

Problem set 7: The business of the art market

Module 7. Total: 100 points. Format: no-code sandbox, with an optional API extension in Problem 4 using a shared class API key from your instructor.

Before you start

This problem set works with the fine art data. In that table the designer field holds the artist or maker name, and the vendor field holds the auction house. You will build the two artifacts art market professionals rely on: an auction house league table that ranks salerooms by what they sell, and an artist-level view of demand through pricing power and sell-through. Choose one major auction house and one well-known artist to use throughout, and confirm the exact spelling of each in the coverage browser before you write any filter, since exact-match filters are spelling-sensitive.

Problem 1 (15 points): Vendor, designer, and the league table

In four to five sentences, explain what the vendor and designer fields represent in the fine art data, how a league table is built from them, and one thing a league table reveals about the art market as a business that a single sale result would not.

Problem 2 (20 points): Build the auction house league table

Write a SQL query against the fine art data that ranks auction houses by total realized value and, alongside it, the number of sold lots. Restrict to sold records, group by vendor, and order so the highest-value houses appear first.

```
-- your query here
```

Problem 3 (20 points): Artist pricing power and sell-through

Write SQL that computes, for your chosen artist, both pricing power (the median of `usd_price_decimal` divided by `sale_estimates_high_usd_price` over sold records with a valid, non-zero high estimate) and sell-through rate (the share of that artist's offered lots that sold, with both sold and unsold in the denominator). You may use one query or two; state which figure each query produces.

```
-- your query (or queries) here
```

Problem 4 (20 points): One house via the API (optional code extension)

Using the shared class API key your instructor has provided, write the endpoint and JSON request body you would send to `POST /v1/tables/{name}/query` to retrieve the raw fields needed to assemble your chosen house's slice of the league table (vendor, designer, `sale_date`, `usd_price_decimal`), filtered to that house and to sold records. Then, in two to three sentences, explain why the query endpoint cannot return the league-table totals directly, and how you would paginate on offset and aggregate the rows yourself in the tutorials notebook or with the Python client.

```
POST /v1/tables/<table_name>/query
Header: X-API-Key: <class key>
Body:
{
  ...
}
```

Problem 5 (25 points): Market concentration and an art market brief

Write a SQL query that lists the top 15 artists by total realized value in the fine art data (sold records, grouped by designer, ordered by total realized value). Then write a short art market brief, 200 to 300 words, in the voice of an adviser to a gallery director or a saleroom's business team. The brief should describe what the concentration of value among a small number of names suggests about the art market as a business, connect it to your Problem 3 artist where relevant, and include an explicit limitations section noting that the data captures the secondary market only, that unsold lots are retained and marked unsold, and that the newest periods are still being ingested, so the brief must not read any recent-period totals as a trend.

-- your query here

Submission. Turn in this file with your SQL, JSON body, and written answers filled in.