]`



The screenshot shows a Jupyter Notebook interface with the following code and output:

```
In [34]: df[filt]
```

Out[34]:

ID	Name	Sex	Age	Height	Weight	Team	NOC	Games	Year	Season	City	Sport	Event	Medal
2	Muhammad Abdul Rashid	M	23.0	181.0	72.0	Pakistan	PAK	1964 Summer	1964	Summer	Tokyo	Hockey	Hockey Men's Hockey	Silver
4	Muhammad Afzal Manna	M	25.0	173.0	76.0	Pakistan	PAK	1964 Summer	1964	Summer	Tokyo	Hockey	Hockey Men's Hockey	Silver
14	Riaz Ahmed	M	30.0	150.0	70.0	Pakistan	PAK	1972 Summer	1972	Summer	Munich	Hockey	Hockey Men's Hockey	Silver
16	Mara Magdalena Aicega Amicarelli	F	26.0	165.0	60.0	Argentina	ARG	2000 Summer	2000	Summer	Sydney	Hockey	Hockey Women's Hockey	Silver
23	Noor Alam	M	26.0	172.0	73.0	Pakistan	PAK	1956 Summer	1956	Summer	Melbourne	Hockey	Hockey Men's Hockey	Silver
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
1149	Munawaruz Zaman	M	22.0	169.0	59.0	Pakistan	PAK	1972 Summer	1972	Summer	Munich	Hockey	Hockey Men's Hockey	Silver
1159	Zhang Yimeng	F	24.0	172.0	63.0	China	CHN	2008 Summer	2008	Summer	Beijing	Hockey	Hockey Women's Hockey	Silver
1160	Zhao Yudiao	F	19.0	174.0	68.0	China	CHN	2008 Summer	2008	Summer	Beijing	Hockey	Hockey Women's Hockey	Silver
1161	Zhou Wanfeng	F	28.0	165.0	59.0	China	CHN	2008 Summer	2008	Summer	Beijing	Hockey	Hockey Women's Hockey	Silver
1165	Jaime Zumalacregui Bentez	M	23.0	176.0	70.0	Spain	ESP	1980 Summer	1980	Summer	Moskva	Hockey	Hockey Men's Hockey	Silver

388 rows x 15 columns

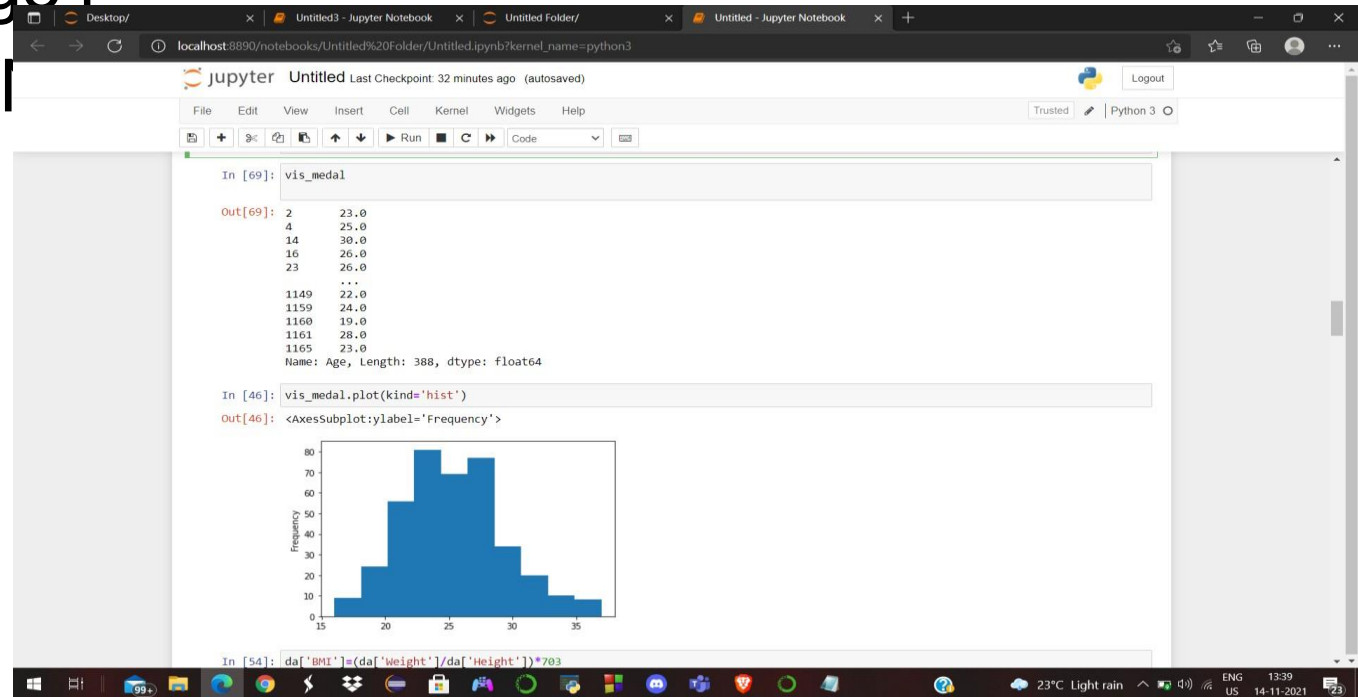
```
In [40]: n=0  
vis_medal=df.loc[filt,'Age']
```

```
In [69]: vis_medal
```

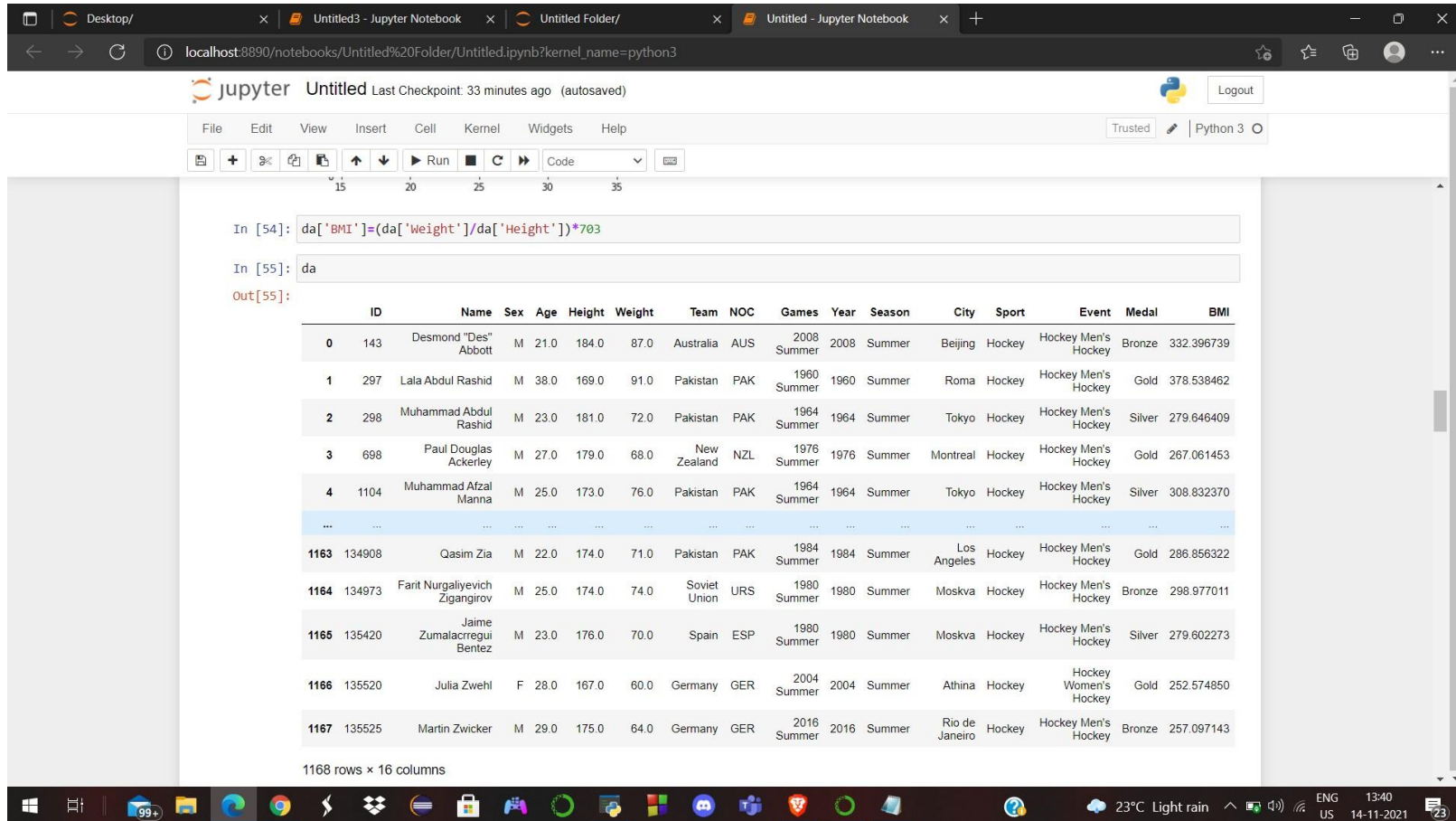
# CONTINUED...

- $n=0$
- `vis_medal=df.loc[filt,'Age']`
- DATA VISUALIZATION

▪  
▪



# CONTINUED....



The screenshot displays a Jupyter Notebook interface with a dark theme. The browser address bar shows the URL `localhost:8890/notebooks/Untitled%20Folder/Untitled.ipynb?kernel_name=python3`. The notebook title is "Untitled" and it indicates the last checkpoint was 33 minutes ago (autosaved). The menu bar includes File, Edit, View, Insert, Cell, Kernel, Widgets, and Help. The toolbar shows various icons for file operations and execution. The code cell contains the following Python code:

```
In [54]: da['BMI']=(da['Weight']/da['Height'])*703
```

The output cell shows the variable `da` and a preview of the data table:

```
In [55]: da
```

**Out[55]:**

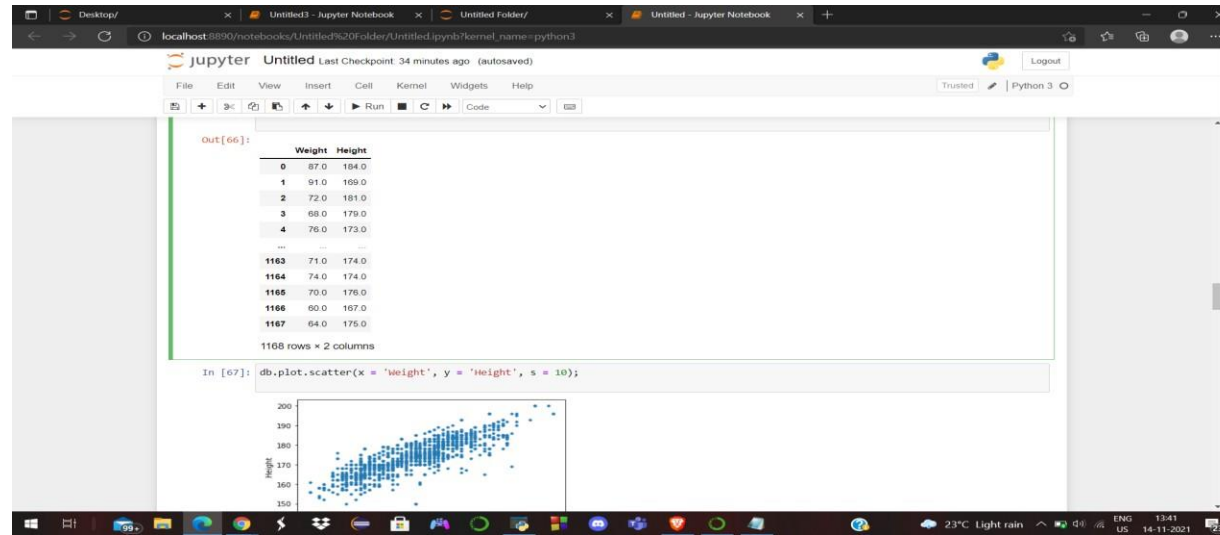
	ID	Name	Sex	Age	Height	Weight	Team	NOC	Games	Year	Season	City	Sport	Event	Medal	BMI
0	143	Desmond "Des" Abbott	M	21.0	184.0	87.0	Australia	AUS	2008 Summer	2008	Summer	Beijing	Hockey	Hockey Men's Hockey	Bronze	332.396739
1	297	Lala Abdul Rashid	M	38.0	169.0	91.0	Pakistan	PAK	1960 Summer	1960	Summer	Roma	Hockey	Hockey Men's Hockey	Gold	378.538462
2	298	Muhammad Abdul Rashid	M	23.0	181.0	72.0	Pakistan	PAK	1964 Summer	1964	Summer	Tokyo	Hockey	Hockey Men's Hockey	Silver	279.646409
3	698	Paul Douglas Ackerley	M	27.0	179.0	68.0	New Zealand	NZL	1976 Summer	1976	Summer	Montreal	Hockey	Hockey Men's Hockey	Gold	267.061453
4	1104	Muhammad Afzal Manna	M	25.0	173.0	76.0	Pakistan	PAK	1964 Summer	1964	Summer	Tokyo	Hockey	Hockey Men's Hockey	Silver	308.832370
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
1163	134908	Qasim Zia	M	22.0	174.0	71.0	Pakistan	PAK	1984 Summer	1984	Summer	Los Angeles	Hockey	Hockey Men's Hockey	Gold	286.856322
1164	134973	Farit Nurgaliyevich Zigangirov	M	25.0	174.0	74.0	Soviet Union	URS	1980 Summer	1980	Summer	Moskva	Hockey	Hockey Men's Hockey	Bronze	298.977011
1165	135420	Jaime Zumalacregui Bentez	M	23.0	176.0	70.0	Spain	ESP	1980 Summer	1980	Summer	Moskva	Hockey	Hockey Men's Hockey	Silver	279.602273
1166	135520	Julia Zwehl	F	28.0	167.0	60.0	Germany	GER	2004 Summer	2004	Summer	Athina	Hockey	Hockey Women's Hockey	Gold	252.574850
1167	135525	Martin Zwicker	M	29.0	175.0	64.0	Germany	GER	2016 Summer	2016	Summer	Rio de Janeiro	Hockey	Hockey Men's Hockey	Bronze	257.097143

1168 rows x 16 columns

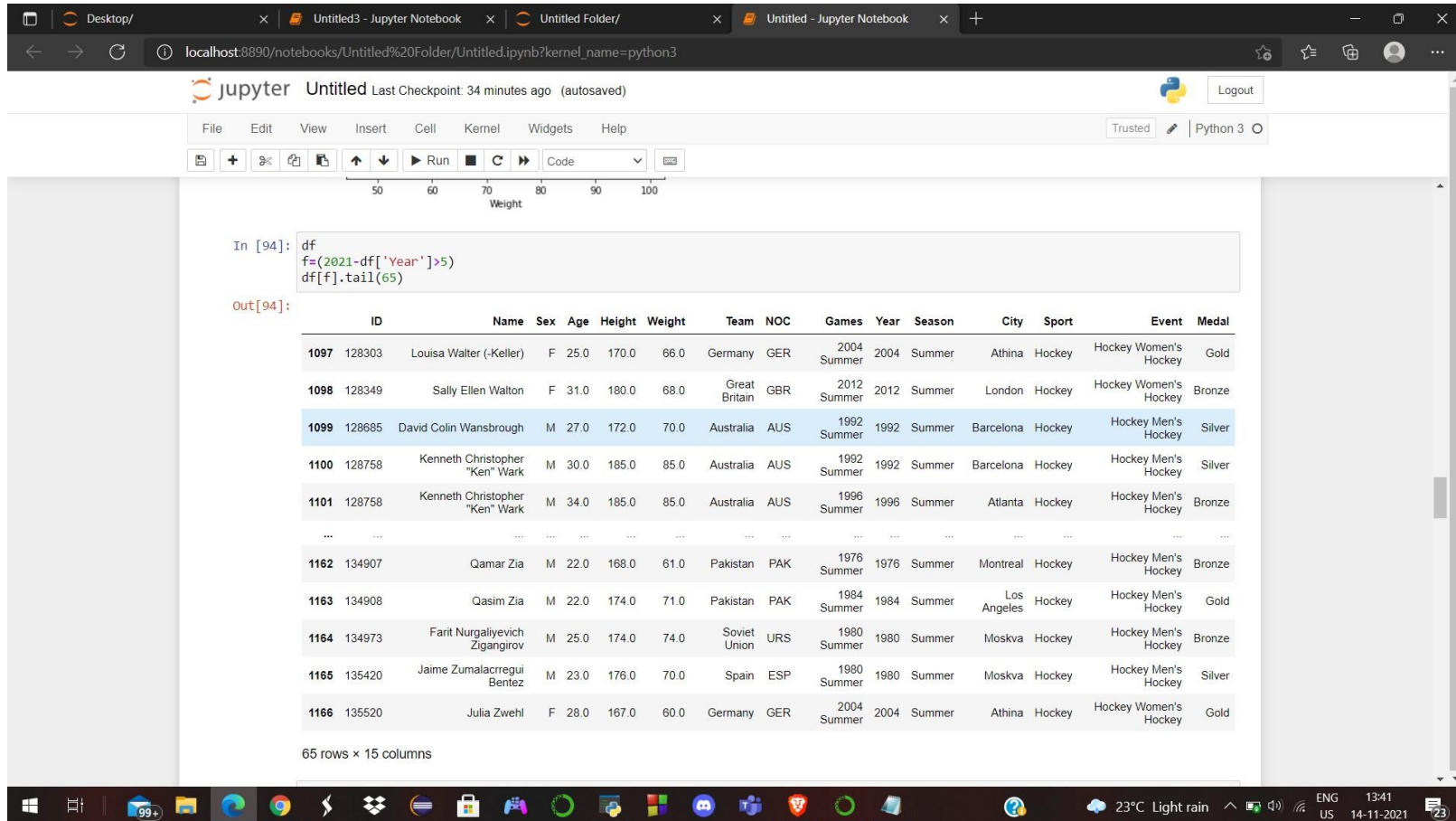
The Windows taskbar at the bottom shows the system clock as 13:40 on 14-11-2021, with weather information for 23°C and light rain.

# TO FILTER OUT THE WEIGHT AND HEIGHT

- `db[['Weight','Height']]`



# TO GET YEAR BELOW 2016



The screenshot shows a Jupyter Notebook interface with a code cell and its output. The code filters a data frame for athletes born after 2021 and displays the last 65 rows. The output is a table with 15 columns: ID, Name, Sex, Age, Height, Weight, Team, NOC, Games, Year, Season, City, Sport, Event, and Medal. The table shows athletes from various countries, including Germany, Great Britain, Australia, Pakistan, Soviet Union, and Spain, competing in Hockey events. The last row shown is for Julia Zwehl, a German female hockey player who won a Gold medal in 2004.

```
In [94]: df
f=(2021-df['Year']>5)
df[f].tail(65)
```

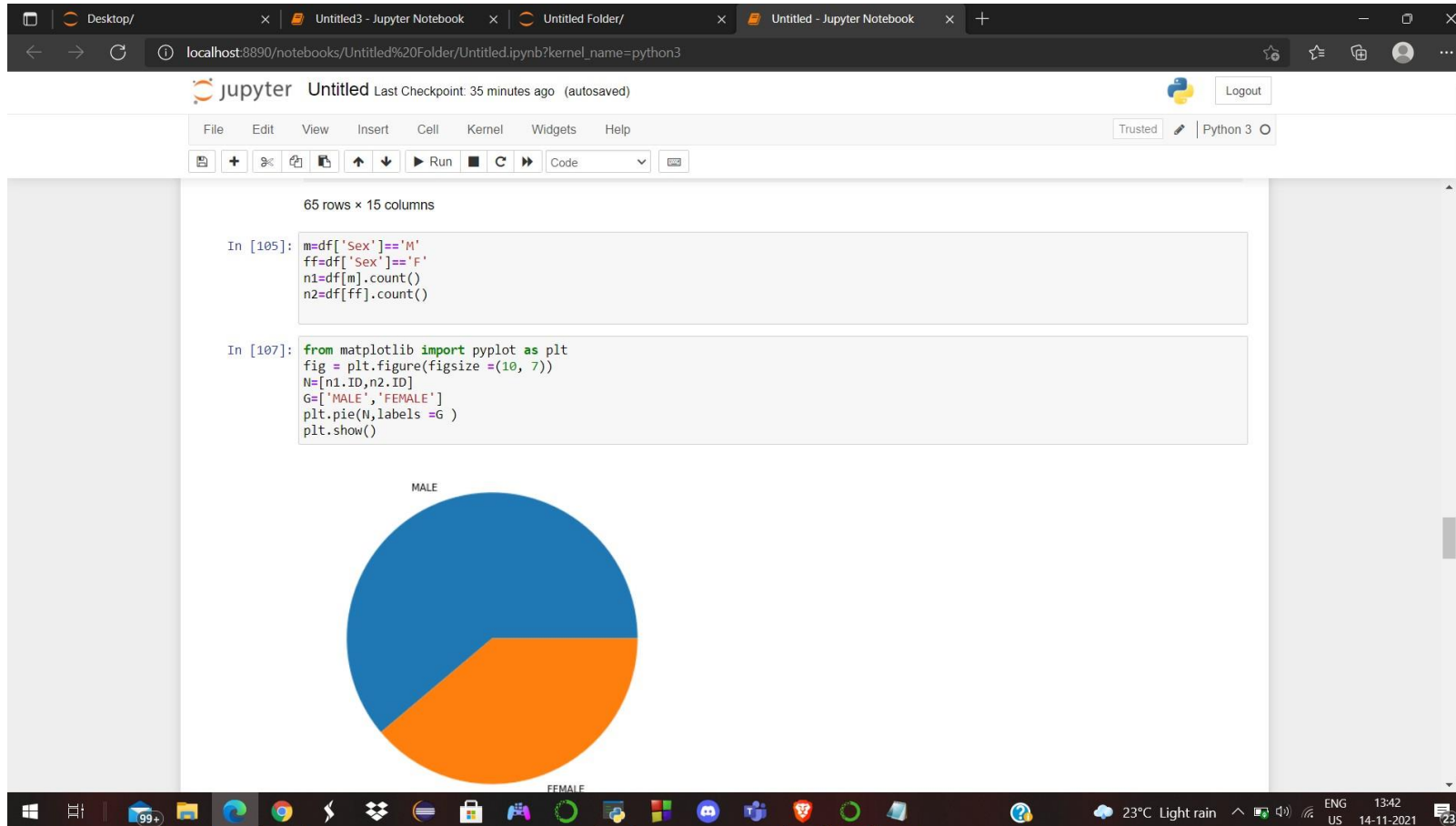
Out[94]:

	ID	Name	Sex	Age	Height	Weight	Team	NOC	Games	Year	Season	City	Sport	Event	Medal
1097	128303	Louisa Walter (-Keller)	F	25.0	170.0	66.0	Germany	GER	2004 Summer	2004	Summer	Athina	Hockey	Hockey Women's Hockey	Gold
1098	128349	Sally Ellen Walton	F	31.0	180.0	68.0	Great Britain	GBR	2012 Summer	2012	Summer	London	Hockey	Hockey Women's Hockey	Bronze
1099	128685	David Colin Wansbrough	M	27.0	172.0	70.0	Australia	AUS	1992 Summer	1992	Summer	Barcelona	Hockey	Hockey Men's Hockey	Silver
1100	128758	Kenneth Christopher "Ken" Wark	M	30.0	185.0	85.0	Australia	AUS	1992 Summer	1992	Summer	Barcelona	Hockey	Hockey Men's Hockey	Silver
1101	128758	Kenneth Christopher "Ken" Wark	M	34.0	185.0	85.0	Australia	AUS	1996 Summer	1996	Summer	Atlanta	Hockey	Hockey Men's Hockey	Bronze
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
1162	134907	Qamar Zia	M	22.0	168.0	61.0	Pakistan	PAK	1976 Summer	1976	Summer	Montreal	Hockey	Hockey Men's Hockey	Bronze
1163	134908	Qasim Zia	M	22.0	174.0	71.0	Pakistan	PAK	1984 Summer	1984	Summer	Los Angeles	Hockey	Hockey Men's Hockey	Gold
1164	134973	Farit Nurgaliyevich Zigangirov	M	25.0	174.0	74.0	Soviet Union	URS	1980 Summer	1980	Summer	Moskva	Hockey	Hockey Men's Hockey	Bronze
1165	135420	Jaime Zumalacregui Bentez	M	23.0	176.0	70.0	Spain	ESP	1980 Summer	1980	Summer	Moskva	Hockey	Hockey Men's Hockey	Silver
1166	135520	Julia Zwehl	F	28.0	167.0	60.0	Germany	GER	2004 Summer	2004	Summer	Athina	Hockey	Hockey Women's Hockey	Gold

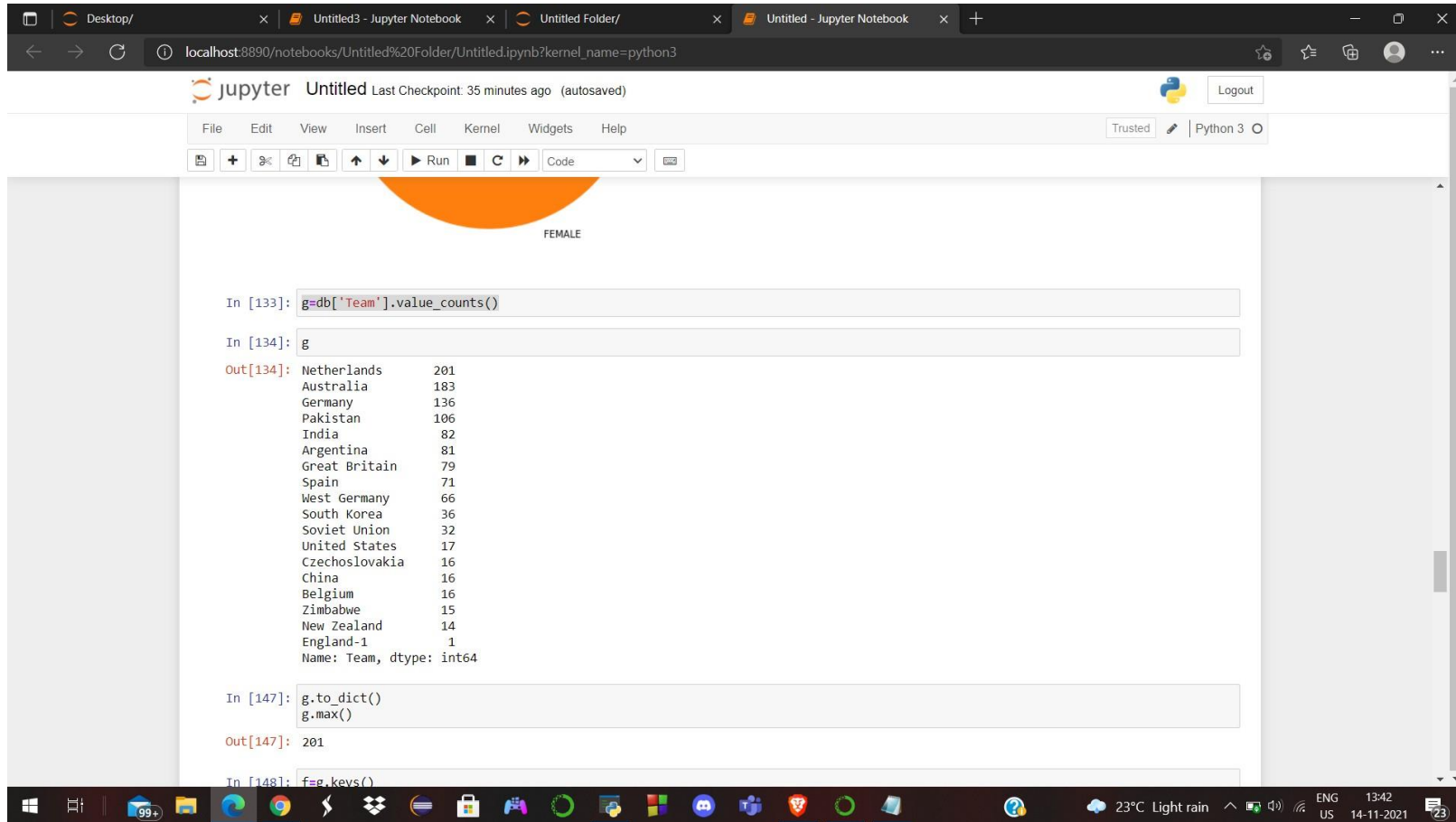
65 rows x 15 columns



# DATA VISUALISATION



# FREQUENCY OF THE TOTAL MEDAL



Desktop/Untitled3 - Jupyter NotebookUntitled Folder/Untitled - Jupyter Notebook

localhost:8890/notebooks/Untitled%20Folder/Untitled.ipynb?kernel\_name=python3

jupyter UntitledrasiC:checkpoint minutes ago (autosaved)

FileEditViewInsertCellKernelWidgetsHelp

RunCode

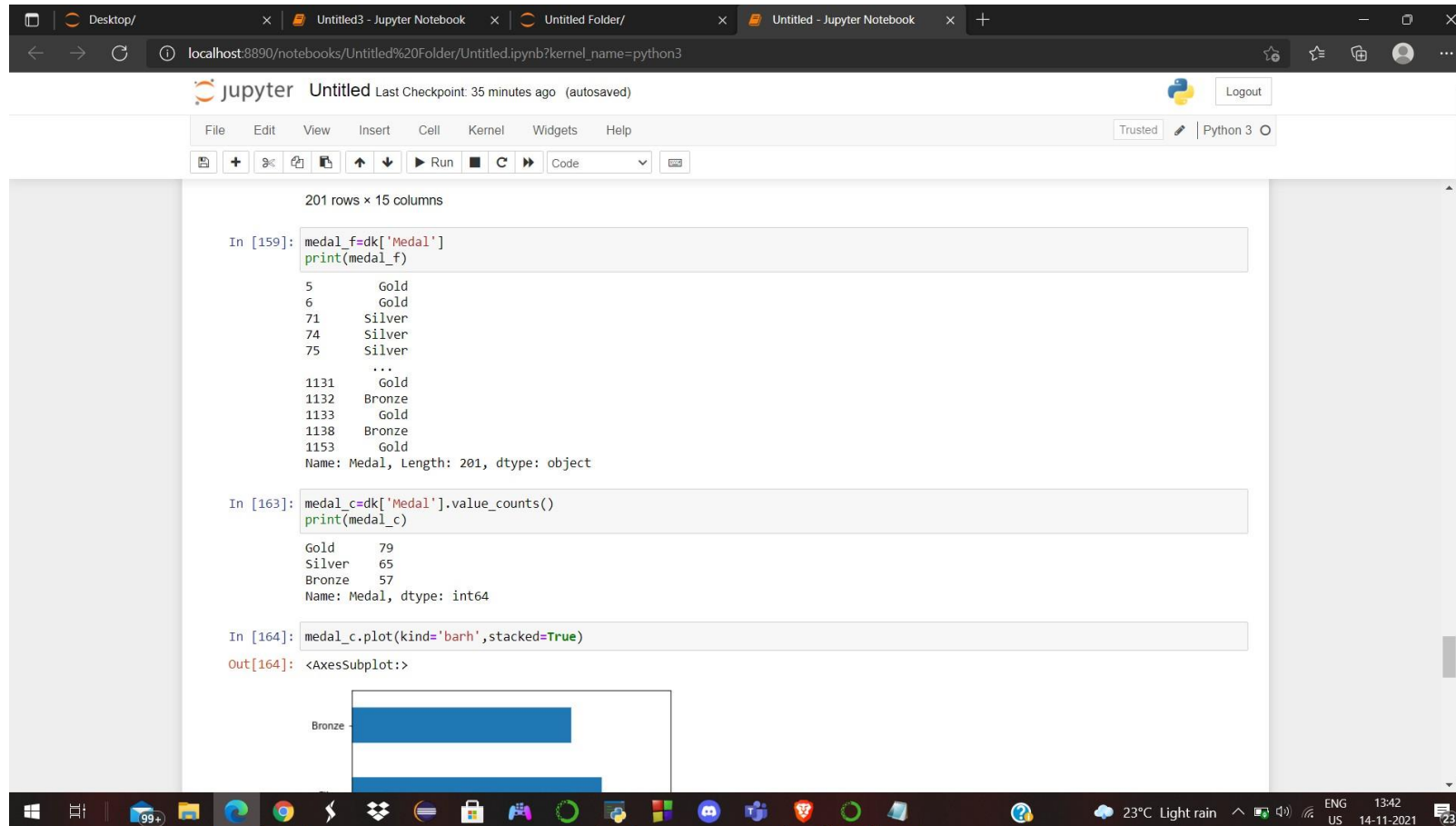
in [158]:

Bill=db['Teas']==f  
d=db[fil11]

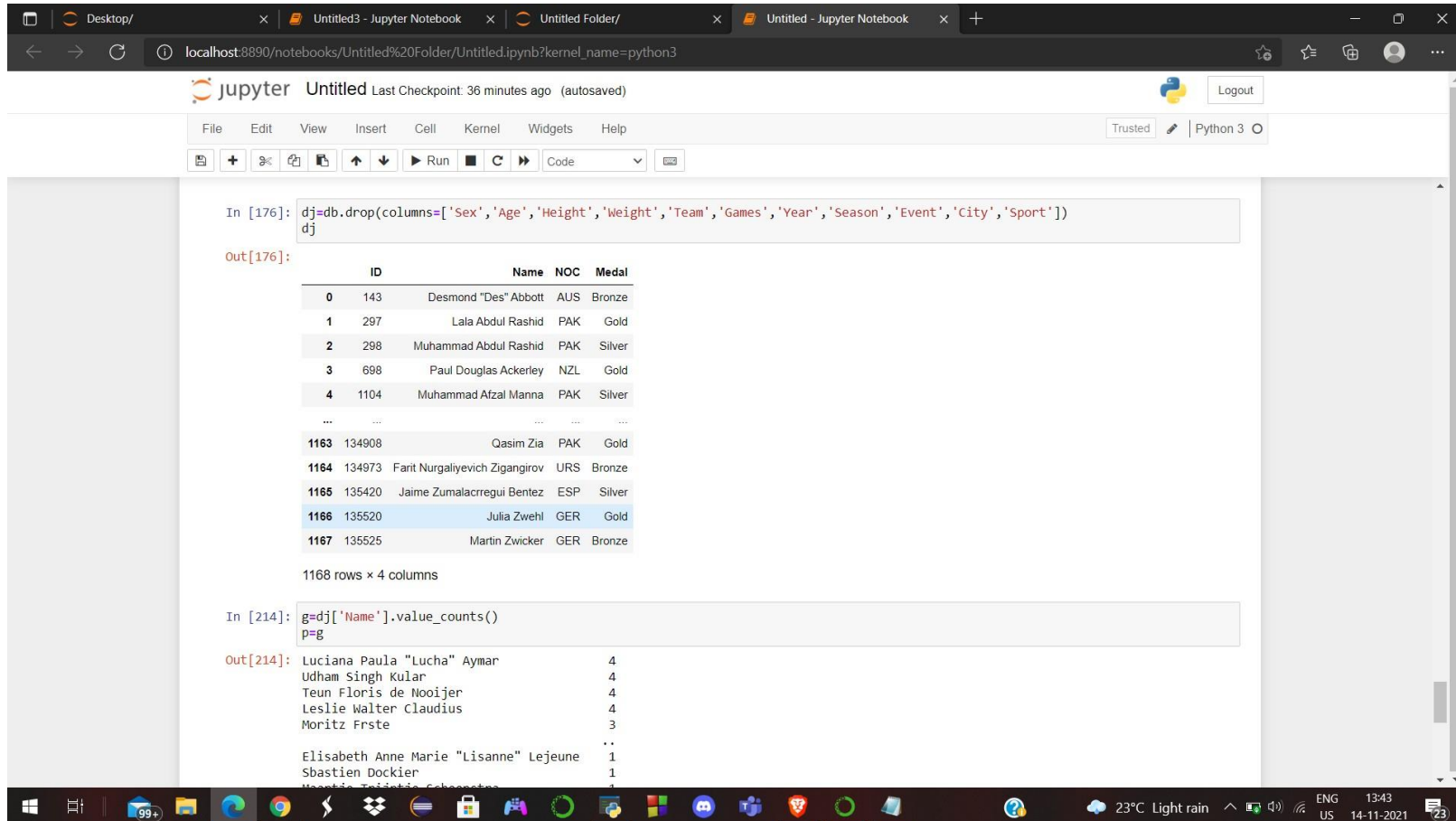
	ID	Name	Sex	Age	Height	Weight	Team	NOC	Games	Year	Season	City	Event	Medal
5	1169	Marilyn Agliotti	F	290	1r20	58.0	Netherlands	NED	2008 Summer	2008	Summer	London	Hockey Women's Hockey	Gold
6	1169	Marilyn Agliotti	F	33.0		58.0	Netherlands	NED	2012 Summer	2012	Summer	London	Hockey Women's Hockey	Gold
71	6397	Alexander German "Sander" Baart	M			76.0	Netherlands	NED	2012 Summer	2012	Summer	London	Hockey Men's Hockey	Silver
74	7072	Billy Pierre Bakker	M	23.0		82.0	Netherlands	NED	2012 Summer	2012	Summer	London	Hockey Men's Hockey	Silver
75	7273	Marcel Gerard Bakkestein	M		1800	80	Netherlands	NED	2012 Summer	2012	Summer	London	Hockey Men's Hockey	Silver
1131	130527	Laura Eveline "Laurien" Willemse (-Gales)	F			59.0	Netherlands	NED	1984 Summer	1984	Summer	Los Angeles	Hockey Women's Hockey	Gold
1142	130527	Laura Eveline "Laurien" Willemse (-Gales)	F	26.0		59.0	Netherlands	NED	1988 Summer	1988	Summer	Seoul	Hockey Women's Hockey	Bronze
1133	130960	Peter Pelegrindt	M	270	1880	160	Netherlands	NED	2000 Summer	2000	Summer	Sydney	Hockey Men's Hockey	Silver
1138	131343	Ingrid Imelda \off	F	240	1650	690	Netherlands	NED	1988 Summer		Summer	Seoul	Hockey Women's Hockey	Beuze
1153	134229	Inge Margriet Zegem (-de Ruiter)	F	0	1700	600	Netherlands	NED	1984 Summer	1984	Summer	Los Angeles	Hockey Women's Hockey	Gold

201 rows ^ 15 columns

# GRAPH TO SHOW THE MEDAL FREQUENCY OF THE HIGHEST MEDAL COUNTRY



# TO CREATE NEW CSV FILE



The screenshot shows a Jupyter Notebook window with the following components:

- Browser Tabs:** Desktop/, Untitled3 - Jupyter Notebook, Untitled Folder/, Untitled - Jupyter Notebook.
- Address Bar:** localhost:8890/notebooks/Untitled%20Folder/Untitled.ipynb?kernel\_name=python3
- Jupyter Header:** jupyter Untitled Last Checkpoint: 36 minutes ago (autosaved) Logout
- Menu Bar:** File Edit View Insert Cell Kernel Widgets Help
- Toolbar:** Includes icons for saving, opening, and running code.
- Code Cell [176]:**

```
dj=db.drop(columns=['Sex','Age','Height','Weight','Team','Games','Year','Season','Event','City','Sport'])  
dj
```
- Output [176]:** A table with 5 columns: ID, Name, NOC, Medal. It displays a subset of data with 1168 rows in total.

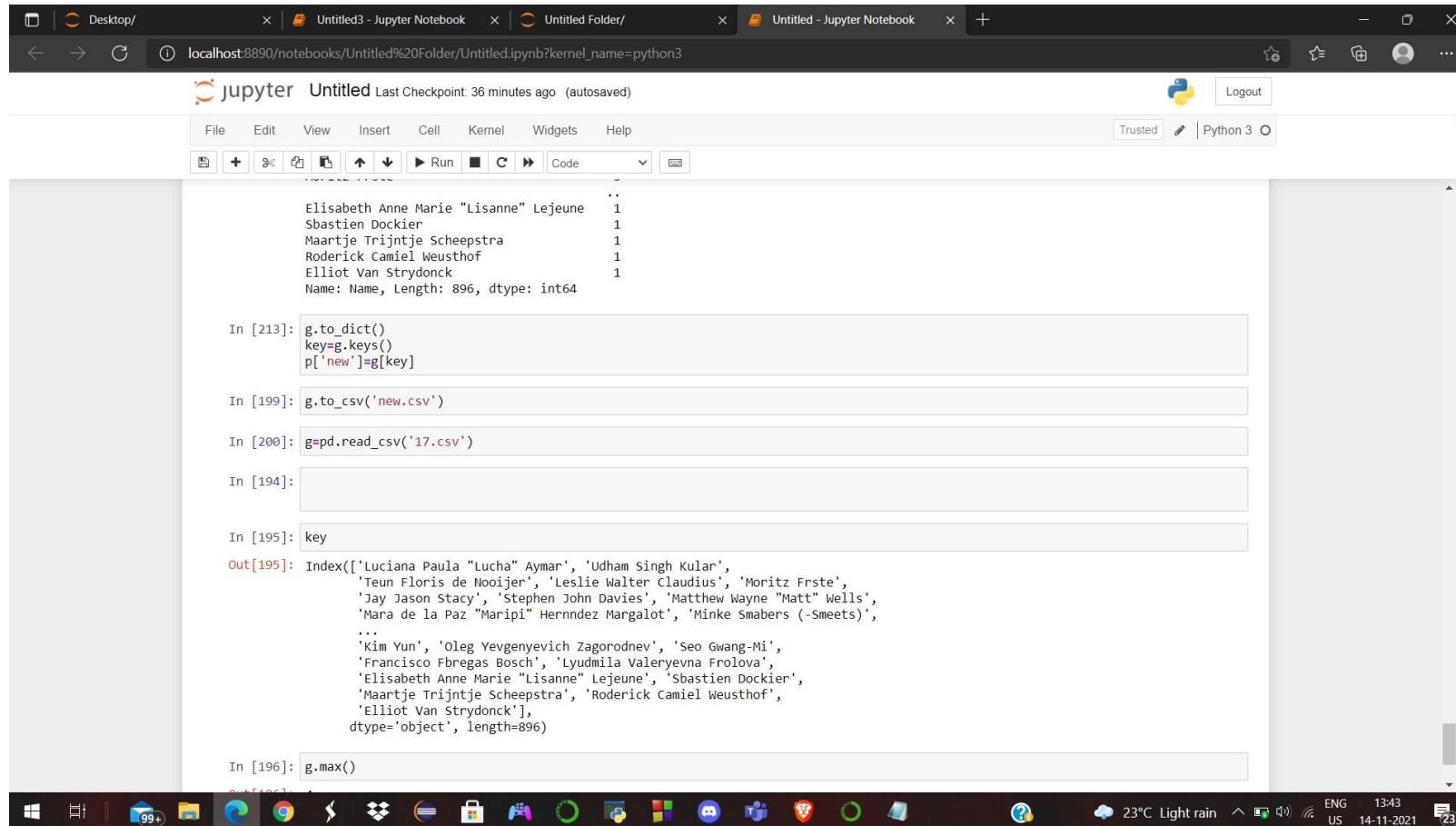
	ID	Name	NOC	Medal
0	143	Desmond "Des" Abbott	AUS	Bronze
1	297	Lala Abdul Rashid	PAK	Gold
2	298	Muhammad Abdul Rashid	PAK	Silver
3	698	Paul Douglas Ackerley	NZL	Gold
4	1104	Muhammad Afzal Manna	PAK	Silver
...	...	...	...	...
1163	134908	Qasim Zia	PAK	Gold
1164	134973	Farit Nurgaliyevich Zigangirov	URS	Bronze
1165	135420	Jaime Zumalacregui Bentez	ESP	Silver
1166	135520	Julia Zwehl	GER	Gold
1167	135525	Martin Zwicker	GER	Bronze

1168 rows x 4 columns
- Code Cell [214]:**

```
g=dj['Name'].value_counts()  
p=g
```
- Output [214]:** A list of names and their counts.

```
Luciana Paula "Lucha" Aymar      4  
Udham Singh Kular                 4  
Teun Floris de Nooijer            4  
Leslie Walter Claudius            4  
Moritz Frste                      3  
..                                 ..  
Elisabeth Anne Marie "Lisanne" Lejeune 1  
Sbastien Dockier                  1
```
- Taskbar:** Windows taskbar at the bottom showing system tray with date 14-11-2021, time 13:43, and weather 23°C Light rain.

# THE FREQUENCY OF THE PLAYER ACCORDING THE GOT MEDALS



```
Elisabeth Anne Marie "Lisanne" Lejeune    1
Sbastien Dockier                          1
Maartje Trijntje Scheepstra                1
Roderick Camiel Weusthof                   1
Elliot Van Strydonck                       1
Name: Name, Length: 896, dtype: int64

In [213]: g.to_dict()
          key=g.keys()
          p['new']=g[key]

In [199]: g.to_csv('new.csv')

In [200]: g=pd.read_csv('17.csv')

In [194]:

In [195]: key
Out[195]: Index(['Luciana Paula "Lucha" Aymar', 'Udham Singh Kular',
                'Teun Floris de Nooijer', 'Leslie Walter Claudius', 'Moritz Frste',
                'Jay Jason Stacy', 'Stephen John Davies', 'Matthew Wayne "Matt" Wells',
                'Mara de la Paz "Maripi" Hernandez Margalot', 'Minke Smabers (-Smeets)',
                ...,
                'Kim Yun', 'Oleg Yevgenyevich Zagorodnev', 'Seo Gwang-Mi',
                'Francisco Fbregas Bosch', 'Lyudmila Valeryevna Frolova',
                'Elisabeth Anne Marie "Lisanne" Lejeune', 'Sbastien Dockier',
                'Maartje Trijntje Scheepstra', 'Roderick Camiel Weusthof',
                'Elliot Van Strydonck'],
                dtype='object', length=896)

In [196]: g.max()
```