

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

DATA CONVERSION TO OPEN FORMATS



```
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")
```

conversion in just 4
lines of code

data-driven
construction.io

RVT | IFC | DWG as DataFrame.py

```
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]
```

 Structured format is ideal for analytics, visualization and automation

two-dimensional
project data



AS

STRUCTURED
DATA 

Column names

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	Level 2
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

Columns axis = 1

Index label

Index axis = 0

Missing value

Data

data-driven
construction.io

Converter

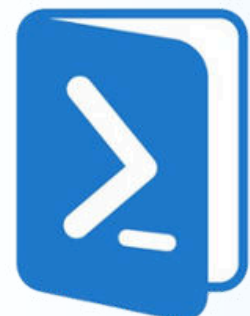
terminal version

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



Command Prompt

```
Command Prompt
C:\DDC\DDC_Converter> RvtExporter.exe D:\sample_basic.exe
```



PowerShell

```
Windows PowerShell
PS C:\DDC\DDC_Converter> RvtExporter.exe D:\sample_basic.rvt
```

 VS Code



kaggle

Google colab



eclipse



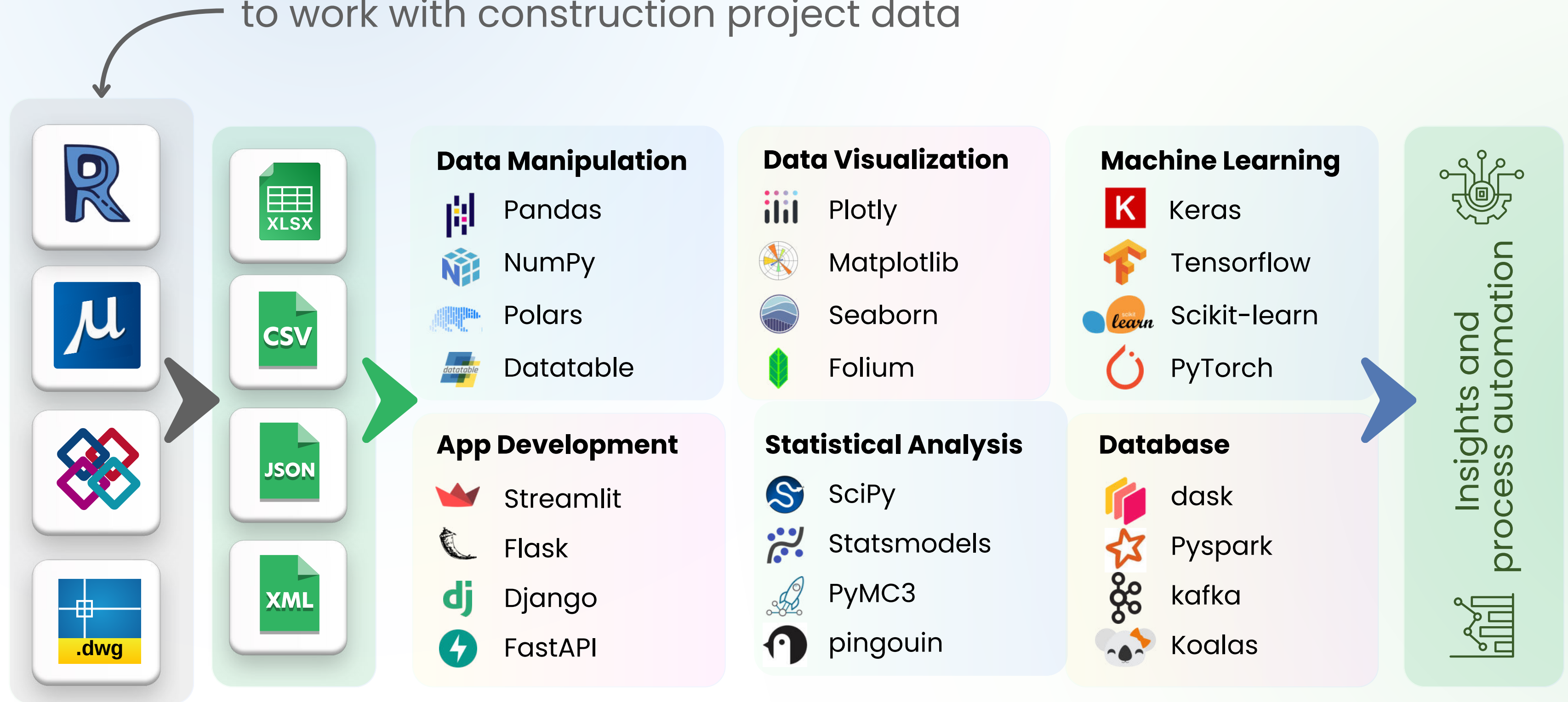
Azure Notebooks



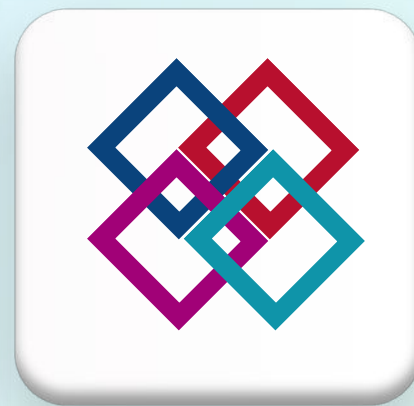
Life Is Short, Use Python

data^{driven}
construction.io

to work with construction project data

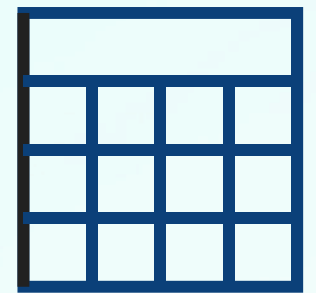


easy to learn, easy to develop



AS

STRUCTURED
DATA



Columns
axis = 1

Index
axis = 0

Index
label

Column names

Missing
value

Data

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	Level 2
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



IFC

STRUCTURED
DATA



ID	Name	Category	Version	Project	Site	Parent	ObjectType
34	0001	IfcProject	IFC2X3	0001	0001	0	0
38274	Default	IfcSite	IFC2X3	0001	Default	0	0
36	9	IfcBuilding	IFC2X3	0001	Default	0	0
39	Level 1	IfcBuildingStorey	IFC2X3	0001	Default	0	0
3797	Basic Wall: Exterior - Brick on Block: 138C IfcWallStandardC	IFC2X3	0001	Default	Level 1	0	0
3999	Basic Wall: Exterior - Brick on Block: 1381 IfcWallStandardC	IFC2X3	0001	Default	Level 1	0	0
4043	Basic Wall: Exterior - Brick on Block: 1382 IfcWallStandardC	IFC2X3	0001	Default	Level 1	0	0
4087	Basic Wall: Exterior - Brick on Block: 1383 IfcWallStandardC	IFC2X3	0001	Default	Level 1	0	0
4131	Basic Wall: Interior - Partition (92mm Stu IfcWallStandardC	IFC2X3	0001	Default	Level 1	0	0
4219	Basic Wall: Interior - Partition (92mm Stu IfcWallStandardC	IFC2X3	0001	Default	Level 1	0	0
4287	Basic Wall: Party Wall - CMU Residential IfcWallStandardC	IFC2X3	0001	Default	Level 1	0	0
4399	Basic Wall: Party Wall - CMU Residential IfcWallStandardC	IFC2X3	0001	Default	Level 1	0	0
4465	Basic Wall: Party Wall - CMU Residential IfcWallStandardC	IFC2X3	0001	Default	Level 1	0	0
4508	Basic Wall: Interior - Partition (92mm Stu IfcWallStandardC	IFC2X3	0001	Default	Level 1	0	0
4553	Basic Wall: Interior - Partition (92mm Stu IfcWallStandardC	IFC2X3	0001	Default	Level 1	0	0
4598	Basic Wall: Interior - Partition (92mm Stu IfcWallStandardC	IFC2X3	0001	Default	Level 1	0	0
5165	Floor: 127mm Slab on Grade: 141232	IfcSlab	IFC2X3	0001	Default	Level 1	0
5267	Floor: 127mm Slab on Grade: 143106	IfcSlab	IFC2X3	0001	Default	Level 1	0
5642	Basic Wall: Interior - Partition (92mm Stu IfcWallStandardC	IFC2X3	0001	Default	Level 1	0	0
5903	Basic Wall: Interior - Partition (92mm Stu IfcWallStandardC	IFC2X3	0001	Default	Level 1	0	0
6426	M_Fixed: 4835mm x 2420mm: 4835mm x IfcWindow	IFC2X3	0001	Default	Level 1	0	0
6531	M_Fixed: 4835mm x 2420mm: 4835mm x IfcWindow	IFC2X3	0001	Default	Level 1	0	0
6652	M_Single-Flush: 1250mm x 2010mm: 125 IfcDoor	IFC2X3	0001	Default	Level 1	0	0
6757	M_Single-Flush: 1250mm x 2010mm: 125 IfcDoor	IFC2X3	0001	Default	Level 1	0	0
6921	M_Fixed: 750mm x 2200mm: 750mm x 22 IfcWindow	IFC2X3	0001	Default	Level 1	0	0
7098	M_Fixed: 750mm x 2200mm: 750mm x 22 IfcWindow	IFC2X3	0001	Default	Level 1	0	0



RVT

STRUCTURED
DATA



ID	Name	Category	Design	IfcGUID	Type IfcGUID	Family and Type
198363	Window - PVC Coating - VOST_Materials	None	31x0gNe59vvExhby0Bf7			
198366	Single Window	OST_Windows	None	31x0gNe59vvExhby0Bf12		
198367	Basic Wall	OST_Walls	None	31x0gNe59vvExhby0Bf13		
198369	Finishes - Interior - Plaste OST_Materials	None	31x0gNe59vvExhby0Bf1z			
198370	Wood - Stud Layer	OST_Materials	None	31x0gNe59vvExhby0Bf1u		
198372	Structure - Timber Insulat OST_Materials	None	31x0gNe59vvExhby0Bf1v			
198373	Structure - Timber Insulat OST_Materials	None	31x0gNe59vvExhby0Bf1w			
198374	Finishes - Exterior - Timbu OST_Materials	None	31x0gNe59vvExhby0Bf1x			
198694	Basic Wall	OST_Walls	None	31x0gNe59vvExhby0Bf1w	38NbW5DL180JLvn67Ze	SIP 202mm Wall - cor
198749	Basic Wall	OST_Walls	None	31x0gNe59vvExhby0Bf1z	31x0gNe59vvExhby0Bf13	Wall - Timber Clad
211807	Sink-Offset-Kohler-Vault - OST_PlumbingFixts	None	283ISWDD8Ju0YHnzXOxNd			
211807	Sink-Offset-Kohler-Vault - OST_PlumbingFixts	None	283ISWDD8Ju0YHnzXOxV			
211850	Sink-Offset-Kohler-Vault - OST_PlumbingFixts	None	283ISWDD8Ju0YHnzXOxS7			
212929	Chrome-Kohler-CP-Polish OST_Materials	None	283ISWDD8Ju0YHnzXOxDC			
212930	Nickel-Kohler-VS-Vibrant OST_Materials	None	283ISWDD8Ju0YHnzXOxDF			
212931	Steel-Kohler-VS-Vibrant OST_Materials	None	283ISWDD8Ju0YHnzXOxDE			
212932	Metal-Kohler-BL-Matte OST_Materials	None	283ISWDD8Ju0YHnzXOxD9			
213558	Faucet-Binck_Reach-Kohl OST_PlumbingFixts	None	283ISWDD8Ju0YHnzXOmxw			
213811	Faucet-Binck_Reach-Kohl OST_PlumbingFixts	None	283ISWDD8Ju0YHnzXOm			
218358	Concrete - Cast-in-Place OST_Materials	None	283ISWDD8Ju0YHnzXOxX			
223462	Door - Frame	OST_Materials	None	283ISWDD8Ju0YHnzXOy1d		
232683	Door - Panel	OST_Materials	None	283ISWDD8Ju0YHnzXOy1c		
232754	Basic Wall	OST_Walls	None	283ISWDD8Ju0YHnzXOy6S		
232758	System Panel	OST_CurtainWallIP	None	283ISWDD8Ju0YHnzXOy6x		
232770	Rectangular Mullion	OST_CurtainWallIM	None	283ISWDD8Ju0YHnzXOy7F		
232780	Single-Flush	OST_Doors	None	283ISWDD8Ju0YHnzXOy71		
232827	Basic Wall	OST_Walls	None	283ISWDD8Ju0YHnzXOy7s		



DWG

STRUCTURED
DATA



ID	Description	Hand	Layer	Locked	Color	Max E	LineW	Backe	Min Extents	Max Extents
1185	<AcDbPolyline>	[4A1]	CL		[352.4 662.9 0.0]	ByLayer	klNwByLayer		[30.7 7.3 0.0]	[352.4 662.9 0.0]
1186	<AcDbPolyline>	[4A2]	ROW		[404.0 237.5 0.0]	ByLayer	klNwByLayer		[8.3 18.3 0.0]	[330.0 673.9 0.0]
1195	<AcDbPolyline>	[4A8]	PL		[421.9 167.5 0.0]	ByLayer	klNwByLayer		[70.9 -46.1 0.0]	[806.1 616.0 0.0]
1741	<AcDbBlockRefere	[6C0]	BUILDING		[424.8 307.5 0.0]	ByLayer	klNwByLayer		[364.0 167.5 0.0]	[404.0 237.5 0.0]
2057	<AcDbPolyline>	[809]	EASEMENT		[504.8 307.5 0.0]	ByLayer	klNwByLayer		[272.3 315.2 0.0]	[510.7 541.2 0.0]
2058	<AcDbPolyline>	[80A]	POND			ByLayer	klNwByLayer		[282.3 325.2 0.0]	[500.7 531.2 0.0]
2412	<AcDbLine>	[96C]	SETBACK			ByLayer	klNwByLayer		[346.1 167.5 0.0]	[421.9 167.5 0.0]
2422	<AcDbLine>	[976]	ROW			ByLayer	klNwByLayer		[148.6 190.8 0.0]	[374.9 651.9 0.0]
2423	<AcDbArc>	[977]	ROW			ByLayer	klNwByLayer		[145.5 147.5 0.0]	[175.5 190.8 0.0]
2433	<AcDbArc>	[981]	ROW			ByLayer	klNwByLayer		[89.8 70.8 0.0]	[116.7 87.5 0.0]
2434	<AcDbLine>	[982]	ROW			ByLayer	klNwByLayer		[53.2 -3.7 0.0]	[89.8 70.8 0.0]
2711	<AcDbLine>	[A97]	CL			ByLayer	klNwByLayer		[84.8 117.5 0.0]	[84.8 117.5 0.0]
3077	<AcDbLine>	[C05]	LOT			ByLayer	klNwByLayer		[344.8 147.5 0.0]	[344.8 307.5 0.0]
3078	<AcDbLine>	[C06]	LOT			ByLayer	klNwByLayer		[264.8 147.5 0.0]	[264.8 307.5 0.0]
3079	<AcDbLine>	[C07]	LOT			ByLayer	klNwByLayer		[424.8 147.5 0.0]	[424.8 307.5 0.0]
3080	<AcDbLine>	[C08]	LOT			ByLayer	klNwByLayer		[504.8 147.5 0.0]	[504.8 307.5 0.0]
3082	<AcDbLine>	[C0A]	LOT			ByLayer	klNwByLayer		[264.8 307.5 0.0]	[344.8 307.5 0.0]
3099	<AcDbLine>	[C18]	EASEMENT			ByLayer	klNwByLayer		[352.3 147.5 0.0]	[352.3 307.1 0.0]
3100	<AcDbLine>	[C1C]	EASEMENT			ByLayer	klNwByLayer		[337.3 147.5 0.0]	[337.3 307.1 0.0]
3101	<AcDbLine>	[C1D]	ROW			ByLayer	klNwByLayer		[175.5 147.5 0.0]	[592.5 147.5 0.0]
3102	<AcDbLine>	[C1E]	ROW			ByLayer	klNwByLayer		[116.7 87.5 0.0]	[592.5 87.5 0.0]
3122	<AcDbRotatedDim	[C32]	*ADSK_CONSTRAINTS			ByLayer	klNwByBlock			
3142	<AcDbLine>	[C46]	EASEMENT			ByLayer	klNwByLayer		[158.9 152.5 0.0]	[592.5 152.5 0.0]
3143	<AcDbLine>	[C47]	EASEMENT			ByLayer	klNwByLayer		[100.2 82.5 0.0]	[592.5 82.5 0.0]
3144	<AcDbRotatedDim	[C48]	*ADSK_CONSTRAINTS			ByLayer	klNwByBlock			



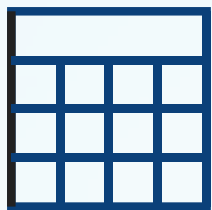
DGN

STRUCTURED
DATA



ID	Name	Category	Design	IfcGUID	Type IfcGUID	Family and Type
198363	Window - PVC Coating - VOST_Materials	None	31x0gNe59vvExhby0Bf7			
198366	Single Window	OST_Windows	None	31x0gNe59vvExhby0Bf12		
198367	Basic Wall	OST_Walls	None	31x0gNe59vvExhby0Bf13		
198369	Finishes - Interior - Plaste OST_Materials	None	31x0gNe59vvExhby0Bf1z			
198370	Wood - Stud Layer	OST_Materials	None	31x0gNe59vvExhby0Bf1u		
198372	Structure - Timber Insulat OST_Materials	None	31x0gNe59vvExhby0Bf1v			
198373	Structure - Timber Insulat OST_Materials	None	31x0gNe59vvExhby0Bf1w			
198374	Finishes - Exterior - Timbu OST_Materials	None	31x0gNe59vvExhby0Bf1x			
198694	Basic Wall	OST_Walls	None	31x0gNe59vvExhby0Bf1w	38NbW5DL180JLvn67Ze	SIP 202mm Wall - cor
198749	Basic Wall	OST_Walls	None	31x0gNe59vvExhby0Bf1z	31x0gNe59vvExhby0Bf13	Wall - Timber Clad
211806	Steel-Kohler-NA-Stainless OST_Materials	None	283ISWDD8Ju0YHnzXOxNd			
211807	Sink-Offset-Kohler-Vault - OST_PlumbingFixts	None	283ISWDD8Ju0YHnzXOxV			
211850	Sink-Offset-Kohler-Vault - OST_PlumbingFixts	None	283ISWDD8Ju0YHnzXOxS7			
212929	Chrome-Kohler-CP-Polish OST_Materials	None	283ISWDD8Ju0YHnzXOxDC			
212930	Nickel-Kohler-VS-Vibrant OST_Materials	None	283ISWDD8Ju0YHnzXOxDF			
212931	Steel-Kohler-VS-Vibrant OST_Materials	None	283ISWDD8Ju0YHnzXOxDE			
212932	Metal-Kohler-BL-Matte OST_Materials	None	283ISWDD8Ju0YHnzXOxD9			
213558	Faucet-Binck_Reach-Kohl OST_PlumbingFixts	None	283ISWDD8Ju0YHnzXOmxw			
213811	Faucet-Binck_Reach-Kohl OST_PlumbingFixts	None	283ISWDD8Ju0YHnzXOm			
218358	Concrete - Cast-in-Place OST_Materials	None	283ISWDD8Ju0YHnzXOxX			
223462	Door - Frame	OST_Materials	None	283ISWDD8Ju0YHnzXOy1d		
232683	Door - Panel	OST_Materials	None	283ISWDD8Ju0YHnzXOy1c		
232754	Basic Wall	OST_Walls	None	283ISWDD8Ju0YHnzXOy6S		
232758	System Panel	OST_CurtainWallIP	None	283ISWDD8Ju0YHnzXOy6x		
232770	Rectangular Mullion	OST_CurtainWallIM	None	283ISWDD8Ju0YHnzXOy7F		
232780	Single-Flush	OST_Doors	None	283ISWDD8Ju0YHnzXOy71		
232827	Basic Wall	OST_Walls	None	283ISWDD8Ju0YHnzXOy7s		

STRUCTURED
DATA



Unnamed: 0	Unnamed: 0.1	Filename	IfcEntity	UniqueID	Ifc version	GlobalId	OwnerHistory	ObjectPlacement	Representation	...	cpIFitMatchKey	Product code	ISOCD3766ShapeCode	ISOCD3766ShapeParameter_b
0	0	1000	beams_ifc	Odffc4:IfcBeamStandardCase	1000.0	IFC4	0juf4qygg5l8rxA20Qwnsj	0.0	1001.0	1010.0	...	NaN	NaN	NaN
1	1	1100	beams_ifc	Odffc4:IfcBeamStandardCase	1100.0	IFC4	0juf4qygg5l8rxA20sznsj	0.0	1101.0	1110.0	...	NaN	NaN	NaN
2	2	1200	beams_ifc	Odffc4:IfcBeamStandardCase	1200.0	IFC4	0juf4qygg5l8s4A20sznsj	0.0	1201.0	1210.0	...	NaN	NaN	NaN
3	3	1300	beams_ifc	Odffc4:IfcBeamStandardCase	1300.0	IFC4	0juf4qygg5l8s4A20sznw6	0.0	1301.0	1310.0	...	NaN	NaN	NaN
4	4	1400	beams_ifc	Odffc4:IfcBeamStandardCase	1400.0	IFC4	0juf4qygg5l8rxA20sznw6	0.0	1401.0	1410.0	...	NaN	NaN	NaN



Excel



PowerBI



Sheets



Google Colab



Python



Kaggle

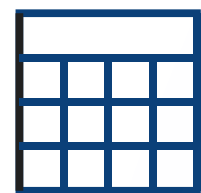


Pandas

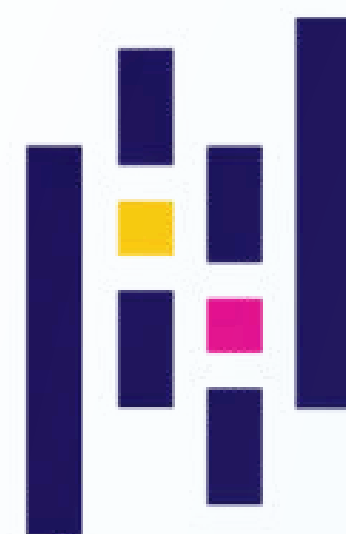


ChatGPT

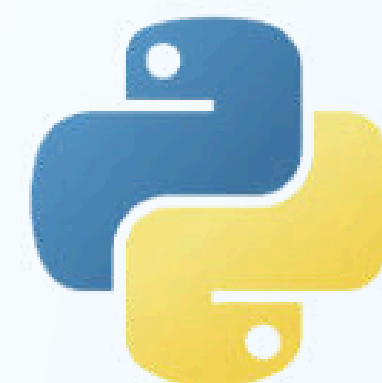
STRUCTURED
DATA



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



pandas



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

snappily.io

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

snappily.io

GROUP

Input

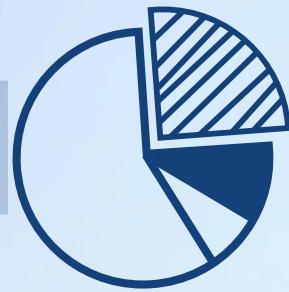
```
GroupBy Revit IFC.py

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

Output

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

snappily.io

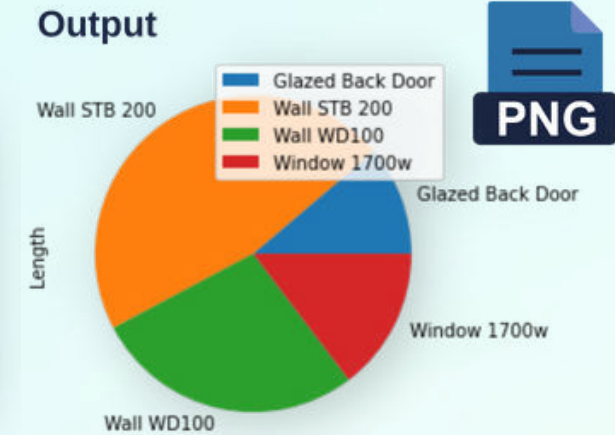


PIE chart

Input

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

Output

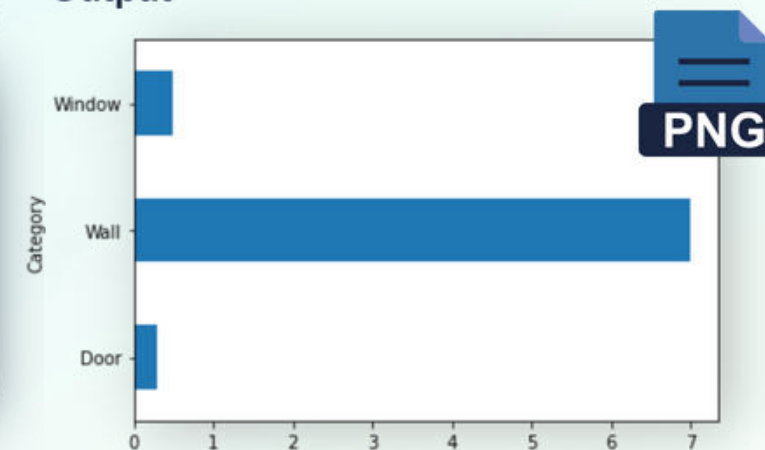


BAR chart

Input

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

Output



Regular Expression

Input

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

Output

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



-	-	-
-	-	-
-	-	-



QTO TakeOff

Input

```
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()
```

snappify.io

Output

	Volume	Length
Category		
Wall	7.0	8600

EXCEL Data Export

Input

```
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')
```

snappify.io

Output

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

Category_estimate

PDF Document

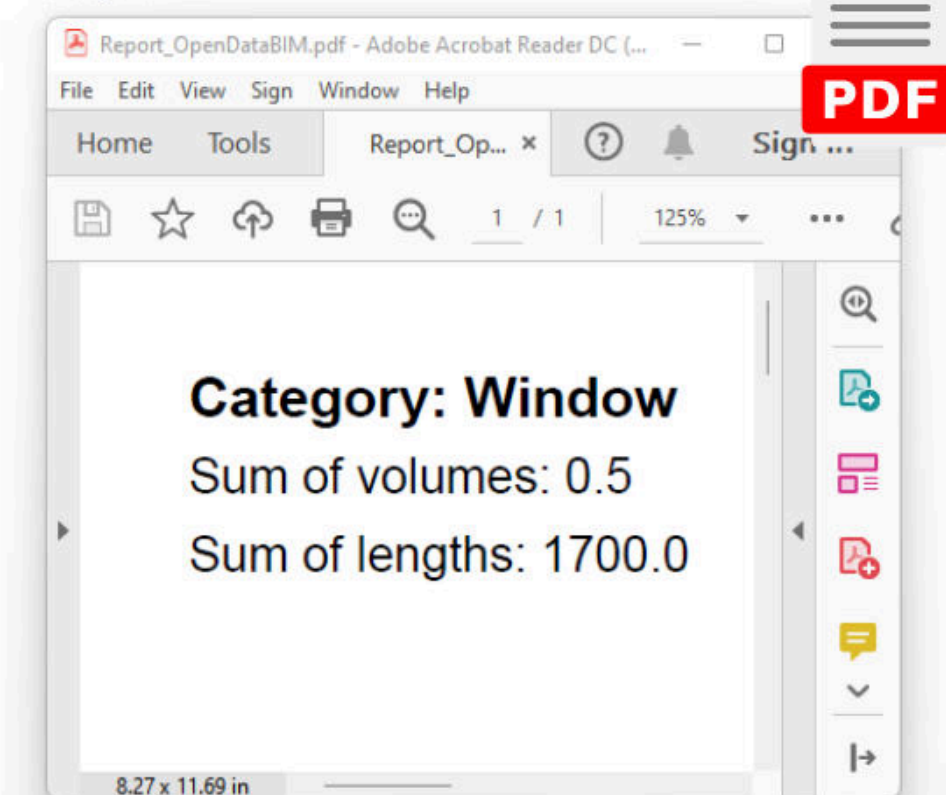
Input

```
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')
```

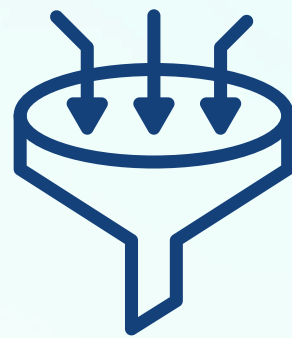
snappify.io

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

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

GROUP



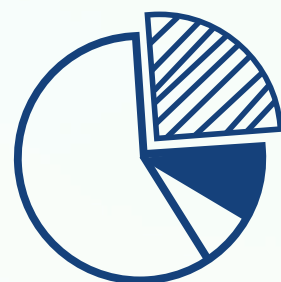
```
GroupBy Revit IFC.py

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

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

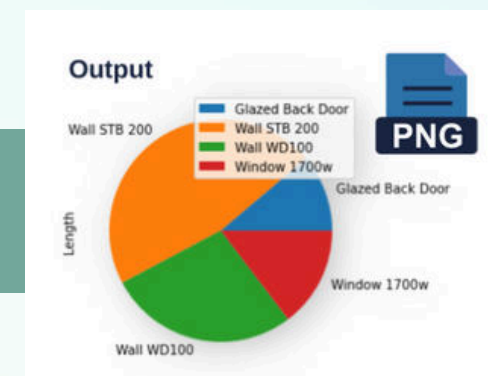
Group the project by the "Type Name" parameter and show the volume of each group

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')
```



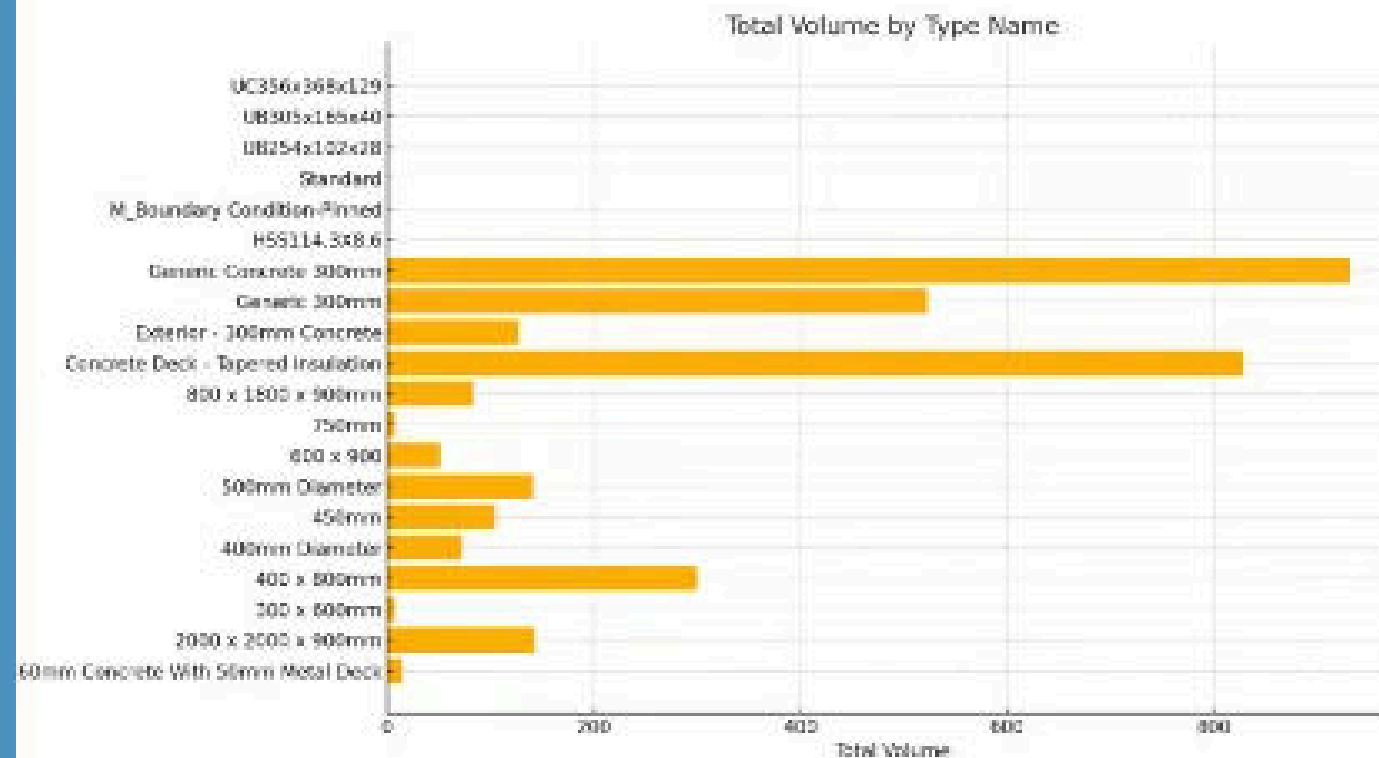
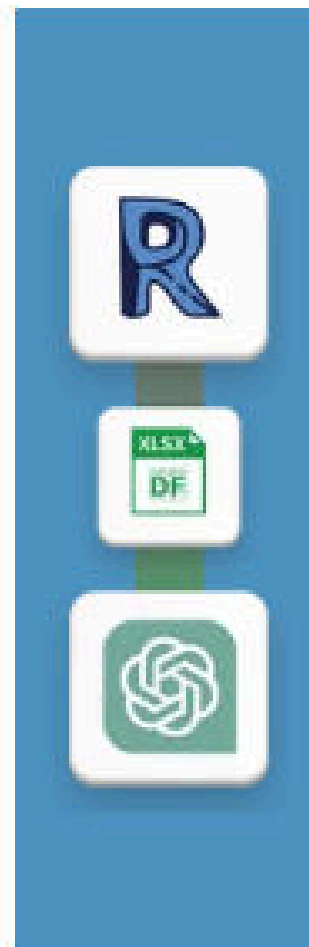
Choose the first 20 types by volume and show the result as a Pie chart



Create a PDF report with a table and a graph

Quick QTO with graph from Revit

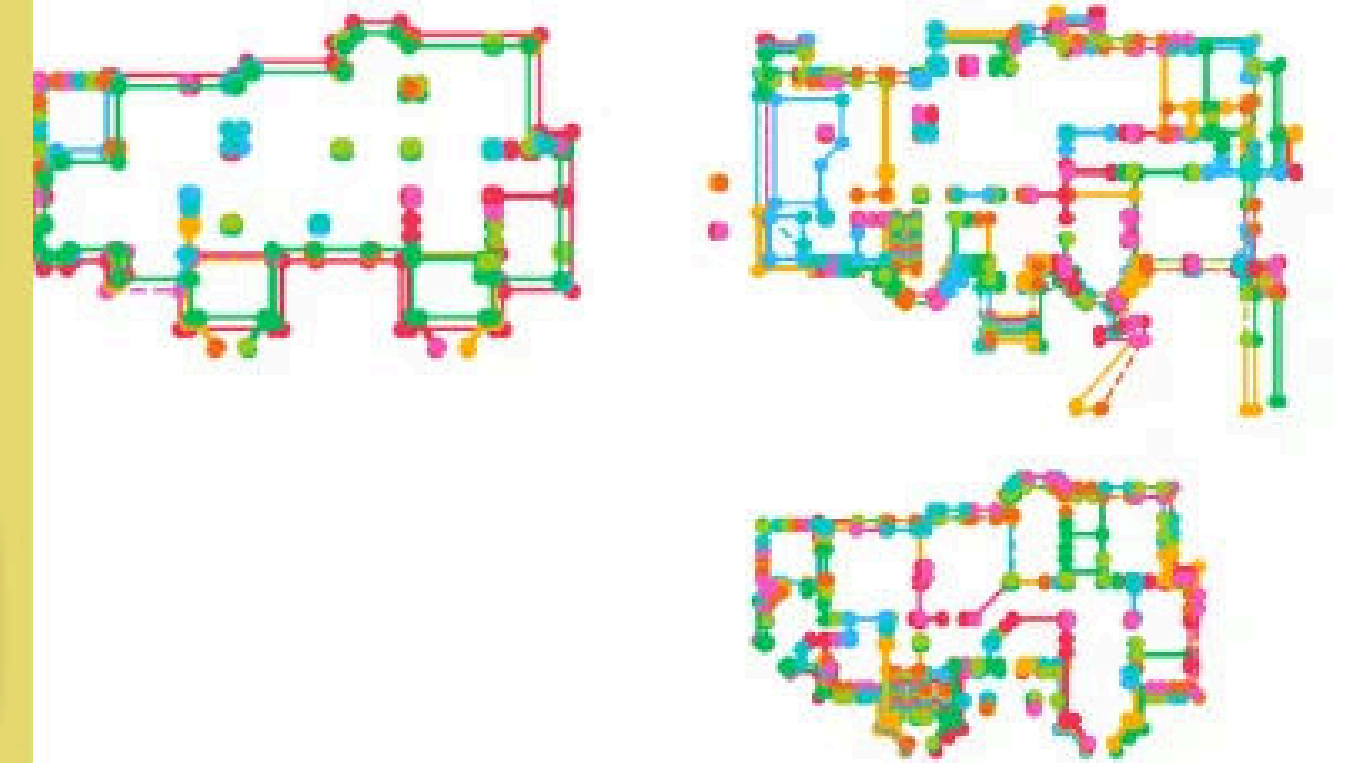
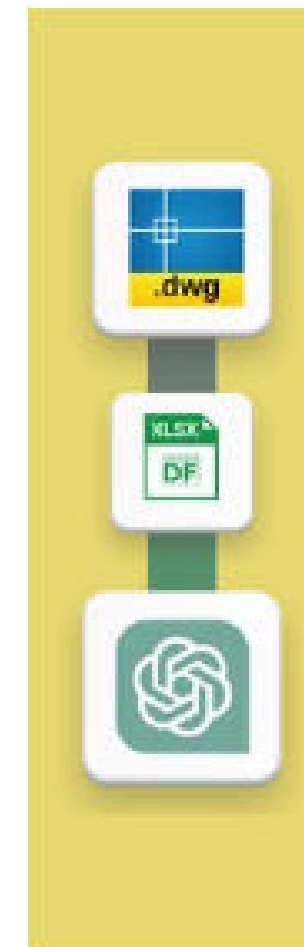
2022_rst_advancedsampleproject.rvt



Group the data in Dataframe by "Type Name" while summarizing the "Volume" parameter and show the number of items in the group. And show it all as a horizontal bar chart without zero values

Plot Polylines from DWG

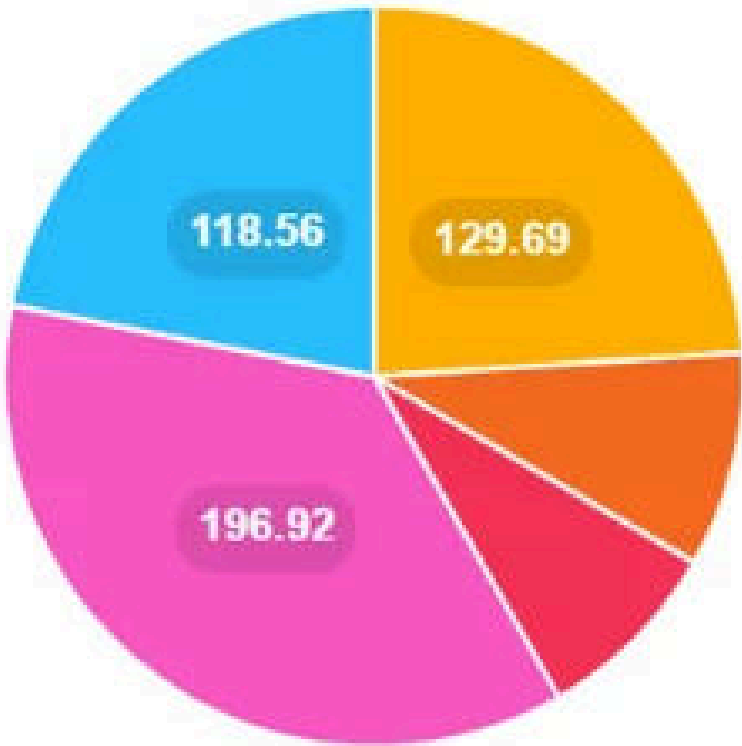
family_house_florida.dwg



Find ids from column "Layer" with value "wall". Get this IDs and find in "ParentID". Then take for each group with "ParentID" column "Point" - x,y,z from each line. Plot separate polylines for each group based on "ParentID" and connects first and last points. Plot all lines with matplotlib without legend

Quick QTO with graph from Revit

Ifc2x3_Duplex_Architecture.ifc



Grouped Wall Data With Area from Revit

2023_rac_basic_sample_project.rvt



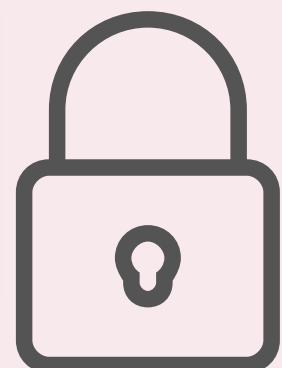
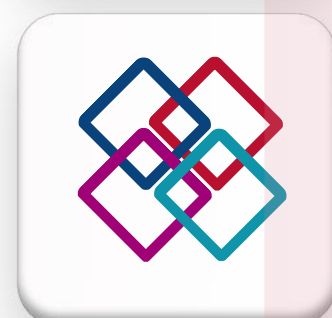
Type Name	Total Area	Count
CL_W1	393.12 sq m	10
Cavity wall_sliders	9.37 sq m	1
Foundation - 300mm Concrete	30.90 sq m	1
Interior - 165 Partition (1-hr)	17.25 sq m	3
Interior - Partition	186.54 sq m	14
Retaining - 300mm Concrete	195.79 sq m	10
SH_Curtain wall	159.42 sq m	9
SIP 202mm Wall - conc clad	114.76 sq m	4
Wall - Timber Clad	162.91 sq m	8



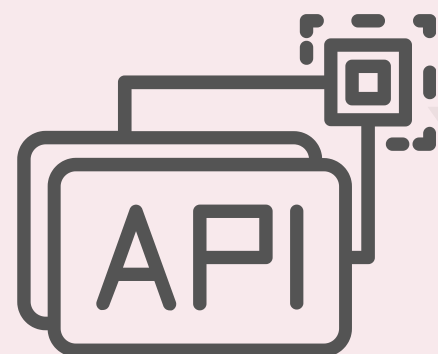
Take only the items that have Level 1 and Level 2 values in the "Parent" parameter and take the items that have IfcSlab values in the "Category" parameter, then group these items by the "ObjectType" parameter and sum the values in the "PSet_Revit_Dimensions Area" parameter and show them as a pie chart



Take only the items that have "OST_Walls" in the "Category" parameter, group them by "Type Name", sum the value of the "Area" column and add the quantity and show them in a table by removing zero values.



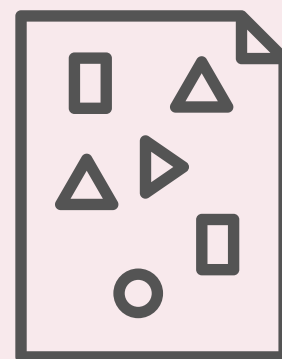
closed
data



closed
api

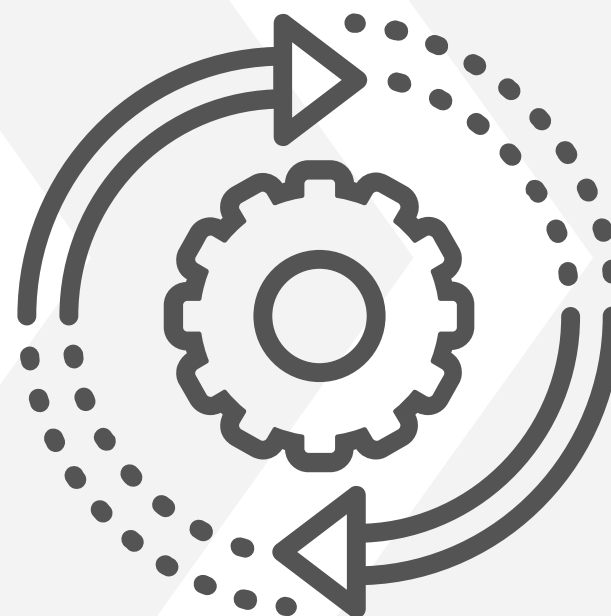


api
cloud

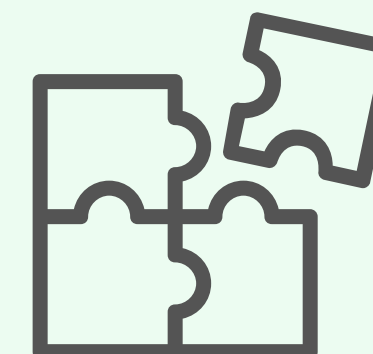


unstructured
data

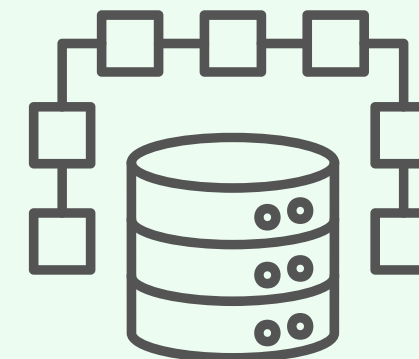
transformation



open
data



common
format



structured
data



open
source





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

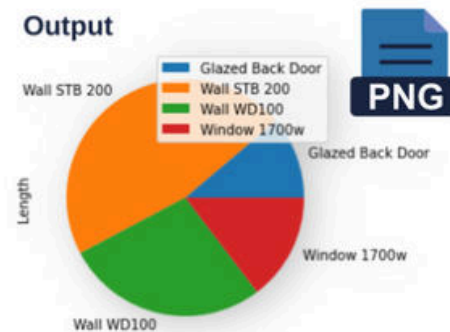
Output

	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

Output



PDF

chatGPT
LLmA, Alpaca

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

Processing

Automated
Workflow

without data processing

post-processed data

EXTRACT

TRANSFORM

LOAD

Selected Project

Company Projects That Are on Servers

Company Projects That Are on Servers

The Data Handling Process

Automated Workflow script

9:00

1
Running a
CAD program

CAD

2
Attribute check
Running the plugin
Setting up the
output

Attribute
Check Report

3
Revision Check
Start
Revision App
Filling in the
settings

4
Storing data
in folders
Revision
Check Report

5
Sending
messages about
created files
Report Creation
Messages

Getting ready for
the next project

17:00

Generated Data and Reports for Project 1

Datamanager
Project Manager
Designer

Mo., Tu.,
We., Th., Fr.
19:00

1
File
Collection

Report on
Collected Files

2
Checking
Changes

data-driven
construction.io

3
Geometric
Collisions
Geometric
Collisions Report

4
Data
Conversion

5
Converting to
Other Formats

Parameters
Geometry
CSV
DAE
IDs

6
Revision
Check

7
Checking the
Correctness of
the Attributes
Attribute
Check Report

05:00

Report Creation
Messages

Automatically Generated Data and Reports for Project 1-X

Datamanager
Project Manager
Designer