Analysis of Funding Trends in the Indian Startup Ecosystem

AMITH KRISHNA PININTY

December 14, 2024

Project Overview

This project focuses on analyzing funding trends in Indian startups from January 2015 to August 2017. Using SQL, insights were derived to identify key patterns, trends, and contributors in the startup funding ecosystem. The database comprises information about funding amounts, investors, cities, industries, and more.

Objective

To utilize SQL for uncovering actionable insights about the Indian startup funding landscape, enabling better understanding of:

- Funding distribution across industries, cities, and investors.
- Temporal trends in funding amounts.
- Key contributors and beneficiaries in the ecosystem.

Database Setup

Database Creation

```
CREATE DATABASE STARTUP;
USE STARTUP;
```

Dataset Table: FUNDING

The 'FUNDING' table contains the following key columns:

- **year**: Year of funding.
- **month**: Month of funding.
- **quarter**: Quarter of funding.
- **citylocation**: Location of the funded startups.
- **industryvertical**: Industry sector of the startups.
- **investmenttype**: Type of investment (e.g., Seed, Series A).
- **inr_amount**: Funding amount in Indian Rupees.
- **investorsname**: Names of investors.

- **startupname**: Name of the funded startup.
- **funding_category**: Category of funding.
- **amount_in_usd**: Funding amount in USD.

Key SQL Queries and Insights

1. Total Funding by Year

Query:

```
SELECT
    year, SUM(INR_amount) AS total_fund
FROM
    funding
GROUP BY year
ORDER BY total_fund DESC;
```

Insight: Identified the year with the highest total funding.

2. Count of Startups Funded by City

Query:

```
SELECT
    citylocation, COUNT(startupname) AS no_of_startups
FROM
    funding
GROUP BY citylocation
ORDER BY no_of_startups DESC;
```

Insight: Highlighted cities that served as major startup hubs.

3. Funding by Industry

Query:

```
SELECT
    industryvertical, SUM(INR_Amount)
FROM
    funding
GROUP BY industryvertical
LIMIT 5;
```

Insight: Revealed the top industries by total funding.

4. Monthly Funding Trend

Query:

```
SELECT
    year, month, SUM(INR_Amount) AS fund_raised
FROM
    funding
GROUP BY year, month
ORDER BY year, month, fund_raised;
```

Insight: Showed seasonal trends in funding.

5. Top Investors by Investment Count

Query:

```
SELECT
    investorsname AS investors, COUNT(*) AS no_of_investments
FROM
    funding
GROUP BY investorsname
ORDER BY no_of_investments DESC;
```

Insight: Identified the most active investors.

6. Yearly Growth Rate of Funding

Query:

```
SELECT year, SUM(inr_amount),
    (SUM(inr_amount) - LAG(SUM(INR_AMOUNT))
    OVER (ORDER BY YEAR)) / LAG(SUM(INR_AMOUNT))
    OVER (ORDER BY YEAR) * 100 AS YOY_Growth
FROM FUNDING
GROUP BY YEAR;
```

Insight: Showed the year-over-year growth in total funding.

7. Industries Receiving Consistent Funding Across All Years

Query:

```
SELECT
    industryvertical, COUNT(DISTINCT year)
FROM
    funding
GROUP BY industryvertical
order by  COUNT(DISTINCT year) desc;
```

Insight: Identified industries that received funding consistently across all years.

8. Rank Industries by Total Funding

Query:

```
SELECT INDUSTRYVERTICAL AS INDUSTRY, SUM(INR_AMOUNT),
    RANK() OVER (ORDER BY SUM(INR_AMOUNT) DESC) AS RANK_
FROM FUNDING
GROUP BY INDUSTRYVERTICAL;
```

Insight: Ranked industries based on total funding received.

9. Cumulative Funding Over Time

Query:

```
SELECT year, month, SUM(inr_amount),
    SUM(SUM(inr_amount)) OVER (ORDER BY year, month) AS cumulative_funding
FROM funding
GROUP BY year, month
ORDER BY year, month;
```

Insight: Showed how funding accumulated over the years.

10. Percentage Contribution of Top Investors

Query:

```
WITH INVESTORSTOTAL AS (
    SELECT
        INVESTORSNAME, SUM(INR_AMOUNT) AS _INVESTMENT
    FROM
        FUNDING
    GROUP BY INVESTORSNAME),
    total_investment AS (
    SELECT SUM(INR_AMOUNT) AS total_amount
    FROM FUNDING)
SELECT
    it.INVESTORSNAME, it._investment,
    (it._investment / ti.total_amount) * 100 AS percentage_contribution
FROM
    INVESTORSTOTAL it, total_investment ti
ORDER BY percentage_contribution DESC;
```

Insight: Identified the investors with the largest percentage contributions to total funding.

11. Average Funding Amount by Investment Type

Query:

```
SELECT
    investmenttype, AVG(INR_amount) AS average_funding
FROM
    funding
GROUP BY investmenttype
ORDER BY average_funding DESC;
```

Insight: Determined the average funding amounts based on investment types.

12. Total Investments in Each Quarter

Query:

```
SELECT
    quarter, SUM(INR_amount) AS total_investment
FROM
    funding
GROUP BY quarter
ORDER BY quarter;
```

Insight: Analyzed total investments made in each quarter.

13. Startups Receiving the Highest Funding

Query:

```
SELECT
    startupname, SUM(INR_amount) AS total_funding
FROM
    funding
GROUP BY startupname
ORDER BY total_funding DESC;
```

Insight: Identified the startups that received the highest total funding.

14. Monthly Average Funding Amount

Query:

```
SELECT
    year, month, AVG(INR_amount) AS average_funding
FROM
    funding
GROUP BY year, month
ORDER BY year, month;
```

Insight: Calculated the average funding amount for each month.

15. Funding Trends by Industry Over Time

Query:

```
SELECT
    year, month, industryvertical, SUM(INR_amount) AS total_funding
FROM
    funding
GROUP BY year, month, industryvertical
ORDER BY year, month, industryvertical;
```

Insight: Showed how funding trends varied by industry over time.

16. Investors Who Funded Multiple Startups

Query:

```
SELECT
    investorsname, COUNT(DISTINCT startupname) AS no_of_startups
FROM
    funding
GROUP BY investorsname
HAVING no_of_startups > 1
ORDER BY no_of_startups DESC;
```

Insight: Identified investors who funded multiple startups.

17. Total Funding by Funding Category

Query:

```
SELECT
    funding_category, SUM(INR_amount) AS total_funding
FROM
    funding
GROUP BY funding_category
ORDER BY total_funding DESC;
```

Insight: Summarized total funding amounts categorized by funding type.

18. Funding Distribution by City and Industry

Query:

```
SELECT
    citylocation, industryvertical, SUM(INR_amount) AS total_funding
FROM
    funding
GROUP BY citylocation, industryvertical
ORDER BY total_funding DESC;
```

Insight: Analyzed funding distribution across different cities and industries.

19. Investors by Total Investment Amount

Query:

```
SELECT
    investorsname, SUM(INR_amount) AS total_investment
FROM
    funding
GROUP BY investorsname
ORDER BY total_investment DESC;
```

Insight: Ranked investors based on total amount invested.

20. Yearly Funding by Industry

Query:

```
SELECT
    year, industryvertical, SUM(INR_amount) AS total_funding
FROM
    funding
GROUP BY year, industryvertical
ORDER BY year, total_funding DESC;
```

Insight: Showed how funding varied across different industries by year.

21. Cities with Most Diverse Startup Funding

Query:

```
SELECT
    citylocation, COUNT(DISTINCT industryvertical) AS diverse_industries
FROM
    funding
GROUP BY citylocation
ORDER BY diverse_industries DESC;
```

Insight: Identified cities with the most diverse range of startup industries.

22. Percentage of Total Funding by Investment Type

Query:

```
SELECT
    investmenttype, (SUM(INR_amount) / (SELECT SUM(INR_amount) FROM funding) * 100)
    AS
    percentage_of_total
FROM
    funding
GROUP BY investmenttype
ORDER BY percentage_of_total DESC;
```

Insight: Calculated the percentage contribution of each investment type to total funding.

23. Startups by Funding Year

Query:

```
SELECT
    year, COUNT(DISTINCT startupname) AS startups_funded
FROM
    funding
GROUP BY year
ORDER BY year;
```

Insight: Counted the number of unique startups funded each year.

24. Most Active Quarters for Funding

Query:

```
SELECT
    quarter, SUM(INR_amount) AS total_funding
FROM
    funding
GROUP BY quarter
ORDER BY total_funding DESC;
```

Insight: Identified which quarters had the highest total funding amounts.

Tools Used

- **SQL:** To execute queries and derive insights.
- **Excel:** For exporting and formatting query results.

Outcomes and Learnings

- **Data Insights:** Successfully identified trends, top contributors, and funding distributions.
- **Skill Development:** Strengthened SQL proficiency with advanced queries, window functions, and subqueries.
- **Data Preparation:** Prepared clean and analysis-ready data.
- **Career Readiness:** Gained practical experience in SQL that prepares for roles involving database management, data analysis, and querying.