

100% OFFLINE · TREE · TABLE · STATS · ZERO DEPENDENCIES

XML Viewer

User Guide & Analyst Handbook

Load any XML file and turn it into a sortable table — explore the tree, flatten nested structures, compute column stats and pivots, and export clean CSV. All in your browser.

THEFAXMINATOR'S · XML VIEWER

What the program does

1

Turns XML into a table

Pick a repeating element as the "row anchor" and every occurrence becomes a table row — nested tags and attributes become columns.

2

100% offline

A single HTML file runs entirely in your browser. No internet, no server — the file you open never leaves your machine.

3

Three ways to look

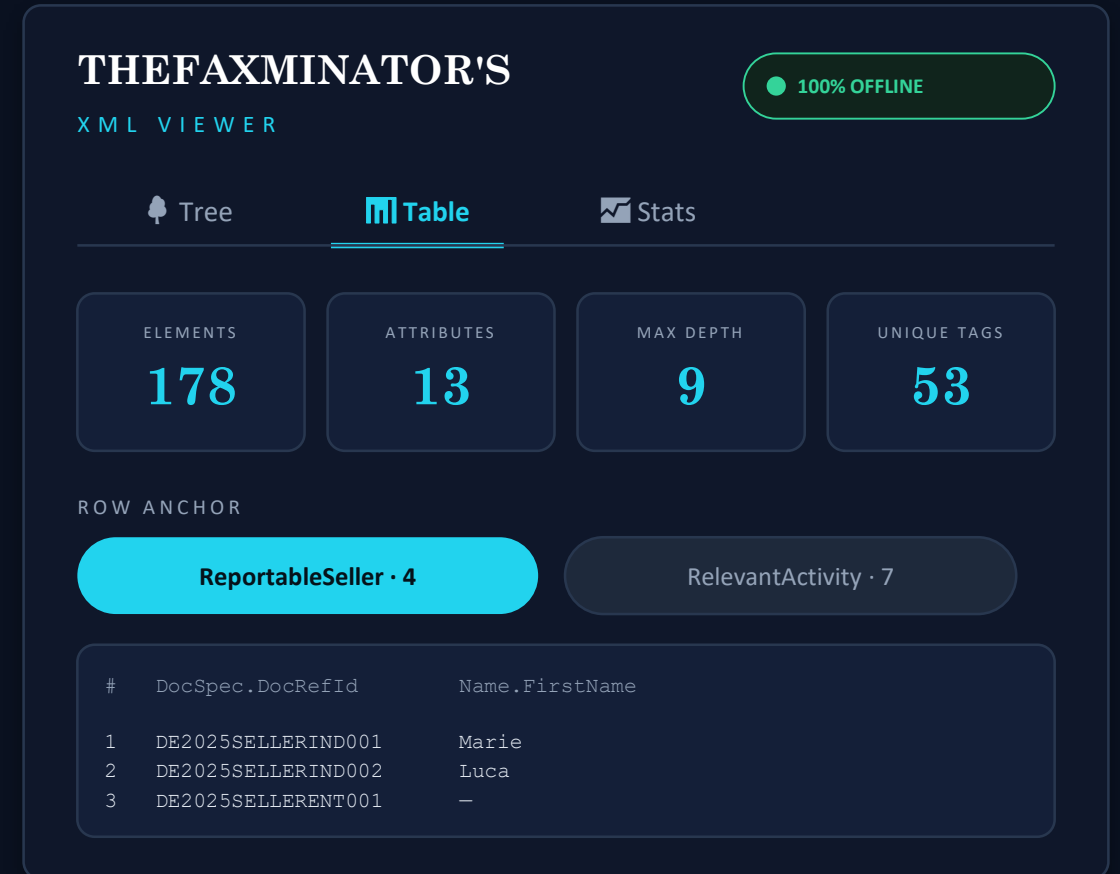
Switch between a collapsible Tree, a virtualized Table, and a Stats view with summaries, pivots and charts.

4

Built for messy data

Handles deep nesting, attributes and repeating children, then exports exactly what you see as clean CSV.

The app on launch — brand header, three tabs, live document stats, and the row-anchor table.



What you need to run it

One file. No install, no dependencies, no build step.

1

XML-Viewer-Offline.html

The entire application — HTML, CSS and JavaScript in a single file. Double-click to open it in any modern browser.

2

A modern browser

Chrome, Edge, Firefox or Safari. Runs client-side using the browser's native XML parser.

3

example.xml

Optional. A small sample XML file so you can try the tool immediately — or just drop in your own XML.

Tip: nothing is uploaded anywhere. The XML you open is read locally and never sent over the network .

How it works — three steps

From a raw XML file to an exportable table in under a minute.

1

Load a file

Drag an XML file onto the drop zone (or click to browse). It parses instantly and shows document stats.



2

Pick a row anchor

Choose which repeating element becomes your rows. The app detects candidates and flattens the rest into columns.



3

Explore & export

Sort, filter, summarise and pivot — then download the current view as CSV.

The Tree, Table and Stats tabs all read the same parsed document — switch freely at any time.

Step 1 — Load an XML file

Drop it in, or click to browse. Parsing happens the moment the file lands.

- 1 Drag your .xml file onto the drop zone — or click it to open a file picker.
- 2 The app parses with the browser's native DOMParser (no library).
- 3 A stats row appears: elements, attributes, max depth and unique tags.
- 4 Use "Load another file" in the info bar to swap files anytime.

⚠ Invalid XML? A red error box shows the parser's message so you can fix the source file.



Drop an XML file here
or click to browse — file never leaves your machine

example.xml

8.9 KB

Load another file

ELEMENTS
178

ATTRIBUTES
13

MAX DEPTH
9

UNIQUE TAGS
53

These four counts give you an instant sense of how large and how deep the document is.

Three views of the same file

The tab bar switches views without re-parsing. Tree, Table, and Stats.



Tree

The raw XML as a collapsible tree. Best for understanding structure and finding where a value lives.

- Search across tags, attributes & values
- Expand / collapse all
- Click a node for its full path



Table

The flattened table. Best for reading many records at once and exporting.

- Choose the row anchor
- Sort, filter rows & columns
- Export to CSV



Stats

Analysis of the current table. Best for data quality and quick aggregation.

- Per-column fill & distinct
- Top-value frequency bars
- Pivot table + bar chart

Tree view — explore the structure

See the document exactly as it is nested. Search jumps straight to matches.

● Search

Type any tag, attribute or value. Matches highlight and their branches auto-expand; a live count shows how many.

● Expand / Collapse all

Two buttons open or close the whole tree. Deep branches start collapsed to keep things readable.

● Path bar

Click any node to see its full ancestry, e.g. Document › Body › ReportableSeller.

Expand all 3 matches

▼ <Document> 2 children

▶ <MessageSpec> 8 children

▼ <Body> 5 children

▶ <PlatformOperator> 8 children

▼ <ReportableSeller> 2 children

· <ResCountryCode> FR

▼ <Identity>

· <TIN issuedBy="FR" FR123456789012

· <FirstName> Marie

· <LastName> Dubois

▶ Document › Body › ReportableSeller › Identity › Individual

The row anchor — XML into rows

Pick the repeating element whose occurrences become your table rows. Everything else flattens into columns.

ROW ANCHOR — PICK THE ELEMENT WHOSE REPETITIONS BECOME ROWS

ReportableSeller (4 total, max 4/parent)

RelevantActivity (7 total, max 2/parent)

Selected anchor → 4 rows

Nested tags → dotted columns

XML

```
<Identity><Individual><Name>  
<FirstName>Marie
```

→ COLUMN

```
Identity.Individual.Name.FirstName
```

Attributes → @ columns

XML

```
<TIN issuedBy="FR">  
FR123456789012
```

→ COLUMN

```
Identity.Individual.TIN.@issuedBy
```

Repeats → joined with " | "

XML

```
<Type>PersonalServices  
<Type>Rental
```

→ COLUMN

```
PersonalServices | Rental
```

Namespace noise (xmlns, xsi:schemaLocation) is stripped automatically. Columns are the union of every row's keys.

Working with the table

Sort, filter and scan large files smoothly — the table only renders the rows you can see.

Rows 1–4 of 4 × 38 of 38 cols

Filter rows...

Filter columns...

↓ Export CSV

#	DocRefId ▲	TIN.@issuedBy	Name.FirstName	Name.LastName	Address.City	BirthDate
1	DE2025SELLERIND001	FR	Marie	Dubois	Paris	1985-03-12
2	DE2025SELLERIND002	IT	Luca	Rossi	Milano	1990-07-24
3	DE2025SELLERENT001	—	—	—	Madrid	—
4	DE2025SELLERENT002	—	—	—	Amsterdam	—

● Click a header to sort

First click ascending, second descending, third clears.

● Filter rows & columns

Row filter matches any cell; column filter hides non-matching columns.

● Virtual scroll

Only visible rows are drawn, so huge files stay smooth. # column stays pinned.

Export to CSV

One click saves exactly what you see — nothing more, nothing less.

● What you see is what you get

The CSV honours the active sort, the row filter, and any columns you've hidden with the column filter.

● Spreadsheet-safe

RFC-4180 quoting escapes commas, quotes and newlines. A UTF-8 BOM makes Excel open accents correctly.

● Named after your file

example.xml exports as example.csv, so downloads stay organised.

example.csv

DocRefId,TIN.@issuedBy,FirstName,City

DE2025SELLERIND001,FR,Marie,Paris

DE2025SELLERIND002,IT,Luca,Milano

DE2025SELLERENT001,,,Madrid

DE2025SELLERENT002,,,Amsterdam

Tip: need Excel formulas or charts? Export CSV, then open it in your spreadsheet tool of choice.

Stats — column summaries

Every column profiled at a glance: how full it is, how varied, and its most common values.

DocSpec.DocTypeIndic

4/4 filled

1 distinct · Text

OECD1

4

Address.PostCode

2/4 filled

2 distinct · min 1,012 · max 28,001

28001

1

1012

1

COLUMN SUMMARIES				
4 rows · 38 columns — toggle Cards / Table, sort & filter				
COLUMN	TYPE	FILL	DISTINCT	TOP VALUE
DocSpec.DocRefId	Text		4	DE2025SELLERIND001 (1)
DocSpec.DocTypeIndic	Text		1	OECD1 (4)
Identity...Address.City	Text		2	Madrid (1)
Identity...Address.PostCode	Numeric		2	1012 (1)
Name.FirstName	Text		2	Marie (1)
ResCountryCode	Text		2	FR (1)

Fill bars are colour-coded — green = 100%, amber ≥ 50%, red below. Click any row to jump to its card.

Stats — pivot & chart

Group rows by one column, aggregate another, and read the result as a table and a bar chart.

Group by

Type

Values

— column —

Count

Sum

Min

Max

Mean

Type	Count
PersonalServices	2
Rental	2
SaleOfGoods	2
RentalOfMeansOfTransport	1
Total (4 groups)	7

Count by Type

A bar chart titled "Count by Type" displays the count for four categories: PersonalSvc, Rental, SaleOfGoods, and RentalTransp. The y-axis represents the count, ranging from 0 to 3. The bars are blue. The counts are: PersonalSvc (2), Rental (2), SaleOfGoods (2), and RentalTransp (1).

Type	Count
PersonalSvc	2
Rental	2
SaleOfGoods	2
RentalTransp	1

Troubleshooting & tips

The most common snags and how to clear them.

● Red "XML parse error" box

The file isn't well-formed XML. The box shows the parser's message — fix the source and reload.

● "No repeating elements found"

The document has no element that repeats. Use the Tree tab to inspect it; there's nothing to tabulate as rows.

● Too many / too few rows

You picked the wrong anchor. Switch the anchor chip — e.g. RelevantActivity (7) instead of ReportableSeller (4).

● A column reads " | " joined

That field repeats within a row. Re-anchor on the child element to give each repeat its own row.

● Numbers won't sum in a pivot

Only clean numeric strings aggregate. Mixed or joined values fall back to Count.

● Everything resets on reload

By design — the app keeps nothing. Re-open the file (or use "Load another file") to start again.

QUICK REFERENCE

You're ready to explore

- **Open** — Double-click XML-Viewer-Offline.html in any modern browser.
- **Load** — Drop an XML file on the zone (or click to browse).
- **Anchor** — Pick the repeating element to use as rows.
- **Explore** — Tree to inspect · Table to sort & filter · Stats to summarise & pivot.
- **Export** — Download CSV — respects your sort, filters and hidden columns.