

From multi-format CAD (BIM) data into a structured format 😊

```
... RVT | IFC | DWG conversion.py  
1 import os, subprocess  
2  
3 # Folder where the DDC converter is located  
4 path_conv = r'C:\DDC_Revit_Community\datadrivenlibs\\'  
5 # Path address RVT | IFC | DWG project are located  
6 file_path = r'C:\DDC\rstadvanced_sample.rvt'  
7  
8 # Conversion of one RVT project  
9 process = subprocess.Popen([os.path.join(path_conv,  
10 'RvtExporter.exe'), file_path], cwd=path_conv)  
11  
12 print("DDC Conversion process finished")
```

DATA CONVERSION TO OPEN FORMATS



conversion in just 4
lines of code

data**driven**
construction.io

```

1 # RVT | IFC | DWG project file name in XLSX format
2 output_file = file_path[:-4] + "_rvt.xlsx"
3 # Read the converted Excel file
4 df = pd.read_excel(output_file)
5 # Update column names to remove storage type in parameter
6 df.columns = [col.split(' : ')[0] for col in df.columns]

```

two-dimensional
project data

data-driven
construction.io

🚀 Structured format is ideal
for analytics, visualization
and automation



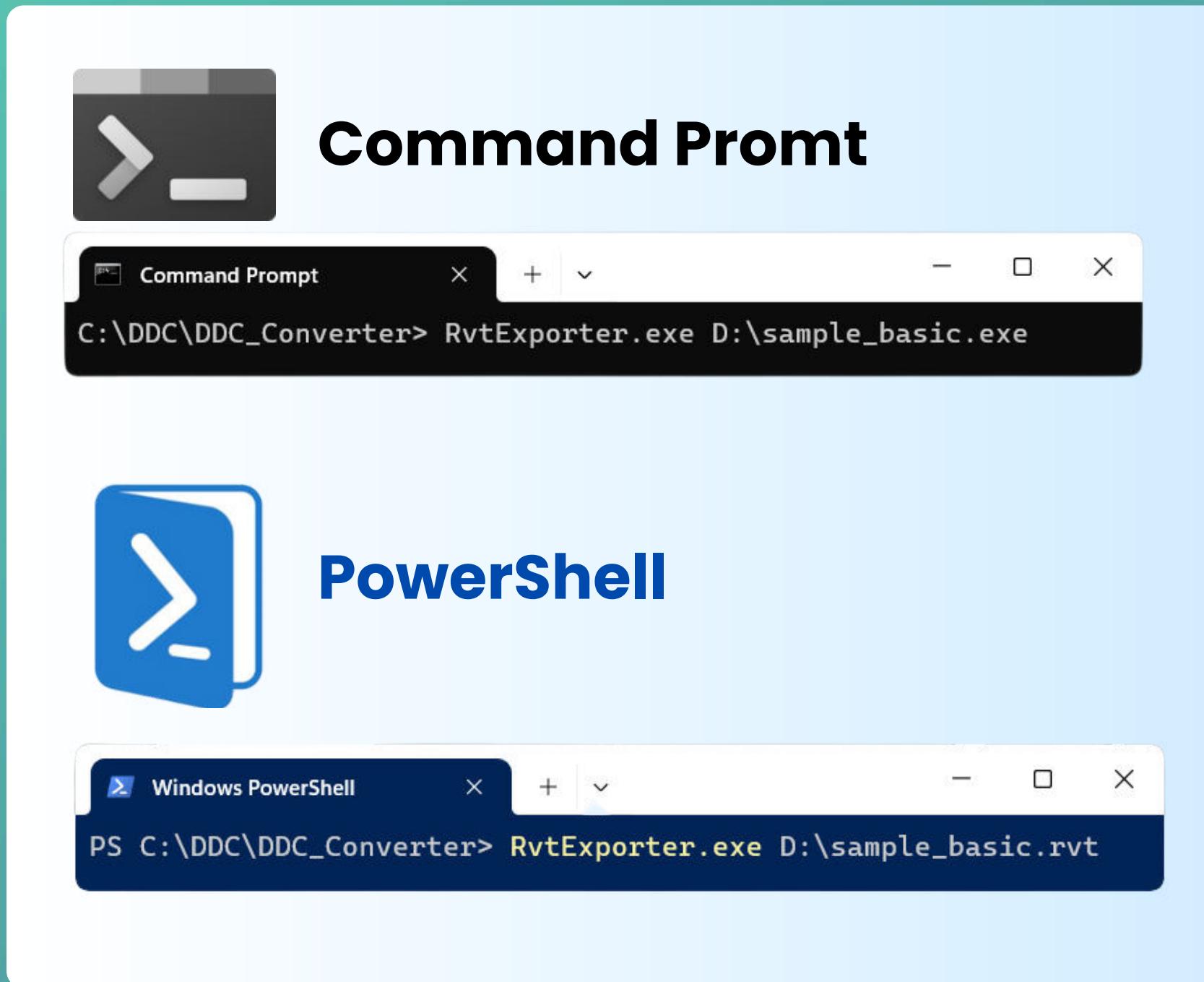
Diagram illustrating the structure of a DataFrame:

- Index label:** Points to the index column labeled "Index axis = 0".
- Column names:** Points to the first row of column headers.
- Data:** Points to the main body of the data.
- Missing value:** Points to a cell containing an empty box.
- Index axis = 1:** Points to the index column labeled "Index axis = 1".

ID	Name	Category	Family Name	Height	BoundingBoxMin_X	BoundingBoxMin_Y	BoundingBoxMin_Z	Level
431144	Single-Flush	OST_Doors	Single-Flush	6.88976378	20.1503	-10.438	9.84252	Level 1
431198	Single-Flush	OST_Doors		6.88976378	13.2281	-1.1207	9.84252	Level 2
457479	Single Window	OST_Windows	Single Window	8.858267717	-11.434	-11.985	9.80971	Level 2
485432	Single Window	OST_Windows	Single Window	8.858267717	-11.434	4.25986	9.80971	
490150	Single-Flush	OST_Doors	Single-Flush	6.88976378	-1.5748	-2.9565	-1E-16	Level 1
493697	Basic Wall	OST_Walls	Basic Wall		-38.15	20.1656	-4.9213	Level 1
497540	Basic Wall	OST_Walls	Basic Wall		-4.5212	-0.0708	9.84252	Level 1

Converter

terminal version



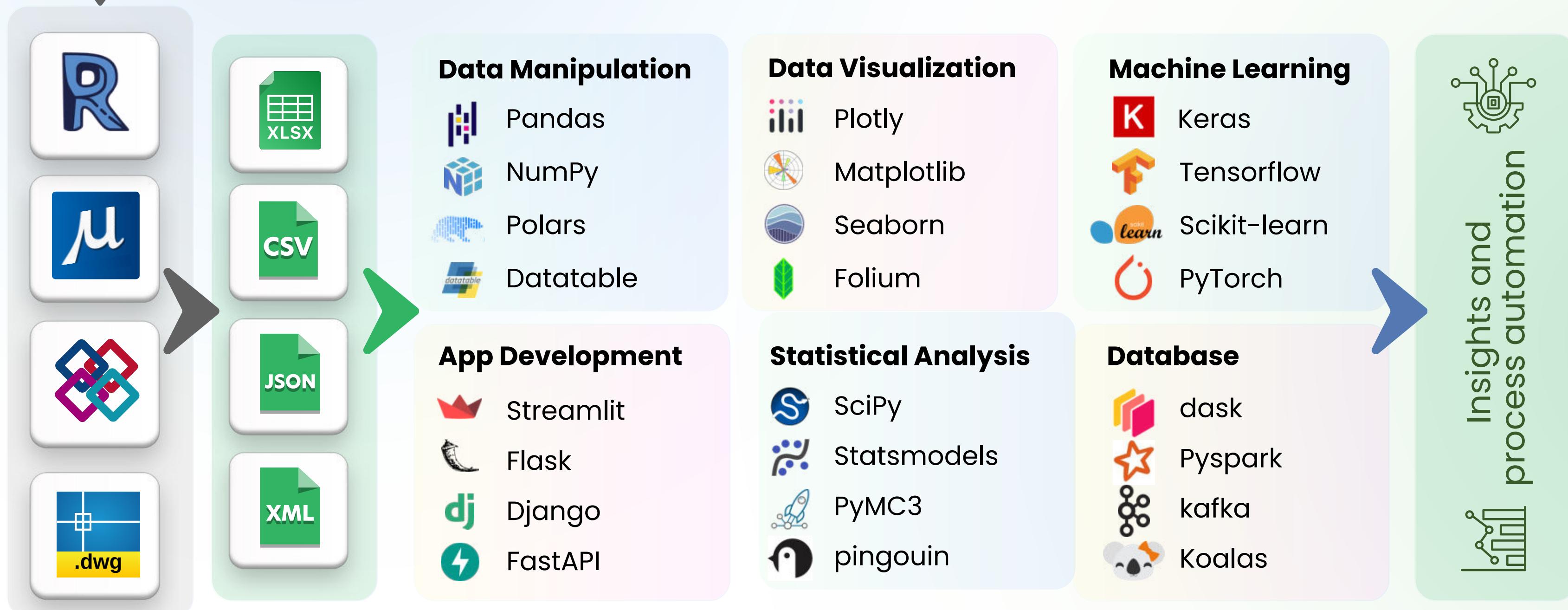
Hundreds of applications allow you to embed the conversion process into your use cases



Life Is Short, Use Python

to work with construction project data

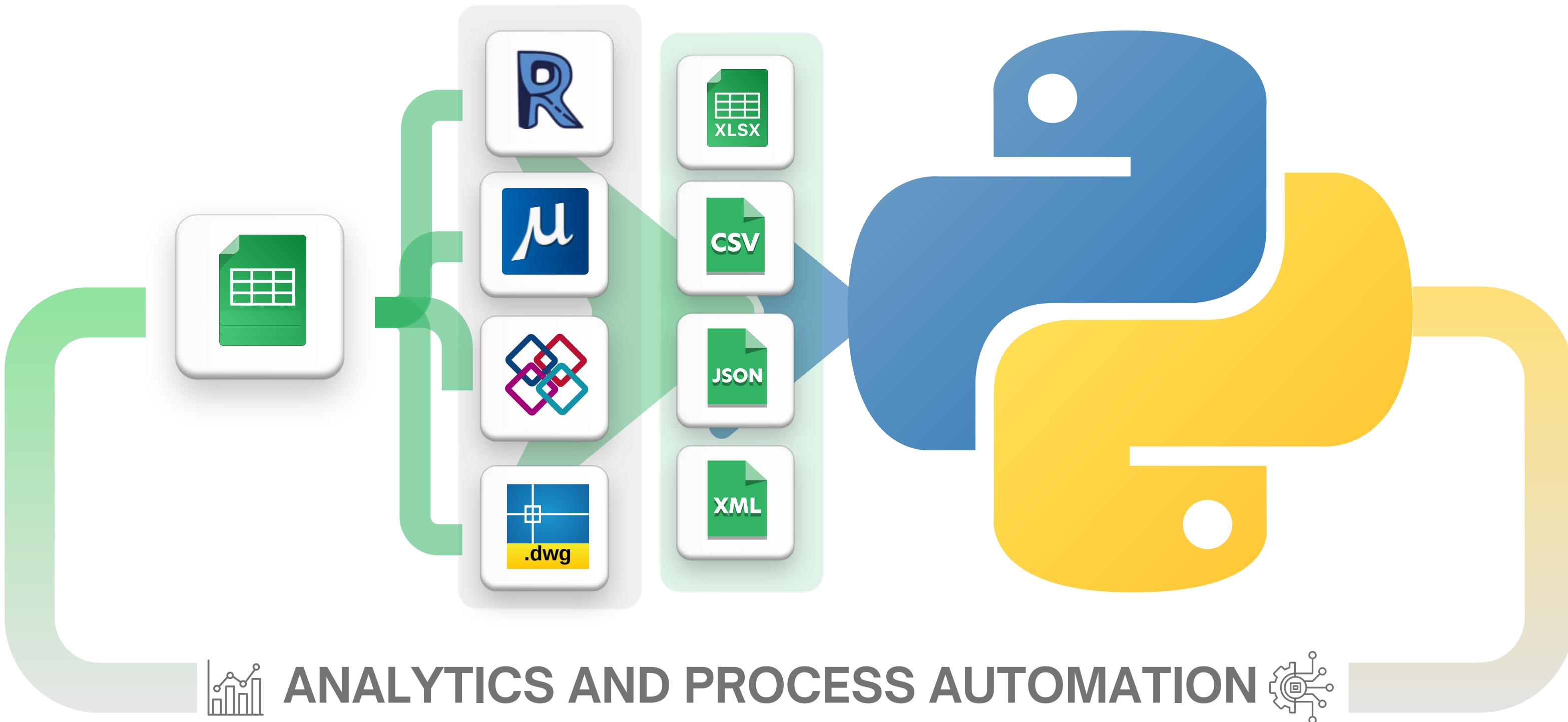
data-driven
construction.io

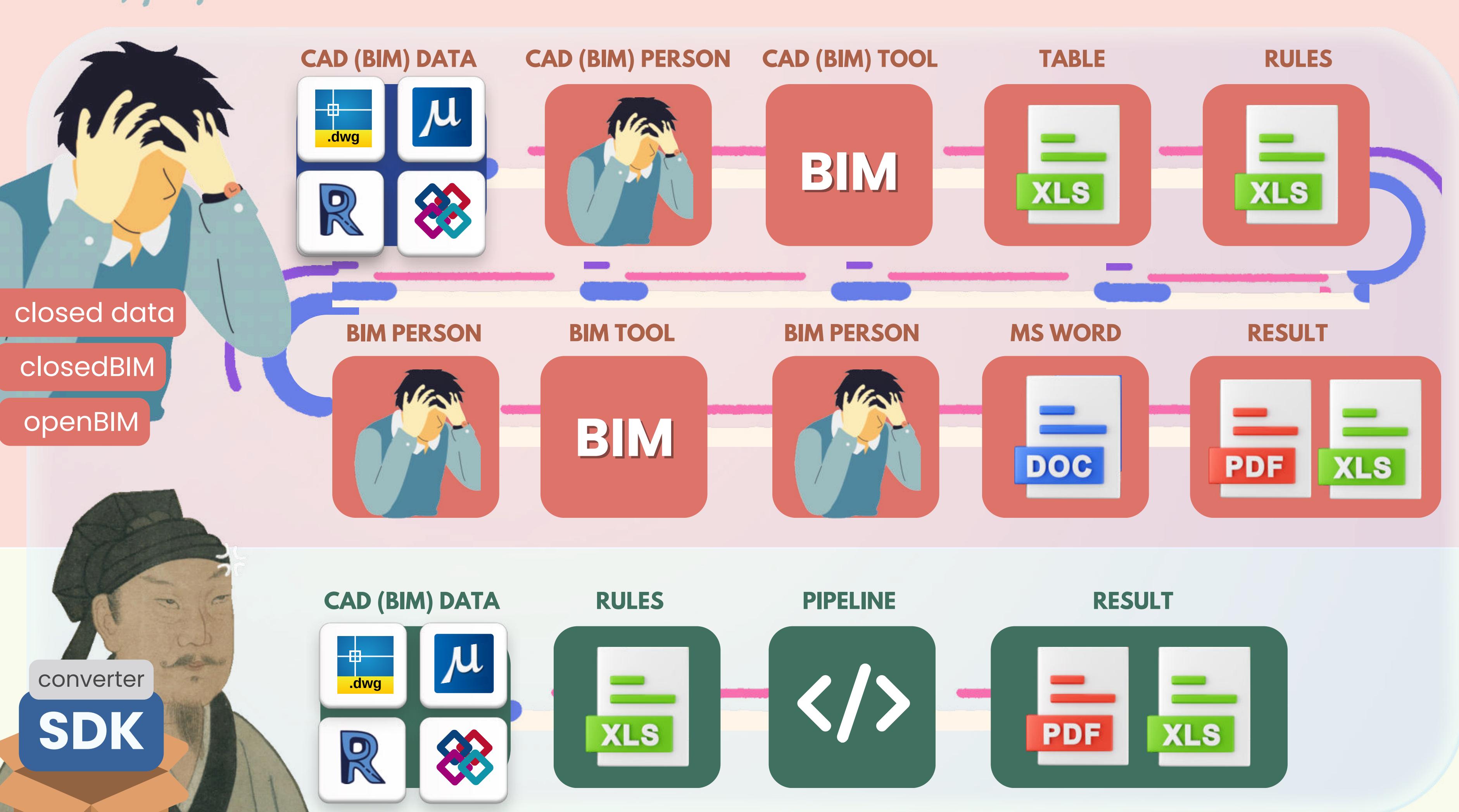


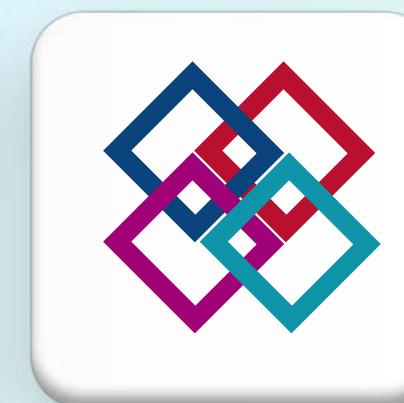
easy to learn, easy to develop

Life Is Short, Use Python

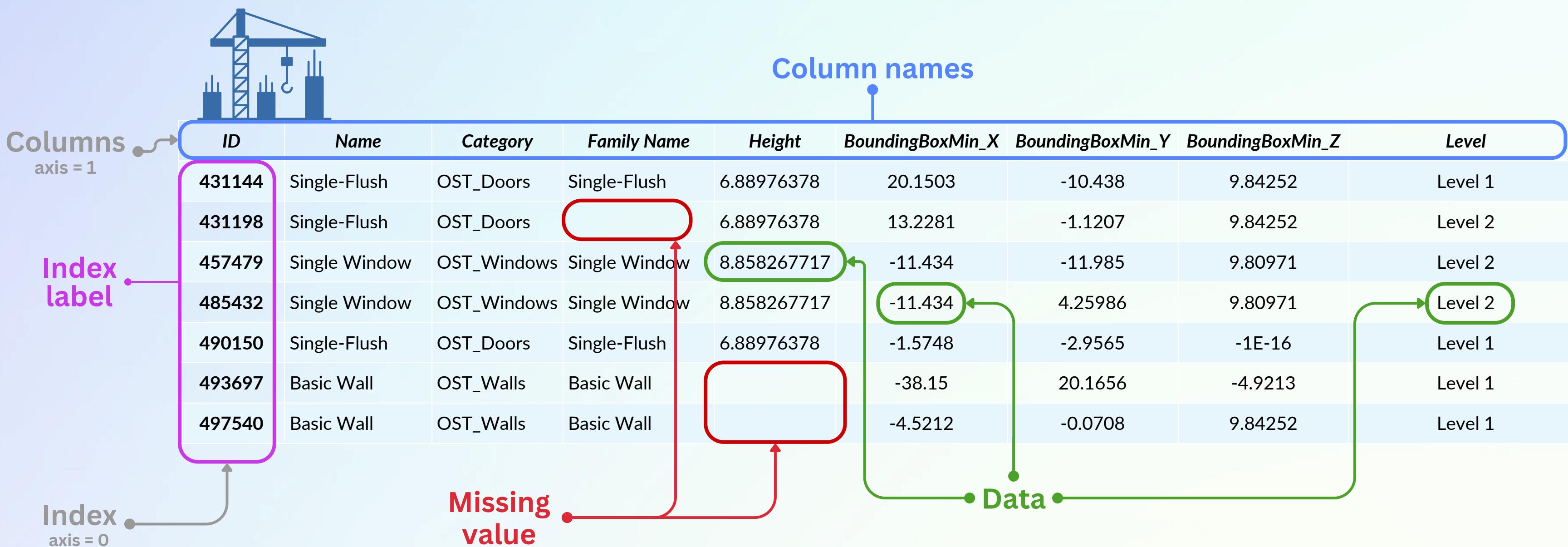
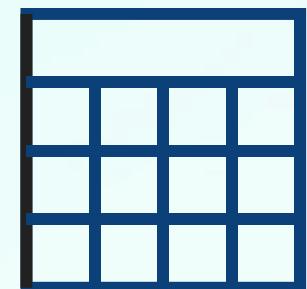
to work with data in construction







STRUCTURED DATA





IFC

STRUCTURED DATA

ID	Name	Category	version	pre	site	Parent	ObjectType
2	34_0001	IfcProject	IFC2X3				Y
3	38274_Default	IfcSite	IFC2X3	0001	0001		
4	36_y	IfcBuilding	IFC2X3	0001	Default	Default	Y
5	39_Level 1	IfcBuildingStorey	IFC2X3	0002	Default	Level 1	Basic Wall-Exterior - Brick on Block:1382 IfcWallStandard;ICF2X3
6	3797_Basic Wall-Exterior - Brick on Block:1382 IfcWallStandard;ICF2X3	IfcBuildingStorey	IFC2X3	0003	Default	Level 1	Basic Wall-Exterior - Brick on B
7	3999_Basic Wall-Exterior - Brick on Block:1382 IfcWallStandard;ICF2X3	IfcBuildingStorey	IFC2X3	0004	Default	Level 1	Basic Wall-Exterior - Brick on B
8	4043_Basic Wall-Exterior - Brick on Block:1382 IfcWallStandard;ICF2X3	IfcBuildingStorey	IFC2X3	0005	Default	Level 1	Basic Wall-Exterior - Brick on B
9	4087_Basic Wall-Exterior - Brick on Block:1382 IfcWallStandard;ICF2X3	IfcBuildingStorey	IFC2X3	0006	Default	Level 1	Basic Wall-Exterior - Brick on B
10	4111_Basic Wall-Interior - Partition (92mm Stu IfcWallStandard;ICF2X3	IfcBuildingStorey	IFC2X3	0007	Default	Level 1	Basic Wall-Interior - Partition (92mm Stu IfcWallStandard;ICF2X3
11	4219_Basic Wall-Interior - Partition (92mm Stu IfcWallStandard;ICF2X3	IfcBuildingStorey	IFC2X3	0008	Default	Level 1	Basic Wall-Interior - Partition (92mm Stu IfcWallStandard;ICF2X3)
12	4287_Basic Wall-Party Wall - CMU Residential IfcWallStandard;ICF2X3	IfcBuildingStorey	IFC2X3	0009	Default	Level 1	Basic Wall-Party Wall - CMU R
13	4399_Basic Wall-Party Wall - CMU Residential IfcWallStandard;ICF2X3	IfcBuildingStorey	IFC2X3	0010	Default	Level 1	Basic Wall-Party Wall - CMU R
14	4445_Basic Wall-Party Wall - CMU Residential IfcWallStandard;ICF2X3	IfcBuildingStorey	IFC2X3	0011	Default	Level 3	Basic Wall-Party Wall - CMU R
15	4508_Basic Wall-Interior - Partition (92mm Stu IfcWallStandard;ICF2X3	IfcBuildingStorey	IFC2X3	0012	Default	Level 1	Basic Wall-Interior - Partition (92mm Stu IfcWallStandard;ICF2X3)
16	4555_Basic Wall-Interior - Partition (92mm Stu IfcWallStandard;ICF2X3	IfcBuildingStorey	IFC2X3	0013	Default	Level 1	Basic Wall-Interior - Partition (92mm Stu IfcWallStandard;ICF2X3)
17	4598_Basic Wall-Interior - Partition (92mm Stu IfcWallStandard;ICF2X3	IfcBuildingStorey	IFC2X3	0014	Default	Level 1	Basic Wall-Interior - Partition (92mm Stu IfcWallStandard;ICF2X3)
18	5165_Floor:127mm Slab on Grade:141232 IfcSlab	IfcBuildingStorey	IFC2X3	0015	Default	Level 1	Floor:127mm Slab on Grade:141232 IfcSlab
19	5267_Floor:127mm Slab on Grade:141208 IfcSlab	IfcBuildingStorey	IFC2X3	0016	Default	Level 1	Floor:127mm Slab on Grade:141208 IfcSlab
20	5642_Basic Wall-Interior - Partition (92mm Stu IfcWallStandard;ICF2X3	IfcBuildingStorey	IFC2X3	0017	Default	Level 1	Basic Wall-Interior - Partition (92mm Stu IfcWallStandard;ICF2X3)
21	5903_Basic Wall-Interior - Partition (92mm Stu IfcWallStandard;ICF2X3	IfcBuildingStorey	IFC2X3	0018	Default	Level 1	Basic Wall-Interior - Partition (92mm Stu IfcWallStandard;ICF2X3)
22	6426_M_Fixed:4835mm x 2420mm:4835mm x IfcWindow	IfcBuildingStorey	IFC2X3	0019	Default	Level 1	4835mm x 2420mm
23	6511_M_Fixed:4835mm x 2420mm:4835mm x IfcWindow	IfcBuildingStorey	IFC2X3	0020	Default	Level 1	4835mm x 2420mm
24	6652_M_Single-Flush:1250mm x 2010mm:1251 IfcDoor	IfcBuildingStorey	IFC2X3	0021	Default	Level 1	1250mm x 2010mm
25	6757_M_Single-Flush:1250mm x 2010mm:1251 IfcDoor	IfcBuildingStorey	IFC2X3	0022	Default	Level 1	1250mm x 2010mm
26	6921_M_Fixed:750mm x 2200mm:750mm x 2 IfcWindow	IfcBuildingStorey	IFC2X3	0023	Default	Level 1	750mm x 2200mm
27	7004_M_Fixed:750mm x 2200mm:750mm x 2 IfcWindow	IfcBuildingStorey	IFC2X3	0024	Default	Level 1	750mm x 2200mm

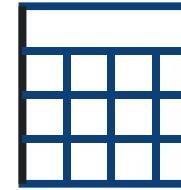


RVT

STRUCTURED DATA

ID	Name	Category	Design	IfcGUID	Type IfcGUID	Family and Type
0	198383_WinView - PVC Coating - OST_Materials	None	31Lx0gNe59vvExhby0f17			
1	198386_Single Window	OST_Windows	None	31Lx0gNe59vvExhby0f12		
2	198387_Basic Wall	OST_Walls	None	31Lx0gNe59vvExhby0f13		
3	198388_Finishes - Interior - Plaster	OST_Materials	None	31Lx0gNe59vvExhby0f12		
4	198389_Finishes - Interior - Paint	OST_Materials	None	31Lx0gNe59vvExhby0f12		
5	198390_Finishes - Interior - Timber Insulat	OST_Materials	None	31Lx0gNe59vvExhby0f12		
6	198392_Structure - Timber Insulat	OST_Materials	None	31Lx0gNe59vvExhby0f12		
7	198393_Structure - Timber Insulat	OST_Materials	None	31Lx0gNe59vvExhby0f12		
8	198394_Finishes - Exterior - Timber	OST_Materials	None	31Lx0gNe59vvExhby0f12		
9	198395_Finishes - Exterior - Tinble	OST_Materials	None	31Lx0gNe59vvExhby0f12		
10	198396_Basic Wall	OST_Walls	None	31Lx0gNe59vvExhby0f12		
11	198397_Basic Wall	OST_Walls	None	31Lx0gNe59vvExhby0f12		
12	198398_Finishes - Interior - Stud Layer	OST_Materials	None	31Lx0gNe59vvExhby0f12		
13	198399_Finishes - Interior - Stud Layer	OST_Materials	None	31Lx0gNe59vvExhby0f12		
14	198400_Finishes - Interior - Stud Layer	OST_Materials	None	31Lx0gNe59vvExhby0f12		
15	198401_Finishes - Exterior - Stud Layer	OST_Materials	None	31Lx0gNe59vvExhby0f12		
16	198402_Finishes - Exterior - Stud Layer	OST_Materials	None	31Lx0gNe59vvExhby0f12		
17	198403_Finishes - Exterior - Stud Layer	OST_Materials	None	31Lx0gNe59vvExhby0f12		
18	198404_Finishes - Exterior - Stud Layer	OST_Materials	None	31Lx0gNe59vvExhby0f12		
19	198405_Sink-Offset-Kohler-Vault	OST_PlumbingFittings	None	28:35WDD08ju0HmzK0fV1		
20	198406_Sink-Offset-Kohler-Vault	OST_PlumbingFittings	None	28:35WDD08ju0HmzK0fV1		
21	198407_Sink-Offset-Kohler-Vault	OST_PlumbingFittings	None	28:35WDD08ju0HmzK0fV1		
22	198408_Sink-Offset-Kohler-Vault	OST_PlumbingFittings	None	28:35WDD08ju0HmzK0fV1		
23	198409_Nickel-Kohler-SN-Vibrant	OST_Materials	None	28:35WDD08ju0HmzK0fV1		
24	198410_Nickel-Kohler-SN-Vibrant	OST_Materials	None	28:35WDD08ju0HmzK0fV1		
25	198411_Steel-Kohler-VS-Vibrant	OST_Materials	None	28:35WDD08ju0HmzK0fV1		
26	198412_Metal-Kohler-BI-Matte	OST_Materials	None	28:35WDD08ju0HmzK0fV1		
27	198413_Metal-Kohler-BI-Matte	OST_Materials	None	28:35WDD08ju0HmzK0fV1		
28	198414_Faucet-Binch_Reach-Kohler	OST_PlumbingFittings	None	28:35WDD08ju0HmzK0mwx		
29	198415_Faucet-Binch_Reach-Kohler	OST_PlumbingFittings	None	28:35WDD08ju0HmzK0mwx		
30	198416_Concrete - Cast-in-Place	OST_Materials	None	28:35WDD08ju0HmzK0mwx		
31	198417_Door - Frame	OST_Materials	None	28:35WDD08ju0HmzK0mwx		
32	198418_Door - Panel	OST_Materials	None	28:35WDD08ju0HmzK0mwx		
33	198419_Door - Panel	OST_Materials	None	28:35WDD08ju0HmzK0mwx		
34	198420_Basic Wall	OST_Walls	None	28:35WDD08ju0HmzK0mwx		
35	198421_System Panel	OST_CurtainWallPanel	None	28:35WDD08ju0HmzK0mwx		
36	198422_Rectangular Mullion	OST_CurtainWallMullion	None	28:35WDD08ju0HmzK0mwx		
37	198423_Single-Flush	OST_Doors	None	28:35WDD08ju0HmzK0mwx		
38	198424_EASEMENT	OST_Materials	None	28:35WDD08ju0HmzK0mwx		
39	198425_EASEMENT	OST_Materials	None	28:35WDD08ju0HmzK0mwx		
40	198426_EASEMENT	OST_Materials	None	28:35WDD08ju0HmzK0mwx		
41	198427_Basic Wall	OST_Walls	None	28:35WDD08ju0HmzK0mwx		
42	198428_Basic Wall	OST_Walls	None	28:35WDD08ju0HmzK0mwx		
43	198429_EASEMENT	OST_Materials	None	28:35WDD08ju0HmzK0mwx		
44	198430_EASEMENT	OST_Materials	None	28:35WDD08ju0HmzK0mwx		
45	198431_EASEMENT	OST_Materials	None	28:35WDD08ju0HmzK0mwx		
46	198432_EASEMENT	OST_Materials	None	28:35WDD08ju0HmzK0mwx		
47	198433_EASEMENT	OST_Materials	None	28:35WDD08ju0HmzK0mwx		
48	198434_EASEMENT	OST_Materials	None	28:35WDD08ju0HmzK0mwx		
49	198435_EASEMENT	OST_Materials	None	28:35WDD08ju0HmzK0mwx		
50	198436_EASEMENT	OST_Materials	None	28:35WDD08ju0HmzK0mwx		
51	198437_EASEMENT	OST_Materials	None	28:35WDD08ju0HmzK0mwx		
52	198438_EASEMENT	OST_Materials	None	28:35WDD08ju0HmzK0mwx		
53	198439_EASEMENT	OST_Materials	None	28:35WDD08ju0HmzK0mwx		
54	198440_EASEMENT	OST_Materials	None	28:35WDD08ju0HmzK0mwx		
55	198441_EASEMENT	OST_Materials	None	28:35WDD08ju0HmzK0mwx		
56	198442_EASEMENT	OST_Materials	None	28:35WDD08ju0HmzK0mwx		
57	198443_EASEMENT	OST_Materials	None	28:35WDD08ju0HmzK0m		

STRUCTURED DATA



Pandas: The leading library for data manipulation and a key tool for building pipelines



 **8811040**

Number of downloads of the Pandas
Pipeline library each day

 **70%**

Data engineers [using](#) Pandas Pipeline as
their primary tool

 **200k**

Questions on Stack Overflow [tagged](#) with
Pandas Pipeline



LOAD

Input

Importing Revit and IFC data.py

```
1 # Importing data for processing
2
3 import pandas as pd
4 df = pd.read_csv('C:\Revit_Sample.csv')
```

Output

	Id	Category	Type	Length	Volume
0	12577	Wall	Wall WD100	3200	1.0
1	15889	Wall	Wall STB 200	5400	6.0
2	76554	Door	Glazed Back Door	1300	0.3
3	74456	Window	Window 1700w	1700	0.5



FILTER

Input

Filtering data in Revit and IFC projects.py

```
1 # Whether each element contains the values
2
3 df[df['Category'].isin(['Wall', 'Window'])]
```

Output

	Id	Category	Type	Length	Volume
0	12577	Wall	Wall WD100	3200	1.0
1	15889	Wall	Wall STB 200	5400	6.0
3	74456	Window	Window 1700w	1700	0.5



GROUP

Input

GroupBy Revit IFC.py

```
1 # Grouping a Revit or IFC project by parameters
2
3 df.groupby('Category')['Volume', 'Length'].sum()
```

Output

Category	Volume	Length
Door	0.3	1300
Wall	7.0	8600
Window	0.5	1700

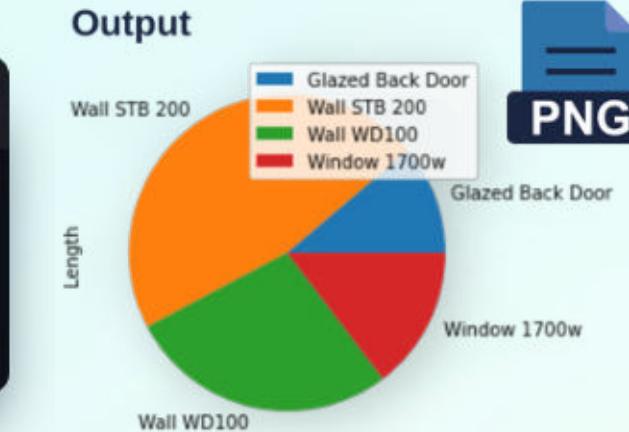


PIE chart



Input

```
- □ × Pie chart.py  
1 # Create a basic pie chart  
2  
3 df.groupby(['Type']).sum().plot.pie(y='Length')
```

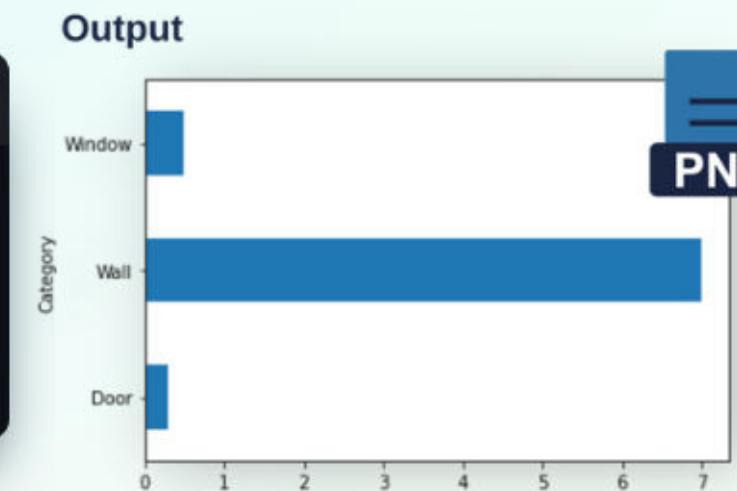


BAR chart



Input

```
- □ × Bar plot.py  
1 # The bar plot can be created as follows  
2  
3 dfp = df.groupby('Category')['Volume'].sum()  
4 dfp.plot(kind='barh')
```



Regular Expression



Input

```
- □ × RegEx.py  
1 #Regular expression in Revit and IFC  
2  
3 df[df['Category'].str.match('Wal*')]
```

Output

	ID	Category	Type	Length	Volume	grid icon
0	12577	Wall	Wall WD100	3200	1.0	
1	15889	Wall	Wall STB 200	5400	6.0	

QTO TakeOff

Input

```
- □ x QTO by RegEx.py  
1 #QTO - Finding volumetric quantities for the group  
2  
3 dfq = df[df['Category'].str.match('Wal*')]  
4 dfq = dfq.groupby('Category')['Volume', 'Length'].sum()
```

Output

Category	Volume	Length
Wall	7.0	8600



EXCEL Data Export

Input

```
- □ x Export to Excel.py  
1 # Creating a grouping and saving as Excel  
2  
3 dfe = df.groupby(['Category'])['Length'].agg(['sum', 'count'])  
4 dfe.to_excel("output.xlsx", sheet_name='Category_estimate')
```

Output

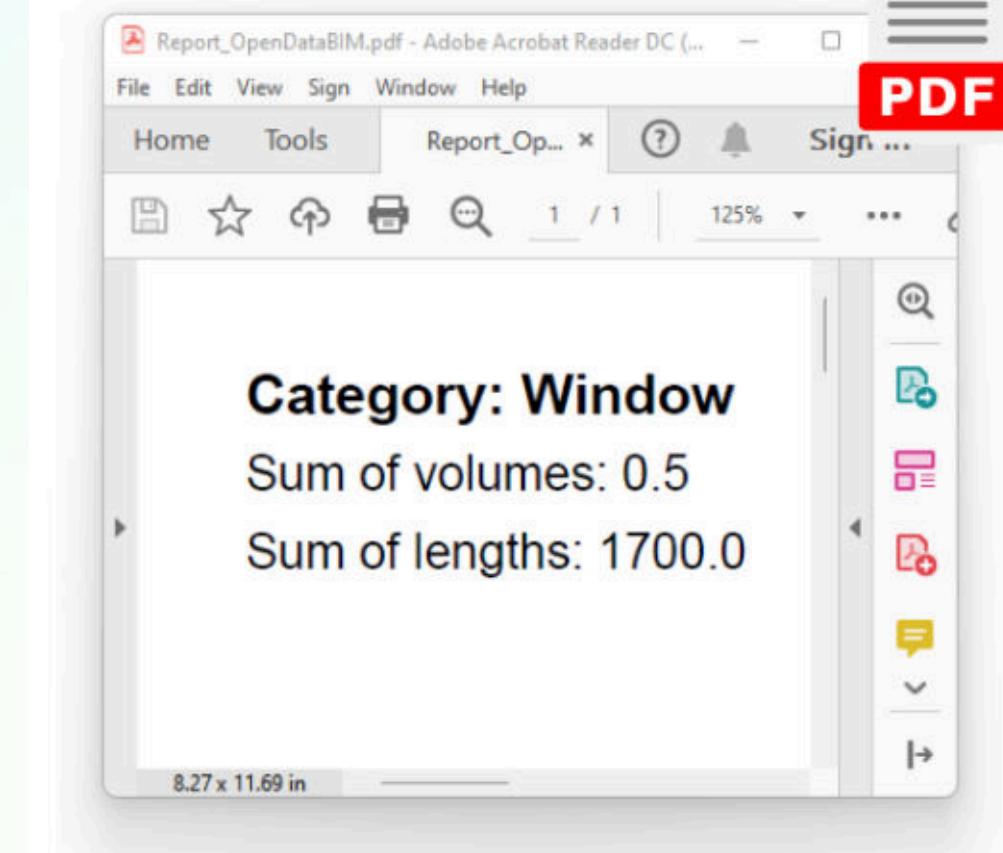
	A	B	C	D
2	Door	1300	1	
3	Wall	8600	2	
4	Window	1700	1	
5				

PDF Document

Input

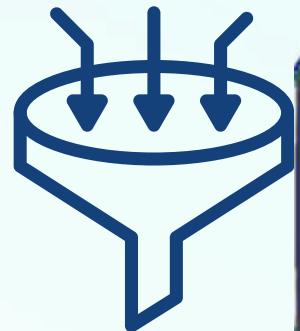
```
- □ x Creating a PDF document.py  
1 from fpdf import FPDF  
2  
3 # Determining the volumetric characteristics of the group  
4 s_cat = 'Window'  
5 dfq= df[df['Category'].str.match(s_cat)]  
6 dfq = dfq.groupby('Category')['Volume', 'Length'].sum()  
7 cat_len = str(dfq.iloc[0]['Length'])  
8 cat_vol = str(dfq.iloc[0]['Volume'])  
9  
10 # Creating a PDF document based on the parameters found  
11 pdf = FPDF()  
12 pdf.add_page()  
13 pdf.set_font('Arial', 'B', 16)  
14 pdf.cell(190, 8, 'Category: ' + s_cat, 2, 1, 'L')  
15 pdf.set_font('Arial', '', 14)  
16 pdf.cell(190, 8, 'Sum of volumes: ' + cat_vol, 2, 1, 'L')  
17 pdf.cell(190, 8, 'Sum of lengths: ' + cat_len, 2, 1, 'L')  
18  
19 # Saving a document in PDF format  
20 pdf.output(' c:\Report_DataDrivenConstruction.pdf ', 'F')
```

Output





FILTER



Filtering data in Revit and IFC projects.py

```

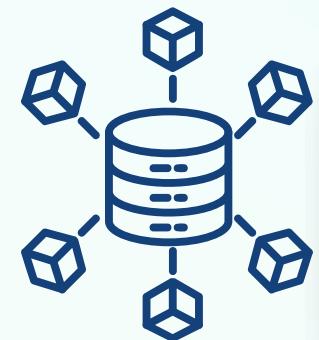
1 # Whether each element contains the values
2
3 df[df['Category'].isin(['Wall', 'Window'])]

```

	Id	Category	Type	Length	Volume
0	12577	Wall	Wall WD100	3200	1.0
1	15889	Wall	Wall STB 200	5400	6.0
3	74456	Window	Window 1700w	1700	0.5



GROUP



GroupBy Revit IFC.py

```

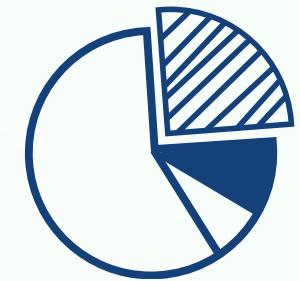
1 # Grouping a Revit or IFC project by parameters
2
3 df.groupby('Category')['Volume', 'Length'].sum()

```

Category	Volume	Length
Door	0.3	1300
Wall	7.0	8600
Window	0.5	1700

Filter the data in the project to keep the wall category items in the project

PDF

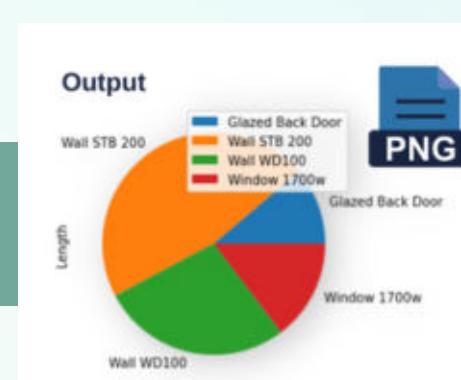


Creating a PDF document.py

```

1 from fpdf import FPDF
2
3 # Determining the volumetric characteristics of the group
4 s_cat = 'Window'
5 dfq= df[df['Category'].str.match(s_cat)]
6 dfq = dfq.groupby('Category')['Volume', 'Length'].sum()
7 cat_len = str(dfq.iloc[0]['Length'])
8 cat_vol = str(dfq.iloc[0]['Volume'])
9
10 # Creating a PDF document based on the parameters found
11 pdf = FPDF()
12 pdf.add_page()
13 pdf.set_font('Arial', 'B', 16)
14 pdf.cell(190, 8, 'Category: ' + s_cat, 2, 1, 'L')
15 pdf.set_font('Arial', '', 14)
16 pdf.cell(190, 8, 'Sum of volumes: ' + cat_vol, 2, 1, 'L')
17 pdf.cell(190, 8, 'Sum of lengths: ' + cat_len, 2, 1, 'L')
18
19 # Saving a document in PDF format
20 pdf.output(' c:\Report_DataDrivenConstruction.pdf ', 'F')

```



Choose the first 20 types by volume and show the result as a Pie chart



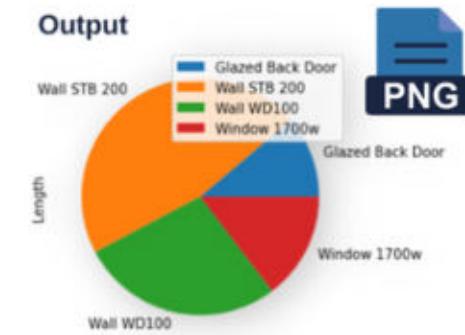
Create a PDF report with a table and a graph

chatGPT

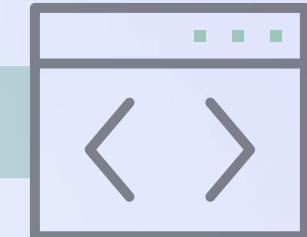
LLmA, Alpaca

Output					
	ID	Category	Type	Length	Volume
0	12577	Wall	Wall WD100	3200	1.0
1	15889	Wall	Wall STB 200	5400	6.0
3	74456	Window	Window 1700w	1700	0.5

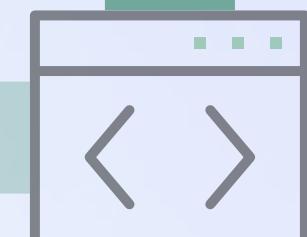
Output		
Category	Volume	Length
Door	0.3	1300
Wall	7.0	8600
Window	0.5	1700



Show the differences between the new version of the project and the latest version



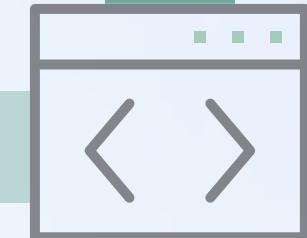
Filter the data in the project to keep the wall category items in the project



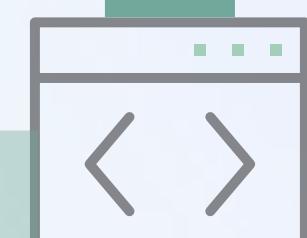
Group the project by the "Type Name" parameter and show the volume of each group



Choose the first 20 types by volume and show the result as a Pie chart



Create a PDF report with a table and a graph



PANDAS



1 Line of code



IDE

QTO.py

```
df[df['Category'].isin(['OST_Walls', 'OST_Columns'])].groupby('Type')['Volume'].sum()
```



Milliseconds



1.5" x 1.5"	0.00
Lamelle 11.5	74.82
MW 11.5	141.28
MW 17.5	67.43
STB 20.0	173.78
STB 25.0 WD 12.0	7.33
STB 30.0	88.57
STB 30.0 Rot	16.82
Standard	0.00
WC Trennwand 5.0	1.61

Effort



Input



Time



Output

1 Sentence



LLM Chat

Sum the 'Volume' column, grouped by
'Type', but only for rows where
'Category' is either 'OST_Walls' or
'OST_Columns'



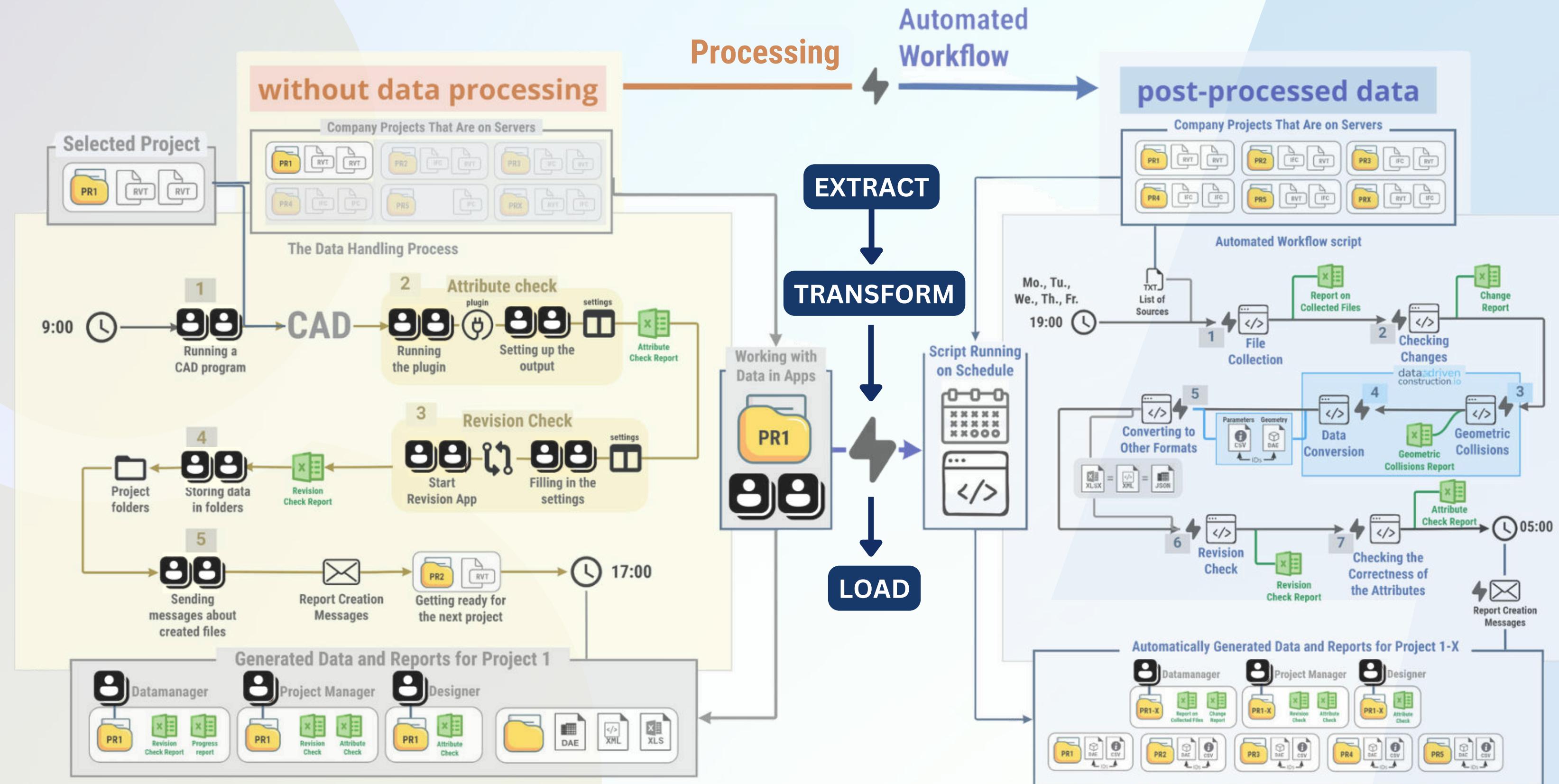
Seconds



1.5" x 1.5"	0.00
Lamelle 11.5	74.82
MW 11.5	141.28
MW 17.5	67.43
STB 20.0	173.78
STB 25.0 WD 12.0	7.33
STB 30.0	88.57
STB 30.0 Rot	16.82
Standard	0.00
WC Trennwand 5.0	1.61

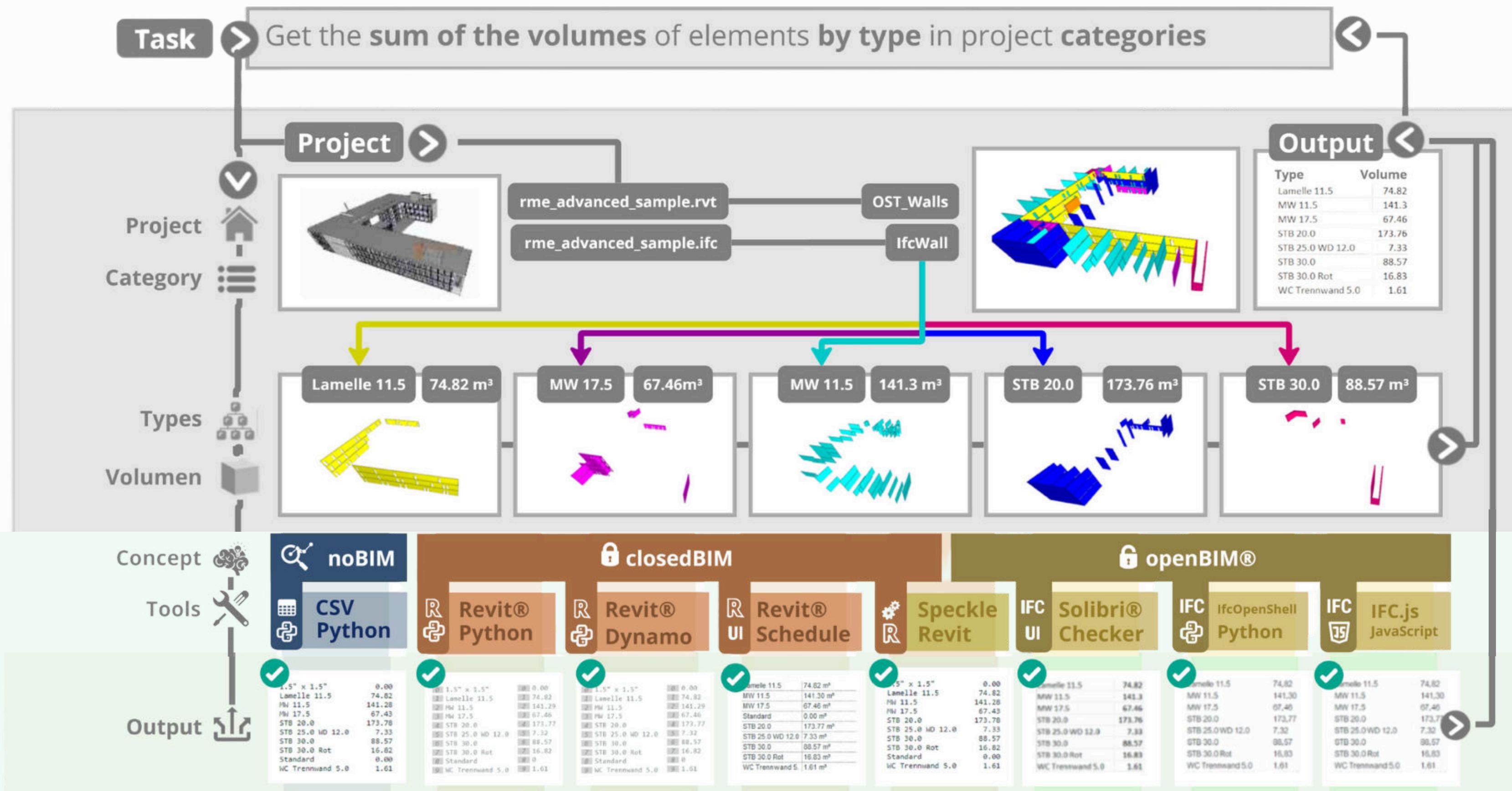
CHATGPT



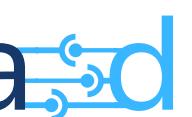


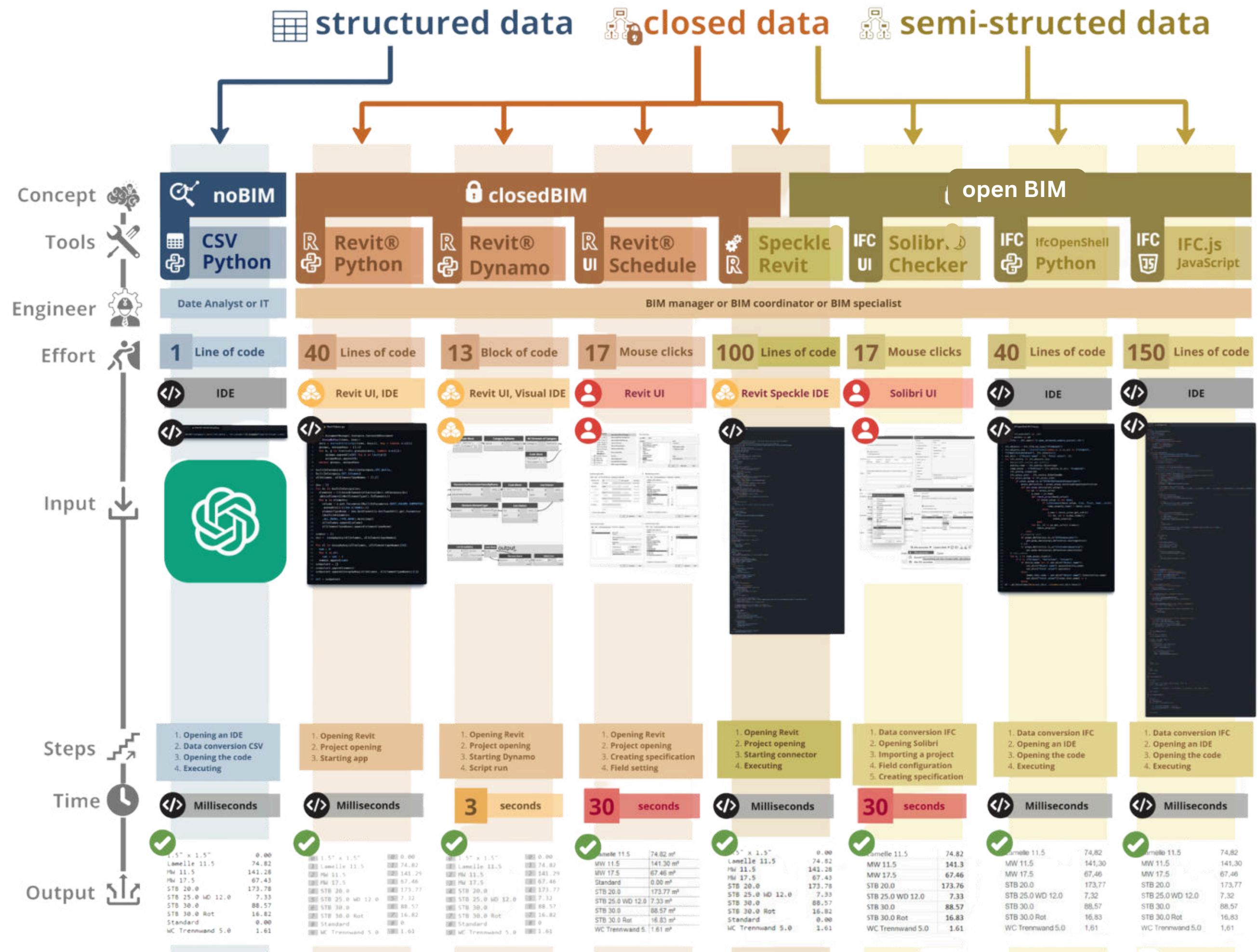
GET DATA FROM A MODEL

The popular case study "Quantitative Takeoff"



Structured data leads the way: simpler, faster, more efficient

data  **driven**
construction.io



**Nicolas Merot**

Ingénieur BIM | Caeli Ingénierie



DataDrivenConstruction products revolutionize data management in construction! Their IFC and RVT to Excel converters enable smooth data analysis and extraction, optimizing...

[Read more](#)**Daniel Glober**

BIM-Manager | SCHOLZE-THOST GmbH



Revit and IFC reports that used to take me almost weeks to create are now updated in just a few minutes. I was able to quickly understand what the DataDrivenConstruction did and thu...

[Read more](#)**Dmitri Garbuzenko**

BIM and AIM Coordinator | RB Rail AS



With the help of Python and especially the pandas library, as the DataDrivenConstruction team does, we are now able to perform delivery checks four times faster....

[Read more](#)**Prof. Dr.-Ing. Michael Bühler**

Co-Owner GemeinWerk Ventures



Be part of the movement with DataDrivenConstruction! Let's make true freedom in data formats a reality and catalyze a new era of productivity and innovation in construction....

[Read more](#)**Abdelrahim (Mohamed) Deghydy**

BIM Manager | Consolidated Contractors Company



DDC converter and Plugin is a fantastic and helpful tool for visualisation and quantification the meta data from Revit. Thanks for sharing such helpful tools!

**Jānis Dzenis**

BIM Coordinator | Merks, SIA



This is a fantastic tool, haven't seen one like this in a long time. In this era, we have countless tools and methods for creating models, drawings, tables, and other forms of data....

[Read more](#)**Valerio Spini**

Settore RVCS



Great experience: Until now, I used to open IFC files in Blocknote to check the parameters and their structure. Thanks to the DataDrivenConstruction converter I can check the parameter...

[Read more](#)**Irina Fischer**

BIM Coordinator | OBERMEYER Group



The decision to use Jupyter Notebook for results verification turned out to be highly beneficial. Our experience with solutions from Data Driven Construction and Jupyter Notebook...

[Read more](#)