Python Project 2:
Excel Automation

Tan Sang Dao
June 17, 2020

Task Flows

Requirement:

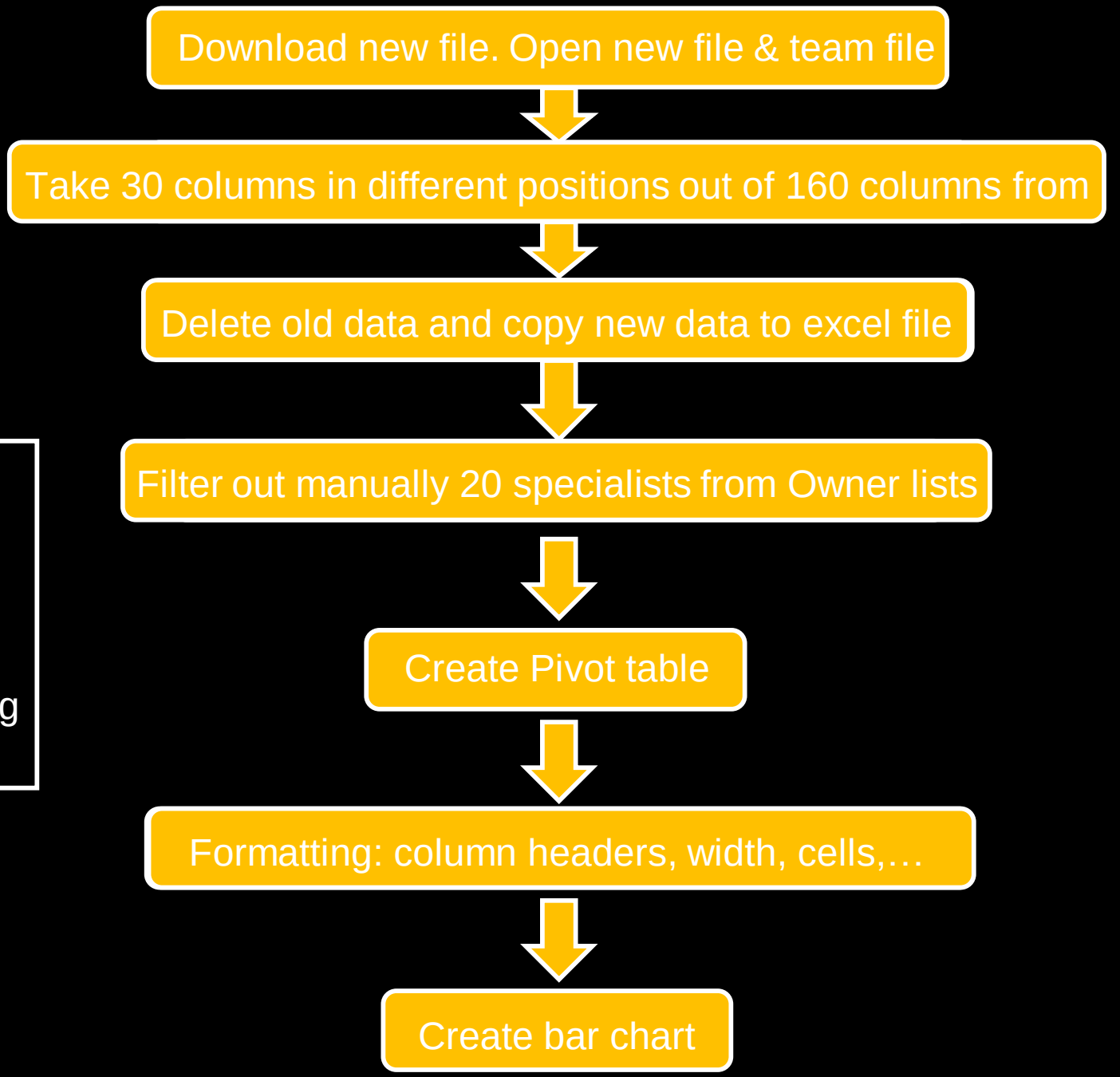
Sales managers want to have a report showing revenue generated from his team specialists.

Problem:

- Cleaning and bringing data to a predefined structure of team‘s excel file take long time, especially when having ad-hoc issues.

- This process ist repeated, consumes time doing more important tasks.

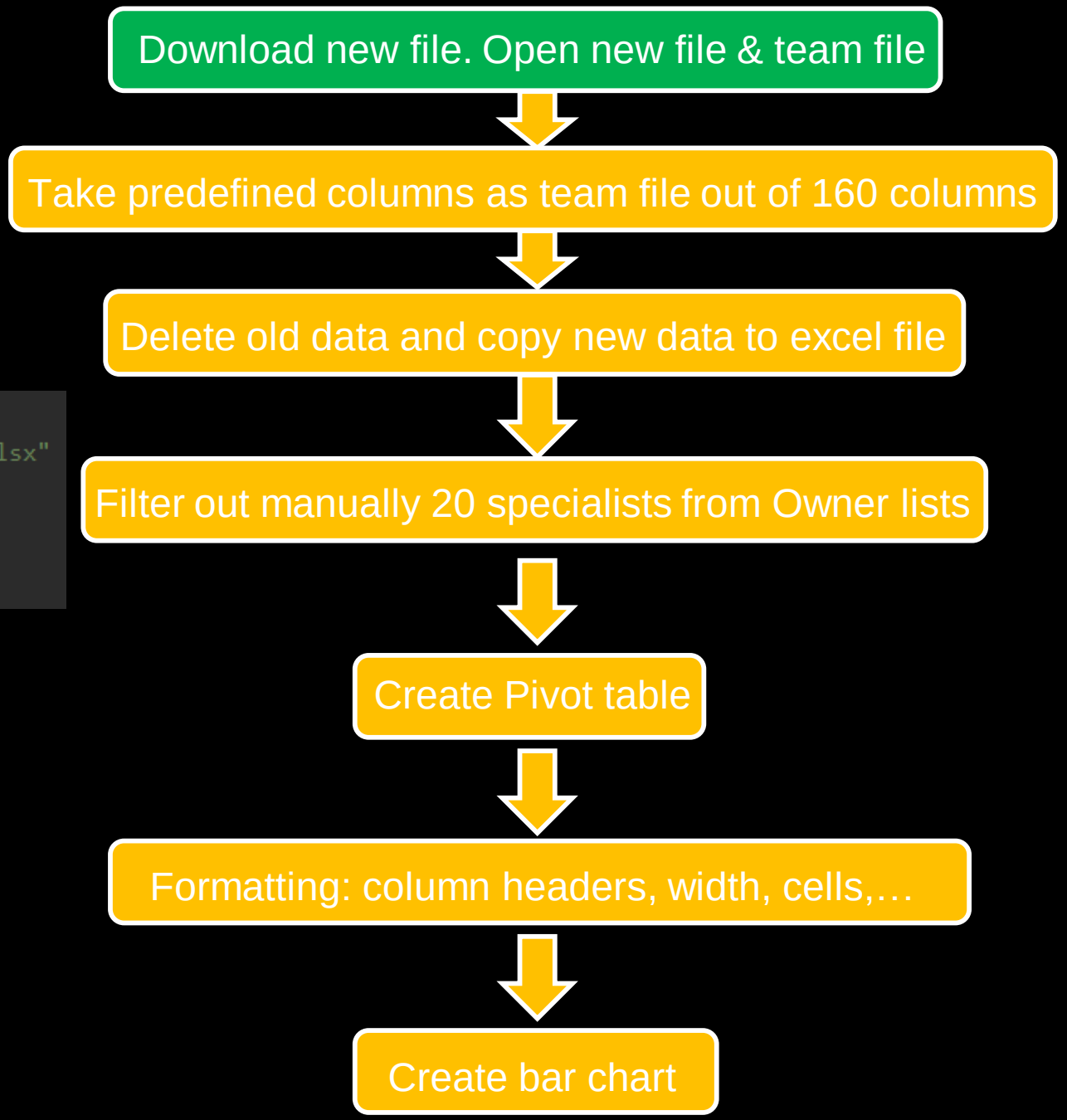
Creating time: 20-25 minutes/file



Python Script

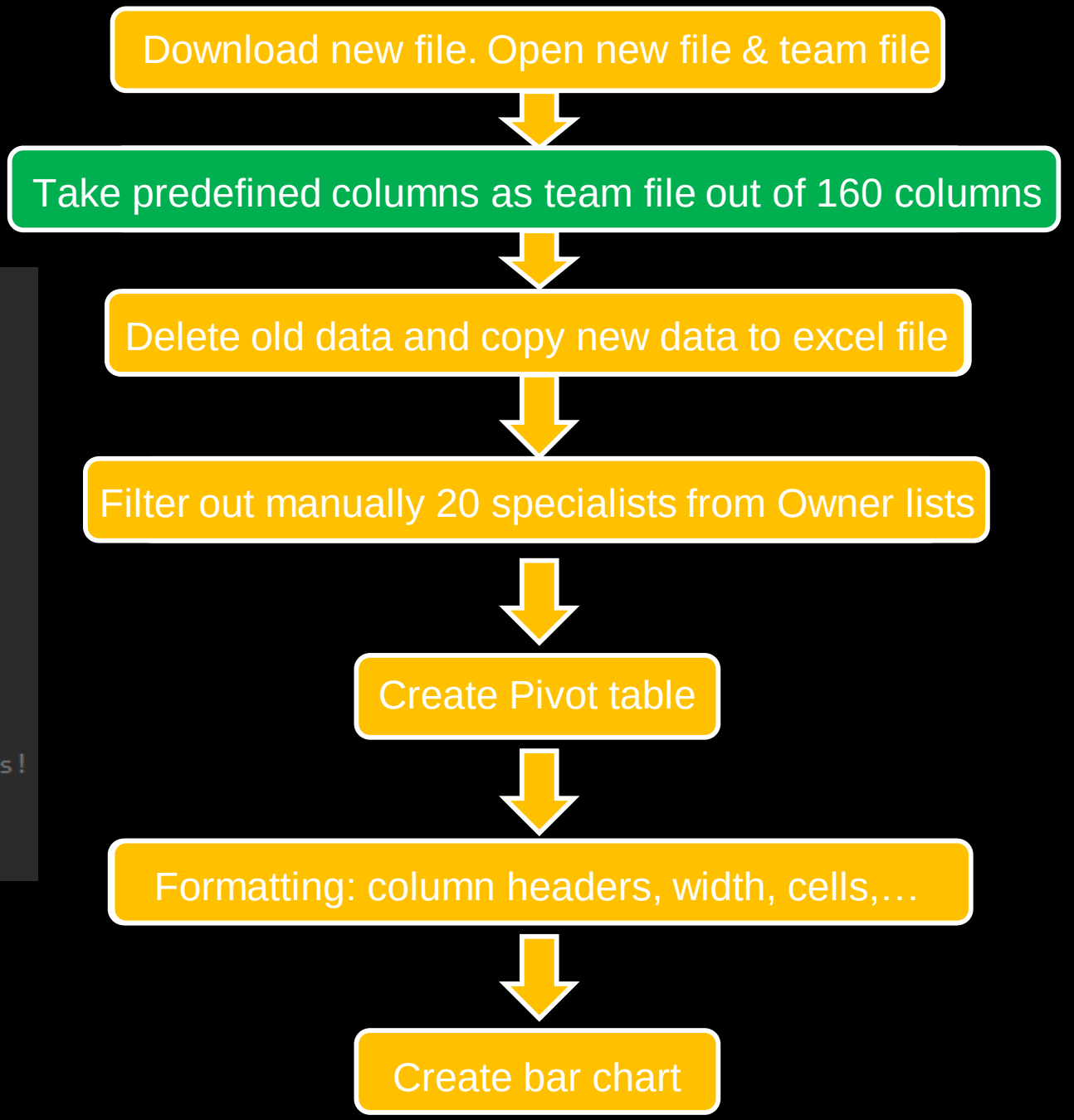
Python Script
(Step-by-Step)

```
framework = r"C:\Users\I517795\Desktop\Excel Project\Framework.xlsx"
cloud_pipe = r"C:\Users\I517795\Desktop\Excel Project\Cloud Pipeline.xlsx"

workbook1 = pd.read_excel(framework, sheet_name = None)
workbook2 = pd.read_excel(cloud_pipe, sheet_name = None)
```



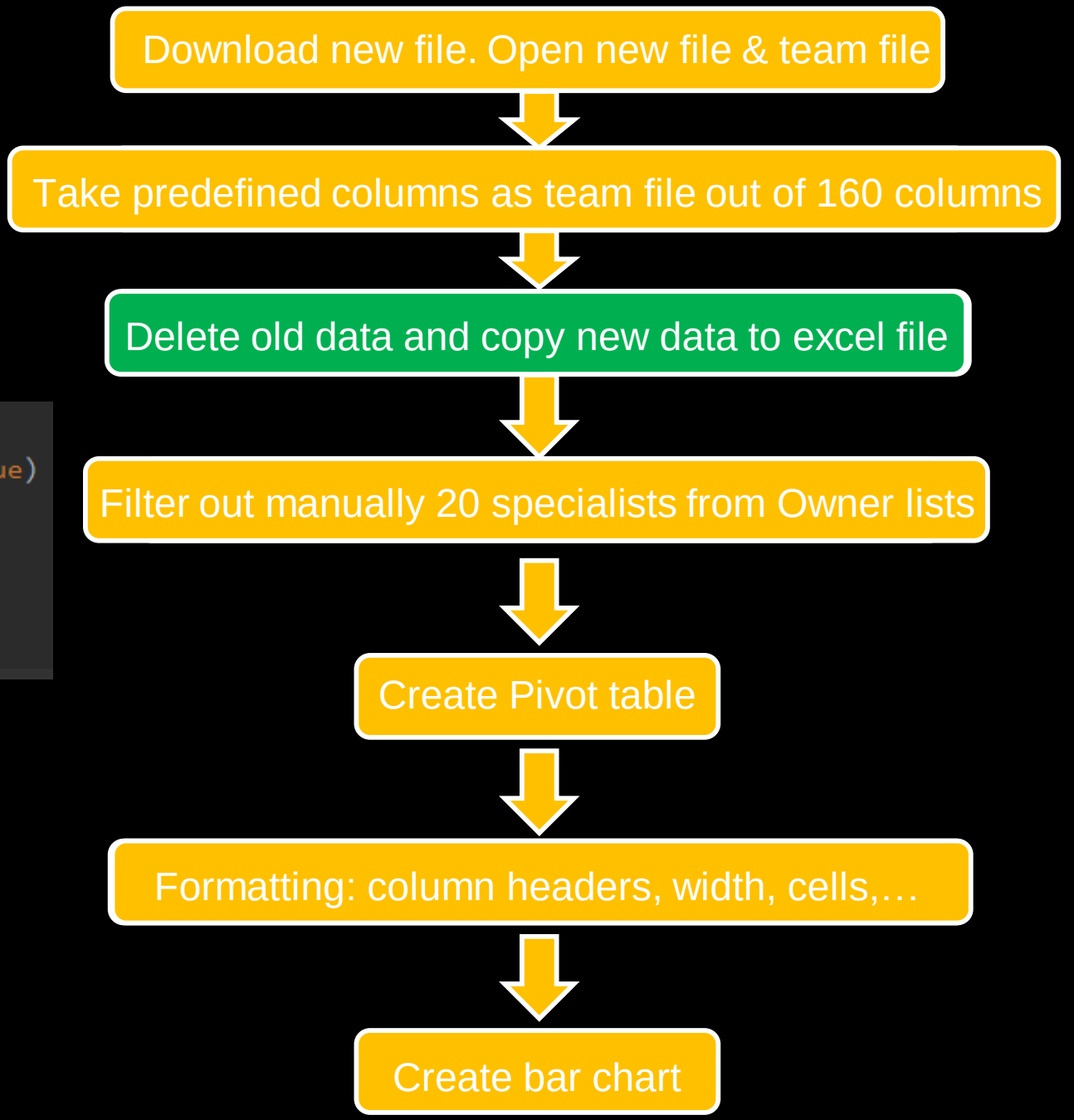
Python Script
(Step-by-Step)

```
sheet1 = workbook1['Raw Data']
sheet2 = workbook2['Sheet0']

for column in sheet2.columns:
    if column not in sheet1.columns:
        sheet2.drop(column, axis=1, inplace=True)

for column in sheet1.columns:
    if column not in sheet2.columns:
        sheet1.drop(column, axis=1, inplace=True)

# Check: if number of columns in sheet 2 = sheet 1 ==> Pass!
print(sheet1.shape)
print(sheet2.shape)
```



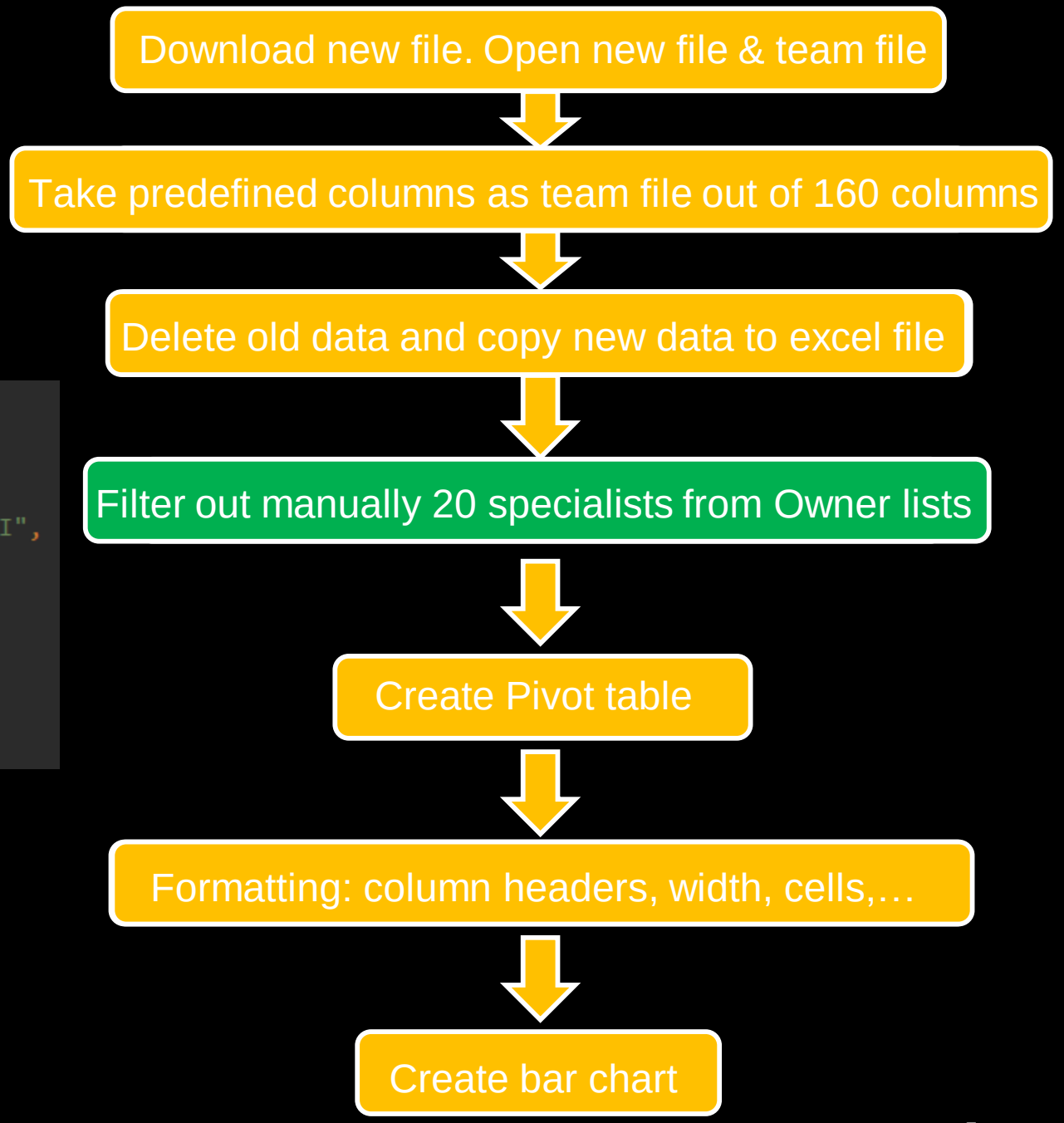
Python Script
(Step-by-Step)

```
# Delete old data in 'Framework':
sheet1.drop(sheet1.index[0:sheet1.shape[0]], inplace = True)

# Append new data from 'Sheet0' to 'Framework':
for column in sheet1.columns:
    sheet1[column] = sheet2[column]
```



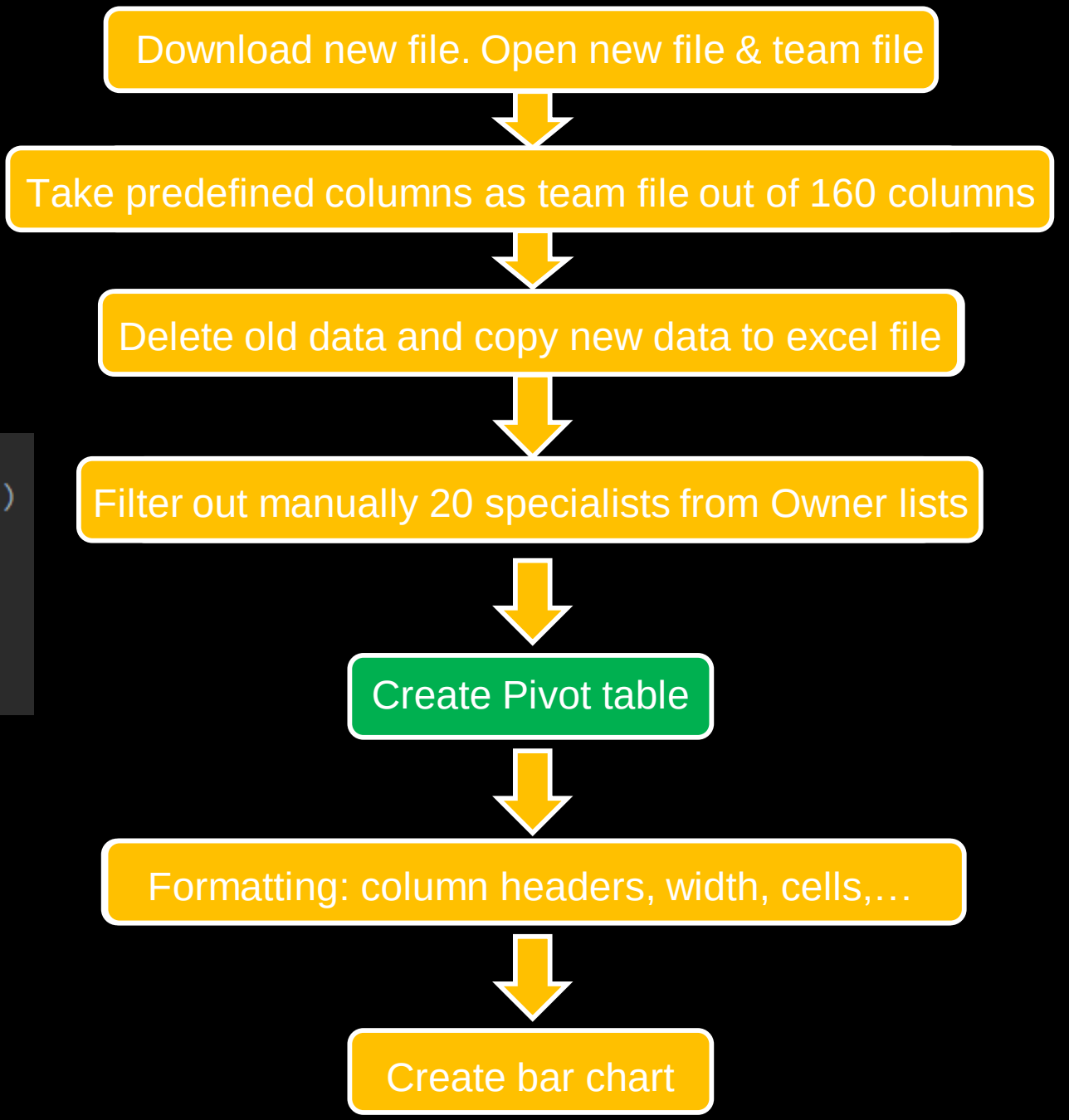
Python Script
(Step-by-Step)

```
owner_list = ["Ali NABIL", "Amrit RAJBANSH", "Cole Molapisane",
              "Eugenio Moya", "GIACOMO COPPI", "Hani ABDULAZIM",
              "Imran AHMED", "Nadim KLAT", "Orestes German",
              "Redha GHANDAL", "Roberto Marcolongo","STEFANO BONADIMANI",
              "Tiago Fernandes", "Wesley Estment", "Eldar Chiproot",
              "Selim Kayacan", "STEFANO BRAGHINI", "Ayse Ozay",
              "Yusuf NAQWI", "Javier Cordoba"]

filter = sheet1.loc[sheet1["Opp Owner Name"].isin(owner_list)]
```



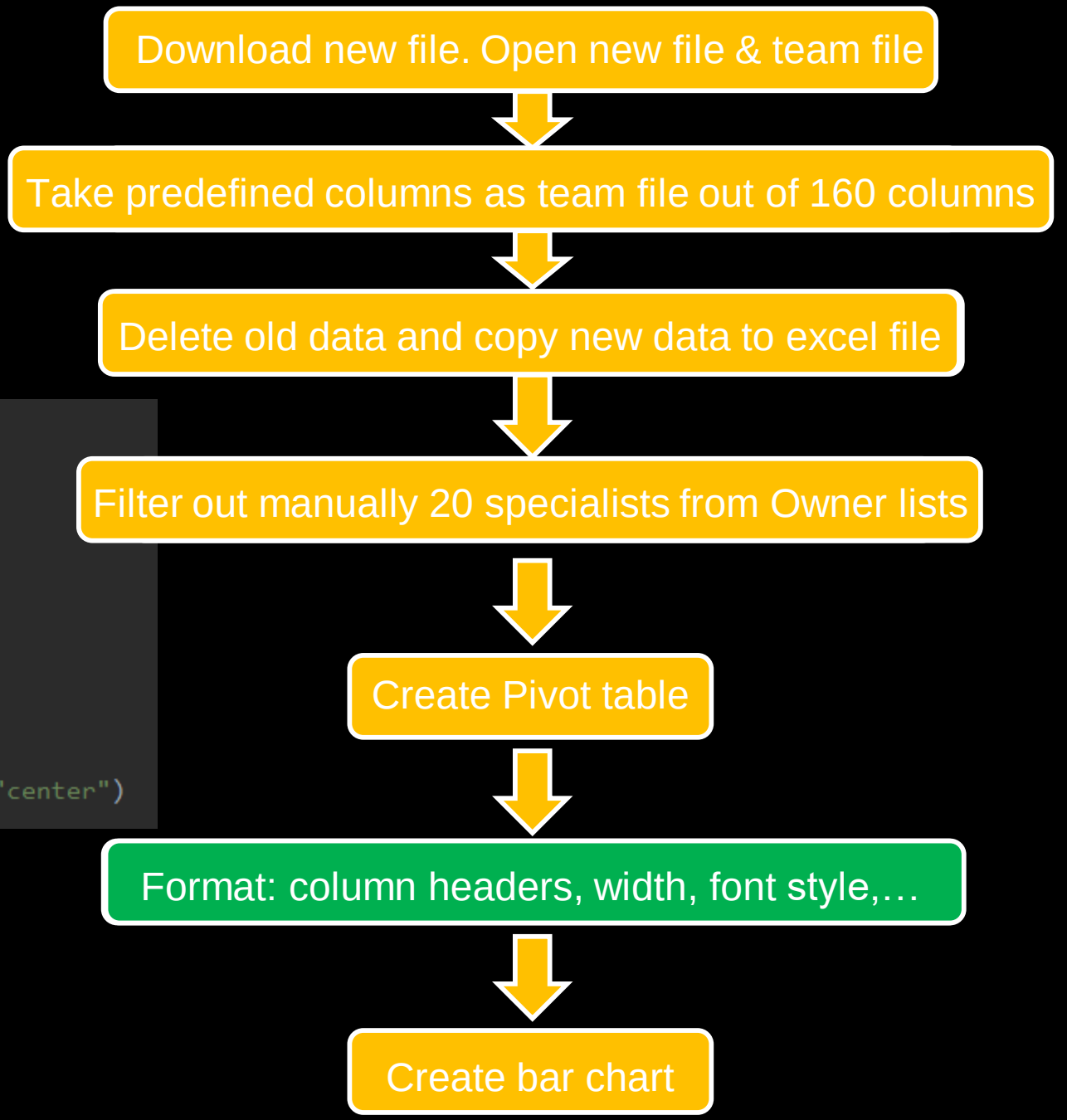
Python Script
(Step-by-Step)

```
#: Create pivot table:
pivot = filter.groupby(["Opp Owner Name", "Closing Qtr"])
result = pivot["ACV kEUR"].sum()

#Convert Series to DataFrame:
result_df = result.to_frame()
```



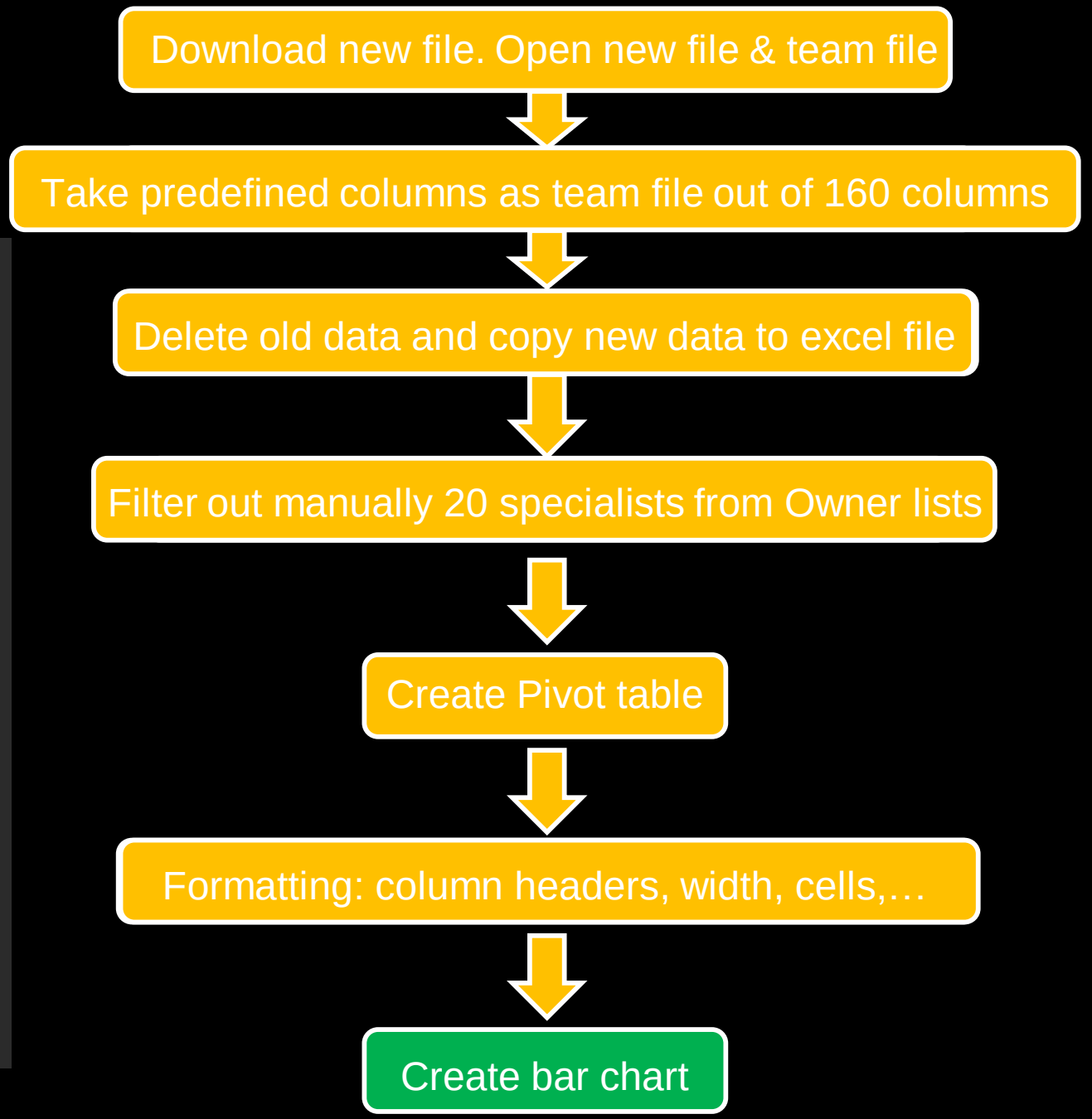
Python Script
(Step-by-Step)

```python
#7: Formatting:
ws = workbook['Pivot Table']
ws2 = workbook['Filter Data']

cell_list = ['A1', 'B1', 'C1']

for x in cell_list:
    ws[x].font = Font(bold = True)
    ws[x].fill = PatternFill("solid", fgColor="FCFF00")
    ws[x].alignment = Alignment(horizontal="center", vertical="center")
```



Python Script
(Step-by-Step)

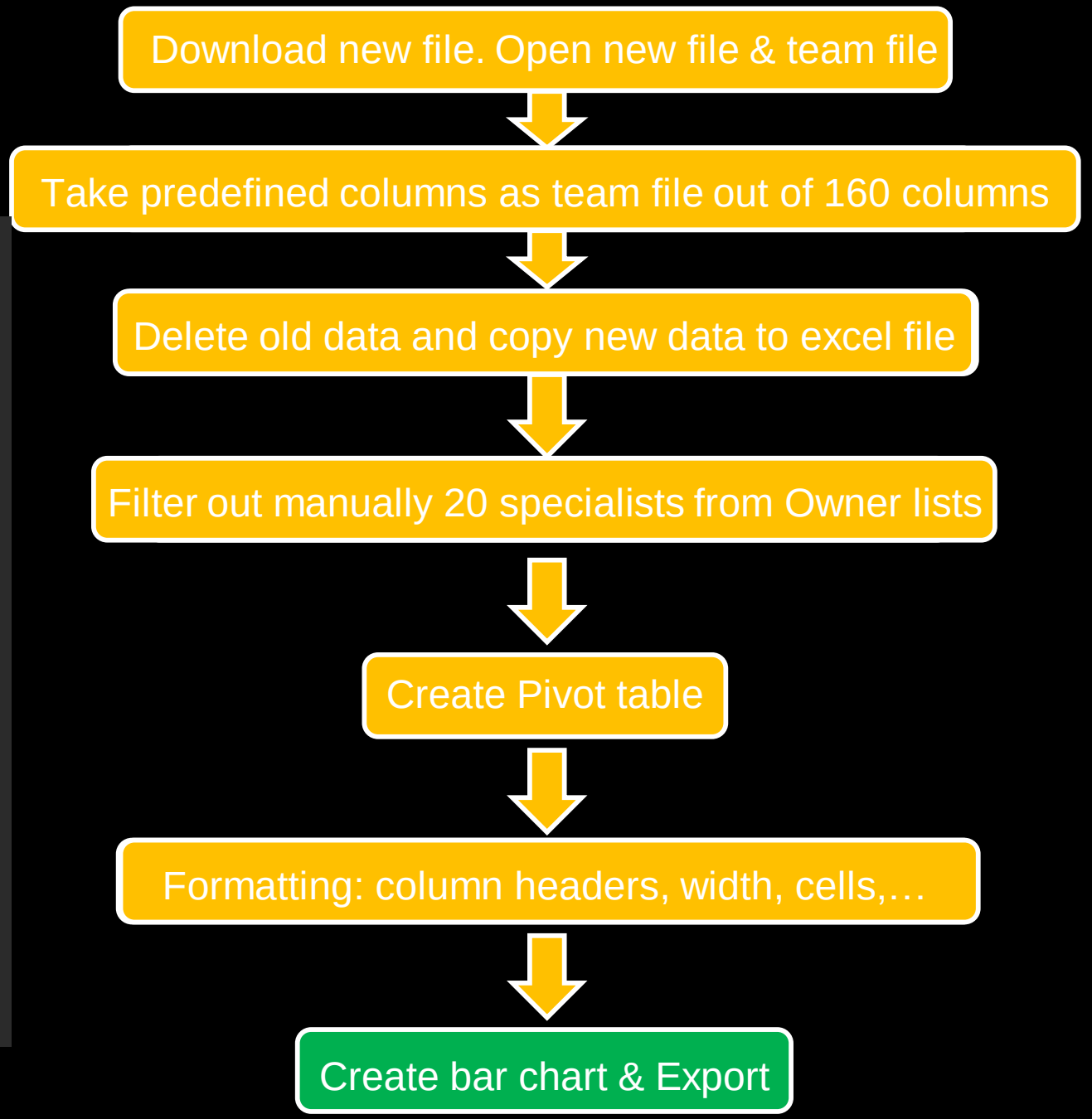
```
# Build chart:
chart1 = BarChart()
chart1.type = 'col'
chart1.style = 3
chart1.title = 'Top 20 of 2020'
chart1.y_axis.title = 'ACV'
chart1.x_axis.title = 'Quarter & Owners'
chart1.layout=Layout(manualLayout=ManualLayout(h=1, w=1))


data = Reference(ws, min_col=1, min_row=1,
                 max_col=3, max_row=65)
cats = Reference(ws, min_col=1, max_col=2,
                  min_row=1, max_row=65)

chart1.add_data(data, titles_from_data=True)
chart1.set_categories(cats)

ws.add_chart(chart1, "A10")
```



Python Script
(Step-by-Step)

```
# Build chart:
chart1 = BarChart()
chart1.type = 'col'
chart1.style = 3
chart1.title = 'Top 20 of 2020'
chart1.y_axis.title = 'ACV'
chart1.x_axis.title = 'Quarter & Owners'
chart1.layout=Layout(manualLayout=ManualLayout(h=1, w=1))


data = Reference(ws, min_col=1, min_row=1,
                 max_col=3, max_row=65)
cats = Reference(ws, min_col=1, max_col=2,
                  min_row=1, max_row=65)

chart1.add_data(data, titles_from_data=True)
chart1.set_categories(cats)

ws.add_chart(chart1, "A10")
```

```
workbook.save('Final Version.xlsx')
```



New creating time:
30 seconds

Full code: https://github.com/zaubers8/Python-Projects.git