

Study sheet: Module 10, time series and market indices

This session teaches students to build a market index the way it is actually built, by bucketing cleared prices into time periods and tracking a summary statistic across those periods, and to treat the exercise as a lesson in methodology rather than a shortcut to a market call. Students construct a quarterly demand index for a brand or category from sold-lot records, using date bucketing in SQL to group transactions and a stable statistic such as median realized price or count of sold lots to summarize each quarter, then compare the shape of two brands' indices side by side. The session's central discipline is one that recurs throughout the course and is unavoidable here: the newest quarters in this dataset are still being ingested, so any index that runs up to the present will show an artificial falloff at its tail, and the session insists, repeatedly and explicitly, that this falloff is a data engineering artifact to be flagged, not a market signal to be reported. Students leave able to build the index and able to say, precisely, where it stops being trustworthy.

What you should be able to do

1. Explain what a market index built from transaction data represents, and distinguish it from a price index built from listed or asking prices.
2. Bucket sold-lot records into calendar quarters using SQL date functions and compute a quarterly summary statistic for each bucket.
3. Build a quarterly demand index for a single brand or category and identify the reference period against which later quarters would be compared.
4. Explain, in specific and concrete terms, why the newest quarters of any such index must be treated as under-ingested rather than as evidence of a market decline, and identify where in a chart that caveat should appear.
5. Compare the index shapes of two brands or categories and describe the comparison as a methodological exercise in relative pattern, not as a claim about which one is appreciating.
6. (Extension) Retrieve the underlying sold-lot records programmatically via the production API and reproduce the quarterly bucketing in Python.

Key terms

- **Demand index:** a time series built from a stable summary statistic, such as median realized price or sold-lot count, tracked across calendar periods.
- **Date bucketing:** grouping transaction records into fixed calendar periods, such as quarters, using a SQL date-truncation function.
- **Recency under-ingestion:** the newest quarters in the dataset have fewer records because ingestion is still catching up, not because market activity fell.
- **Reference period:** the baseline quarter or period against which later values in an index are implicitly or explicitly compared.
- **Seasonality:** a recurring pattern tied to the calendar, such as concentrated spring and fall auction seasons, that can shape an index independent of underlying demand.

- **Volume-based index:** an index built on the count of sold lots per period, as distinct from a price-based index built on a price statistic.
- **Trailing artifact:** a distortion that appears specifically at the most recent end of a time series because of how the underlying data was collected, not because of any real change in the world.
- **Overlay comparison:** placing two time series on the same chart to compare their shapes directly, without claiming a causal or predictive relationship between them.

Core sandbox query

The query the module is built around. Run it, then change the brand or category and compare.

```
SELECT date_trunc('quarter', CAST(sale_date AS date)) AS sale_quarter,  
       approx_percentile(usd_price_decimal, 0.5) AS median_price_usd,  
       COUNT(*) AS sold_lots  
FROM all_fine_art_data  
WHERE status = 'sold'  
       AND designer LIKE '%warhol%'  
GROUP BY date_trunc('quarter', CAST(sale_date AS date))  
ORDER BY sale_quarter;
```

Full lesson, more queries, and the homework are in Module 10 at academy.altfndata.com. Questions: info@altfndata.com.