Exploratory Data Analysis (EDA)

Overview of Job Openings

```
CREATE TABLE tbl_overview_job_openings AS
SELECT
   COUNT(*) AS total_jobs,
   SUM(number_of_openings) AS total_openings,
   AVG(demand_index) AS avg_demand_index,
   ROUND(AVG(salary_median)) AS avg_salary
FROM startup_hiring_clean;
```

Hiring Trends by Year and Month

```
CREATE TABLE tbl_hiring_trends_year_month AS
SELECT
   year,
   month,
   month_name,
   SUM(number_of_openings) AS total_openings
FROM startup_hiring_clean
GROUP BY year, month, month_name;
```

Industry-Wise Hiring Distribution

```
CREATE TABLE tbl_industry_hiring_dist AS
SELECT
   industry,
   SUM(number_of_openings) AS total_openings
FROM startup_hiring_clean
GROUP BY industry;
```

Location-Based Hiring Analysis

```
CREATE TABLE tbl_location_hiring AS
SELECT
   city,
   state,
```

```
    SUM(number_of_openings) AS total_openings
FROM startup_hiring_clean
GROUP BY city, state;
```

Talent Demand Analysis

Identification of High-Demand Job Roles

```
CREATE TABLE tbl_high_demand_roles AS
SELECT
    job_title,
    SUM(number_of_openings) AS total_openings,
    ROUND(AVG(demand_index),2) AS avg_demand
FROM startup_hiring_clean
GROUP BY job_title
HAVING AVG(demand_index) > 70;
```

Skill Demand Analysis

```
CREATE TABLE tbl_skill_demand AS
SELECT
    skills_required,
    COUNT(*) AS job_count,
    SUM(number_of_openings) AS total_openings
FROM startup_hiring_clean
GROUP BY skills_required;
```

Industry-Wise Talent Demand

```
CREATE TABLE tbl_industry_talent_demand AS
SELECT
    industry,
    ROUND(AVG(demand_index),2) AS avg_demand_index,
    SUM(number_of_openings) AS total_openings
FROM startup_hiring_clean
GROUP BY industry;
```

Demand Index & Market Pressure Analysis

```
CREATE TABLE tbl_market_pressure AS
SELECT
   job_title,
   ROUND(AVG(demand_index),2) AS avg_demand_index,
   ROUND(AVG(job_market_pressure),2) AS avg_market_pressure
FROM startup_hiring_clean
GROUP BY job_title;
```

Salary & Experience Analysis

Salary Distribution Across Job Roles

```
CREATE TABLE tbl_salary_by_job_role AS
SELECT
   job_title,
   ROUND(AVG(salary_median)) AS avg_salary
FROM startup_hiring_clean
GROUP BY job_title;
```

Salary Comparison by Experience Level

```
CREATE TABLE tbl_salary_by_experience AS
SELECT
   experience_level,
   ROUND(AVG(salary_median)) AS avg_salary
FROM startup_hiring_clean
GROUP BY experience_level;
```

Relationship Between Demand and Salary

```
CREATE TABLE tbl_demand_vs_salary AS
SELECT
   job_title,
   ROUND(AVG(demand_index),2) AS avg_demand,
   ROUND(AVG(salary_median)) AS avg_salary
FROM startup_hiring_clean
GROUP BY job_title;
```

High-Paying & High-Demand Roles

```
CREATE TABLE tbl_high_pay_high_demand AS
SELECT
   job_title,
   ROUND(AVG(salary_median)) AS avg_salary,
   ROUND(AVG(demand_index),2) AS avg_demand
FROM startup_hiring_clean
GROUP BY job_title
HAVING AVG(salary_median) >
     (SELECT AVG(salary_median) FROM startup_hiring_clean)
  AND AVG(demand_index) >
     (SELECT AVG(demand_index) FROM startup_hiring_clean);
```

Remote vs On-Site Hiring Analysis

Remote vs On-Site Distribution

```
CREATE TABLE tbl_remote_distribution AS
SELECT
   remote_availability,
   COUNT(*) AS job_count,
   SUM(number_of_openings) AS total_openings
FROM startup_hiring_clean
GROUP BY remote_availability;
```

Salary Comparison: Remote vs On-Site

```
CREATE TABLE tbl_remote_salary_comparison AS
SELECT
   remote_availability,
   ROUND(AVG(salary_median)) AS avg_salary
FROM startup_hiring_clean
GROUP BY remote_availability;
```

Role & Industry Trends in Remote Hiring

```
CREATE TABLE tbl_remote_role_industry AS
```

```
SELECT
   industry,
   job_title,
   COUNT(*) AS job_count
FROM startup_hiring_clean
WHERE remote_flag = 1
GROUP BY industry, job_title;
```

Ranking Top In-Demand & High-Paying Roles

```
CREATE TABLE tbl_ranked_roles AS
SELECT
   job_title,
   ROUND(AVG(demand_index),2) AS avg_demand,
   ROUND(AVG(salary_median)) AS avg_salary,
   RANK() OVER (ORDER BY AVG(demand_index) DESC) AS demand_rank,
   RANK() OVER (ORDER BY AVG(salary_median) DESC) AS salary_rank
FROM startup_hiring_clean
GROUP BY job_title;
```

Trend Analysis Using Window Functions

```
CREATE TABLE tbl_opening_trend_window AS
SELECT
   year,
   month,
   SUM(number_of_openings) AS total_openings,
   SUM(SUM(number_of_openings))
      OVER (ORDER BY year, month) AS cumulative_openings
FROM startup_hiring_clean
GROUP BY year, month;
```