

Investment Portfolio Analysis Report

Investment Portfolio Analysis Report

1. Executive Summary

This report presents the results of an investment portfolio analysis project that integrates **SQL analytics**, **Python & Jupyter Notebook**, and **Power BI** to evaluate portfolio performance, risk, diversification, and client behavior through comprehensive dashboards and KPI metrics.

2. Project Overview

Objectives

- Provide actionable insights into investment portfolio performance
- Compare portfolios against market benchmarks
- Visualize risk-adjusted returns and client activity
- Deliver interactive dashboards for decision-makers

Methodology

- Constructed a **normalized relational schema** in MySQL via Docker
- Executed analytical queries using PostgreSQL-style SQL to compute performance, risk, fees, and allocation metrics
- Combined SQL output with **Python data analysis** in Jupyter
- Built a **Power BI dashboard** for visualization and KPI tracking

3. Technical Implementation

3.1 Database & SQL Queries

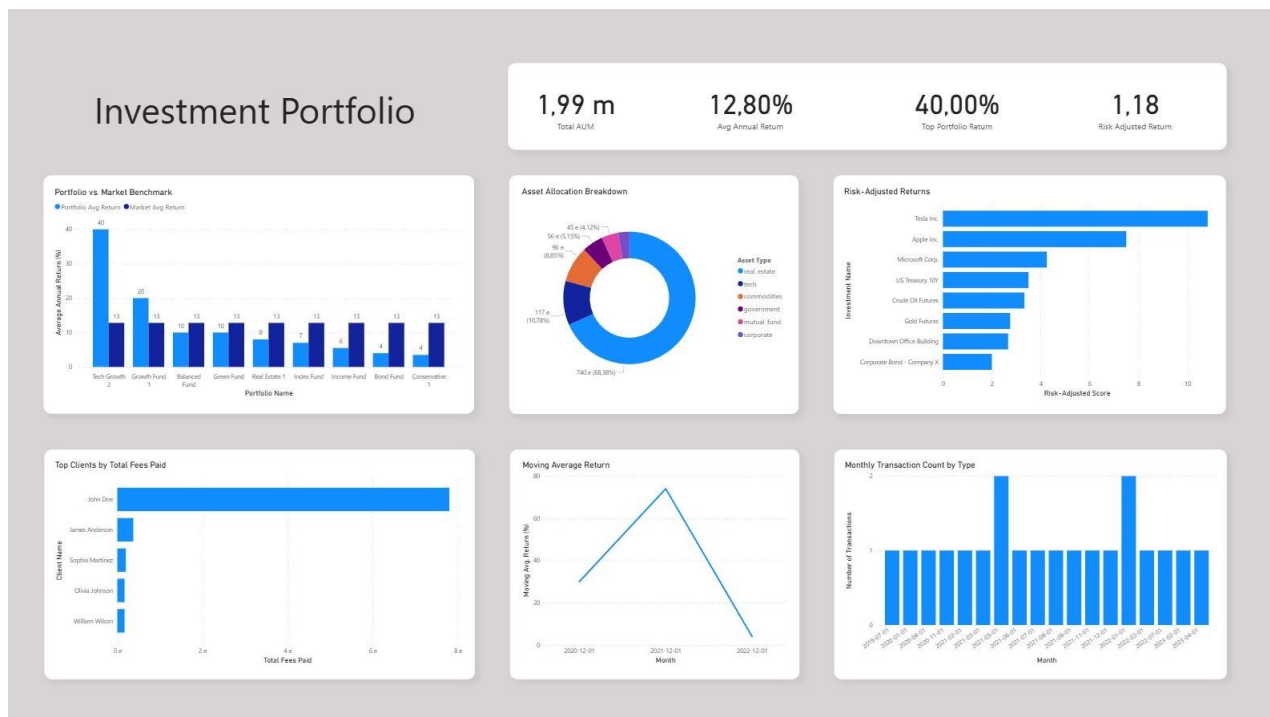
- MySQL 8 containerized with Docker Compose, initialized using init.sql
- Organized SQL-level analytics in scripts/queries/ with subfolders:
 - client_insights/ (e.g. top fee payers, client-owned value)
 - portfolio_analysis/ (weighted returns, diversification, rankings)
 - investment_analysis/ (risk-adjusted return calculations)
 - dashboards/ (monthly transaction trends)
 - advanced/ (moving averages, benchmark comparison)

3.2 Python & Jupyter Notebook

- Python scripts and a Jupyter Notebook (analysis_investment_portfolio.ipynb) used:
 - to load data from MySQL
 - run data exploration and statistics
 - produce deliverable outputs for Power BI integration

3.3 Power BI Dashboard

- Visual KPI tracks created in Power BI including:
 - Total Assets Under Management (AUM)
 - Average annual return
 - Risk-adjusted return
 - Top portfolio return
- Main visuals implemented:
 - Clustered column chart for Portfolio vs. Market Benchmark
 - Line chart for Moving Average Return
 - Donut chart for Asset Allocation Breakdown
 - Bar chart for Risk-Adjusted Returns
 - Stacked column chart for Monthly Transaction Trend
 - Bar chart for Top Clients by Fees



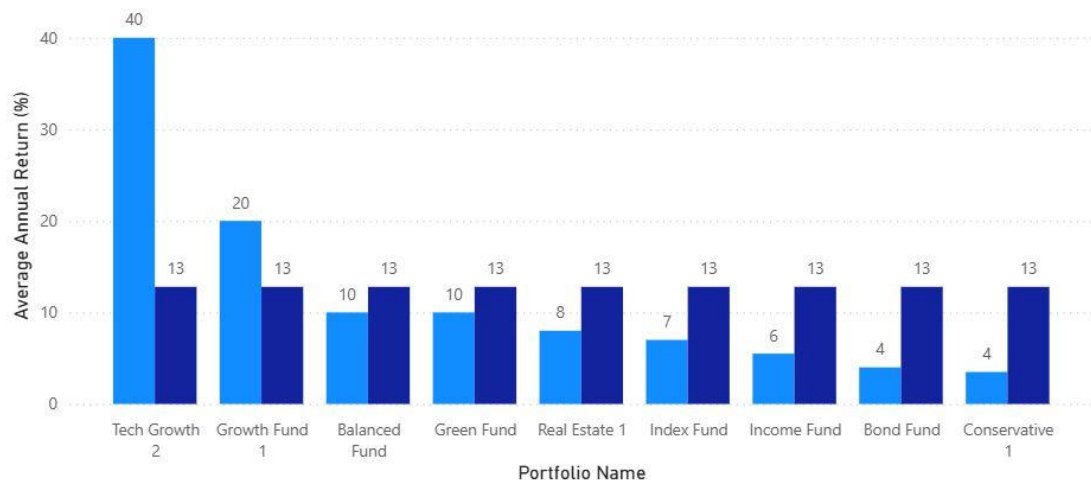
4. Key Findings & Insights

AUM & Performance

- Total AUM across all portfolios: 1,99 m
- Average annual return: 12,8 %
- Best-performing portfolio: *Tech Growth 2* with an annual return of 40%

Portfolio vs. Market Benchmark

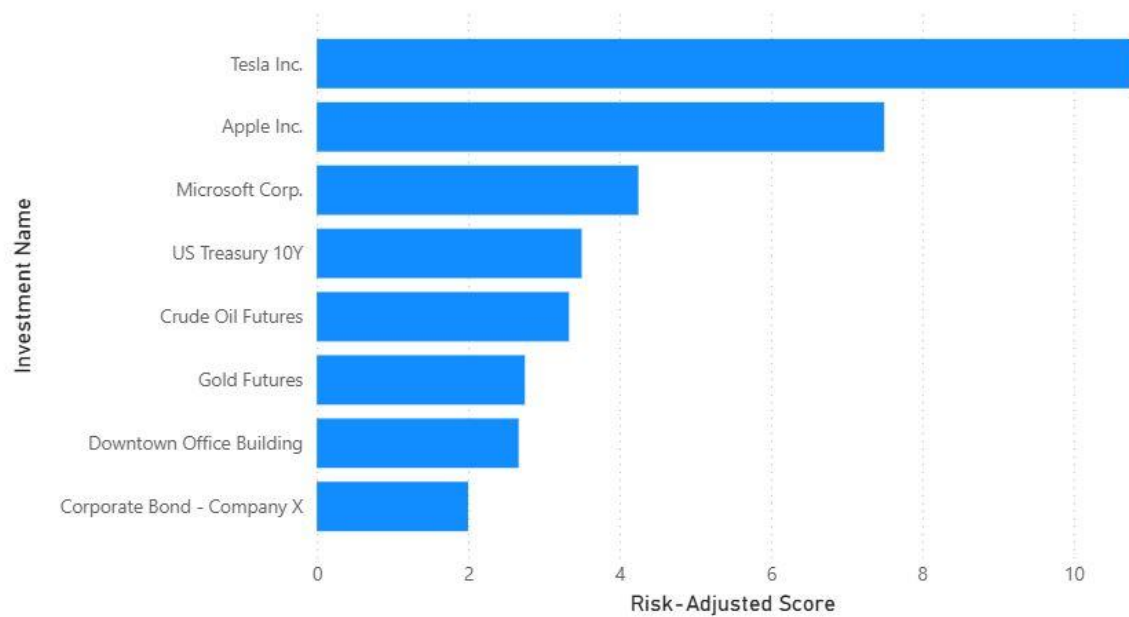
● Portfolio Avg Return ● Market Avg Return



Risk-Adjusted Returns

- Investments with high volatility underperformed once adjusted for risk
- Top risk-adjusted return observed investment: Tesla Inc.

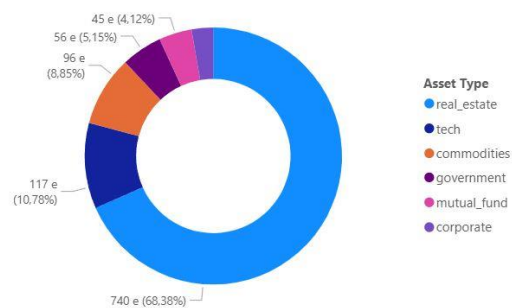
Risk-Adjusted Returns



Allocation & Diversification

- Portfolios generally diverse across assets, though some show concentration in a few holdings
- Benchmarked portfolios outperform similar client-risk-profile groups

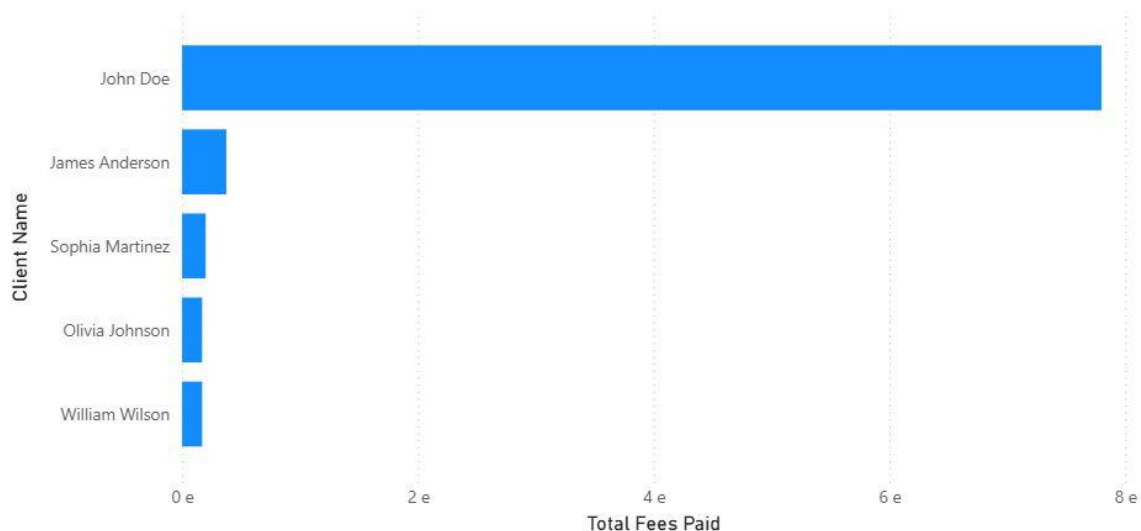
Asset Allocation Breakdown



Client Behavior & Fees

- Clients with highest transaction volume also paid the most in fees
- Some high-fee clients appear to have lower net return after fees

Top Clients by Total Fees Paid



5. Recommendations

- Encourage optimization for **risk-adjusted performance**, not just raw return
- Reduce fee leakage where clients churn frequently
- Enhance diversification in under-diversified portfolios
- Incorporate benchmarking practices to drive competitive advantage

6. Supplementary Materials

- **Power BI Dashboard File:** `investment_portfolio_dashboard.pbix`
- **Dashboard Snapshot:** `benchmark_snapshot.jpg` (includes visual preview and KPI summary)
- **HTML Overview:** `docs/index.html` — interactive project overview and user guidance

7. Tools & Technology Stack

- **Database:** MySQL 8 containerized via Docker
 - **SQL:** PostgreSQL-qualified SQL queries
 - **Python:** Jupyter with pandas, sqlalchemy, python-dotenv, mysql-connector-python, matplotlib
 - **Visualization:** Power BI Desktop for dashboards and data storytelling
-

8. Project Files & Structure

```
investment_portfolio/
├── docker/           — Docker Compose and init.sql schema/data
├── scripts/          — Organized SQL queries
├── notebooks/        — Python/Jupyter analytics
├── powerbi/          — PBIX dashboard and snapshot PDF
├── docs/             — HTML documentation
├── requirements.txt   — Python dependencies
└── .gitignore
```

9. License & Contact Information

This project is provided for **educational use**. Please retain attribution if reused or referenced

10. Contact Information

Created by Gabor Barta

 **Fiverr profile:** [fiverr.com/gaborbarta1](https://www.fiverr.com/gaborbarta1)

 **Contact:** barta.gabor1992@email.com