



Querying NCREIF Property Database

Joint Research and Performance Measurement Committee Presentation

April 9, 2025

Jeff Fisher

Accessing NCREIF Property Data

NCREIF Property Database

Legacy Web Service (API)

New API

Query Tool

Client Programs

- Python, R, C# etc.
- Excel & Power Query
- Microsoft Power BI
- User's Database
- Etc.

- Property data only
 - Frozen Classic
 - Classic Research
 - Expanded
- Not fund returns (NFI)
- Not PDF files

- SOAP Based
- Will eventually be bypassed

- Client program on NCREIF Website
- Built in queries
- Advanced mode
- XML files to save and upload queries

- REST Based
- Much easier to access with client programs
- Added security
- Several "endpoints" to accommodate different ways of inputting and extracting data.

- No need to log in to website
- Pull data into your application
- Must still get authenticated using NCREIF Username and Password

Building a Data Warehouse?



Like the Query Tool, you must define the filter criteria (where parameter) **before** requesting data.

The data still must be aggregated to meet the masking criteria.

Therefore, it needs any appropriate filter criteria before it is aggregated and downloaded. E.g., should the partial sale quarter be included or not for a property?

Recommended steps:

1. Decide what data you want and the appropriate filter criteria.
2. Download into your application or data warehouse.

Downloading all data and then querying from your data warehouse is not likely to work.

Query Parameters (AKA “Payload”)

p_SelectQuery – e.g., Sum (NOI) / Sum (Denom) as ‘IncomeReturn’*

p_WhereClause – e.g., [NPI_Plus] = 1 AND [PropertyType] = ‘Office’

p_GroupbyClause – e.g., [YYYYQ],[CBSAorDiv] (Default is YYYYQ if left out. Do not put in Select if in Groupby)

p_DataTypeId – e.g., 1 for Classic Research Database; 2 for Classic Research; 3 for Expanded NPI Database

p_QueryData – e.g., 0 for all managers (default if left out); 1 for “My Manager”; 2 for “All but My Manager”

Optional:

KPI – e.g., “CapRates” or “PercentLease”. Optional. Automatically uses correct Select and Where parameters like the query tool. Other KPIs currently available are “NPIClassic” and “NPIExpanded” with more to come.

*Avoid double quotes, e.g., as “IncomeReturn”. Also avoid spaces such as ‘Income Return’ or you will get ‘Income%20Return’ back from the API.

Classic or Expanded; Frozen or Not!

Database Properties selected

Parameter	Fields in Where Parameter			
DataTypeID	NPI	NPI_Plus	Result	PropertyType Field Names
1	1		Classic Research	A, I, O, R, etc.
2	1		Classic Frozen	A, I, O, R, etc.
3	1		Classic Research	Residential, Industrial, etc.*
3		1	Expanded	Residential, Industrial, etc.*
*Old names available in PropertyType_Old and PropertySubType_Old fields with DataTypeID = 3				

- Property subtype names are also different between the different databases.
- The old property type names are available in the expanded database if needed.

Endpoints

(Different URLs depending on how you want to request the data and get the results.)

- Execute Query from JSON Parameters and Return XML Object
- Execute Query from XML String and Return XML Object
- Execute Query from JSON Parameters and Return JSON Object
- Execute Query from JSON Parameters and Return an Excel Spreadsheet
- Execute Query from XML String and Return JSON Object
- Execute Query from Uploaded XML File (Compatible with Query Tool) and Return JSON Object

Sample Endpoint

URL: <https://qt-api.ncreif.org/QT/ExecuteQuery>

Sample Request:

```
{
  "p_DataTypeId":1,
  "p_SelectQuery": "Sum(NOI) / Sum(Denom) as 'Income Return', Count(Denom) as 'Props' etc. ",
  "p_WhereClause": "NPI_Plus = 1",
  "p_GroupbyClause": "Period, YYYYQ, Year, Quarter",
  "kpi": "",
  "p_QueryData": 0
}
```

Parameters
encapsulated in a
JSON object.

Sample Response

```
<NewDataSet>
  <Result1>
    <Period>2</Period>
    <YYYYQ>19781</YYYYQ>
    <Year>1978</Year>
    <Quarter>1</Quarter>
    <IncomeReturn>0.0218</IncomeReturn>
    <CapitalReturn>0.0073</CapitalReturn>
    <TotalReturn>0.029</TotalReturn>
    <Props>233</Props>
  </Result1>
  ...
</NewDataSet>
```

XML file that many
programs recognize,
e.g., it can be imported
into Excel as a table.

Query Throttle

6:00 am to 6 pm Central Time

8 queries per minute

Other times

30 queries per minute

Batch Query Example Using Power Query OR VBA in Excel

	A	B	C	D	E	F	G	H	I	
1			Query Parameters Table							
2			QueryName	p_SelectQuery	p_WhereClause	p_DataTypeId	p_GroupbyClause	p_QueryData	KPI	
3			ClassicNPI	SUM(NOI) AS NOI, SUM(CapEx) AS CapEx, SUM(MV) AS MV, SUM(MVLag1) AS MVLag1, SUM(PSales) AS PSales, SUM(Denom) AS Denom, SUM(NOI) / SUM(Denom) AS 'Income_Return', (SUM(MV) - SUM(MVLag1) - SUM(CapEx) + SUM(PSALES)) / SUM(Denom) AS 'Capital_Return', (SUM(NOI) + SUM(MV) - SUM(MVLag1) -	NPI=1		3 Year, YYYYQ		0	
4			ExpandedNPI	SUM(MVLag1) AS MVLag1, SUM(PSales) AS PSales, SUM(Denom) AS Denom, SUM(NOI) / SUM(Denom) AS 'Income_Return', (SUM(MV) - SUM(MVLag1) - SUM(CapEx) + SUM(PSALES)) / SUM(Denom) AS 'Capital_Return', (SUM(NOI) + SUM(MV) - SUM(MVLag1) - SUM(CapEx) + SUM(PSALES)) / SUM(Denom) AS 'Total_Return',	NPI_Plus=1		3 Year, YYYYQ		0	
5			ODCE_Props	SUM(NOI) AS NOI, SUM(CapEx) AS CapEx, SUM(MV) AS MV, SUM(MVLag1) AS MVLag1, SUM(PSales) AS PSales, SUM(Denom) AS Denom, SUM(NOI) / SUM(Denom) AS 'Income_Return', (SUM(MV) - SUM(MVLag1) - SUM(CapEx) + SUM(PSALES)) / SUM(Denom) AS	NPI_Plus=1 AND FundType = 'D'		3 Year, YYYYQ		0	
6			Percent_Leased	Avg(PercentLeased) AS Occupancy, Count(PercentLeased) AS Props	PercentLeased is not Null and NPI_Plus = 1		3 Year, YYYYQ		0	
7			Cap_Rates	Count(AppCapRate) AS CountOfAppCapRate, Avg(AppCapRate) AS	NPI_Plus = 1		3 Year, YYYYQ		0	
					NPI_Plus = 1 AND					

QueryParameters | Classic NPI | Expanded NPI | ODCE Props | Percent Leased | Appraisal Cap Rates

Ready Accessibility: Investigate

Clipboard: Paste, Cut, Copy, Format Painter

Font: Calibri, 11, Bold, Italic, Underline, Text Color, Background Color

Alignment: Center, Left, Right, Indent, Wrap Text, Merge & Center

Number: General, Currency, Percentage, Thousand Separator, Decimal Places

Styles: Conditional Formatting, Format as Table, Cell Styles

Cells: Insert, Delete

D9 : 25295067

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	QueryName	Year	YYYYQ	NOI	CapEx	MV	MVLag1	PSales	Denom	Income_Return	Capital_Return	Total_Return	Prop_Count
2	ClassicNPI	1978	19781	12460361	1045924	580990824	575792295	0	572161803.3	0.021777688	0.007257746	0.029035434	233
3	ClassicNPI	1978	19782	13549340	3512157	618339141	609771269	0	607010900.8	0.022321411	0.00832887	0.030650282	259
4	ClassicNPI	1978	19783	14311574	1283255	688675929	678802395	0	674673497.8	0.021212593	0.012732498	0.033945091	283
5	ClassicNPI	1978	19784	14228940	2744192	729986518	700435921	0	697065037	0.020412643	0.038456103	0.058868747	291
6	ClassicNPI	1979	19791	16853367	2865251	817048986	800677850	0	796492686.5	0.021159475	0.016956697	0.038116172	309
7	ClassicNPI	1979	19792	20769432	1935049	933456479	913108775	0	907153155.5	0.022895177	0.020297185	0.043192361	331
8	ClassicNPI	1979	19793	21710589	3759226	1070684326	1040414815	956820	1034579155	0.020984947	0.026549061	0.047534008	356
9	ClassicNPI	1979	19794	25295067	8943206	1231133627	1176664071	1676393	1171865789	0.021585294	0.040279991	0.061865284	370
10	ClassicNPI	1980	19801	28406387	4335061	1430115931	1378295675	0	1370994410	0.02071955	0.034635586	0.055355136	405
11	ClassicNPI	1980	19802	32426249	7085424	1582310228	1570794466	0	1563528428	0.020739149	0.002833551	0.0235727	428
12	ClassicNPI	1980	19803	34872795	9615441	1775851147	1735900359	247327	1728960151	0.020169808	0.017688478	0.037858287	465
13	ClassicNPI	1980	19804	37664990	8149203	1976014374	1904991793	286173	1896368311	0.019861643	0.03330553	0.053167172	493
14	ClassicNPI	1981	19811	43628932	11446531	2290643221	2257880610	1532335	2248294731	0.019405344	0.010162553	0.029567897	526

	A	B	C	D
1				Query Parameters Table
2			QueryName	p_SelectQuery
3			ClassicNPI	SUM(NOI) AS NOI, SUM(CapEx) AS CapEx, SUM(MV) AS MV, SUM(MVLag1) AS MVI PSales, SUM(Denom) AS Denom, SUM(NOI) / SUM(Denom) AS 'Income_Return', (S SUM(CapEx) + SUM(PSALES)) / SUM(Denom) AS 'Capital_Return', (SUM(NOI) + SUI SUM(CapEx) + SUM(PSALES)) / SUM(Denom) AS 'Total_Return', COUNT(MV) AS 'Pr
4			ExpandedNPI	SUM(NOI) AS NOI, SUM(CapEx) AS CapEx, SUM(MV) AS MV, SUM(MVLag1) AS MVI PSales, SUM(Denom) AS Denom, SUM(NOI) / SUM(Denom) AS 'Income_Return', (S SUM(CapEx) + SUM(PSALES)) / SUM(Denom) AS 'Capital_Return', (SUM(NOI) + SUI SUM(CapEx) + SUM(PSALES)) / SUM(Denom) AS 'Total_Return', COUNT(MV) AS 'Pr
5			ODCE_Props	SUM(NOI) AS NOI, SUM(CapEx) AS CapEx, SUM(MV) AS MV, SUM(MVLag1) AS MVI PSales, SUM(Denom) AS Denom, SUM(NOI) / SUM(Denom) AS 'Income_Return', (S SUM(CapEx) + SUM(PSALES)) / SUM(Denom) AS 'Capital_Return', (SUM(NOI) + SUI SUM(CapEx) + SUM(PSALES)) / SUM(Denom) AS 'Total_Return', COUNT(MV) AS 'Pr
6			Percent_Leased	Avg(PercentLeased) AS Occupancy, Count(PercentLeased) AS Props
7			Cap_Rates	Count(AppCapRate) AS CountOfAppCapRate, Avg(AppCapRate) AS CapRate
8			NOI_Growth	Sum(NOI) AS NOI, Sum(NOILag1) AS NOILag1, Sum(NOI) / Sum(NOILag1) - 1 as 'N 'Prop_Count'

Queries & Connections

Queries | Connections
✕

17 queries

- ⌘ Classic NPI
188 rows loaded.
- ⌘ Expanded NPI
188 rows loaded.
- ⌘ ODCE Props
188 rows loaded.
- ⌘ Percent Leased
167 rows loaded.
- ⌘ Appraisal Cap Rates
188 rows loaded.
- ⌘ NOI Growth
187 rows loaded.
- ⌘ Leveraged Returns
168 rows loaded.
- ⌘ MVI Value Weighted
188 rows loaded.

Adding Usage or Clusters to GroupbyClause

Query Parameters Table						
QueryName	p_SelectQuery	p_WhereClause	p_DataType	p_GroupbyClause	p_QueryData	KPI
Retail Usage	SUM(CapEx) AS CapEx, SUM(MV) AS MV, SUM(MVLag1) AS MVLag1, SUM(PSales) AS PSales, SUM(Denom) AS Denom, COUNT(MV) AS 'Prop_Count', AVG(PercentLeased) as PercentLeased	NPI_Plus=1 AND [PropertyType]='Retail'		[Period],[YYYYQ],[Year],[Quarter],[Usage]		0
Residential Clusters	SUM(CapEx) AS CapEx, SUM(MV) AS MV, SUM(MVLag1) AS MVLag1, SUM(PSales) AS PSales, SUM(Denom) AS Denom, COUNT(MV) AS 'Prop_Count', AVG(PercentLeased) as PercentLeased	NPI_Plus=1 AND [PropertyType]='Residential'		[Period],[YYYYQ],[Year],[Quarter],[PropertySubType],[clusters]		0

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	QueryName	Period	YYYYQ	Year	Quarter	PropertySubType	Clusters	CapEx	MV	MVLag1	PSales	Denom	Prop_Count	PercentLeased
2	Residential Clusters	43	19882	1988	2	Residential: Apartment	Garden	114124	113776541	114004810	0	113403224	7	0.95
3	Residential Clusters	44	19883	1988	3	Residential: Apartment	Garden	407462	178424336	175656704	0	174924829.7	11	0.955555556
4	Residential Clusters	45	19884	1988	4	Residential: Apartment	Garden	2387379	207039748	211150336	0	211160384.2	14	0.91
5	Residential Clusters	46	19891	1989	1	Residential: Apartment	Garden	347027	248831620	244005248	0	242788207.8	16	0.907333333
6	Residential Clusters	46	19891	1989	1	Residential: Apartment	Low-Rise	-462306	86130739	86101302	0	85392004	4	0.943333333
7	Residential Clusters	47	19892	1989	2	Residential: Apartment	Garden	853844	436555289	435112480	0	433201772.3	28	0.913703704

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	QueryName	Period	YYYYQ	Year	Quarter	Usage	CapEx	MV	MVLag1	PSales	Denom	Prop_Count	PercentLeased
2	Retail Usage	72	19953	1995	3	Retail: Not High-End with Grocer	-14791	111462843	111147843	0	110411234.8	5	0.98
3	Retail Usage	73	19954	1995	4	Retail: Not High-End with Grocer	72059	109668300	111462843	0	110790403.8	5	1
4	Retail Usage	74	19961	1996	1	Retail: Not High-End with Grocer	34162	108575949	109668300	0	109022428.3	5	0.97
5	Retail Usage	74	19961	1996	1	Retail: Not High-End without Grocer	-208246	42902092	42882772	0	42454964	3	0.95
6	Retail Usage	75	19962	1996	2	Retail: Not High-End with Grocer	153917	160779203	160855186	0	159640580.8	8	0.9875
7	Retail Usage	75	19962	1996	2	Retail: Not High-End without Grocer	163911	73709354	73166849	0	72686989.5	4	0.96
8	Retail Usage	76	19963	1996	3	Retail: Not High-End with Grocer	71204	166937798	166109203	0	165125019.3	9	0.98
9	Retail Usage	76	19963	1996	3	Retail: Not High-End without Grocer	32685	73742039	73709354	0	73154031.5	4	0.975

Expanded NPI

Display Options



```
let
// Step 1: Load other parameters from the Excel table
ParameterTable = Excel.CurrentWorkbook(){[Name="QueryParameters"]}[Content],

// Step 2: Filter the row based on the Query Name (modify the name per query)
QueryName = "ExpandedNPI",
// Change this to match the query's row in the table
CurrentParams = Table.SelectRows(ParameterTable, each [QueryName] = QueryName),

// Step 3: Extract parameters from the selected row
p_SelectQuery = CurrentParams{0}[p_SelectQuery],
p_WhereClause = CurrentParams{0}[p_WhereClause],
p_DataTypeId = CurrentParams{0}[p_DataTypeId],
p_GroupbyClause = CurrentParams{0}[p_GroupbyClause],
p_QueryData = CurrentParams{0}[p_QueryData],
KPI_Param = CurrentParams{0}[KPI],

// Step 4: Define the base URL
BaseUrl = "https://qt-api.ncreif.org",

// Step 5: Login to get a fresh token.
// Use the Power BI parameters "LoginEmail" and "LoginPassword" (which you must create separately)
LoginUrl = BaseUrl & "/Login/Login?cachebuster=" & DateTime.ToText(DateTime.LocalNow(), "yyyyMMddHHmmss"),
LoginPayload = Json.FromValue([
    Email = LoginEmail, // This parameter should contain the email (e.g., "MyName@ncreif.org")
    Password = LoginPassword // This parameter should contain the password (e.g., "MyPassword")
]),
LoginHeaders = [
    #"Accept" = "text/plain",
    #"Content-Type" = "application/json"
],
LoginResponse = Web.Contents(LoginUrl, [
    Headers = LoginHeaders,
    Content = LoginPayload
```

✓ No syntax errors have been detected.

Done

Cancel

Only thing that needs changed for different queries.

Queries read from Query Parameter table.

Username and password for authentication stored in parameter or separate text file (or in the M code here).

Merge Multiple Queries using Power Query

NPI Query

Cap Rate Query

	A	B	C	D	E	F	G
1	QueryName	Year	YYYYQ	NPI Total Return	Prop_Count	Count of Cap Rate	Appraisal Cap rate
2	ExpandedNPI	1978	19781	0.029035434	233	56	0.088459087
3	ExpandedNPI	1978	19782	0.030650282	259	59	0.073820778
4	ExpandedNPI	1978	19783	0.033945091	283	69	0.084709616
5	ExpandedNPI	1978	19784	0.058868747	291	128	0.076583172
6	ExpandedNPI	1979	19791	0.038116172	309	103	0.084068682
7	ExpandedNPI	1979	19792	0.043192361	331	101	0.084227484
8	ExpandedNPI	1979	19793	0.047534008	356	85	0.085562837
9	ExpandedNPI	1979	19794	0.061865284	370	147	0.075917697
10	ExpandedNPI	1980	19801	0.055355136	405	123	0.080871027
11	ExpandedNPI	1980	19802	0.0235727	428	93	0.083218301
12	ExpandedNPI	1980	19803	0.037858287	465	120	0.07542697
13	ExpandedNPI	1980	19804	0.053167172	493	201	0.0761552
14	ExpandedNPI	1981	19811	0.029567897	526	144	0.081092853
15	ExpandedNPI	1981	19812	0.04233486	569	154	0.077742491
16	ExpandedNPI	1981	19813	0.032137229	621	221	0.077603599
17	ExpandedNPI	1981	19814	0.052935961	681	343	0.071515258
18	ExpandedNPI	1982	19821	0.024920829	724	225	0.073532037
19	ExpandedNPI	1982	19822	0.020697653	757	244	0.079920755
20	ExpandedNPI	1982	19823	0.015216439	776	263	0.079903139

Using API with Microsoft Power BI

The screenshot displays the Microsoft Power BI desktop application interface. At the top, the title bar shows the file name "NPI - Copy" and the last saved time "2/2/2025 at 2:39 PM". A search bar is located in the top right corner. Below the title bar is the ribbon menu with tabs for File, Home, Insert, Modeling, View, Optimize, and Help. The Home tab is active, showing various toolbars for data sources (Get data, Excel, OneLake, SQL Server, Enter data, Dataverse, Recent sources), queries (Transform data, Refresh data), insert (New visual, Text box, More visuals), calculations (New visual calculation, New measure, Quick measure), sensitivity, publish, and Copilot.

The main workspace contains a line chart titled "NPI Total Return". The y-axis is labeled "Average of Table.Total_Return" and ranges from -0.10 to 0.05. The x-axis is labeled "Table.YYYYQ" and ranges from 1980 to 2020. The chart shows a blue line representing the total return over time, with a significant dip around 2008.

On the right side, there are three panels: "Filters", "Visualizations", and "Data". The "Filters" panel has a search bar and two sections for adding data fields. The "Visualizations" panel has a search bar and a grid of visualization icons. The "Data" panel has a search bar and shows the table "NPI".

AI-ASSISTED NCREIF QUERY TOOL

Example: What are the office returns for the past 5 years in Chicago?

Create Query

Select Parameter:

```
SUM(NOI) AS NOI, SUM(CapEx) AS CapEx, SUM(MV) AS MV, SUM(MVLag1) AS MVLag1,
SUM(PSales) AS PSales, SUM(Denom) AS Denom, SUM(NOI) / SUM(Denom) AS
'Income_Return', (SUM(MV) - SUM(MVLag1) - SUM(CapEx)) / SUM(Denom) AS
'Capital_Return', (SUM(NOI) + SUM(MV) - SUM(MVLag1) - SUM(CapEx)) / SUM(Denom) AS
'Total_Return', COUNT(MV) AS 'Prop_Count'
```

Where Parameter:

```
PropertyType = 'Office' AND CBSAorDiv = '16974' AND [YYYYQ] >= 20201
```

GroupBy Parameter:

```
[YYYYQ]
```

Run Query

Save Query to XML

MyQuery.xml

Clear Query

Upload a Saved XML Query File

Choose File

No file chosen

Load Query from XML

Summarize the data

What data do you want? (Returns, cap rates, Percent Leased, etc.)

Created by ChatGPT from the question in the Example above. You can edit the query if needed.

Send query to new API

Query can be saved that is compatible with the Query Tool and new API.

Have ChatGPT Summarize the query results.

Or load a saved query.

[Download CSV File](#)

YYYYQ	NOI	CapEx	MV	MVLag1	PSales	Denom	Income_Return	Capital_Return	Total_Return	Prop_Count
20201	119393441	219991450	13811810364	13637711768	0	13707909679	0.0087	-0.0033	0.0054	73
20202	129289997	237414728	13941116176	13973859902	0	14049470600	0.0092	-0.0192	-0.01	74
20203	123249042	167408285	13672743975	13956459470	372937797	13812611700	0.0089	-0.0327	-0.0237	75
20204	131417255	152189115	13479502492	13794243975	411917309	13620574126	0.0096	-0.0343	-0.0246	75
20211	138415028	104849149	13702882573	13672422492	72136592	13642640428	0.0101	-0.0055	0.0047	76
20212	146910666	90251332	13490091222	13702882573	352132270	13522971882	0.0109	-0.0224	-0.0115	75
20213	124154702	123241586	13607202278	13490091222	196938846	13411857691	0.0093	-0.0005	0.0088	74
20214	127557794	135639400	13742619663	13676784772	0	13702085207	0.0093	-0.0051	0.0042	74
20221	136345559	113250822	13542380808	14039425728	482602035	13809301602	0.0099	-0.0442	-0.0343	77
20222	139120298	120322016	12813821808	13044990808	0	13058778383	0.0107	-0.0269	-0.0163	73
20223	131178219	176105681	13127125766	13283753908	0	13328080676	0.0098	-0.025	-0.0151	77
20224	136873553	179143826	12785041798	13278116623	9425438	13317351299	0.0103	-0.0505	-0.0402	82
20231	139930401	197245215	12186723117	12805945744	0	12857924885	0.0109	-0.0635	-0.0526	83
20232	127729991	175856906	10567946186	12135927469	0	12181279258	0.0105	-0.1432	-0.1327	83
20233	145750253	244809153	11063091670	11420946186	0	11494767345	0.0127	-0.0524	-0.0397	83
20234	129712382	82208386	9953434489	11063091670	148542698	10986687053	0.0118	-0.1085	-0.0967	83
20241	98909356	251210372	8839587845	9980434489	55212993	10045463393	0.0098	-0.1386	-0.1287	80
20242	106309675	124197282	8642842317	8871975752	59773682	8868750994	0.012	-0.0398	-0.0279	77
20243	97766804	90879678	8578314268	8642842317	0	8655693221	0.0113	-0.018	-0.0067	74
20244	110763496	89275878	7935290180	8609914268	513079039	8361091522	0.0132	-0.0914	-0.0781	75

The provided HTML table contains data on the office returns for Chicago over the past several years, broken down by quarter (YYYYQ). The table includes columns for Net Operating Income (NOI), Capital Expenditures (CapEx), Market Value (MV), Market Value Lagged by 1 quarter (MVLag1), Property Sales (PSales), Denominator for returns calculation (Denom), Income Return, Capital Return, Total Return, and Property Count (Prop_Count). Here is a summary of the trends observed in the data:

- Decreasing Market Value (MV):** There is a noticeable decline in the market value of office properties from 2020 to 2024. This indicates a downward trend in the valuation of office spaces in Chicago over the past five years.
- Negative Capital Returns:** Capital returns have been negative for most quarters since 2020, with a particularly sharp decline in 2023 and 2024. This suggests that the value of office properties has been depreciating, which could be due to a variety of factors such as oversupply, reduced demand, or economic downturns.
- Fluctuating Income Returns:** Income returns have fluctuated but remained positive throughout the period. This indicates that while property values have declined, rental incomes have generally provided a positive return.
- Total Returns:** The total return, which is the sum of income return and capital return, has been negative in several quarters, particularly from 2023 onwards. This suggests that the overall investment performance of office properties in Chicago has been poor in recent times.
- Property Sales (PSales):** There have been periods with no property sales (e.g., 2020Q1, 2020Q2, 2021Q4, 2022Q2, 2022Q3, 2023Q1, 2023Q2), which could indicate low market activity or investor caution.

Correlation with other economic trends:

- Interest Rates:** Rising interest rates can lead to higher borrowing costs, which can negatively impact property investment returns. If there has been an upward trend in interest rates during this period, it could explain some of the negative capital returns and decreased market values.
- Economic Activity (GNP/GDP):** Economic downturns or slow growth in Gross National Product (GNP) or Gross Domestic Product (GDP) can lead to reduced demand for office space as businesses scale back operations. This, in turn, would affect both the income and capital returns on office properties.
- Remote Work Trends:** The shift towards remote work, accelerated by the COVID-19 pandemic, may have had an impact on the demand for office spaces, potentially contributing to the declining market values and negative capital returns observed. Overall, the trends in the data suggest that the office property market in Chicago has been facing challenges, with declining values and negative returns in recent years. These trends could be influenced by broader economic factors, shifts in work culture, and local market dynamics. Investors and stakeholders would need to consider these factors when making future investment decisions in the Chicago office market.