

Title Page

Project Title : The Startup Ecosystem

Group ID : 39

Student ID : 202412061

Student's Name : Parmar Shenal

Student ID : 202412068

Student's Name : Patel Mahek

Institution : Dhirubhai Ambani Institute of Information and Communication Technology (DA-IICT)

Date of Submission : 13/11/2024

Table of Contents

Table of Contents

Chapter 1: Software Requirements Specification (SRS)	4
1. Problem Description.....	4
2. Requirement Collection.....	7
3. Fact-Finding Chart	18
4. Requirements List	19
5. User Privileges	22
Chapter 2 : Database Design	25
1. Noun Analysis.....	25
2. Schema and ER Diagram Design.....	28
3. ER Diagram Improvement	30
4. Mapping ER Model to Relational Model	34
5. Create DDL Scripts.....	40
Chapter 3: Normalization of Database	45
1. Normalization and Schema Refinement	45
2. Redundancy and Anomalies Documentation	53
3. Normalization Process	56
Chapter 4: Implementation of Database	62
1. Revised DDL Scripts	62
2. Database Population.....	67
3. SQL Queries	131
Chapter 5: Interface Implementation	164
1. Setup JDBC and Basic GUI	164
2. CRUD Operations in GUI.....	188
Chapter 6: Technical Issues and Solution.....	196
1. Technical Issues	196
2. Solution.....	197

Chapter 1: Software Requirements Specification (SRS)

1. Problem Description

Case Study : The Startup Ecosystem Database

Meaning of Startup

A startup ecosystem is a network of people, organizations, and businesses that work together to create and grow new startups. These ecosystems can be physical or virtual, and can include

- Startups at different stages
- Entrepreneurs
- Investors
- Mentors and advisors
- Educational institutions
- Incubators and accelerators

Startup Ecosystem Database Meaning

A startup ecosystem database is a collection of information about a region's startup ecosystem, including startups, investors, and other stakeholders. These databases can help identify growth opportunities, encourage collaboration, and highlight a specific area's potential.

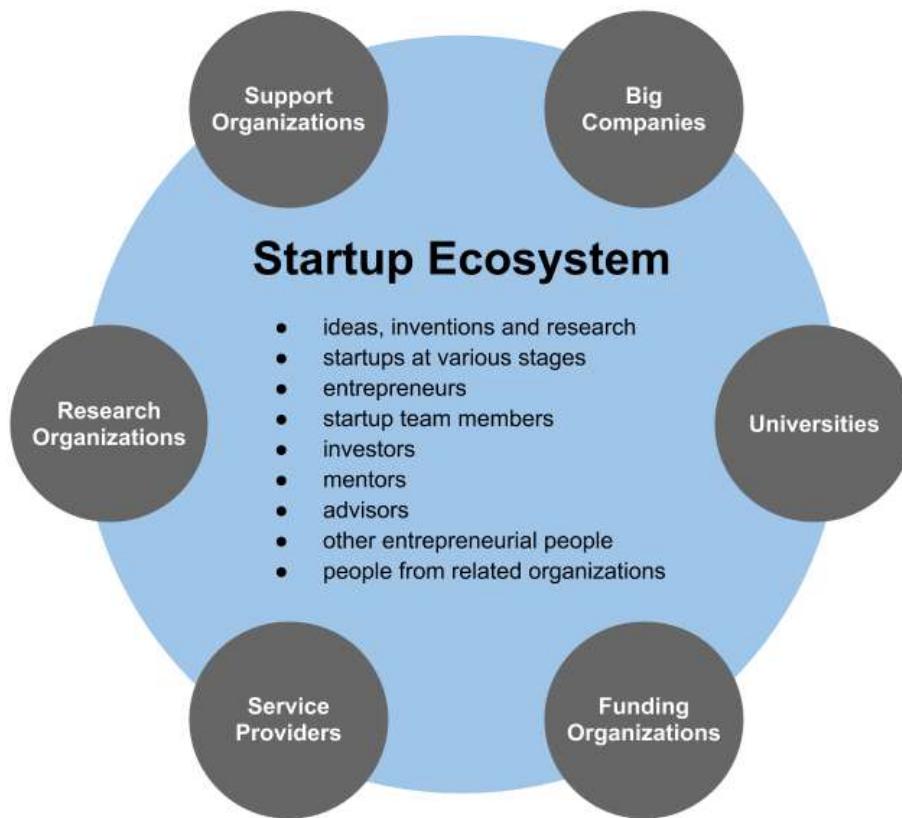
Suggestions

- Wright says : “Startup ecosystems rely on many factors, but people are by far the most important.”

- Startup Genome research shows that founders have unique skills and knowledge that can have outsized impact on the local economy when their talent is supported by key local stakeholders.”

Ref :- <https://www.startupblink.com/blog/building-a-database-for-your-startup-ecosystem/>

Key Elements Of Startup Ecosystem



Ref : <https://www.startupcommons.org/what-is-startup-ecosystem.html>

Challenges for Startups

1. Hiring and Managing Teams

- Difficulty in finding and retaining skilled employees due to competition from larger firms.
- Many founders lack business management experience, impacting operational efficiency.
- Startups often rely on bootstrapped funds or family support, making scaling challenging.

2. Diversity and the Digital Divide

- Understanding and catering to India's diverse customer base is complex.
- Regional differences limit startups' ability to scale nationally and effectively.
- Founders from urban areas may have limited insight into rural customer needs.

3. Market Penetration and Pricing

- High competition and presence of copycats make market entry difficult.
- Indian customers are highly price-sensitive, affecting startups' pricing strategies.
- Challenges in payment collection, especially without digital payment methods.

4. Hiring Qualified Employees

- Difficulty attracting talent due to perceived instability compared to larger corporations.
- Skills gap between academic training and industry requirements necessitates significant training.
- Limited ability to attract international talent due to bureaucratic hurdles.

5. Complex Regulatory Environment

- Navigating bureaucratic and regulatory processes is time-consuming and unpredictable.
- Issues with tax policies, including GST and "Angel Tax," create compliance challenges.

- Lengthy and costly setup processes act as barriers to starting and scaling businesses.

Ref : <https://www.orfonline.org/public/uploads/posts/pdf/20230725154621.pdf>

2. Requirement Collection

Background Reading

Intended Audience

1. Entrepreneurial Talent

- Entrepreneurs are the core of the ecosystem.
- They drive innovation and bring new ideas to life.
- Their vision and ambition push the ecosystem forward.

2. Investment Resources

- Comprises venture capital firms, angel investors, and other funding sources.
- Provides the financial backing necessary for startup growth and scaling.
- Enables startups to access crucial capital at different stages of their journey.

3. Support Infrastructure

- Includes incubators, accelerators, coworking spaces, and service providers.
- Offers physical infrastructure, mentorship, and networking opportunities.
- Provides essential services like legal, marketing, and technical support.

4. Educational Institutions and Research Organizations

- Supply new knowledge, innovation, and a talented workforce.
- Collaborate with startups for research support and technology transfer.
- Play a critical role in the ongoing development of entrepreneurial skills.

5. Government and Policy Framework

- Local government policies shape the ecosystem's success.
- Includes support like intellectual property protection and economic incentives.
- Affects startups through regulatory environments and policy support.

Ref: <https://augment.org/blog/startup-ecosystem>

Strategies to Build a Startup Ecosystem

- Evaluate the data of your ecosystem
- Provide a Startup-Business Friendly Environment
- Ensure the ecosystem has hubs for networking
- Boost the investment ecosystem
- Promote your ecosystem

Stages in Building a Startup Ecosystem



Product Scope

Startup ecosystems provide a variety of goods and services upon which other people and companies depend on. Thus, the principles of start-up ecosystem management suggest that rather than managing individual people or organizations, resources should be managed at the level of the startup ecosystem itself.

Ref : https://en.wikipedia.org/wiki/Startup_ecosystem

Indicators of growth in the startup ecosystem

- The pace of growth in the startup ecosystem has increased to 15% year-on-year in 2018, while the growth of the number of incubators and accelerators has grown to 11%.
- Significantly, the number of women entrepreneurs stood at 14%, up from 10% and 11% in the previous two years.
- Startups in the country have been able to create an estimated 40,000 new jobs over the year, taking the total jobs in the start-up ecosystem to 1.6-1.7 lakh.
- Bangalore has been listed within the world's 20 leading startup cities in the 2019 Startup Genome Project ranking. It is also ranked as one of the world's five fastest growing startup cities.

Ref : <https://www.startupindia.gov.in/content/sih/en/international/go-to-market-guide/indian-startup-ecosystem.html>

Building a Local Startup Ecosystem

1. Understanding Ecosystem Processes

- Comprehensive knowledge of the ecosystem's dynamics is crucial for its development.
- Failure to identify these processes can hinder the ecosystem's functionality.

2. Availability of Funding and Investment Opportunities

- Access to venture capital, angel investors, and other funding sources is essential for startup growth.
- Investment opportunities fuel innovation and scalability.

3. Presence of a Skilled and Educated Workforce

- A talented workforce with relevant skills is necessary for startup success.
- Collaboration with educational institutions can provide a steady stream of qualified individuals.

4. Access to Resources and Infrastructure

- Startups need access to resources such as technology, office space, and business services.
- A robust infrastructure supports operational efficiency and business growth.

5. Strategic Approach to Ecosystem Development

- Building a local startup ecosystem requires a well-planned strategy to address key factors.
- A coordinated effort among stakeholders helps promote a thriving entrepreneurial environment.

Ref : <https://www.ramotion.com/blog/startup-ecosystem/#section-elements-of-a-startup-ecosystem>

Analyzation of background reading

- **Vision and mission:** A strong vision and mission is essential for a successful startup ecosystem.

- **Stakeholders and programs:** A startup ecosystem needs a variety of stakeholders and supporting programs.
- **Measurements:** A startup ecosystem needs clear measurements to track its growth over time.
- **Local community:** Consider the needs of the local community.
- **Physical, social, and cultural environment:** Consider the physical, social, and cultural environment.
- **Political environment:** Consider the political environment.
- **Economic policies:** Consider the economic policies.
- **Availability of resources and talent:** Consider the availability of resources and talent.
- **Government initiatives:** The government can play a critical role by offering incentives, enabling connectivity, facilitating funding, and implementing a favorable regulatory environment.
- **Funding:** Startup funding is essential, and can come from loans, grants, or family and friends.
- **Ecosystem analysis and knowledge sharing:** Sharing stories of success can help create a culture of success and attract more entrepreneurs, investors, and talent.

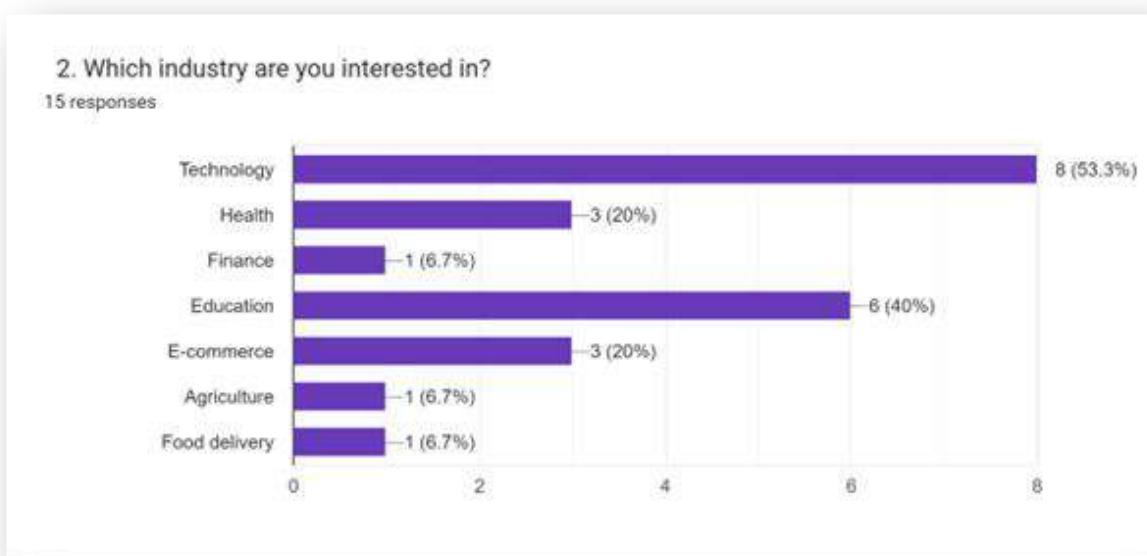
Interviews

- **Mentorship and Expertise:** Develop mentorship programs pairing founders with experienced industry professionals to guide strategic decisions.
- **Improved Funding Access:** Facilitate connections between startups and investors, focusing on seed and growth-stage funding.
- **Resource Sharing Platforms:** Create shared resources to reduce overhead costs for startups.
- **Streamlined Regulations:** Advocate for simplified regulatory requirements for startups to ease compliance and reduce entry barriers.
- **Networking Events:** Host networking events for startups to connect with investors, potential partners, and other entrepreneurs for collaboration opportunities.
- **Talent Acquisition Support:** Assist startups in finding skilled talent through partnerships with universities and job fairs targeting key skills.
- **Innovation and R&D Grants:** Provide or facilitate grants for startups engaged in innovative projects to encourage R&D.
- **Market Access Programs:** Create initiatives that help startups enter new markets, both locally and internationally.
- **Continuous Ecosystem Feedback:** Develop feedback loops with startups to understand their challenges and adapt ecosystem support services accordingly.
- **Data-Driven Insights:** Use analytics to identify trends and success patterns within the ecosystem, sharing these insights to benefit new startups and investors.

Questionnaires/Surveys

Summary Of Responses

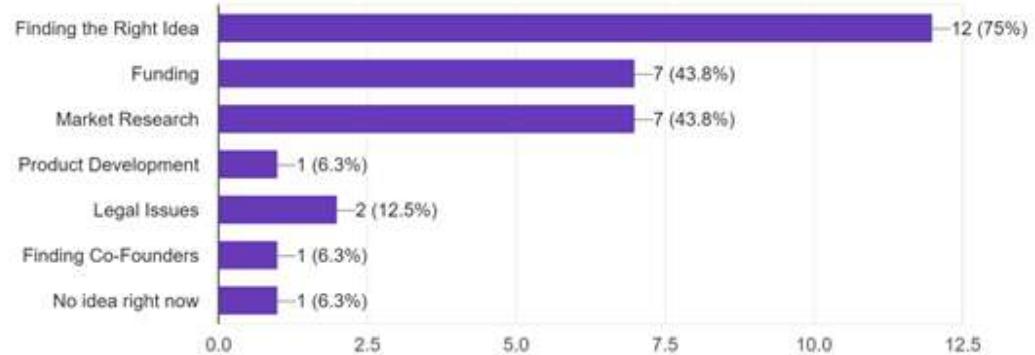
1. Interested Startup Industry



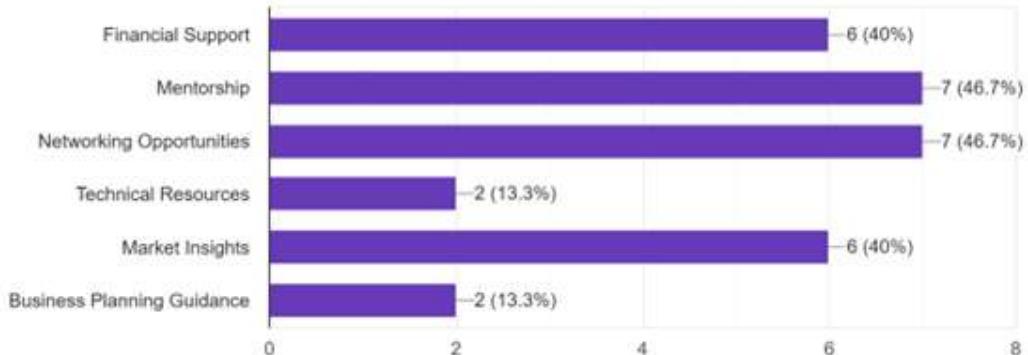
2. Challenges Faced by Startups

5. What are the biggest challenges you face in starting your business?

16 responses

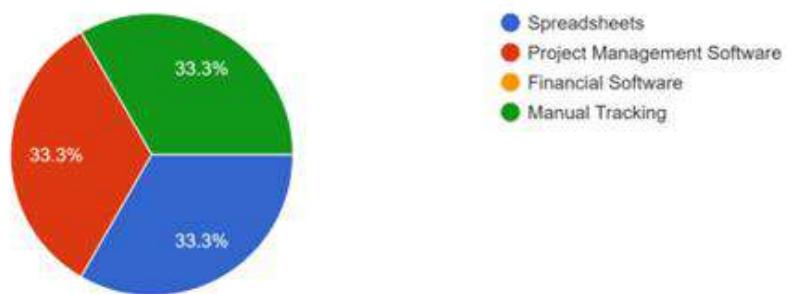
**3. Support Needed by Startups****6. What type of support do you need most to start your business?**

15 responses

**4. Tracking the Startup Progress**

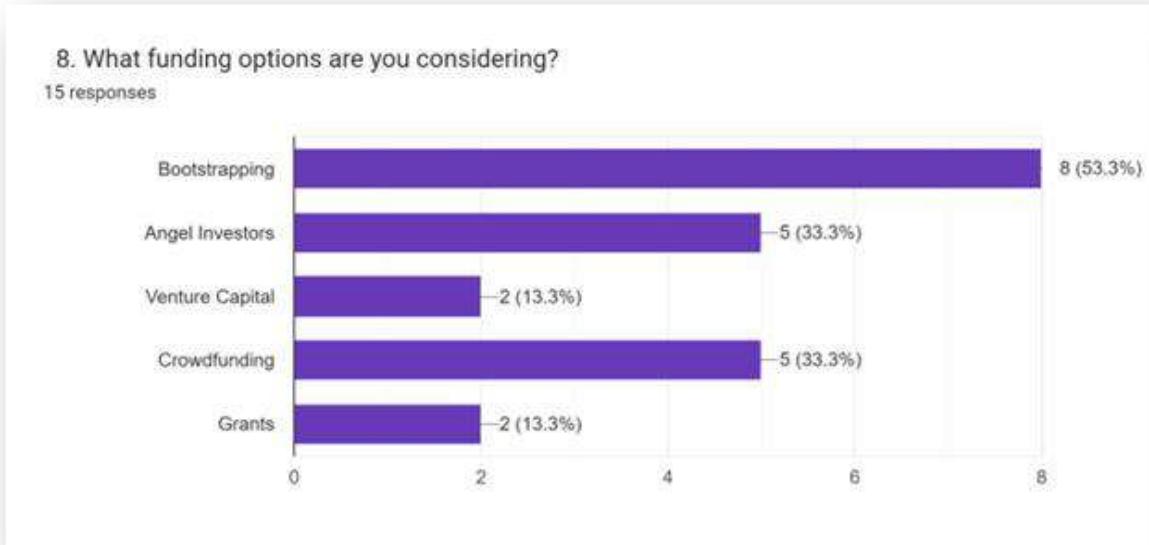
10. How do you currently track your startup's progress towards goals?

3 responses

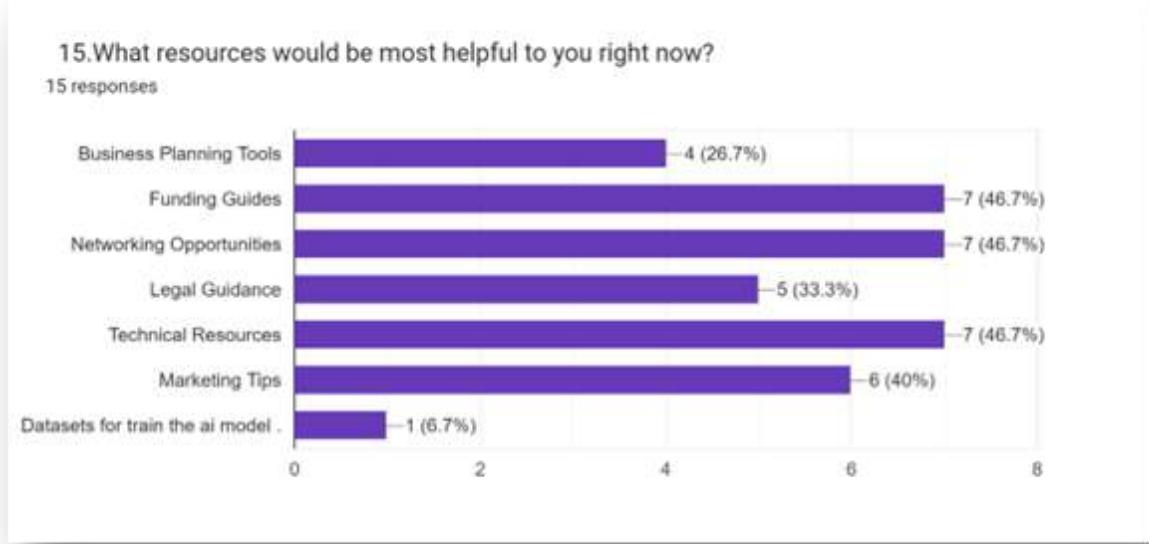
**5. Biggest barrier growth****10. What is your biggest concern about starting a startup?**

16 responses

**6. Funding Options**



7. Helpful Resources



Combined Summary from overall responses

- **Funding:** Bootstrapping is the most popular funding option among respondents,

followed by angel investors and crowdfunding. Venture capital and grants are less favored.

- **Resources:** Respondents expressed a strong need for Funding Guides, Networking Opportunities, and Technical Resources. Legal Guidance was also in high demand, while Business Planning Tools and Marketing Tips were less frequently requested.
- **Challenges:** The biggest challenges faced by startups include finding the right idea, securing funding, and navigating market research, legal compliance, and regulations.
- **Support:** Respondents sought various types of support, with Financial Support, Mentorship, and Networking Opportunities being the most requested.
- **Industry Interest:** Technology is the most popular industry for startups, followed by Education and Health.
- **Progress Tracking:** Respondents use a variety of methods to track their startup's progress, including spreadsheets, project management software, and manual tracking.

Overall, the data suggests that startups are seeking a combination of financial resources, mentorship, networking, and industry-specific knowledge to overcome challenges and achieve success.

Observation

Observation Summary

- **Sector Focus:** High interest in technology and education sectors, indicating demand for innovative solutions in these fields.
- **Funding Challenges:** Securing initial capital and ongoing funding is a significant hurdle for many startups, emphasizing the need for better access to investors.
- **Mentorship Needs:** Founders seek mentorship in business strategy, scaling, and market entry, highlighting the importance of connecting with experienced mentors.

- **Support Services:** Startups require support in legal matters, marketing, and strategic guidance, pointing to a need for accessible service platforms.
- **Networking and Community:** Strong demand for networking opportunities to build connections and gain insights from other founders and experts.
- **Market Insights:** Startups value tools that provide competitive analysis and market trend updates to make informed business decisions.
- **Comprehensive Support System:** Overall, the ecosystem needs a robust support structure encompassing funding access, mentorship, market intelligence, and community resources to drive growth and innovation.

3. Fact-Finding Chart

Objective	Technique	Subject(s)	Time Commitment
To gather information about the startup ecosystem	Background reading	Industry reports, case studies, news Articles	1-2 days
To identify key stakeholders and their roles	Interviews	Founders, investors, mentors, partners, industry experts	2-4 hours each
To understand the challenges faced by startups	Interviews, surveys	Founders, industry experts	1-2 hours each
To assess the current state of the ecosystem	Interviews, surveys	Founders, investors, mentors, industry experts	1-2 hours each

To identify gaps in existing resources and support	Interviews, surveys	Founders, industry experts	1-2 hours each
To explore potential areas for innovation and growth	Interviews, surveys	Founders, industry experts	1-2 hours each
To assess the effectiveness of existing initiatives	Interviews, surveys	Founders, investors, mentors, industry experts	1-2 hours each
To identify potential partnerships and collaborations	Interviews, networking events	Founders, investors, mentors, industry experts	1-2 hours each
To understand the regulatory environment	Legal research, interviews	Legal experts, government officials	1-2 hours
To assess the availability of funding and support resources	Interviews, research	Investors, funding organizations, support networks	1-2 hours each

4. Requirements List

1. Startup Information

- **Store startup details:** Name, industry, country, date of founding, stage, revenue, number of employees.
- **Track lifecycle stages:** Capture stages such as seed, growth, expansion, exit, etc.

2. Startup Owners Information

- **Founder information:** Capture personal details (name, email, expertise, LinkedIn profile, nationality).
- **Maintain relationships:** Establish and track the relationship between startups and their founders.

3. Investment and Funding Information

- **Funding rounds:** Record each funding round including investment type, amount raised, and funding date.
- **Investor information:** Capture details about investors, their investments, and equity stakes.
- **Valuation and equity:** Track the valuation of startups post-funding and calculate equity stake for each investor.

4. Accelerators and Incubators

- **Accelerator/incubator details:** Store information like name, location, focus area, and contact info.
- **Startup-accelerator relationship:** Track startup participation in accelerators or incubators, including joining and exit dates and program details.

5. Mentors and Advisory Support

- **Mentor information:** Store mentor details (name, email, area of expertise, LinkedIn profile).
- **Mentorship relationships:** Track mentorship relationships between startups and mentors, including start date, end date, and type of mentorship.

6. Partnerships

- **Partnership tracking:** Track partnerships between startups and other organizations, including partnership type and start and end dates.

7. Stakeholders

- **Stakeholder information:** Store information about other stakeholders in the startup ecosystem.

- **Track stakeholder impact:** Track the involvement and impact of stakeholders on the ecosystem.

8. Ecosystem Metrics

- **Metrics and reports:** Provide metrics and reports on the health and growth of the startup ecosystem, such as the number of startups, investments, jobs created, etc.
- **Filtering:** Allow filtering of data based on region, industry, or stage of development.

9. Events and Networking Opportunities

- **Event tracking:** Allow startups to register for and track events, workshops, and networking opportunities.
- **Notifications:** Provide notifications or reminders for relevant events in the ecosystem.

10. Challenges and Barriers

- **Track startup challenges:** Capture challenges startups face, including regulatory issues, talent shortage, or market penetration barriers.
- **Solutions:** Log potential solutions or strategies to overcome these challenges.

11. Government and Regulatory Support

- **Government support:** Track government frameworks and policies that support startups.
- **Legal compliance:** Capture legal and regulatory compliance details for startups.

12. Knowledge Sharing and Resources

- **Educational materials:** Provide resources like educational content, reports, success stories, and guidelines for entrepreneurs.
- **Knowledge sharing:** Allow startups to share their journey, best practices, and success stories with the community.

5. User Privileges

User Classes and Privileges

1. Startup CEO

- **Role:** Chief Executive Officer
- **Privileges:**
 - Full access to all startup data, including financials, team members, and strategic documents.
 - Ability to add, modify, or remove team members.
 - Authority to manage funding rounds and interact with investors.
 - Access to performance analytics of the startup, including revenue, growth metrics, and investor engagement.
 - Control over partnership agreements and contractual obligations.

2. Team Lead

- **Role:** Head of a specific department
- **Privileges:**
 - Access to department-specific data, including project progress, team performance, and resources.
 - Authority to manage tasks and assign work to team members within their department.
 - Ability to view team member performance metrics and provide feedback.
 - Limited access to financial data relevant to their department, such as budgets for projects or initiatives.

3. Team Member

Role: Employee contributing to the startup's projects

- **Privileges:**
 - Access to tasks, project plans, and collaborative tools related to their work.
 - Ability to update their progress on tasks and projects.

- o View access to relevant performance metrics for their role.
- o No access to sensitive financial data or information about other team members beyond their immediate team.

4. Investor

- **Role:** Individual or entity that provides funding to the startup
- **Privileges:**
 - o Access to investment portfolios and details about funding rounds they are involved in.
 - o Ability to view financial reports, including revenue and growth projections of the startup.
 - o Limited access to startup performance metrics, focusing primarily on their investment interests.
 - o No access to internal team operations or proprietary project details.

5. Accelerator Program Manager

- **Role:** Overseer of accelerator programs that support startups
- **Privileges:**
 - o Access to information about startups enrolled in the program, including progress reports and mentorship assignments.
 - o Ability to assign mentors and review feedback from mentors about the startups.
 - o View access to financial metrics and success metrics of the participating startups.
 - o Can track and manage resources allocated to startups within the accelerator.

6. Mentor

- **Role:** Experienced professional providing guidance to startups
- **Privileges:**
 - o Access to mentorship records and startup progress in areas of their expertise.

- o Ability to provide feedback and advice to the startups they are mentoring.
- o Can view performance metrics related to their mentees' growth and achievements.
- o No access to financial data or sensitive operational information outside their mentorship scope.

7. System Administrator

- **Role:** Responsible for overall system maintenance and user management
- **Privileges:**
 - o Full access to all system data, including user management and permissions.
 - o Authority to add, modify, or remove users from the system, including changing roles and permissions.
 - o Ability to run data backups and restore data as needed.
 - o Monitor system usage and enforce security protocols to protect sensitive information.

8. Government Stakeholder

- **Role:** Government officials interested in the startup ecosystem
- **Privileges:**
 - o Access to aggregate data on startup performance, funding, and job creation within the ecosystem.
 - o Ability to view reports on startup health and trends affecting the ecosystem.
 - o Limited access to information on initiatives supporting startups, without access to individual startup financial data.

9. Ecosystem Analyst

- **Role:** Individuals analyzing the startup ecosystem for research and development
- **Privileges:**
 - o Access to aggregate data and analytics regarding startup growth, investment trends, and ecosystem performance.
 - o Ability to create reports and present findings based on available data.

- o No access to sensitive or proprietary data of individual startups or personal data of users.

Chapter 2 : Database Design

1. Noun Analysis

Table - Accepted Noun & Verbs list

Candidate Entity Set	Attribute Set	Candidate Relationship Set
Startup	Startup_ID (PK), Name, Industry, Stage, Founded_Date, Location, Funding_Amount, Employee_Count	Connected to Founders, Investments, Mentors, Accelerators, Partnerships

Founder	Founder_ID (PK), fName, lName, Role, LinkedIn_Profile, Startup_ID (FK)	Connects to Startups through Founding
Investor	Investor_ID (PK), Name, Type, Investment_Stage_Preference, Location	Connects to Startups via Investments
Investment	Investment_ID (PK), Investor_ID (FK), Startup_ID (FK), Amount, Date, Stage	Connects Investors and Startups
Mentor	Mentor_ID (PK), Name, Expertise, Affiliation, LinkedIn_Profile	Connects to Startups for Mentorship
Accelerator	Accelerator_ID (PK), Name, Location, Industry_Focus, Batch_Size, Program_Duration	Connects to Startups through Enrollment
Partnership	Partnership_ID (PK), Startup_ID (FK), Partner_Organization_ID (FK), Type, Start_Date, End_Date	Connects Startups with Partner Organizations
Ecosystem Resource	Resource_ID (PK), Name, Type, Provider, Contact_Info	Connected to Startups for support
Industry	Industry_ID (PK), Name, Sector, Description	Categorizes Startups by Industry
Mentor Contact Info	Mentor_ID (FK), phone_no	Holds Mentor Contact Details
Accelerator Contact Info	Accelerator_ID (FK), phone_no	Holds Accelerator Contact Details
Investor Contact Info	Investor_ID (FK), phone_no	Holds Investor Contact Details

Startup-Mentor	Startup_ID (FK), Mentor_ID (FK), Date_Joined	Connects Startups and Mentors
Startup-Accelerator	Startup_ID (FK), Accelerator_ID (FK), Date_Joined	Connects Startups and Accelerators

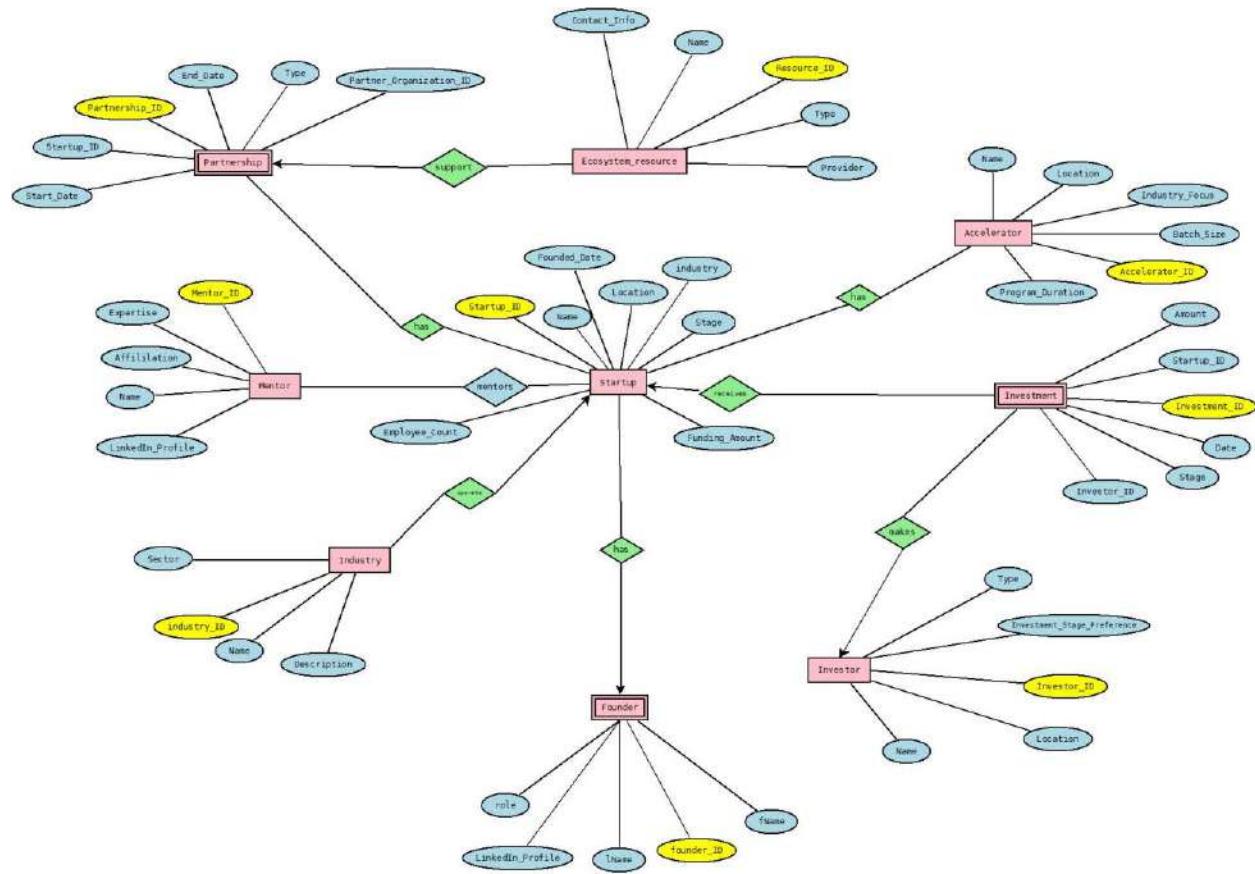
Relationships For ER Diagram Design

Entity 1	Entity 2	Relationship Type	Intermediate Table (if any)	Description
Startup	Founder	One-to-Many	-	Each startup can have multiple founders, and each founder is associated with only one startup.
Startup	Investor	Many-to-Many	Investment	An investor can invest in multiple startups, and a startup can receive funding from multiple investors.
Startup	Mentor	Many-to-Many	Startup_Mentor	Mentors can work with multiple startups, and startups can have guidance from multiple mentors.
Startup	Accelerator	Many-to-Many	Startup_Accelerator	Each startup can join multiple accelerators, and each accelerator can have multiple startups.
Startup	Partnership	One-to-Many	Partnership	Each partnership involves one startup and one partner organization,

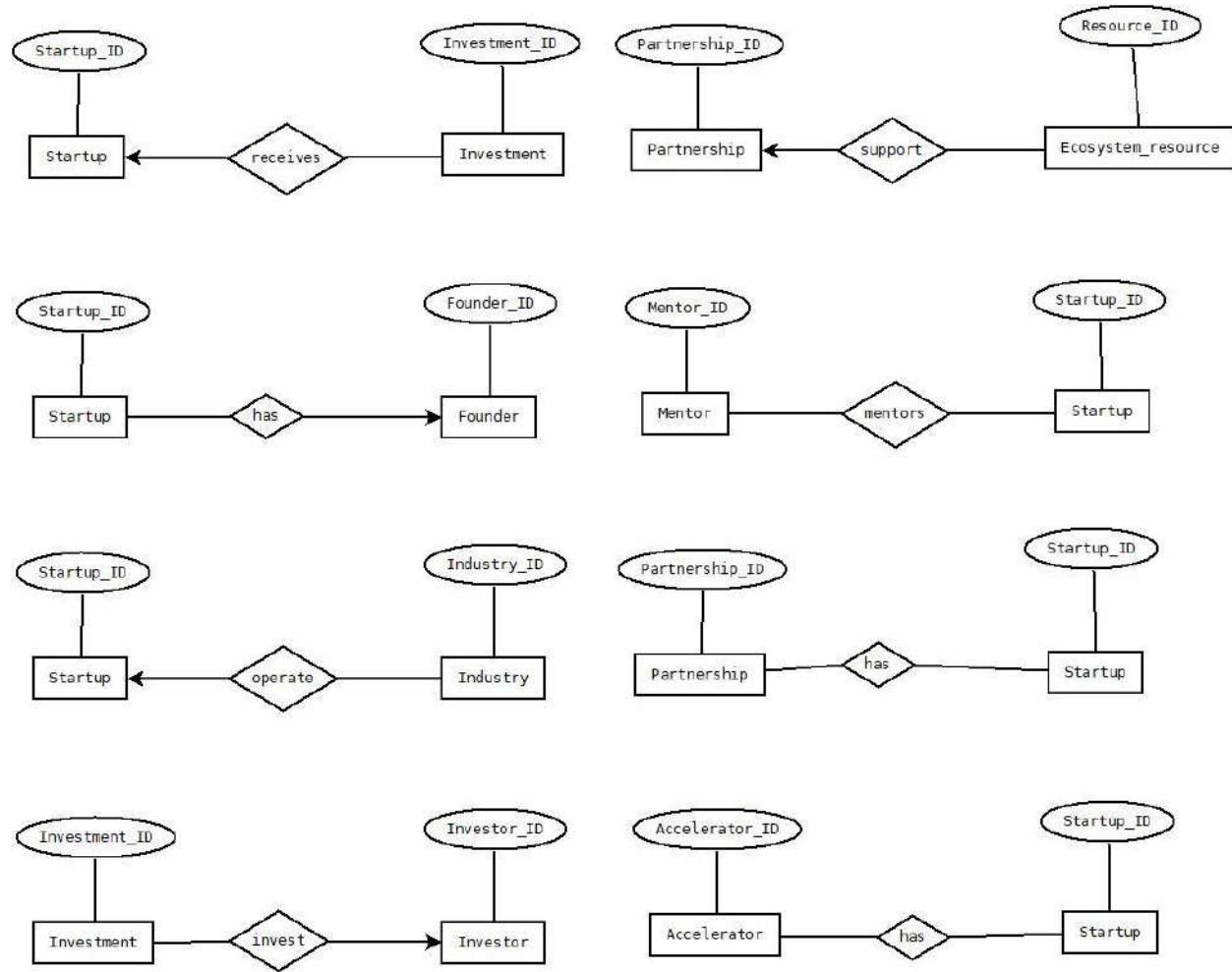
				but a startup can have multiple partnerships.
Accelerator	Ecosystem Resource	Implicit	-	Accelerators can be part of an ecosystem and provide specific resources to startups.
Industry	Startup	One-to-Many	-	Each startup is associated with an industry, while an industry can contain multiple startups.
Mentor	Mentor_contact_info	One-to-Many	-	Each mentor can have multiple contact entries stored in the Mentor_contact_info table.
Accelerator	Accelerator_contact_info	One-to-Many	-	Each accelerator can have multiple contact entries stored in the Accelerator_contact_info table.
Investor	Investor_contact_info	One-to-Many	-	Each investor can have multiple contact entries stored in the Investor_contact_info table.

2. Schema and ER Diagram Design

Entity Relationship Diagram



Schema Diagram



3. ER Diagram Improvement

Identify Entity Types

Entity	Type	Reason
Startup	Strong Entity	Has a primary key (Startup_ID) and exists independently.
Founder	Strong Entity	

		Has a primary key (Founder_ID) and is directly associated with Startup.
Investor	Strong Entity	Has a primary key (Investor_ID) and exists independently.
Investment	Weak Entity	Depends on both Investor and Startup entities for context, forming a many-to-many relationship.
Mentor	Strong Entity	Has a primary key (Mentor_ID) and exists independently.
Accelerator	Strong Entity	Has a primary key (Accelerator_ID) and exists independently.
Partnership	Weak Entity	Relies on the Startup and Ecosystem_Resource entities, forming a one-to-many relationship with Startup.
Ecosystem_Resource	Strong Entity	Has a primary key (Resource_ID) and exists independently.
Industry	Strong Entity	Has a primary key (Industry_ID) and exists independently.
Mentor_contact_info	Weak Entity	Dependent on Mentor, holding additional contact info, forming a one-to-many relationship.
Accelerator_contact_info	Weak Entity	

		Dependent on Accelerator, holding additional contact info, forming a one-to-many relationship.
Investor_contact_info	Weak Entity	Dependent on Investor, holding additional contact info, forming a one-to-many relationship.
Startup_Mentor	Associative Entity	Represents a many-to-many relationship between Startup and Mentor.
Startup_Accelerator	Associative Entity	Represents a many-to-many relationship between Startup and Accelerator.

Identify Relationship Types

Entity 1	Entity 2	Relationship Type	Details
Startup	Founder	One-to-Many	Each Startup can have multiple Founders, but each Founder is associated with only one Startup.
Startup	Investor	Many-to-Many	Represented by the Investment table, allowing multiple investors for a startup and multiple startups for an investor.
Startup	Mentor	Many-to-Many	Represented by Startup_Mentor, where a mentor can work with multiple startups, and startups can have multiple mentors.

Startup	Accelerator	Many-to-Many	Represented by <i>Startup_Accelerator</i> , where each startup can join multiple accelerators, and each accelerator can have multiple startups.
Startup	Partnership	One-to-Many	Each partnership involves one startup and one partner organization. A startup can have multiple partnerships.
Accelerator	Ecosystem_Resource	Implicit/Associative	Implicit relationship where accelerators are part of the ecosystem providing resources to startups.
Industry	Startup	One-to-Many	Each startup belongs to one industry, but an industry can have multiple startups.
Mentor	Mentor_contact_info	One-to-Many	Each mentor can have multiple contact entries.
Accelerator	Accelerator_contact_info	One-to-Many	Each accelerator can have multiple contact entries.
Investor	Investor_contact_info	One-to-Many	Each investor can have multiple contact entries.

ER Diagram Analysis

- Enhanced Relationships:** Associative tables (*Startup_Mentor*) track many-to-many relationships and can include additional details like mentor roles or timestamps.

2. **Attribute Refinement:** Entities like *Investment* and *Accelerator* could benefit from attributes like "Equity_Percentage" (for *Investment*) and "Success_Rate" (for *Accelerator*) to capture critical data insights.
3. **Optional Relationships:** Certain relationships, such as *Startup-Mentor*, are marked as optional to reflect varying startup stages, improving flexibility.

4. Mapping ER Model to Relational Model

Each entity and relationship mapped to a relational schema.

Startup

Attributes	DataTypes	Constraints
Startup_ID	int	PK
Industry	varchar	NOT NULL
Location	varchar	NOT NULL
Stage	varchar	NOT NULL
Funding_Amount	int	NOT NULL
Name	varchar	NOT NULL, UNIQUE
Founded_Date	date	NOT NULL, CHECK
Employee_Count	int	NOT NULL

Founder

Attributes	DataTypes	Constraints

Founder_ID	int	PK
fName	varchar	NOT NULL
IName	varchar	NOT NULL
Role	varchar	NOT NULL
Startup_ID	int	FK ref startup
LinkedIn_Profile	varchar	NOT NULL,UNIQUE

Investor

Attributes	DataTypes	Constraints
Investor_id	int	PK
Type	varchar	NOT NULL
Investment_Stage_Preference	varchar	NOT NULL
Name	varchar	NOT NULL,UNIQUE
Location	int	NOT NULL

Investment

Attributes	DataTypes	Constraints
Investment_ID	int	PK
Investor_ID	int	FK ref investor
Amount	int	NOT NULL
Date	date	NOT NULL
Stage	varchar	NOT NULL

Mentor

Attributes	DataTypes	Constraints
Mentor_id	int	PK
Name	varchar	NOT NULL
Expertise	varchar	NOT NULL
Affiliation	varchar	NOT NULL
LinkedIn_Profile	varchar	NOT NULL, UNIQUE

Accelerator

Attributes	DataTypes	Constraints
Accelerator_ID	int	PK
Name	varchar	NOT NULL, UNIQUE
Location	varchar	NOT NULL
Industry_focus	varchar	NOT NULL
Batch_Size	int	NOT NULL
Program_Duration	int	NOT NULL

Partnership

Attributes	DataTypes	Constraints
Partnership_ID	int	PK
Startup_ID	varchar	FK ref startup

Partner_Organization_ID	INT	FK ref partnership_organization
Type	varchar	NOT NULL
Start_Date	date	NOT NULL
End_Date	date	NOT_NULL,check

Ecosystem Resource

Attributes	DataTypes	Constraints
Resource_ID	int	PK
Name	varchar	NOT NULL
Type	varchar	NOT NULL
Provider	varchar	NOT NULL
Contact_info	int	NOT NULL, UNIQUE

Industry

Attributes	DataTypes	Constraints
Industry_ID	int	PK
Name	varchar	NOT NULL
Sector	varchar	NULL
Description	text	NOT NULL

Mentor_Contact_Info

Attributes	DataTypes	Constraints
Mentor_ID	int	Composite PK
Phone_no	int	Composite PK
Join_date	date	NOT NULL

Accelerator_Contact_Info

Attributes	DataTypes	Constraints
Accelerator_ID	int	Composite PK
Phone_no	int	Composite PK
Join_date	date	NOT NULL

Investor_Contact_Info

Attributes	DataTypes	Constraints
Investor_ID	int	Composite PK
Phone_no	int	Composite PK
Join_date	date	NOT NULL

Startup_Mentor

Attribute	Data Type	Constraints
Startup_ID	INT	Composite PK, FK (References Startup.Startup_ID)
Mentor_ID	INT	Composite PK, FK (References Mentor.Mentor_ID)
Date_joined	DATE	NOT NULL

Startup_Accelerator

Schemas written in format R1(A1,A2,...,An)R1(A1, A2, ..., An)R1(A1,A2,...,An)
with primary keys underlined.

- **R1: Startup** (Startup_ID, Name, Industry, Stage, Founded_Date, Location, Funding_Amount, Employee_Count)
- **R2: Founder** (Founder_ID, fName, lName, Role, LinkedIn_Profile, Startup_ID)
- **R3: Investor** (Investor_ID, Name, Type, Investment_Stage_Preference, Location)
- **R4: Investment** (Investment_ID, Investor_ID, Startup_ID, Amount, Date, Stage)
- **R5: Mentor** (Mentor_ID, Name, Expertise, Affiliation, LinkedIn_Profile)
- **R6: Accelerator** (Accelerator_ID, Name, Location, Industry_Focus, Batch_Size, Program_Duration)
- **R7: Partnership** (Partnership_ID, Startup_ID, Partner_Organization_ID, Type, Start_Date, End_Date)
- **R8: Ecosystem_Resource** (Resource_ID, Name, Type, Provider, Contact_Info)
- **R9: Industry** (Industry_ID, Name, Sector, Description)
- **R10: Mentor_contact_info** (Mentor_ID, phone_no)
- **R11: Accelerator_contact_info** (Accelerator_ID, phone_no)
- **R12: Investor_contact_info** (Investor_ID, phone_no)

- **R13: Startup_Mentor** (Startup_ID, Mentor_ID, Date_joined)
- **R14: Startup_Accelerator** (Startup_ID, Accelerator_ID, Date_joined)

5. Create DDL Scripts

1. Startup Table

```
CREATE TABLE Startup (
    Startup_ID INT PRIMARY KEY,
    Name VARCHAR(255) NOT NULL,
    Industry VARCHAR(100),
    Stage VARCHAR(50),
    Founded_Date DATE,
    Location VARCHAR(100),
    Funding_Amount DECIMAL(15, 2) CHECK (Funding_Amount >= 0),
    Employee_Count INT CHECK (Employee_Count >= 0)
);
```

2. Founder Table

```
CREATE TABLE Founder (
    Founder_ID INT PRIMARY KEY,
    fName VARCHAR(255) NOT NULL,
    lName VARCHAR(255) NOT NULL,
    Role VARCHAR(100),
    LinkedIn_Profile VARCHAR(255),
```

```

Startup_ID INT,
FOREIGN KEY (Startup_ID) REFERENCES Startup(Startup_ID)
);

```

3. Investor Table

```

CREATE TABLE Investor (
Investor_ID INT PRIMARY KEY,
Name VARCHAR(255) NOT NULL,
Type VARCHAR(50) NOT NULL,
Investment_Stage_Preference VARCHAR(50),
Location VARCHAR(100)
);

```

4. Investment Table

```

CREATE TABLE Investment (
Investment_ID INT PRIMARY KEY,
Investor_ID INT,
Startup_ID INT,
Amount DECIMAL(15, 2) CHECK (Amount >= 0),
Date DATE,
Stage VARCHAR(50),
FOREIGN KEY (Investor_ID) REFERENCES Investor(Investor_ID),
FOREIGN KEY (Startup_ID) REFERENCES Startup(Startup_ID)
);

```

5. Mentor Table

```

CREATE TABLE Mentor (

```

```

Mentor_ID INT PRIMARY KEY,
Name VARCHAR(255) NOT NULL,
Expertise VARCHAR(100),
Affiliation VARCHAR(100),
LinkedIn_Profile VARCHAR(255)
);

```

6. Accelerator Table

```

CREATE TABLE Accelerator (
    Accelerator_ID INT PRIMARY KEY,
    Name VARCHAR(255) NOT NULL,
    Location VARCHAR(100),
    Industry_Focus VARCHAR(100),
    Batch_Size INT CHECK (Batch_Size > 0),
    Program_Duration INT CHECK (Program_Duration > 0)
);

```

7. Partnership Table

```

CREATE TABLE Partnership (
    Partnership_ID INT PRIMARY KEY,
    Startup_ID INT,
    Partner_Organization_ID INT,
    Type VARCHAR(50),
    Start_Date DATE,
    End_Date DATE CHECK (End_Date >= Start_Date),
    FOREIGN KEY (Startup_ID) REFERENCES Startup(Startup_ID),
    FOREIGN KEY (Partner_Organization_ID) REFERENCES
        Ecosystem_Resource(Resource_ID)
);

```

```
);
```

8. Ecosystem Resource Table

```
CREATE TABLE Ecosystem_Resource (  
    Resource_ID INT PRIMARY KEY,  
    Name VARCHAR(255) NOT NULL,  
    Type VARCHAR(50),  
    Provider VARCHAR(100),  
    Contact_Info VARCHAR(255)  
);
```

9. Industry Table

```
CREATE TABLE Industry (  
    Industry_ID INT PRIMARY KEY,  
    Name VARCHAR(100) NOT NULL,  
    Sector VARCHAR(100),  
    Description TEXT  
);
```

10. Mentor Contact Info Table

```
CREATE TABLE Mentor_Contact_Info (  
    Mentor_ID INT,  
    Phone_No VARCHAR(15),  
    PRIMARY KEY (Mentor_ID, Phone_No),  
    FOREIGN KEY (Mentor_ID) REFERENCES Mentor(Mentor_ID)  
);
```

11. Accelerator Contact Info Table

```
CREATE TABLE Accelerator_Contact_Info (
    Accelerator_ID INT,
    Phone_No VARCHAR(15),
    PRIMARY KEY (Accelerator_ID, Phone_No),
    FOREIGN KEY (Accelerator_ID) REFERENCES Accelerator(Accelerator_ID)
);
```

12. Investor_Contact_Info Table

```
CREATE TABLE Investor_Contact_Info (
    Investor_ID INT,
    Phone_No VARCHAR(15),
    PRIMARY KEY (Investor_ID, Phone_No),
    FOREIGN KEY (Investor_ID) REFERENCES Investor(Investor_ID)
);
```

13. Startup_Mentor Table

```
CREATE TABLE Startup_Mentor (
    Startup_ID INT,
    Mentor_ID INT,
    Date_Joined DATE,
    PRIMARY KEY (Startup_ID, Mentor_ID),
    FOREIGN KEY (Startup_ID) REFERENCES Startup(Startup_ID),
    FOREIGN KEY (Mentor_ID) REFERENCES Mentor(Mentor_ID)
);
```

14. Startup_Accelerator Table

```
CREATE TABLE Startup_Accelerator (
    Startup_ID INT,
    Accelerator_ID INT,
    Date_Joined DATE,
    PRIMARY KEY (Startup_ID, Accelerator_ID),
    FOREIGN KEY (Startup_ID) REFERENCES Startup(Startup_ID),
    FOREIGN KEY (Accelerator_ID) REFERENCES Accelerator(Accelerator_ID)
);
```

Chapter 3: Normalization of Database

1. Normalization and Schema Refinement

Original Design of Database

1. Startup

- **Schema:** Startup(Startup_ID, Name, Industry, Stage, Founded_Date, Location, Funding_Amount, Employee_Count)

- **Primary Key:** Startup_ID
- **Description:** The Startup table stores details about individual startups, including their industry, stage, funding, and location.

2. Founder

- **Schema:** Founder(Founder_ID, fName, lName, Role, LinkedIn_Profile, Startup_ID)
- **Primary Key:** Founder_ID
- **Foreign Key:** Startup_ID references Startup(Startup_ID)
- **Description:** The Founder table keeps records of founders associated with each startup, linking them to the Startup table.

3. Investor

- **Schema:** Investor(Investor_ID, Name, Type, Investment_Stage_Preference, Location)
- **Primary Key:** Investor_ID
- **Description:** The Investor table contains information on investors, their types (e.g., angel investor, venture capital), preferred investment stages, and location.

4. Investment

- **Schema:** Investment(Investment_ID, Investor_ID, Startup_ID, Amount, Date, Stage)
- **Primary Key:** Investment_ID
- **Foreign Keys:**
 - Investor_ID references Investor(Investor_ID)
 - Startup_ID references Startup(Startup_ID)
- **Description:** The Investment table records investment transactions between investors and startups, including the amount, date, and stage of investment.

5. Mentor

- **Schema:** Mentor(Mentor_ID, Name, Expertise, Affiliation, LinkedIn_Profile)
- **Primary Key:** Mentor_ID

- **Description:** This table holds information about mentors, their expertise, affiliations, and LinkedIn profiles.

6. Accelerator

- **Schema:** Accelerator(Accelerator_ID, Name, Location, Industry_Focus, Batch_Size, Program_Duration)
- **Primary Key:** Accelerator_ID
- **Description:** The Accelerator table contains details about accelerators and incubators, including their industry focus, batch size, and program duration.

7. Partnership

- **Schema:** Partnership(Partnership_ID, Startup_ID, Partner_Organization_ID, Type, Start_Date, End_Date)
- **Primary Key:** Partnership_ID
- **Foreign Keys:**
 - Startup_ID references Startup(Startup_ID)
 - Partner_Organization_ID references Ecosystem_Resource(Resource_ID)
- **Description:** The Partnership table documents partnerships between startups and external ecosystem resources, specifying the type of partnership, start and end dates.

8. Ecosystem_Resource

- **Schema:** Ecosystem_Resource(Resource_ID, Name, Type, Provider, Contact_Info)
- **Primary Key:** Resource_ID
- **Description:** This table stores information on various resources available in the ecosystem, such as funding, legal, or marketing support, with contact details for each provider.

9. Industry

- **Schema:** Industry(Industry_ID, Name, Sector, Description)
- **Primary Key:** Industry_ID

- **Description:** The Industry table categorizes industries, listing their names, sectors, and descriptions.

10. Mentor_Contact_Info

- **Schema:** Mentor_Contact_Info(Mentor_ID, Phone_No)
- **Primary Key:** (Mentor_ID, Phone_No)
- **Foreign Key:** Mentor_ID references Mentor(Mentor_ID)
- **Description:** This table records multiple contact numbers for each mentor.

11. Accelerator_Contact_Info

- **Schema:** Accelerator_Contact_Info(Accelerator_ID, Phone_No)
- **Primary Key:** (Accelerator_ID, Phone_No)
- **Foreign Key:** Accelerator_ID references Accelerator(Accelerator_ID)
- **Description:** This table allows for multiple contact numbers for each accelerator.

12. Investor_Contact_Info

- **Schema:** Investor_Contact_Info(Investor_ID, Phone_No)
- **Primary Key:** (Investor_ID, Phone_No)
- **Foreign Key:** Investor_ID references Investor(Investor_ID)
- **Description:** This table holds contact numbers for each investor, enabling multiple contacts per investor.

13. Startup_Mentor

- **Schema:** Startup_Mentor(Startup_ID, Mentor_ID, Date_Joined)
- **Primary Key:** (Startup_ID, Mentor_ID)
- **Foreign Keys:**
 - Startup_ID references Startup(Startup_ID)
 - Mentor_ID references Mentor(Mentor_ID)
- **Description:** This join table represents a many-to-many relationship between startups and mentors, with the date each mentor joined the startup.

14. Startup_Accelerator

- **Schema:** Startup_Accelerator(Startup_ID, Accelerator_ID, Date_Joined)
- **Primary Key:** (Startup_ID, Accelerator_ID)
- **Foreign Keys:**
 - Startup_ID references Startup(Startup_ID)
 - Accelerator_ID references Accelerator(Accelerator_ID)
- **Description:** This table represents a many-to-many relationship between startups and accelerators, tracking when each startup joined an accelerator program.

Dependency Analysis

■ Identification of primary keys, foreign keys, and functional dependencies.

1. Startup Table

- **Primary Key:** Startup_ID
- **Functional Dependencies:**
 - $\text{Startup_ID} \rightarrow (\text{Name}, \text{Industry}, \text{Stage}, \text{Founded_Date}, \text{Location}, \text{Funding_Amount}, \text{Employee_Count})$
 - Explanation: Each Startup_ID uniquely determines all other attributes in the Startup table, such as Name, Industry, and Funding_Amount.
- **Foreign Keys:** None

2. Founder Table

- **Primary Key:** Founder_ID
- **Foreign Key:** Startup_ID references Startup(Startup_ID)
- **Functional Dependencies:**
 - $\text{Founder_ID} \rightarrow (\text{fName}, \text{lName}, \text{Role}, \text{LinkedIn_Profile}, \text{Startup_ID})$
 - Explanation: Each founder is uniquely identified by Founder_ID, which determines their first name, last name, role, LinkedIn profile, and associated startup (Startup_ID).

3. Investor Table

- **Primary Key:** Investor_ID
- **Functional Dependencies:**
 - Investor_ID → (Name, Type, Investment_Stage_Preference, Location)
 - Explanation: Each Investor_ID uniquely determines attributes like Name, Type, and Location.
- **Foreign Keys:** None

4. Investment Table

- **Primary Key:** Investment_ID
- **Foreign Keys:**
 - Investor_ID references Investor(Investor_ID)
 - Startup_ID references Startup(Startup_ID)
- **Functional Dependencies:**
 - Investment_ID → (Investor_ID, Startup_ID, Amount, Date, Stage)
 - Explanation: Each investment is uniquely identified by Investment_ID, determining the associated Investor_ID, Startup_ID, investment Amount, date, and stage.

5. Mentor Table

- **Primary Key:** Mentor_ID
- **Functional Dependencies:**
 - Mentor_ID → (Name, Expertise, Affiliation, LinkedIn_Profile)
 - Explanation: Each mentor is uniquely identified by Mentor_ID, which determines all other mentor attributes.
- **Foreign Keys:** None

6. Accelerator Table

- **Primary Key:** Accelerator_ID

- **Functional Dependencies:**
 - Accelerator_ID → (Name, Location, Industry_Focus, Batch_Size, Program_Duration)
 - Explanation: Each Accelerator_ID uniquely determines attributes like Name, Location, Industry_Focus, etc.
- **Foreign Keys:** None

7. Partnership Table

- **Primary Key:** Partnership_ID
- **Foreign Keys:**
 - Startup_ID references Startup(Startup_ID)
 - Partner_Organization_ID references Ecosystem_Resource(Resource_ID)
- **Functional Dependencies:**
 - Partnership_ID → (Startup_ID, Partner_Organization_ID, Type, Start_Date, End_Date)
 - Explanation: Each partnership is uniquely identified by Partnership_ID, determining the associated startup and partner organization, as well as the type, start, and end dates.

8. Ecosystem Resource Table

- **Primary Key:** Resource_ID
- **Functional Dependencies:**
 - Resource_ID → (Name, Type, Provider, Contact_Info)
 - Explanation: Each ecosystem resource is uniquely identified by Resource_ID, determining attributes like Name, Type, and Provider.
- **Foreign Keys:** None

9. Industry Table

- **Primary Key:** Industry_ID
- **Functional Dependencies:**

- Industry_ID → (Name, Sector, Description)
- Explanation: Each industry is uniquely identified by Industry_ID, which determines attributes such as Name, Sector, and Description.
- **Foreign Keys:** None

10. Mentor Contact Info Table

- **Primary Key:** (Mentor_ID, Phone_No)
- **Foreign Key:** Mentor_ID references Mentor(Mentor_ID)
- **Functional Dependencies:**
 - (Mentor_ID, Phone_No) → (Mentor_ID, Phone_No)
 - Explanation: Mentor_ID and Phone_No together form a composite primary key, identifying each unique phone number associated with a mentor.

11. Accelerator Contact Info Table

- **Primary Key:** (Accelerator_ID, Phone_No)
- **Foreign Key:** Accelerator_ID references Accelerator(Accelerator_ID)
- **Functional Dependencies:**
 - (Accelerator_ID, Phone_No) → (Accelerator_ID, Phone_No)
 - Explanation: Accelerator_ID and Phone_No together form a composite primary key, uniquely identifying each phone number associated with an accelerator.

12. Investor Contact Info Table

- **Primary Key:** (Investor_ID, Phone_No)
- **Foreign Key:** Investor_ID references Investor(Investor_ID)
- **Functional Dependencies:**
 - (Investor_ID, Phone_No) → (Investor_ID, Phone_No)
 - Explanation: Investor_ID and Phone_No together form a composite primary key, uniquely identifying each phone number associated with an investor.

13. Startup Mentor Table

- **Primary Key:** (Startup_ID, Mentor_ID)

- **Foreign Keys:**
 - Startup_ID references Startup(Startup_ID)
 - Mentor_ID references Mentor(Mentor_ID)
- **Functional Dependencies:**
 - $(\text{Startup_ID}, \text{Mentor_ID}) \rightarrow (\text{Startup_ID}, \text{Mentor_ID}, \text{Date_Joined})$
 - Explanation: The combination of Startup_ID and Mentor_ID uniquely determines the mentor's joining date for each startup.

14. Startup_Accelerator Table

- **Primary Key:** (Startup_ID, Accelerator_ID)
- **Foreign Keys:**
 - Startup_ID references Startup(Startup_ID)
 - Accelerator_ID references Accelerator(Accelerator_ID)

2. Redundancy and Anomalies Documentation

Redundancies

Schema	Table	Columns with Redundancy	Type of Redundancy	Description
SE	Founder	Startup_ID	Repetition	Multiple founders can reference the same Startup_ID, repeating information about the startup.
SE	Investment	Investor_ID, Startup_ID	Partial Dependency	The Investor_ID and Startup_ID might be repeated for each investment transaction.
SE	Accelerator	Location	Transitive Dependency	Accelerators in the same location may

				repeatedly list the same location.
SE	Partnership	Startup_ID, Partner_Organization_ID	Repetition	Partnerships may involve multiple instances with the same Startup_ID or Partner_Organization_ID
SE	Ecosystem_Resource	Contact_Info	Repetition	Contact information is limited to one entry, potentially causing duplication if multiple contacts are required.
SE	Mentor_contact_info	Mentor_ID, phone_no	Repetition	A mentor may have multiple phone numbers, causing repetition of Mentor_ID
SE	Investor_contact_info	Investor_ID, phone_no	Repetition	Investors with multiple contacts will have repeating Investor_ID entries.
SE	Startup_Mentor	Startup_ID, Mentor_ID	Repetition	Mentors associated with multiple startups cause Mentor_ID And Startup_ID repetition.
SE	Startup_Accelerator	Startup_ID, Accelerator_ID	Repetition	The same startup may be linked to multiple accelerators, causing repetition of Startup_ID

Anomalies

Table	Type of Anomaly	Description
Founder	Insert	

		Adding a new founder requires associating them with an existing Startup_ID , even if the startup has not yet been added.
Investment	Delete	Deleting a single investment can cause the loss of historical funding data tied to Investor_ID And Startup_ID
Accelerator	Update	Updating Location for one accelerator does not automatically update similar locations for other accelerators in the same area
Partnership	Insert	A new partnership may be difficult to add if the Partner_Organization_ID does not exist in Ecosystem_Resource
Ecosystem_Resource	Delete	Removing a resource could delete key data needed by Partnership and cause loss of partnerships.
Mentor_contact_info	Insert	Adding multiple phone numbers for the same mentor requires duplicating Mentor_ID
Startup_Mentor	Delete	Removing a relationship between a mentor and startup

		could result in a loss of mentorship history.
Startup_Accelerator	Update	Changing Accelerator_ID for a startup's history would require updating all related entries manually.

3. Normalization Process

1NF – Enforcing scalar values.

1. Startup Table

The Startup table already meets 1NF requirements, as each attribute (e.g., Startup_ID, Name, Industry, etc.) holds only a single value per row. No adjustments needed.

2. Founder Table

This table also meets 1NF, as each attribute holds only one value per record (e.g., Founder_ID, fName, lName, etc.).

3. Investor Table

The Investor table is already in 1NF, with attributes like Investor_ID, Name, Type, and Location containing atomic values.

4. Investment Table

The Investment table adheres to 1NF, with attributes like Investment_ID, Investor_ID, Startup_ID, and Amount each containing scalar values.

5. Mentor Table

The Mentor table is in 1NF with single-valued attributes like Mentor_ID, Name, Expertise, and Affiliation.

6. Mentor_Contact_Info Table

Initially, Mentor could have had multiple phone numbers (a multi-valued attribute). By creating a separate Mentor_Contact_Info table, we achieve 1NF by storing each phone number as a separate row.

7. Accelerator_Contact_Info Table

Similar to Mentor, the Accelerator entity could have multiple contact numbers. This was addressed by separating contact information into the Accelerator_Contact_Info table, thus maintaining 1NF.

8. Investor_Contact_Info Table

To ensure 1NF, the Investor table's contact numbers were moved into a separate Investor_Contact_Info table, allowing each contact number to be recorded in a single-valued format.

All other tables (e.g., Accelerator, Partnership, Ecosystem_Resource, Startup_Mentor, and Startup_Accelerator) already adhere to 1NF, as each field holds atomic values.

2NF – Eliminating partial dependencies.

1. Tables with Composite Primary Keys:

- **Startup_Mentor** and **Startup_Accelerator** tables both have composite primary keys, with (Startup_ID, Mentor_ID) and (Startup_ID, Accelerator_ID), respectively. In these tables, all non-key attributes should depend on the entire composite primary key.
- **No Partial Dependencies Identified:** Each non-key attribute in these tables, such as Date_Joined, depends on the entire composite primary key, so they already meet 2NF requirements.

2. Other Tables:

- **Single-Attribute Primary Keys:** Tables with single-attribute primary keys (e.g., Startup, Founder, Investor, Mentor, etc.) automatically meet 2NF requirements because there are no composite primary keys, meaning no partial dependencies can exist.
- **Dependency Structure:** Each non-key attribute depends entirely on the primary key in tables like Startup, Investment, and Mentor, so no additional changes are necessary for 2NF compliance.

Redundancy Analysis for 2NF

Startup Table

- **Redundancy:** The Location attribute in the Startup table might be repeated for multiple startups situated in the same location.
- **Description:** Multiple startups may be located in the same city, which leads to repeated entries of the Location attribute. This redundancy does not violate 2NF, as each non-key attribute depends on the primary key (Startup_ID), but it can be further optimized in 3NF.

Accelerator Table

- **Redundancy:** The Location attribute might also be repeated for multiple accelerators in the same city or region.
- **Description:** Just as with the Startup table, having the Location field in the Accelerator table can lead to data duplication if multiple accelerators are located in the same place. This redundancy does not violate 2NF but will be addressed when moving to 3NF.

Investment Table

- **Redundancy:** The Stage attribute, which indicates the investment stage (e.g., Seed, Series A), might be repeated for each investment that falls under the same stage.

- **Description:** Investment stages are often shared across multiple investments, leading to redundancy in the Stage attribute. This does not violate 2NF but could be further normalized by creating a separate Investment_Stage table in 3NF to avoid repetition.

Mentor Contact Info, Accelerator Contact Info, and Investor Contact Info Tables

- **Redundancy:** In each of these tables, the entity IDs (Mentor_ID, Accelerator_ID, and Investor_ID) are repeated for every unique phone number.
- **Description:** Each mentor, accelerator, or investor may have multiple phone numbers, resulting in repeated entries for the entity ID in these contact information tables. While this structure adheres to 2NF, it could be further refined in 3NF by using unique identifiers for each contact entry.

Partnership Table

- **Redundancy:** The Type attribute, which specifies the partnership type (e.g., Strategic, R&D), may appear multiple times if similar partnerships are formed across different startups.
- **Description:** Repetition of the Type attribute can occur if multiple partnerships of the same type are recorded for different startups. This redundancy can be minimized by creating a Partnership_Type table in 3NF to store unique partnership types.

Investor Table

- **Redundancy:** The Location attribute of investors may be repeated if multiple investors are from the same location.
- **Description:** The Location attribute, while not violating 2NF, could lead to data duplication if multiple investors reside in the same city or region. This redundancy could be removed by creating a Location table and referencing it in the Investor table in 3NF.

3NF/BCNF – Removing transitive dependencies.

Startup Table

- **Transitive Dependency:** Location could be a transitive dependency if multiple startups share the same location.
- **Solution:** Create a separate Location table to store unique locations and reference it in the Startup table.
- **New Structure:**
 - **Location Table:** Location(Location_ID, City, State, Country)
 - **Startup Table (updated):** Startup(Startup_ID, Name, Industry, Stage, Founded_Date, Location_ID, Funding_Amount, Employee_Count)
 - **Dependencies:**
 - Location_ID → (City, State, Country) in the Location table
 - Startup_ID → (Name, Industry, Stage, Founded_Date, Location_ID, Funding_Amount, Employee_Count) in the updated Startup table

Accelerator Table

- **Transitive Dependency:** Location is a transitive dependency here as well, potentially duplicated across multiple records.
- **Solution:** Use the new Location table and reference Location_ID in the Accelerator table.
- **New Structure:**
 - **Accelerator Table (updated):** Accelerator(Accelerator_ID, Name, Location_ID, Industry_Focus, Batch_Size, Program_Duration)

Investor Table

- **Transitive Dependency:** The Location attribute can also be duplicated in the Investor table.
- **Solution:** Use Location_ID in Investor to reference the Location table.
- **New Structure:**
 - **Investor Table (updated):** Investor(Investor_ID, Name, Type, Investment_Stage_Preference, Location_ID)

Investment Table

- **Transitive Dependency:** Stage is a transitive dependency if multiple investments share the same stage (e.g., Seed, Series A).
- **Solution:** Create an Investment_Stage table and reference it in the Investment table.
- **New Structure:**
 - **Investment_Stage Table:** Investment_Stage(Stage_ID, Stage_Name)
 - **Investment Table** (updated): Investment(Investment_ID, Investor_ID, Startup_ID, Amount, Date, Stage_ID)
- **Dependencies:**
 - Stage_ID → Stage_Name in the Investment_Stage table
 - Investment_ID → (Investor_ID, Startup_ID, Amount, Date, Stage_ID) in the updated Investment table

Partnership Table

- **Transitive Dependency:** Type is a transitive dependency if multiple partnerships share the same type (e.g., Strategic, R&D).
- **Solution:** Create a Partnership_Type table and reference it in the Partnership table.
- **New Structure:**
 - **Partnership_Type Table:** Partnership_Type(Type_ID, Type_Name)
 - **Partnership Table** (updated): Partnership(Partnership_ID, Startup_ID, Partner_Organization_ID, Type_ID, Start_Date, End_Date)
- **Dependencies:**
 - Type_ID → Type_Name in the Partnership_Type table
 - Partnership_ID → (Startup_ID, Partner_Organization_ID, Type_ID, Start_Date, End_Date) in the updated Partnership table

Chapter 4: Implementation of Database

1. Revised DDL Scripts

1. Startup Table

```
CREATE TABLE Startup (
    Startup_ID INT PRIMARY KEY,
    Name VARCHAR(255) NOT NULL,
    Industry VARCHAR(100),
    Stage VARCHAR(50),
    Founded_Date DATE,
    Location VARCHAR(100),
    Funding_Amount DECIMAL(15, 2) CHECK (Funding_Amount >= 0),
    Employee_Count INT CHECK (Employee_Count >= 0)
);
```

2. Founder Table

```
CREATE TABLE Founder (
    Founder_ID INT PRIMARY KEY,
    fName VARCHAR(255) NOT NULL,
    lName VARCHAR(255) NOT NULL,
    Role VARCHAR(100),
    LinkedIn_Profile VARCHAR(255),
    Startup_ID INT,
    FOREIGN KEY (Startup_ID) REFERENCES Startup(Startup_ID)
);
```

3. Investor Table

```
CREATE TABLE Investor (
    Investor_ID INT PRIMARY KEY,
    Name VARCHAR(255) NOT NULL,
    Type VARCHAR(50) NOT NULL,
    Investment_Stage_Preference VARCHAR(50),
    Location VARCHAR(100)
);
```

4. Investment Table

```
CREATE TABLE Investment (
    Investment_ID INT PRIMARY KEY,
    Investor_ID INT,
    Startup_ID INT,
    Amount DECIMAL(15, 2) CHECK (Amount >= 0),
```

```
Date DATE,  
Stage VARCHAR(50),  
FOREIGN KEY (Investor_ID) REFERENCES Investor(Investor_ID),  
FOREIGN KEY (Startup_ID) REFERENCES Startup(Startup_ID)  
);
```

5. Mentor Table

```
CREATE TABLE Mentor (  
Mentor_ID INT PRIMARY KEY,  
Name VARCHAR(255) NOT NULL,  
Expertise VARCHAR(100),  
Affiliation VARCHAR(100),  
LinkedIn_Profile VARCHAR(255)  
);
```

6. Accelerator Table

```
CREATE TABLE Accelerator (  
Accelerator_ID INT PRIMARY KEY,  
Name VARCHAR(255) NOT NULL,  
Location VARCHAR(100),  
Industry_Focus VARCHAR(100),  
Batch_Size INT CHECK (Batch_Size > 0),  
Program_Duration INT CHECK (Program_Duration > 0)  
);
```

7. Partnership Table

```
CREATE TABLE Partnership (
```

```

Partnership_ID INT PRIMARY KEY,
Startup_ID INT,
Partner_Organization_ID INT,
Type VARCHAR(50),
Start_Date DATE,
End_Date DATE CHECK (End_Date >= Start_Date),
FOREIGN KEY (Startup_ID) REFERENCES Startup(Startup_ID),
FOREIGN KEY (Partner_Organization_ID) REFERENCES
    Ecosystem_Resource(Resource_ID)
);

```

8. Ecosystem Resource Table

```

CREATE TABLE Ecosystem_Resource (
    Resource_ID INT PRIMARY KEY,
    Name VARCHAR(255) NOT NULL,
    Type VARCHAR(50),
    Provider VARCHAR(100),
    Contact_Info VARCHAR(255)
);

```

9. Industry Table

```

CREATE TABLE Industry (
    Industry_ID INT PRIMARY KEY,
    Name VARCHAR(100) NOT NULL,
    Sector VARCHAR(100),
    Description TEXT
);

```

10. Mentor_Contact_Info Table

```
CREATE TABLE Mentor_Contact_Info (
    Mentor_ID INT,
    Phone_No VARCHAR(15),
    PRIMARY KEY (Mentor_ID, Phone_No),
    FOREIGN KEY (Mentor_ID) REFERENCES Mentor(Mentor_ID)
);
```

11. Accelerator_Contact_Info Table

```
CREATE TABLE Accelerator_Contact_Info (
    Accelerator_ID INT,
    Phone_No VARCHAR(15),
    PRIMARY KEY (Accelerator_ID, Phone_No),
    FOREIGN KEY (Accelerator_ID) REFERENCES Accelerator(Accelerator_ID)
);
```

12. Investor_Contact_Info Table

```
CREATE TABLE Investor_Contact_Info (
    Investor_ID INT,
    Phone_No VARCHAR(15),
    PRIMARY KEY (Investor_ID, Phone_No),
    FOREIGN KEY (Investor_ID) REFERENCES Investor(Investor_ID)
);
```

13. Startup_Mentor Table

```
CREATE TABLE Startup_Mentor (
    Startup_ID INT,
```

```
Mentor_ID INT,  
Date_Joined DATE,  
PRIMARY KEY (Startup_ID, Mentor_ID),  
FOREIGN KEY (Startup_ID) REFERENCES Startup(Startup_ID),  
FOREIGN KEY (Mentor_ID) REFERENCES Mentor(Mentor_ID)  
);
```

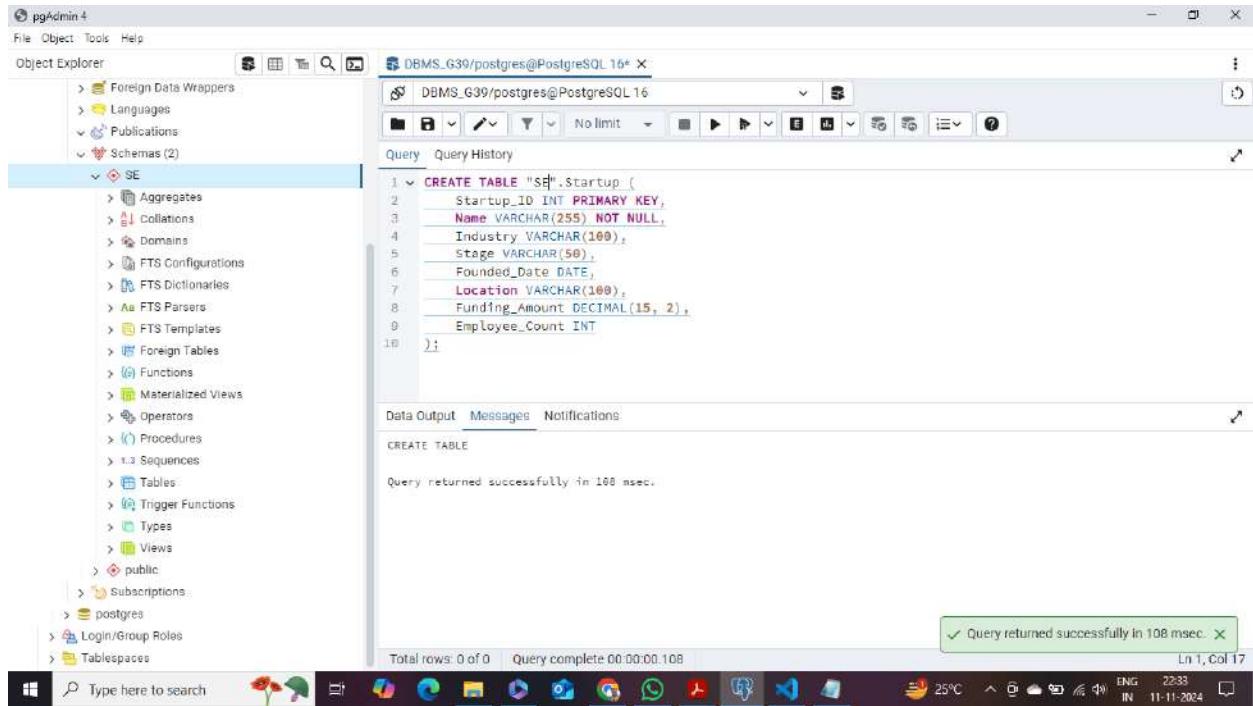
14. Startup Accelerator Table

```
CREATE TABLE Startup_Accelerator (  
    Startup_ID INT,  
    Accelerator_ID INT,  
    Date_Joined DATE,  
    PRIMARY KEY (Startup_ID, Accelerator_ID),  
    FOREIGN KEY (Startup_ID) REFERENCES Startup(Startup_ID),  
    FOREIGN KEY (Accelerator_ID) REFERENCES Accelerator(Accelerator_ID)  
);
```

2. Database Population

CREATE Table

Startup Table



The screenshot shows the pgAdmin 4 interface with the following details:

- File Object Tools Help**
- Object Explorer** pane on the left, expanded to show the **SE** schema, which contains:
 - Foreign Data Wrappers
 - Languages
 - Publications
 - Schemas (2)
 - Aggregates
 - Collations
 - Domains
 - FTS Configurations
 - FTS Dictionaries
 - FTS Parsers
 - FTS Templates
 - Foreign Tables
 - Functions
 - Materialized Views
 - Operators
 - Procedures
 - Sequences
 - Tables
 - Trigger Functions
 - Types
 - Views
 - public
 - Subscriptions
 - postgres
 - Login/Group Roles
 - Tablespaces
- DBMS_G39/postgres@PostgreSQL 16** tab selected.
- Query** tab active, containing the SQL code for creating the **Startup** table:

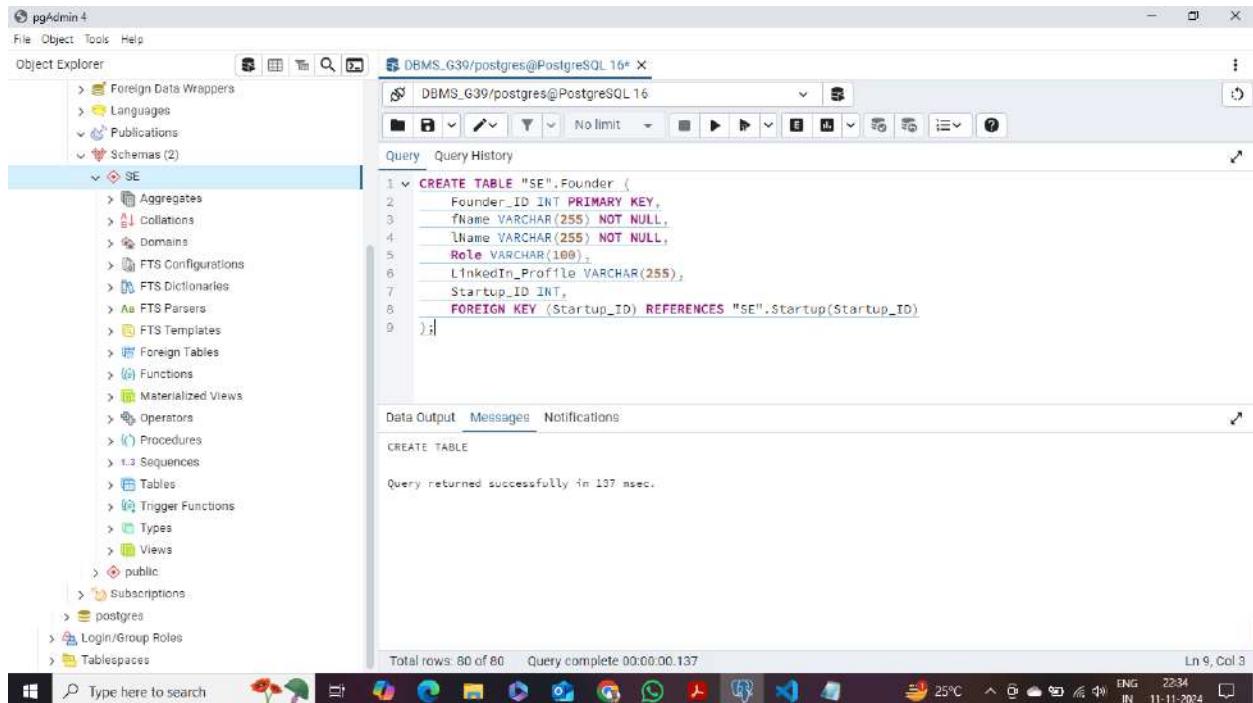
```

1 v CREATE TABLE "SE".Startup (
2   Startup_ID INT PRIMARY KEY,
3   Name VARCHAR(255) NOT NULL,
4   Industry VARCHAR(100),
5   Stage VARCHAR(50),
6   Founded_Date DATE,
7   Location VARCHAR(100),
8   Funding_Amount DECIMAL(15, 2),
9   Employee_Count INT
10 );

```

- Data Output** tab shows the message: "Query returned successfully in 100 msec."
- Messages** tab shows the message: "Total rows: 0 of 0 Query complete 00:00:00.108"
- Notifications** tab is empty.
- Bottom status bar: "Query returned successfully in 108 msec. Ln 1, Col 17"

Founder Table



The screenshot shows the pgAdmin 4 interface with the following details:

- File Object Tools Help**
- Object Explorer** pane on the left, expanded to show the **SE** schema, which contains:
 - Foreign Data Wrappers
 - Languages
 - Publications
 - Schemas (2)
 - Aggregates
 - Collations
 - Domains
 - FTS Configurations
 - FTS Dictionaries
 - FTS Parsers
 - FTS Templates
 - Foreign Tables
 - Functions
 - Materialized Views
 - Operators
 - Procedures
 - Sequences
 - Tables
 - Trigger Functions
 - Types
 - Views
 - public
 - Subscriptions
 - postgres
 - Login/Group Roles
 - Tablespaces
- DBMS_G39/postgres@PostgreSQL 16** tab selected.
- Query** tab active, containing the SQL code for creating the **Founder** table:

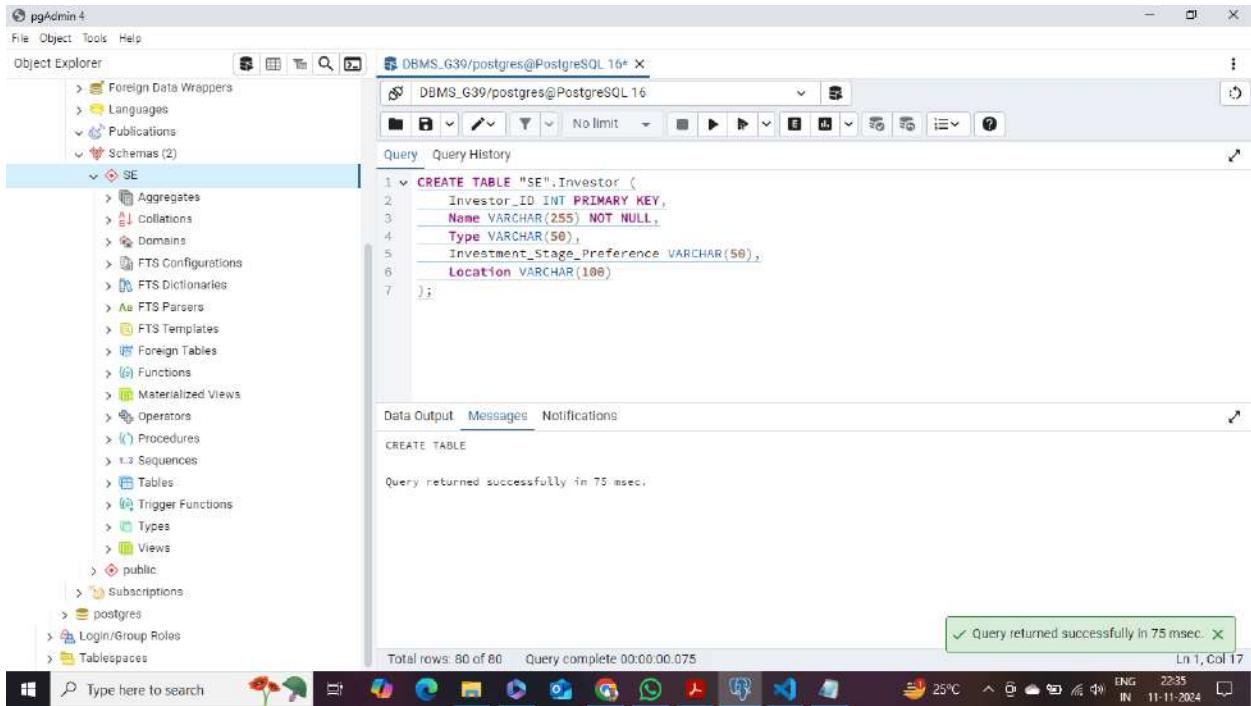
```

1 v CREATE TABLE "SE".Founder (
2   Founder_ID INT PRIMARY KEY,
3   fName VARCHAR(255) NOT NULL,
4   lName VARCHAR(255) NOT NULL,
5   Role VARCHAR(100),
6   LinkedIn_Profile VARCHAR(255),
7   Startup_ID INT,
8   FOREIGN KEY (Startup_ID) REFERENCES "SE".Startup(Startup_ID)
9 );

```

- Data Output** tab shows the message: "Query returned successfully in 137 msec."
- Messages** tab shows the message: "Total rows: 80 of 80 Query complete 00:00:00.137"
- Notifications** tab is empty.
- Bottom status bar: "Query returned successfully in 137 msec. Ln 9, Col 3"

Investor Table

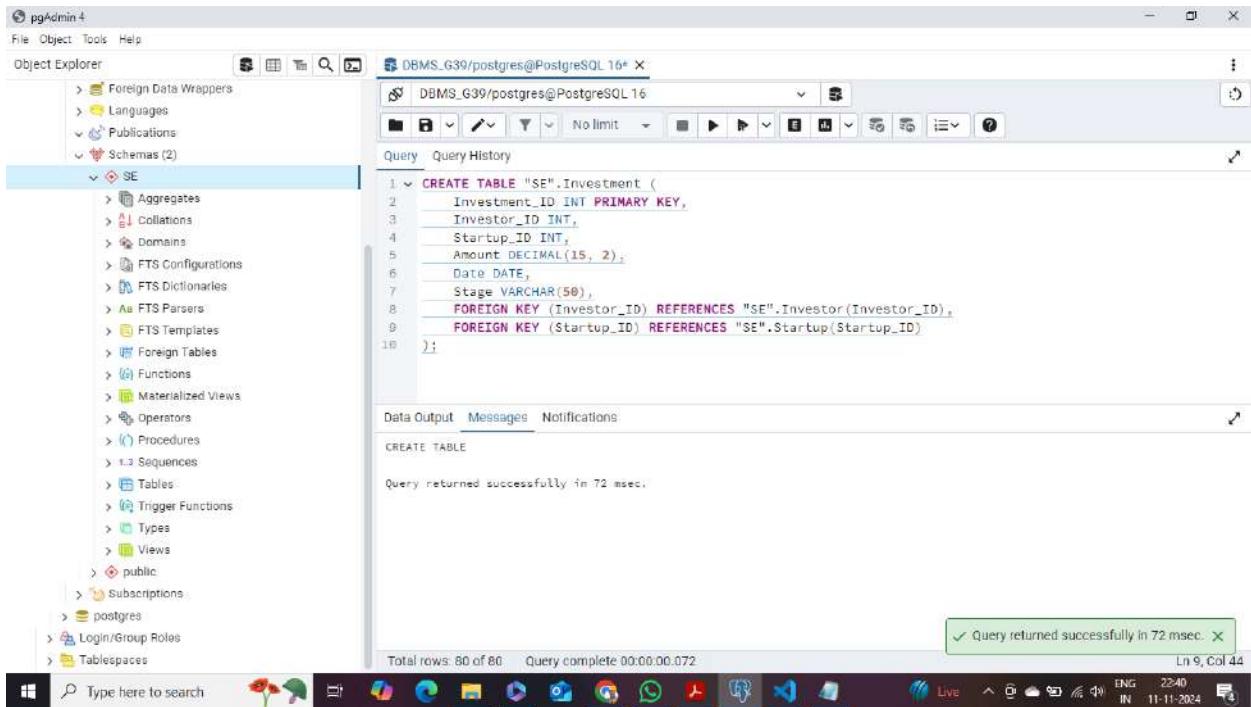


The screenshot shows the pgAdmin 4 interface with the following details:

- Object Explorer:** Shows the schema "SE" expanded, containing various database objects like Aggregates, Collations, Domains, FTS Configurations, etc.
- Query Editor:** Displays the SQL code for creating the "Investor" table:


```
1 v CREATE TABLE "SE".Investor (
2   Investor_ID INT PRIMARY KEY,
3   Name VARCHAR(255) NOT NULL,
4   Type VARCHAR(50),
5   Investment_Stage_Preference VARCHAR(50),
6   Location VARCHAR(100)
7 );
```
- Messages:** Shows a success message: "Query returned successfully in 75 msec."
- System Bar:** Includes the Windows taskbar with icons for File Explorer, Edge, and other applications, along with system status information (25°C, ENG IN, 11-11-2024).

Investment Table

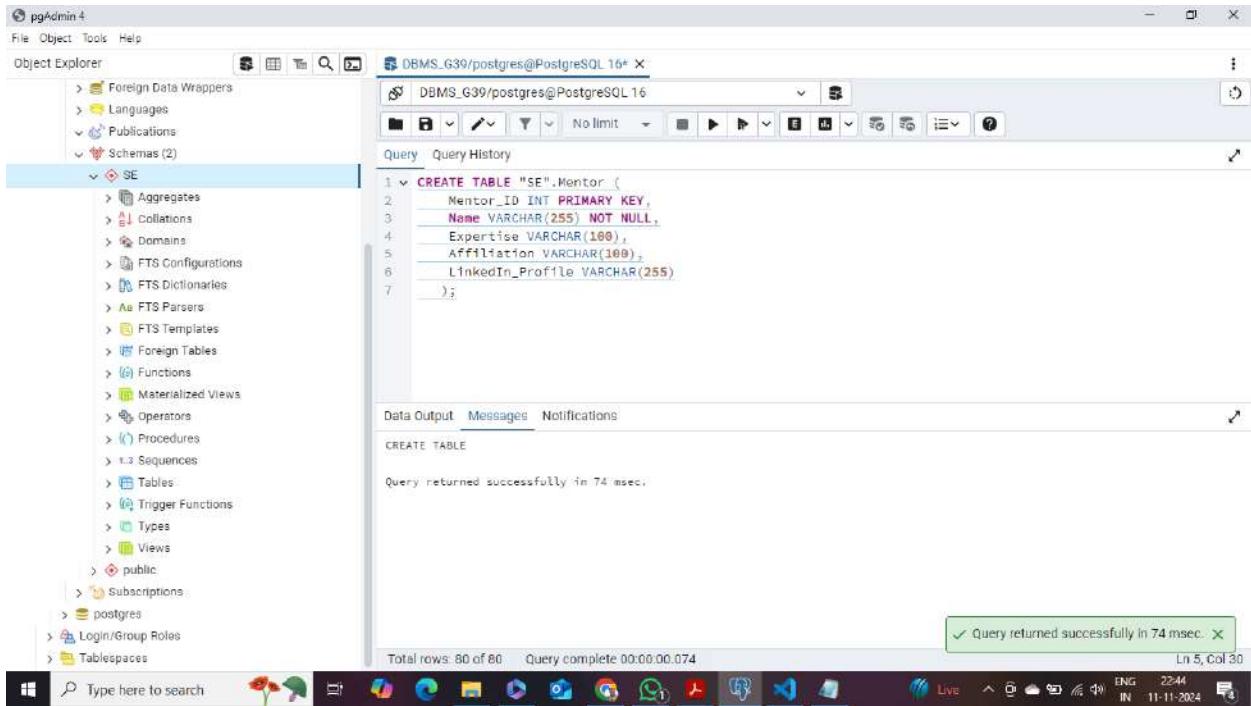


The screenshot shows the pgAdmin 4 interface with the following details:

- Object Explorer:** Shows the schema "SE" expanded, containing various database objects like Aggregates, Collations, Domains, FTS Configurations, etc.
- Query Editor:** Displays the SQL code for creating the "Investment" table:


```
1 v CREATE TABLE "SE".Investment (
2   Investment_ID INT PRIMARY KEY,
3   Investor_ID INT,
4   Startup_ID INT,
5   Amount DECIMAL(15, 2),
6   Date DATE,
7   Stage VARCHAR(50),
8   FOREIGN KEY (Investor_ID) REFERENCES "SE".Investor(Investor_ID),
9   FOREIGN KEY (Startup_ID) REFERENCES "SE".Startup(Startup_ID)
10 );
```
- Messages:** Shows a success message: "Query returned successfully in 72 msec."
- System Bar:** Includes the Windows taskbar with icons for File Explorer, Edge, and other applications, along with system status information (22:40, ENG IN, 11-11-2024).

Mentor Table

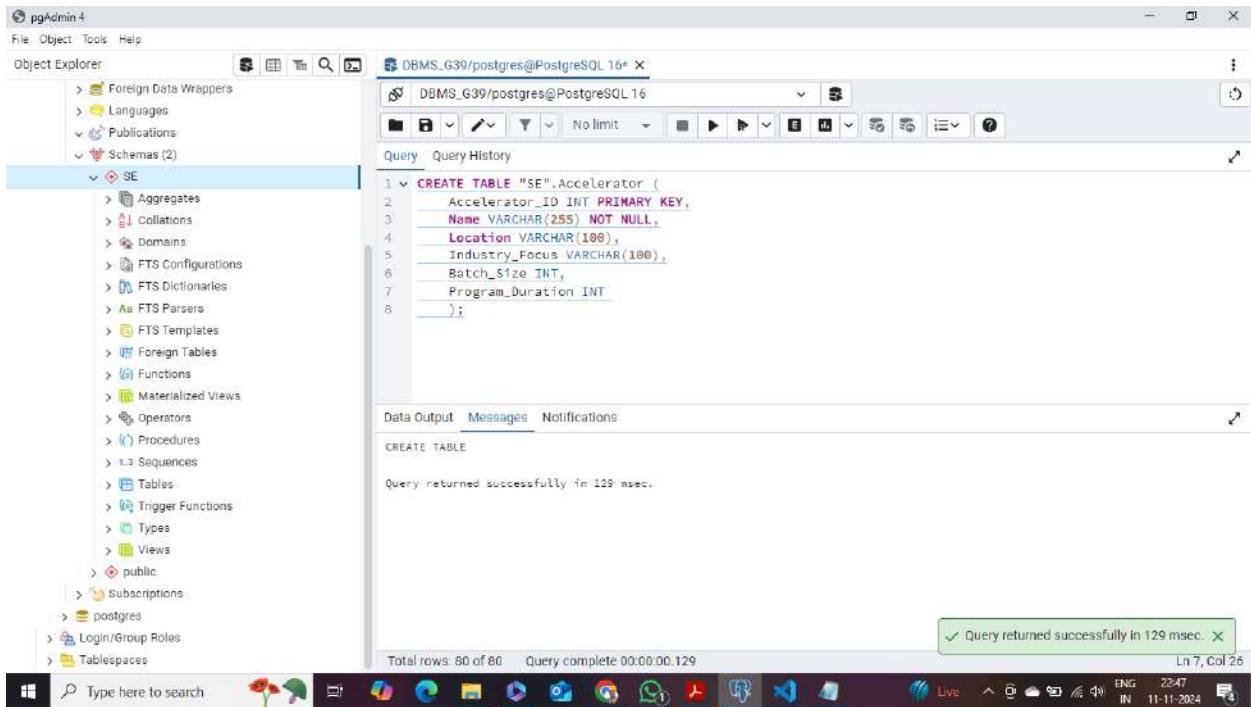


The screenshot shows the pgAdmin 4 interface with the following details:

- Object Explorer:** Shows the schema "SE" expanded, containing tables like Aggregates, Collations, Domains, FTS Configurations, FTS Dictionaries, FTS Parsers, FTS Templates, Foreign Tables, Functions, Materialized Views, Operators, Procedures, Sequences, Tables, Trigger Functions, Types, Views, and public.
- Query Editor:** Displays the SQL code for creating the "Mentor" table:


```
1 v CREATE TABLE "SE".Mentor (
2   Mentor_ID INT PRIMARY KEY,
3   Name VARCHAR(255) NOT NULL,
4   Expertise VARCHAR(100),
5   Affiliation VARCHAR(100),
6   LinkedIn_Profile VARCHAR(255)
7 );
```
- Data Output:** Shows the message "Query returned successfully in 74 msec."
- System Bar:** Shows the status bar with "Total rows: 80 of 80" and "Query complete 00:00:00.074".
- Taskbar:** Shows the Windows taskbar with various pinned icons.

Accelerator Table

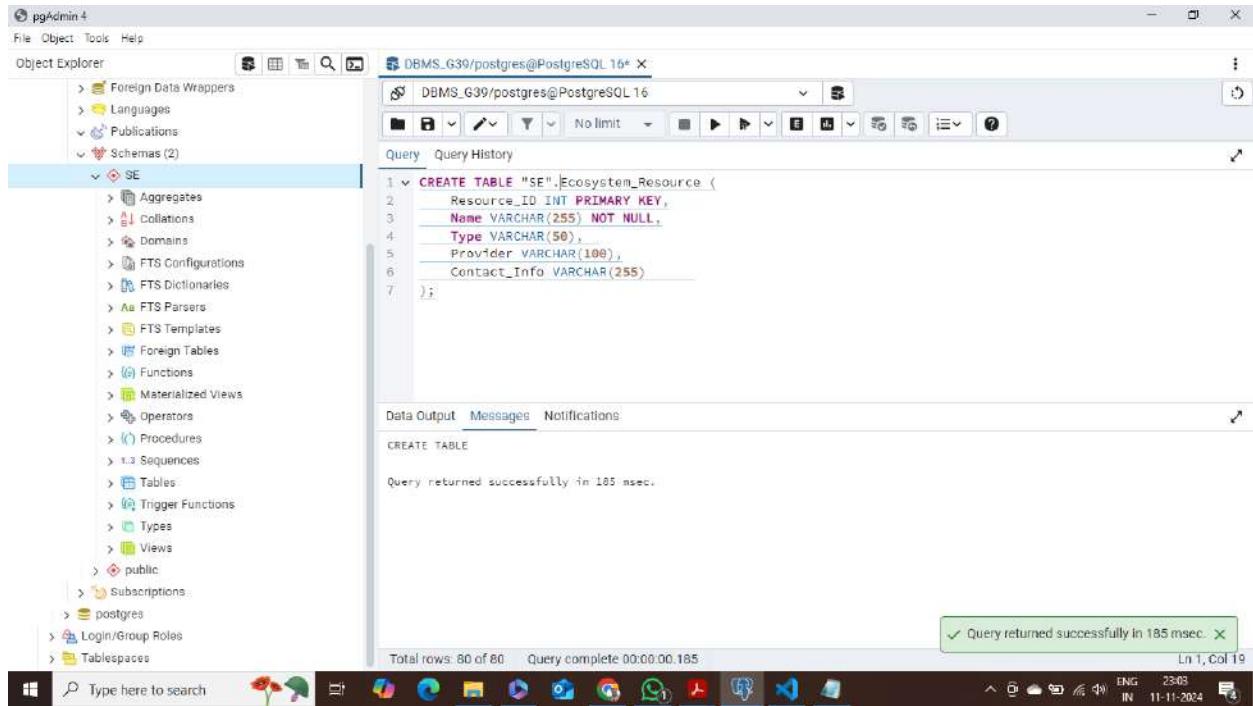


The screenshot shows the pgAdmin 4 interface with the following details:

- Object Explorer:** Shows the schema "SE" expanded, containing tables like Aggregates, Collations, Domains, FTS Configurations, FTS Dictionaries, FTS Parsers, FTS Templates, Foreign Tables, Functions, Materialized Views, Operators, Procedures, Sequences, Tables, Trigger Functions, Types, Views, and public.
- Query Editor:** Displays the SQL code for creating the "Accelerator" table:


```
1 v CREATE TABLE "SE".Accelerator (
2   Accelerator_ID INT PRIMARY KEY,
3   Name VARCHAR(255) NOT NULL,
4   Location VARCHAR(100),
5   Industry_Focus VARCHAR(100),
6   Batch_Size INT,
7   Program_Duration INT
8 );
```
- Data Output:** Shows the message "Query returned successfully in 129 msec."
- System Bar:** Shows the status bar with "Total rows: 80 of 80" and "Query complete 00:00:00.129".
- Taskbar:** Shows the Windows taskbar with various pinned icons.

Ecosystem_Resource Table

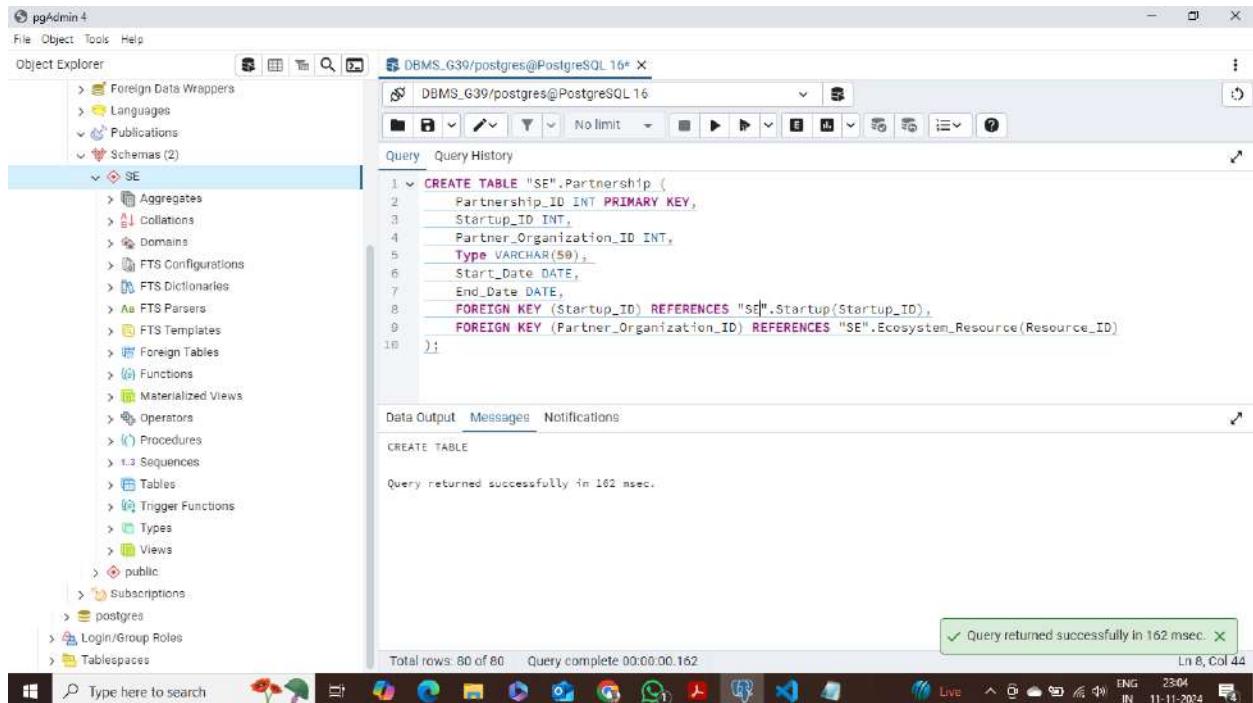


The screenshot shows the pgAdmin 4 interface with the following details:

- File Menu:** File, Object, Tools, Help.
- Object Explorer:** Shows various database objects like Foreign Data Wrappers, Languages, Publications, Schemas (2), SE, and others.
- Query Editor:** Displays the SQL code for creating the `Ecosystem_Resource` table:


```
1 v CREATE TABLE "SE".Ecosystem_Resource (
2   Resource_ID INT PRIMARY KEY,
3   Name VARCHAR(255) NOT NULL,
4   Type VARCHAR(50),
5   Provider VARCHAR(100),
6   Contact_Info VARCHAR(255)
7 );
```
- Data Output:** Shows the message: "Query returned successfully in 185 msec."
- Messages:** Shows the message: "Query returned successfully in 185 msec. X"
- Notifications:** No notifications are present.
- System Bar:** Shows "Total rows: 80 of 80" and "Query complete 00:00:00.185".
- Bottom Bar:** Shows the Windows taskbar with various pinned icons.

Partnership Table



The screenshot shows the pgAdmin 4 interface with the following details:

- File Menu:** File, Object, Tools, Help.
- Object Explorer:** Shows various database objects like Foreign Data Wrappers, Languages, Publications, Schemas (2), SE, and others.
- Query Editor:** Displays the SQL code for creating the `Partnership` table:


```
1 v CREATE TABLE "SE".Partnership (
2   Partnership_ID INT PRIMARY KEY,
3   Startup_ID INT,
4   Partner_Organization_ID INT,
5   Type VARCHAR(50),
6   Start_Date DATE,
7   End_Date DATE,
8   FOREIGN KEY (Startup_ID) REFERENCES "SE".Startup(Startup_ID),
9   FOREIGN KEY (Partner_Organization_ID) REFERENCES "SE".Ecosystem_Resource(Resource_ID)
10 );
```
- Data Output:** Shows the message: "Query returned successfully in 182 msec."
- Messages:** Shows the message: "Query returned successfully in 162 msec. X"
- Notifications:** No notifications are present.
- System Bar:** Shows "Total rows: 80 of 80" and "Query complete 00:00:00.162".
- Bottom Bar:** Shows the Windows taskbar with various pinned icons.

Industry Table

The screenshot shows the pgAdmin 4 interface. On the left, the Object Explorer pane displays various database objects under the 'SE' schema, including Foreign Data Wrappers, Languages, Publications, Schemas, SE, and others. The right pane shows a SQL query window with the following code:

```

1 v CREATE TABLE "SE".Industry (
2   Industry_ID INT PRIMARY KEY,
3   Name VARCHAR(100) NOT NULL,
4   Sector VARCHAR(100),
5   Description TEXT
6 );

```

Below the code, the status bar indicates 'Total rows: 0 of 0' and 'Query complete 00:00:00.147'. A message box at the bottom right says 'Query returned successfully in 147 msec.' and 'Ln 1, Col 17'.

Mentor_Contact_Info Table

The screenshot shows the pgAdmin 4 interface. On the left, the Object Explorer pane displays various database objects under the 'SE' schema. The right pane shows a SQL query window with the following code:

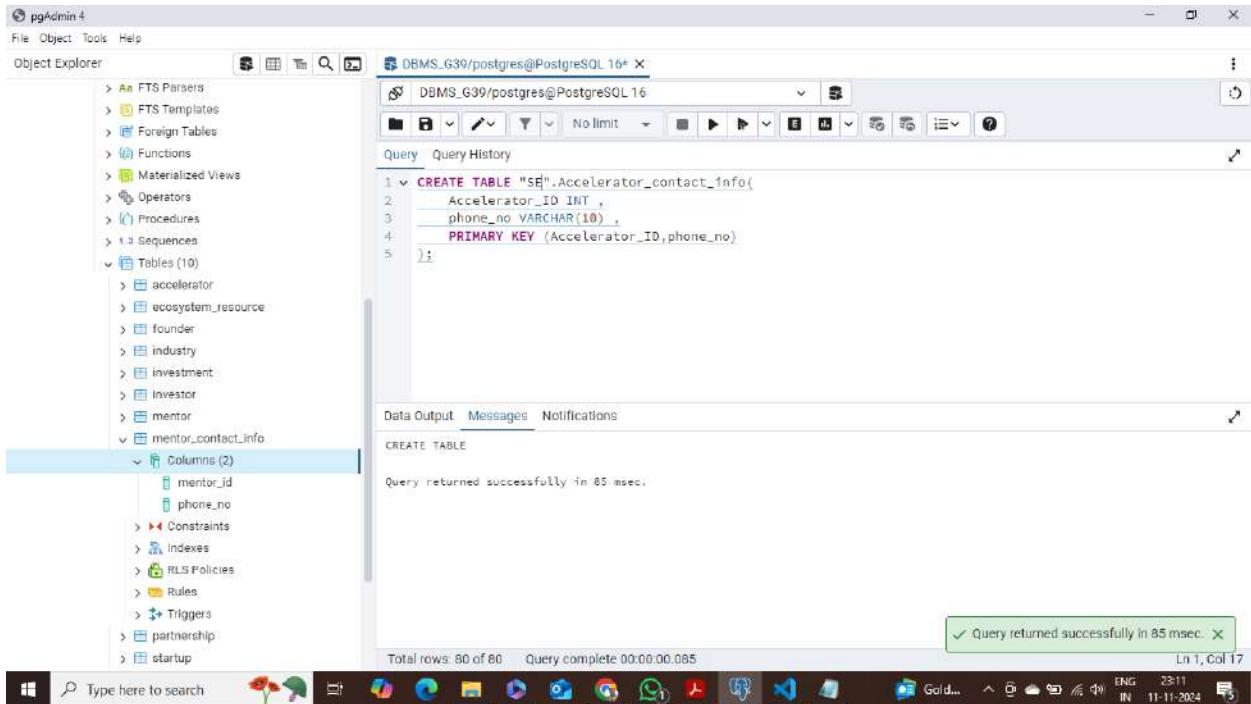
```

1 v CREATE TABLE "SE".Mentor_contact_info (
2   Mentor_ID INT ,
3   phone_no VARCHAR(20) ,
4   PRIMARY KEY (Mentor_ID,phone_no)
5 );

```

Below the code, the status bar indicates 'Total rows: 0 of 0' and 'Query complete 00:00:00.069'. A message box at the bottom right says 'Query returned successfully in 69 msec.' and 'Ln 1, Col 17'.

Accelerator_Contact_Info Table



The screenshot shows the pgAdmin 4 interface. In the Object Explorer, under the 'Tables (10)' section, the 'mentor_contact_info' table is selected. The 'Columns (2)' section shows two columns: 'mentor_id' and 'phone_no'. The 'Query' tab contains the SQL code for creating the table:

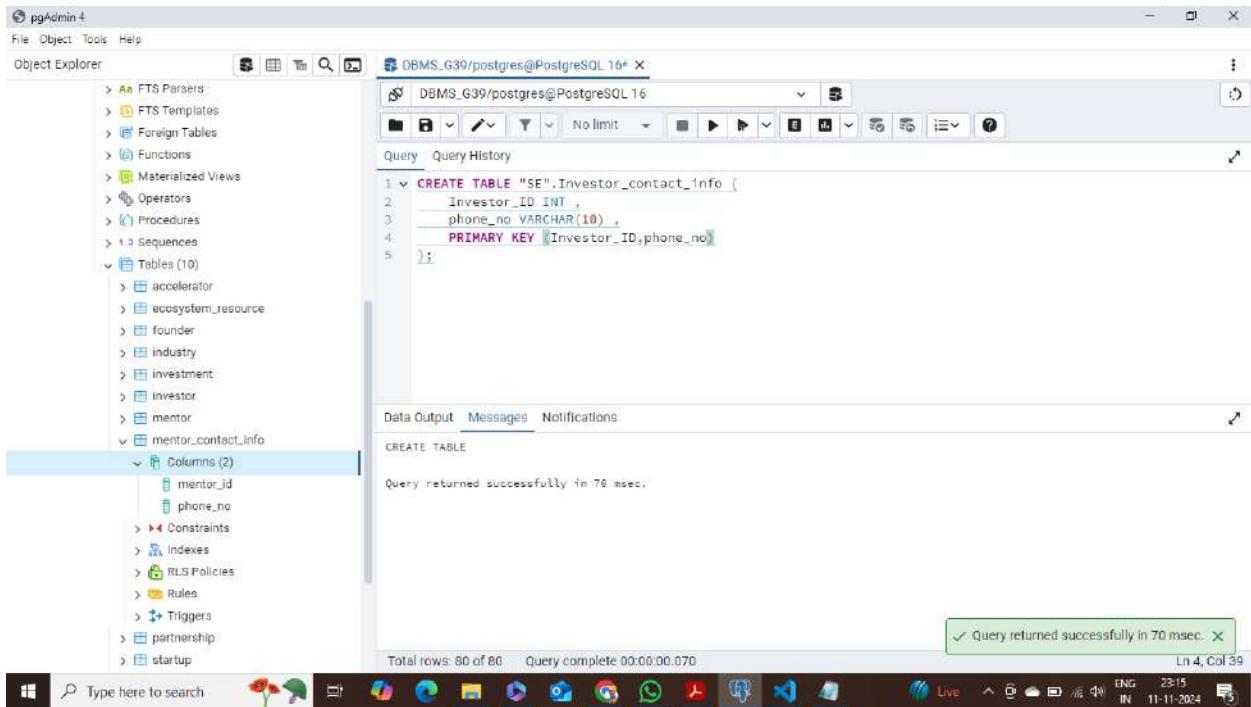
```

1 v CREATE TABLE "SE".Accelerator_contact_info(
2   Accelerator_ID INT ,
3   phone_no VARCHAR(10) ,
4   PRIMARY KEY (Accelerator_ID,phone_no)
5 );

```

The 'Messages' tab shows a success message: 'Query returned successfully in 85 msec.'

Investor_Contact_Info Table



The screenshot shows the pgAdmin 4 interface. In the Object Explorer, under the 'Tables (10)' section, the 'investor_contact_info' table is selected. The 'Columns (2)' section shows two columns: 'Investor_ID' and 'phone_no'. The 'Query' tab contains the SQL code for creating the table:

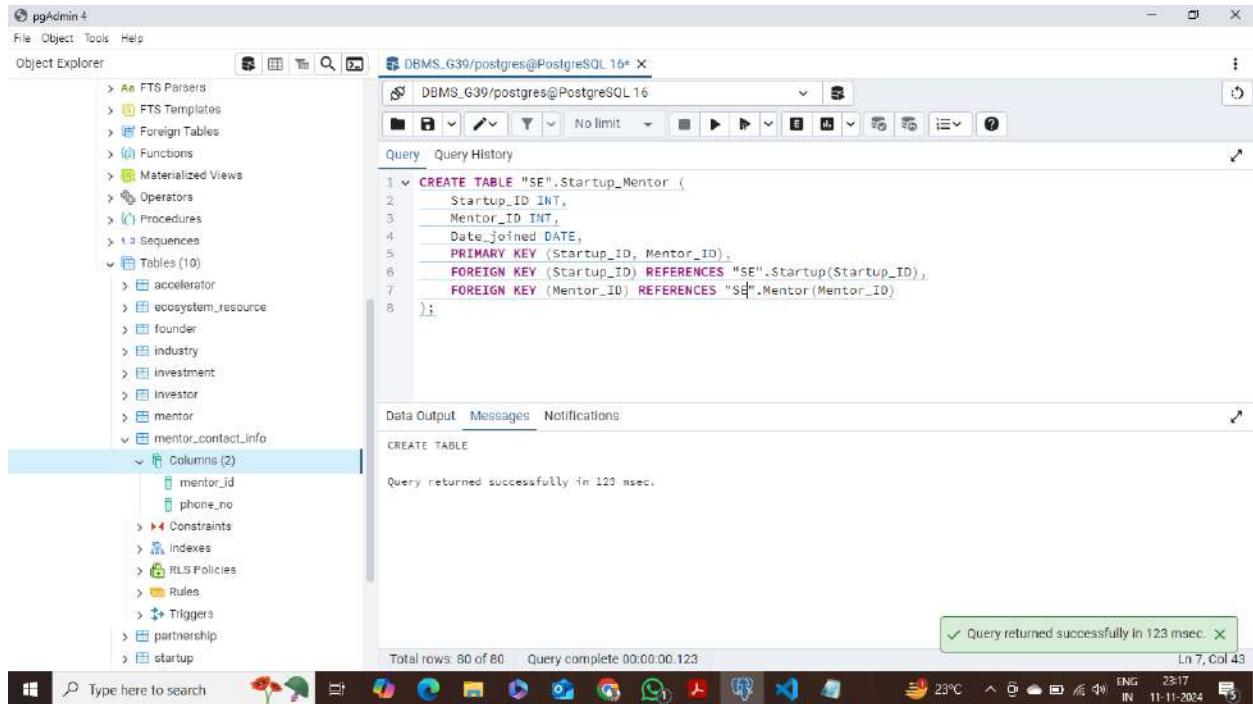
```

1 v CREATE TABLE "SE".Investor_contact_info (
2   Investor_ID INT ,
3   phone_no VARCHAR(10) ,
4   PRIMARY KEY (Investor_ID,phone_no)
5 );

```

The 'Messages' tab shows a success message: 'Query returned successfully in 70 msec.'

Startup_Mentor Table



The screenshot shows the pgAdmin 4 interface. The left pane is the Object Explorer, displaying a tree structure of database objects. The right pane is the Query Editor, showing the SQL code for creating the `Startup_Mentor` table. The code includes primary key and foreign key constraints referencing other tables.

```

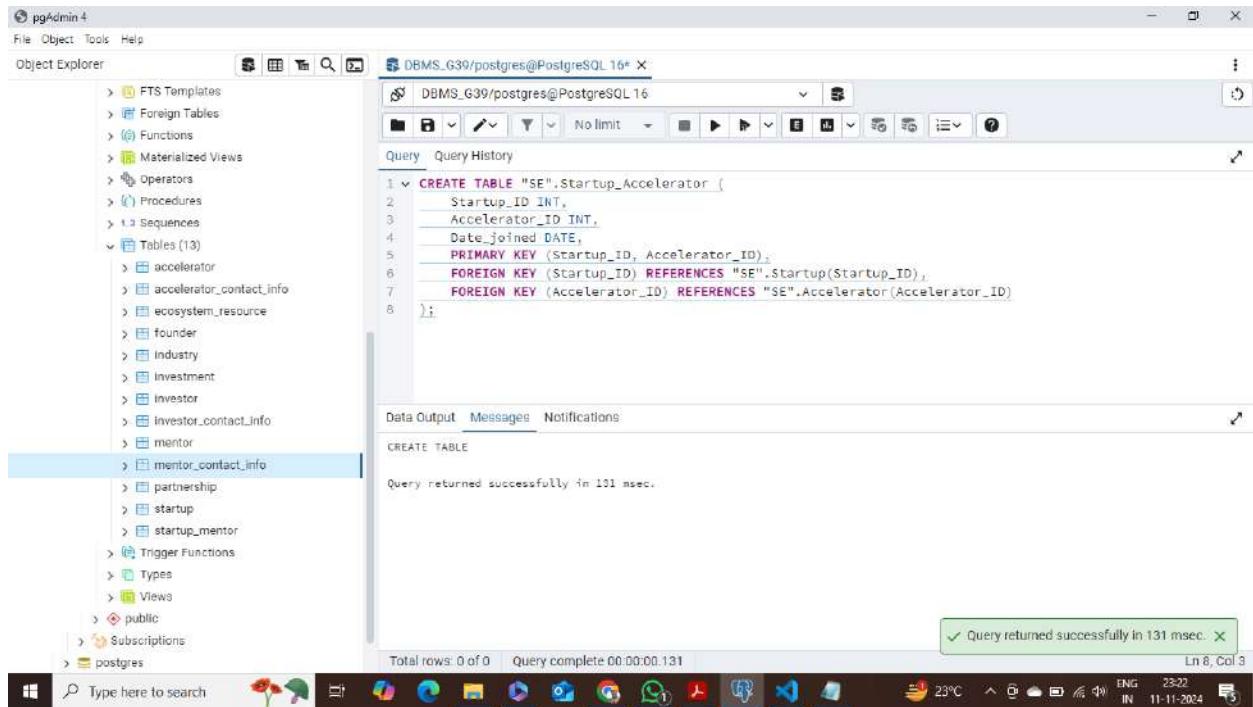
CREATE TABLE "SE".Startup_Mentor (
    Startup_ID INT,
    Mentor_ID INT,
    Date_joined DATE,
    PRIMARY KEY (Startup_ID, Mentor_ID),
    FOREIGN KEY (Startup_ID) REFERENCES "SE".Startup(Startup_ID),
    FOREIGN KEY (Mentor_ID) REFERENCES "SE".Mentor(Mentor_ID)
);

```

Query returned successfully in 120 msec.

Total rows: 0 of 0 Query complete 00:00:00.123

Startup_Accelerator Table



The screenshot shows the pgAdmin 4 interface. The left pane is the Object Explorer, displaying a tree structure of database objects. The right pane is the Query Editor, showing the SQL code for creating the `Startup_Accelerator` table. The code includes primary key and foreign key constraints referencing other tables.

```

CREATE TABLE "SE".Startup_Accelerator (
    Startup_ID INT,
    Accelerator_ID INT,
    Date_joined DATE,
    PRIMARY KEY (Startup_ID, Accelerator_ID),
    FOREIGN KEY (Startup_ID) REFERENCES "SE".Startup(Startup_ID),
    FOREIGN KEY (Accelerator_ID) REFERENCES "SE".Accelerator(Accelerator_ID)
);

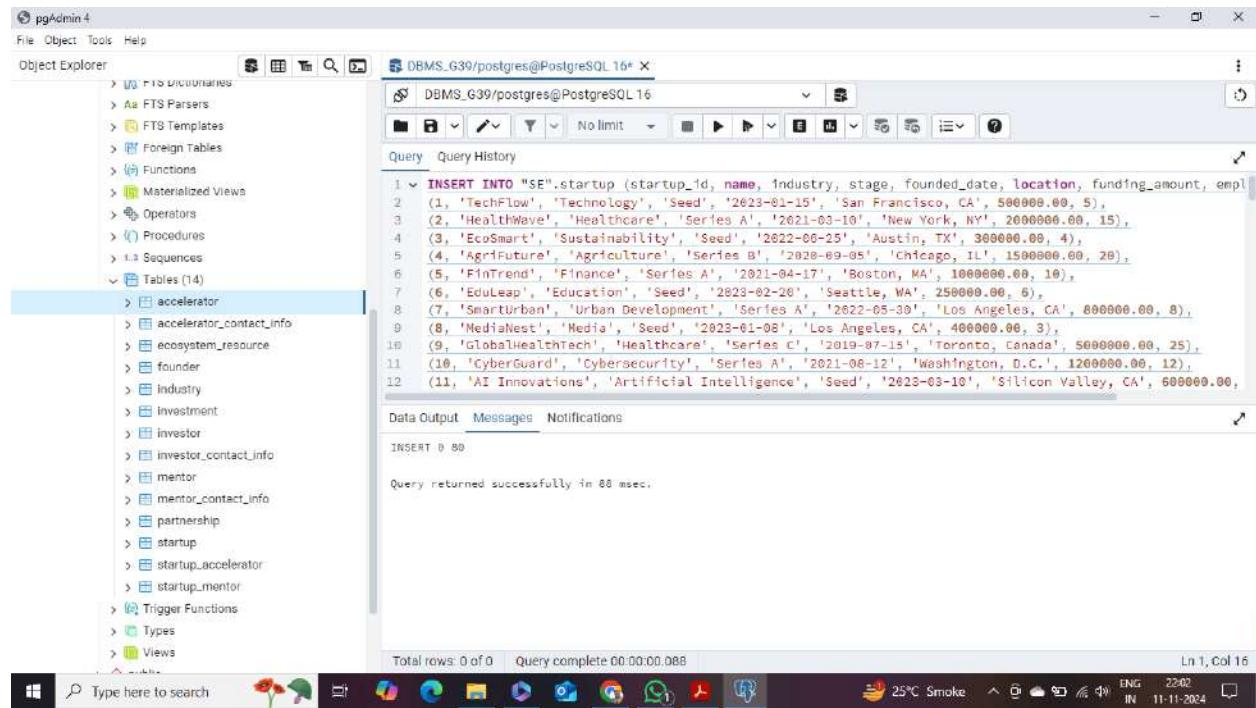
```

Query returned successfully in 101 msec.

Total rows: 0 of 0 Query complete 00:00:00.131

INSERT Statements

Startup Table



```

INSERT INTO startup (startup_id, name, industry, stage, founded_date, location, funding_amount, employee_count) VALUES
(1, 'TechFlow', 'Technology', 'Seed', '2023-01-15', 'San Francisco, CA', 500000.00, 5),
(2, 'HealthWave', 'Healthcare', 'Series A', '2021-03-10', 'New York, NY', 2000000.00, 15),
(3, 'EcoSmart', 'Sustainability', 'Seed', '2022-06-25', 'Austin, TX', 300000.00, 4),
(4, 'AgriFuture', 'Agriculture', 'Series B', '2020-09-05', 'Chicago, IL', 1500000.00, 20),
(5, 'FinTrend', 'Finance', 'Series A', '2021-04-17', 'Boston, MA', 1000000.00, 10),
(6, 'EduLeap', 'Education', 'Seed', '2023-02-20', 'Seattle, WA', 250000.00, 6),
(7, 'SmartUrban', 'Urban Development', 'Series A', '2022-05-30', 'Los Angeles, CA', 800000.00, 8),
(8, 'MediaNest', 'Media', 'Seed', '2023-01-08', 'Los Angeles, CA', 400000.00, 3),
(9, 'GlobalHealthTech', 'Healthcare', 'Series C', '2019-07-15', 'Toronto, Canada', 5000000.00, 25),
(10, 'CyberGuard', 'Cybersecurity', 'Series A', '2021-08-12', 'Washington, D.C.', 1200000.00, 12),
(11, 'AI Innovations', 'Artificial Intelligence', 'Seed', '2023-03-10', 'Silicon Valley, CA', 600000.00, 7),
(12, 'FoodieTech', 'Food Technology', 'Series A', '2020-11-21', 'San Diego, CA', 900000.00, 9),
(13, 'IoT Connect', 'Internet of Things', 'Seed', '2023-04-05', 'New York, NY', 350000.00, 5),
(14, 'Space Ventures', 'Aerospace', 'Series B', '2019-10-30', 'Houston, TX', 2500000.00, 15),

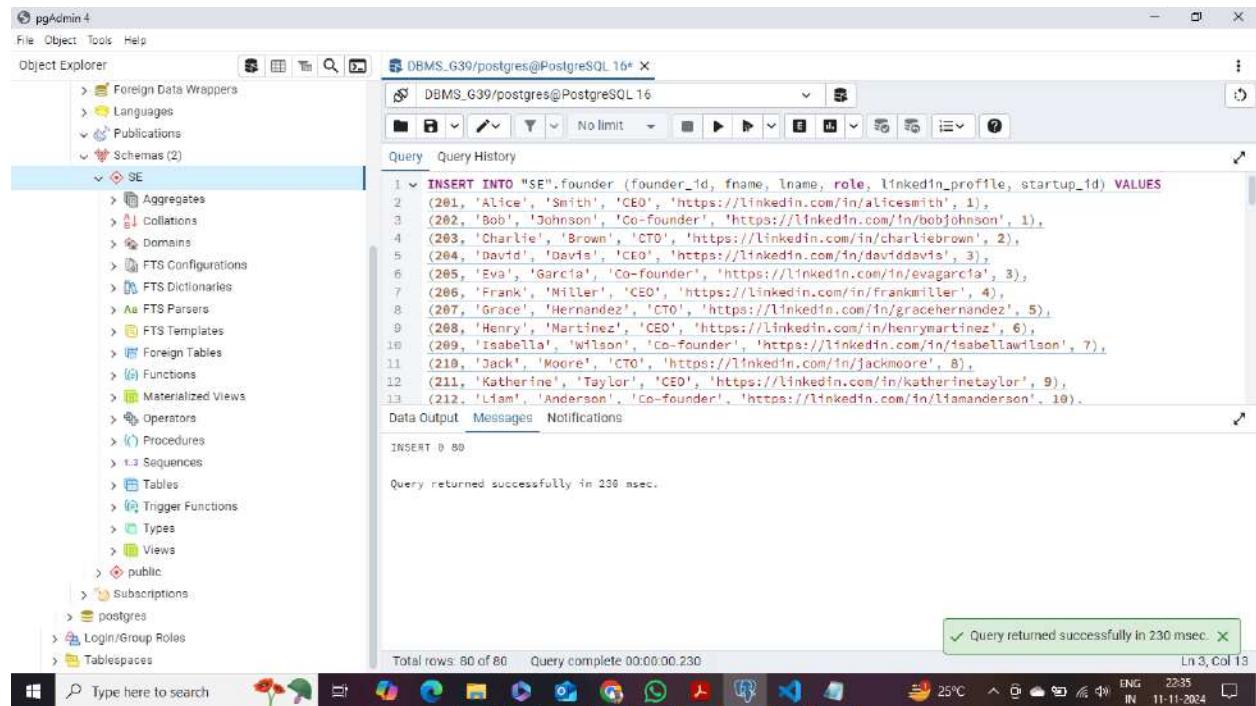
```

(15, 'EcomMagic', 'E-commerce', 'Series A', '2021-05-02', 'Miami, FL', 850000.00, 10),
(16, 'CreativeMinds', 'Media & Entertainment', 'Seed', '2023-02-15', 'Los Angeles, CA', 200000.00, 4),
(17, 'Blockchain Experts', 'Blockchain', 'Series A', '2022-06-17', 'Singapore', 3000000.00, 20),
(18, 'VR Experiences', 'Virtual Reality', 'Seed', '2023-03-28', 'Los Angeles, CA', 500000.00, 6),
(19, 'Digital Health Solutions', 'Healthcare', 'Series B', '2020-08-14', 'Berlin, Germany', 1800000.00, 14),
(20, 'Impact Ventures', 'Social Impact', 'Seed', '2023-05-01', 'London, UK', 400000.00, 5),
(21, 'TravelGenie', 'Travel & Tourism', 'Series A', '2021-07-22', 'Barcelona, Spain', 700000.00, 8),
(22, 'FinBuddy', 'Finance', 'Seed', '2022-12-05', 'Dublin, Ireland', 250000.00, 3),
(23, 'CleanEnergy Solutions', 'Clean Technology', 'Series A', '2020-09-10', 'Vancouver, Canada', 1300000.00, 11),
(24, 'Startup Hub', 'Various', 'Seed', '2023-01-20', 'Austin, TX', 300000.00, 6),
(25, 'FashionTech', 'Fashion', 'Series A', '2021-04-12', 'Paris, France', 600000.00, 5),
(26, 'TechNest', 'Technology', 'Seed', '2023-02-22', 'Tokyo, Japan', 750000.00, 9),
(27, 'EduTech Innovations', 'Education', 'Series B', '2021-05-30', 'Melbourne, Australia', 1200000.00, 12),
(28, 'Wearable Innovations', 'Wearables', 'Seed', '2023-04-15', 'Los Angeles, CA', 400000.00, 4),
(29, 'SportsConnect', 'Sports', 'Series A', '2022-11-02', 'New York, NY', 900000.00, 10),
(30, 'Mobility Solutions', 'Transportation', 'Seed', '2023-03-12', 'San Francisco, CA', 600000.00, 7),
(31, 'Robotics Innovations', 'Robotics', 'Series A', '2021-07-18', 'Bangalore, India', 1500000.00, 15),
(32, 'Telecom Innovations', 'Telecommunications', 'Seed', '2022-10-05', 'Chicago, IL', 800000.00, 8),
(33, 'GameTech', 'Gaming', 'Series A', '2021-09-30', 'Seattle, WA', 1100000.00, 9),
(34, 'SustainableLiving', 'Sustainability', 'Seed', '2023-01-11', 'Toronto, Canada', 450000.00, 6),
(35, 'MediaSpark', 'Media', 'Series B', '2020-12-01', 'San Diego, CA', 2000000.00, 15),
(36, 'BioInnovations', 'Biotechnology', 'Seed', '2023-05-10', 'Boston, MA', 300000.00, 4),
(37, 'SmartHome Solutions', 'Home Automation', 'Series A', '2021-08-20', 'San Francisco, CA', 950000.00, 10),

- (38, 'AgriTech Hub', 'Agriculture', 'Seed', '2023-04-30', 'Atlanta, GA', 500000.00, 5),
 (39, 'Impact Accelerator', 'Social Entrepreneurship', 'Series B', '2019-05-15', 'Amsterdam, Netherlands', 2200000.00, 20),
 (40, 'VR/AR Innovations', 'Virtual/Augmented Reality', 'Seed', '2023-03-07', 'Los Angeles, CA', 350000.00, 7),
 (41, 'Investment Network', 'Finance', 'Series A', '2021-11-12', 'New York, NY', 1400000.00, 12),
 (42, 'HealthTech Innovators', 'Healthcare', 'Seed', '2022-10-22', 'Dublin, Ireland', 300000.00, 8),
 (43, 'TechGlobal', 'Technology', 'Series A', '2020-06-19', 'Singapore', 1800000.00, 14),
 (44, 'ElderCare Innovations', 'Healthcare', 'Seed', '2023-02-28', 'Sydney, Australia', 400000.00, 5),
 (45, 'SmartFashion', 'Fashion', 'Series B', '2021-03-05', 'Paris, France', 1000000.00, 10),
 (46, 'EnergyHub', 'Energy', 'Seed', '2022-09-17', 'Berlin, Germany', 600000.00, 6),
 (47, 'PetCare Innovations', 'Pets', 'Series A', '2021-08-30', 'Austin, TX', 800000.00, 9),
 (48, 'Youth Ventures', 'Social Impact', 'Seed', '2023-01-25', 'Mumbai, India', 500000.00, 3),
 (49, 'AI Accelerator', 'Artificial Intelligence', 'Series B', '2021-06-14', 'Los Angeles, CA', 3000000.00, 20),
 (50, 'FinTech Pioneers', 'Finance', 'Seed', '2023-04-12', 'New York, NY', 350000.00, 11),
 (51, 'InnovateTech', 'Technology', 'Series A', '2022-11-21', 'San Francisco, CA', 1300000.00, 12),
 (52, 'HealthTech Solutions', 'Healthcare', 'Seed', '2023-05-15', 'Toronto, Canada', 700000.00, 9),
 (53, 'SmartEnergy', 'Energy', 'Seed', '2023-02-18', 'Los Angeles, CA', 400000.00, 5),
 (54, 'GreenFuture', 'Clean Technology', 'Series A', '2021-10-04', 'New York, NY', 1200000.00, 10),
 (55, 'EducationPlus', 'Education', 'Seed', '2022-12-25', 'Sydney, Australia', 600000.00, 7),
 (56, 'Health4All', 'Healthcare', 'Series C', '2018-05-12', 'Toronto, Canada', 8000000.00, 60),
 (57, 'GoGreen', 'Sustainability', 'Series A', '2019-07-25', 'Paris, France', 2000000.00, 25),
 (58, 'FinGen', 'Finance', 'Seed', '2022-02-18', 'Dublin, Ireland', 300000.00, 5),
 (59, 'CleanAir Tech', 'Clean Technology', 'Series B', '2020-10-05', 'San Francisco, CA', 2000000.00, 40),
 (60, 'EduQuest', 'Education', 'Seed', '2023-03-15', 'New York, NY', 450000.00, 10),
 (61, 'WellnessSpace', 'Healthcare', 'Series A', '2021-09-30', 'Chicago, IL', 1000000.00, 30),
 (62, 'RetailX', 'Retail', 'Seed', '2023-01-22', 'Miami, FL', 600000.00, 15),

(63, 'SpaceXplore', 'Aerospace', 'Series C', '2018-11-30', 'Austin, TX', 7000000.00, 100),
(64, 'FoodInnovate', 'Food Technology', 'Seed', '2023-05-25', 'Los Angeles, CA', 550000.00, 8),
(65, 'RoboAutomate', 'Robotics', 'Series A', '2020-07-22', 'San Francisco, CA', 1200000.00, 20),
(66, 'EduInnovate', 'Education', 'Seed', '2023-03-02', 'New York, NY', 350000.00, 5),
(67, 'TravelAssist', 'Travel & Tourism', 'Series B', '2019-09-10', 'Barcelona, Spain', 2500000.00, 45),
(68, 'CyberDefend', 'Cybersecurity', 'Seed', '2023-02-01', 'Los Angeles, CA', 700000.00, 8),
(69, 'SmartFinance', 'Finance', 'Series A', '2020-08-15', 'New York, NY', 1500000.00, 35),
(70, 'GreenTech Solutions', 'Clean Technology', 'Seed', '2022-01-25', 'Berlin, Germany', 800000.00, 12),
(71, 'FitWell', 'Wellness', 'Series B', '2019-11-01', 'Chicago, IL', 1300000.00, 30),
(72, 'AIAssist', 'Artificial Intelligence', 'Seed', '2023-04-08', 'San Francisco, CA', 550000.00, 15),
(73, 'SmartGrid', 'Energy', 'Series C', '2018-07-10', 'Los Angeles, CA', 6000000.00, 50),
(74, 'EcoUrban', 'Urban Development', 'Seed', '2022-09-14', 'New York, NY', 500000.00, 10),
(75, 'AgroImpact', 'Agriculture', 'Series B', '2020-06-25', 'Austin, TX', 2100000.00, 40),
(76, 'MediaConnect', 'Media', 'Seed', '2023-02-16', 'San Diego, CA', 300000.00, 5),
(77, 'TechBridge', 'Technology', 'Series A', '2021-10-05', 'Silicon Valley, CA', 1000000.00, 18),
(78, 'GreenFinance', 'Finance', 'Seed', '2023-01-15', 'London, UK', 550000.00, 7),
(79, 'Biotech Innovate', 'Biotechnology', 'Series A', '2021-03-10', 'Boston, MA', 1250000.00, 12),
(80, 'TravelTech', 'Travel & Tourism', 'Seed', '2023-05-18', 'San Francisco, CA', 600000.00, 10);

Founder Table



```
INSERT INTO founder (founder_id, fname, lname, role, linkedin_profile, startup_id) VALUES
```

```

(201, 'Alice', 'Smith', 'CEO', 'https://linkedin.com/in/alicesmith', 1),
(202, 'Bob', 'Johnson', 'Co-founder', 'https://linkedin.com/in/bobjohnson', 1),
(203, 'Charlie', 'Brown', 'CTO', 'https://linkedin.com/in/charliebrown', 2),
(204, 'David', 'Davis', 'CEO', 'https://linkedin.com/in/daviddavis', 3),
(205, 'Eva', 'Garcia', 'Co-founder', 'https://linkedin.com/in/evagarcia', 3),
(206, 'Frank', 'Miller', 'CEO', 'https://linkedin.com/in/frankmiller', 4),
(207, 'Grace', 'Hernandez', 'CTO', 'https://linkedin.com/in/gracehernandez', 5),
(208, 'Henry', 'Martinez', 'CEO', 'https://linkedin.com/in/henrymartinez', 6),
(209, 'Isabella', 'Wilson', 'Co-founder', 'https://linkedin.com/in/isabellawilson', 7),
(210, 'Jack', 'Moore', 'CTO', 'https://linkedin.com/in/jackmoore', 8),
(211, 'Katherine', 'Taylor', 'CEO', 'https://linkedin.com/in/katherinetaylor', 9),
(212, 'Liam', 'Anderson', 'Co-founder', 'https://linkedin.com/in/liamanderson', 10),
(213, 'Mia', 'Thomas', 'CTO', 'https://linkedin.com/in/miathomas', 11),
(214, 'Noah', 'Jackson', 'CEO', 'https://linkedin.com/in/noahjackson', 12),
(215, 'Olivia', 'White', 'Co-founder', 'https://linkedin.com/in/oliviawhite', 13),

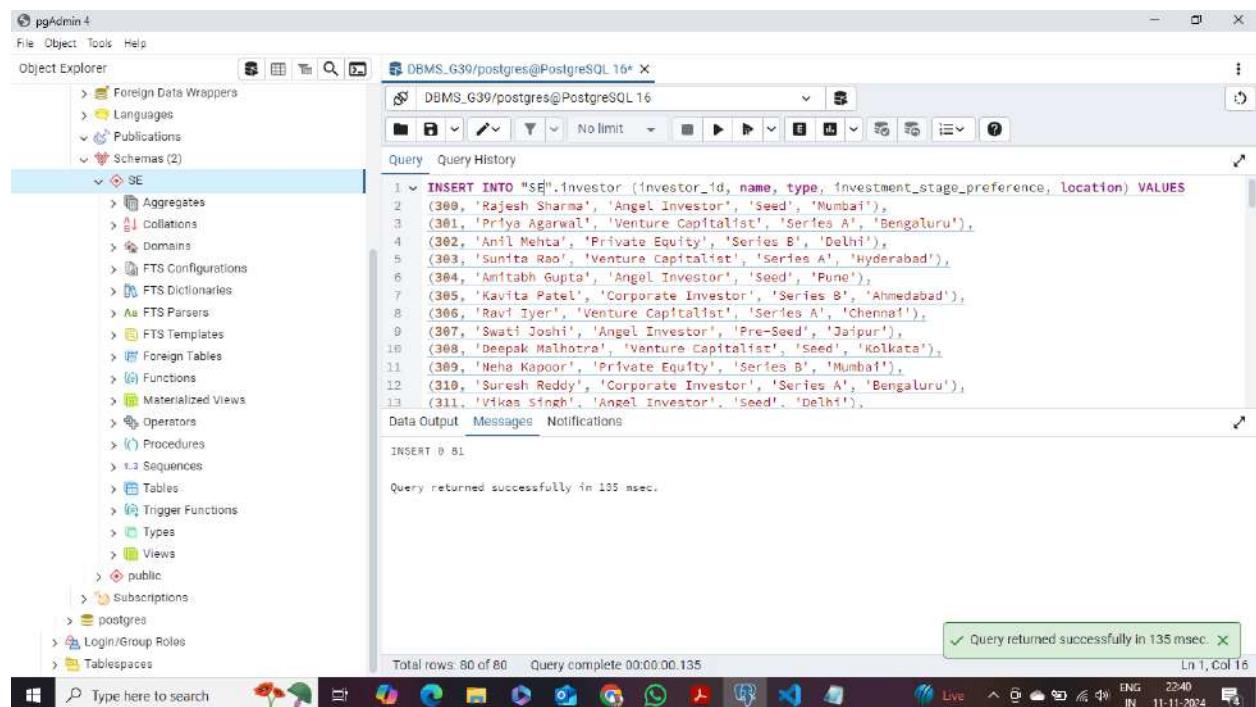
```

- (216, 'Peter', 'Harris', 'CTO', 'https://linkedin.com/in/peterharris', 14),
(217, 'Quinn', 'Clark', 'CEO', 'https://linkedin.com/in/quinnclark', 15),
(218, 'Rita', 'Lewis', 'Co-founder', 'https://linkedin.com/in/ritalewis', 16),
(219, 'Sam', 'Robinson', 'CTO', 'https://linkedin.com/in/samrobinson', 17),
(220, 'Tina', 'Walker', 'CEO', 'https://linkedin.com/in/tinawalker', 18),
(221, 'Ulysses', 'Hall', 'Co-founder', 'https://linkedin.com/in/ulysseshall', 19),
(222, 'Vera', 'Young', 'CTO', 'https://linkedin.com/in/verayoung', 20),
(223, 'Will', 'King', 'CEO', 'https://linkedin.com/in/willking', 21),
(224, 'Xena', 'Scott', 'Co-founder', 'https://linkedin.com/in/xenascott', 22),
(225, 'Yara', 'Green', 'CTO', 'https://linkedin.com/in/yaragreen', 23),
(226, 'Zane', 'Adams', 'CEO', 'https://linkedin.com/in/zaneadams', 24),
(227, 'Anita', 'Nelson', 'Co-founder', 'https://linkedin.com/in/anitanelson', 25),
(228, 'Brian', 'Carter', 'CTO', 'https://linkedin.com/in/briancarter', 26),
(229, 'Cindy', 'Mitchell', 'CEO', 'https://linkedin.com/in/cindymitchell', 27),
(230, 'Daniel', 'Perez', 'Co-founder', 'https://linkedin.com/in/danielperez', 28),
(231, 'Elena', 'Roberts', 'CTO', 'https://linkedin.com/in/elenaroberts', 29),
(232, 'Felix', 'Turner', 'CEO', 'https://linkedin.com/in/felixturner', 30),
(233, 'Gina', 'Phillips', 'Co-founder', 'https://linkedin.com/in/ginaphillips', 31),
(234, 'Henry', 'Campbell', 'CTO', 'https://linkedin.com/in/henrycampbell', 32),
(235, 'Ivy', 'Parker', 'CEO', 'https://linkedin.com/in/ivymarker', 33),
(236, 'James', 'Evans', 'Co-founder', 'https://linkedin.com/in/jamesevans', 34),
(237, 'Kara', 'Edwards', 'CTO', 'https://linkedin.com/in/karaedwards', 35),
(238, 'Leo', 'Stewart', 'CEO', 'https://linkedin.com/in/leostewart', 36),
(239, 'Maya', 'Morris', 'Co-founder', 'https://linkedin.com/in/mayamorris', 37),
(240, 'Nina', 'Murphy', 'CTO', 'https://linkedin.com/in/ninamurphy', 38),
(241, 'Oscar', 'Cook', 'CEO', 'https://linkedin.com/in/oscarcook', 39),
(242, 'Paula', 'Rogers', 'Co-founder', 'https://linkedin.com/in/paularogers', 40),

(243, 'Quentin', 'Reed', 'CTO', 'https://linkedin.com/in/quentinreed', 41),
(244, 'Rosa', 'Bell', 'CEO', 'https://linkedin.com/in/rosabell', 42),
(245, 'Steve', 'Garcia', 'Co-founder', 'https://linkedin.com/in/stevegarcia', 43),
(246, 'Tara', 'Long', 'CTO', 'https://linkedin.com/in/taralong', 44),
(247, 'Uma', 'Ward', 'CEO', 'https://linkedin.com/in/umaward', 45),
(248, 'Victor', 'Russell', 'Co-founder', 'https://linkedin.com/in/victorrussell', 46),
(249, 'Wendy', 'Foster', 'CTO', 'https://linkedin.com/in/wendyfoster', 47),
(250, 'Xander', 'Sanders', 'CEO', 'https://linkedin.com/in/xandersanders', 48),
(251, 'Yasmine', 'Fleming', 'Co-founder', 'https://linkedin.com/in/yasminefleming', 49),
(252, 'Zach', 'Stone', 'CTO', 'https://linkedin.com/in/zachstone', 50),
(253, 'Alan', 'Bennett', 'CEO', 'https://linkedin.com/in/alanbennett', 51),
(254, 'Brenda', 'Hodges', 'Co-founder', 'https://linkedin.com/in/brendahodges', 51),
(255, 'Carl', 'Stevens', 'CTO', 'https://linkedin.com/in/carlstevens', 52),
(256, 'Diana', 'Cook', 'CEO', 'https://linkedin.com/in/dianacook', 53),
(257, 'Edward', 'Ford', 'Co-founder', 'https://linkedin.com/in/edwardford', 53),
(258, 'Fiona', 'Bishop', 'CTO', 'https://linkedin.com/in/fionabishop', 54),
(259, 'George', 'Fisher', 'CEO', 'https://linkedin.com/in/georgefisher', 55),
(260, 'Holly', 'Cross', 'Co-founder', 'https://linkedin.com/in/hollycross', 55),
(261, 'Ian', 'Ellison', 'CTO', 'https://linkedin.com/in/ianellison', 56),
(262, 'Jessica', 'Brock', 'CEO', 'https://linkedin.com/in/jessicabrock', 57),
(263, 'Kevin', 'Chavez', 'Co-founder', 'https://linkedin.com/in/kevincchavez', 57),
(264, 'Laura', 'Owens', 'CTO', 'https://linkedin.com/in/lauraowens', 58),
(265, 'Mark', 'Larson', 'CEO', 'https://linkedin.com/in/marklarson', 59),
(266, 'Nadia', 'Kelley', 'Co-founder', 'https://linkedin.com/in/nadiakelley', 59),
(267, 'Oscar', 'Wagner', 'CTO', 'https://linkedin.com/in/oscarwagner', 60),
(268, 'Paige', 'Dixon', 'CEO', 'https://linkedin.com/in/paigedixon', 61),
(269, 'Quinn', 'Snyder', 'Co-founder', 'https://linkedin.com/in/quinnsnyder', 61),

(270, 'Ryan', 'Peters', 'CTO', 'https://linkedin.com/in/ryanpeters', 62),
 (271, 'Sophie', 'George', 'CEO', 'https://linkedin.com/in/sophiegeorge', 63),
 (272, 'Tom', 'Mason', 'Co-founder', 'https://linkedin.com/in/tommason', 63),
 (273, 'Uma', 'George', 'CTO', 'https://linkedin.com/in/umageorge', 64),
 (274, 'Violet', 'Lowe', 'CEO', 'https://linkedin.com/in/violetlowe', 65),
 (275, 'Will', 'Ray', 'Co-founder', 'https://linkedin.com/in/willray', 66),
 (276, 'Xena', 'Bryant', 'CTO', 'https://linkedin.com/in/xenabryant', 67),
 (277, 'Yara', 'Fernandez', 'CEO', 'https://linkedin.com/in/yarafrnandez', 68),
 (278, 'Zoe', 'Bradley', 'Co-founder', 'https://linkedin.com/in/zoebradley', 68),
 (279, 'Aaron', 'Fitzgerald', 'CTO', 'https://linkedin.com/in/aaronfitzgerald', 69),
 (280, 'Beth', 'Leonard', 'CEO', 'https://linkedin.com/in/bethleonard', 70);

Investor Table



```

pgAdmin 4
File Object Tools Help
Object Explorer DBMS_G39/postgres@PostgreSQL 16e
Foreign Data Wrappers Languages Publications Schemas (2)
  SE Aggregates Collations Domains FTS Configurations FTS Dictionaries FTS Parsers FTS Templates Foreign Tables Functions Materialized Views Operators Procedures Sequences Tables Trigger Functions Types Views public Subscriptions
  postgres Login/Group Roles Tablespaces
  Type here to search  Live 22:40 11-11-2024
  Total rows: 80 of 80  Query complete 00:00:00.135
  ✓ Query returned successfully in 135 msec. Ln 1, Col 16
  1 ✓ INSERT INTO "SE".investor (investor_id, name, type, investment_stage_preference, location) VALUES
  2 (300, 'Rajesh Sharma', 'Angel Investor', 'Seed', 'Mumbai'),
  3 (301, 'Priya Agarwal', 'Venture Capitalist', 'Series A', 'Bengaluru'),
  4 (302, 'Anil Mehta', 'Private Equity', 'Series B', 'Delhi'),
  5 (303, 'Sunita Rao', 'Venture Capitalist', 'Series A', 'Hyderabad'),
  6 (304, 'Amitabh Gupta', 'Angel Investor', 'Seed', 'Pune'),
  7 (305, 'Kavita Patel', 'Corporate Investor', 'Series B', 'Ahmedabad'),
  8 (306, 'Ravi Tyer', 'Venture Capitalist', 'Series A', 'Chennai'),
  9 (307, 'Swati Joshi', 'Angel Investor', 'Pre-Seed', 'Jaipur'),
  10 (308, 'Deepak Malhotra', 'Venture Capitalist', 'Seed', 'Kolkata'),
  11 (309, 'Neha Kapoor', 'Private Equity', 'Series B', 'Mumbai'),
  12 (310, 'Suresh Reddy', 'Corporate Investor', 'Series A', 'Bengaluru'),
  13 (311, 'Vikas Singh', 'Angel Investor', 'Seed', 'Delhi')
  Data Output Messages Notifications
  INSERT 0 81
  Query returned successfully in 105 msec.
  ✓ Query returned successfully in 135 msec. X
  
```

INSERT INTO investor (investor_id, name, type, investment_stage_preference, location) VALUES
 (300, 'Rajesh Sharma', 'Angel Investor', 'Seed', 'Mumbai'),

- (301, 'Priya Agarwal', 'Venture Capitalist', 'Series A', 'Bengaluru'),
(302, 'Anil Mehta', 'Private Equity', 'Series B', 'Delhi'),
(303, 'Sunita Rao', 'Venture Capitalist', 'Series A', 'Hyderabad'),
(304, 'Amitabh Gupta', 'Angel Investor', 'Seed', 'Pune'),
(305, 'Kavita Patel', 'Corporate Investor', 'Series B', 'Ahmedabad'),
(306, 'Ravi Iyer', 'Venture Capitalist', 'Series A', 'Chennai'),
(307, 'Swati Joshi', 'Angel Investor', 'Pre-Seed', 'Jaipur'),
(308, 'Deepak Malhotra', 'Venture Capitalist', 'Seed', 'Kolkata'),
(309, 'Neha Kapoor', 'Private Equity', 'Series B', 'Mumbai'),
(310, 'Suresh Reddy', 'Corporate Investor', 'Series A', 'Bengaluru'),
(311, 'Vikas Singh', 'Angel Investor', 'Seed', 'Delhi'),
(312, 'Manisha Desai', 'Venture Capitalist', 'Series A', 'Hyderabad'),
(313, 'Pankaj Chawla', 'Private Equity', 'Series B', 'Chandigarh'),
(314, 'Anita Ghosh', 'Angel Investor', 'Seed', 'Pune'),
(315, 'Rohit Menon', 'Corporate Investor', 'Series B', 'Ahmedabad'),
(316, 'Meera Khanna', 'Venture Capitalist', 'Series A', 'Chennai'),
(317, 'Siddharth Kulkarni', 'Angel Investor', 'Seed', 'Mumbai'),
(318, 'Preeti Sethi', 'Private Equity', 'Series B', 'Delhi'),
(319, 'Abhinav Das', 'Corporate Investor', 'Series A', 'Hyderabad'),
(320, 'Lakshmi Pillai', 'Angel Investor', 'Seed', 'Jaipur'),
(321, 'Nitin Kapoor', 'Venture Capitalist', 'Series A', 'Pune'),
(322, 'Jyoti Bhat', 'Private Equity', 'Series B', 'Ahmedabad'),
(323, 'Rohan Naik', 'Angel Investor', 'Pre-Seed', 'Kolkata'),
(324, 'Sarika Nair', 'Venture Capitalist', 'Series A', 'Chennai'),
(325, 'Mohit Goel', 'Private Equity', 'Series B', 'Bengaluru'),
(326, 'Sneha Shetty', 'Corporate Investor', 'Seed', 'Hyderabad'),
(327, 'Rajiv Sinha', 'Venture Capitalist', 'Series A', 'Mumbai'),

- (328, 'Rina Verma', 'Angel Investor', 'Pre-Seed', 'Delhi'),
(329, 'Kunal Roy', 'Corporate Investor', 'Series B', 'Pune'),
(330, 'Shivani Rathi', 'Private Equity', 'Series A', 'Bengaluru'),
(331, 'Arjun Malik', 'Angel Investor', 'Seed', 'Ahmedabad'),
(332, 'Megha Purohit', 'Venture Capitalist', 'Series A', 'Kolkata'),
(333, 'Vikram Pandey', 'Corporate Investor', 'Series B', 'Hyderabad'),
(334, 'Rashmi Tiwari', 'Private Equity', 'Seed', 'Delhi'),
(335, 'Ashish Patil', 'Angel Investor', 'Seed', 'Chennai'),
(336, 'Komal Thakur', 'Corporate Investor', 'Series A', 'Mumbai'),
(337, 'Harish Choudhary', 'Venture Capitalist', 'Series B', 'Jaipur'),
(338, 'Tina Kaur', 'Angel Investor', 'Seed', 'Pune'),
(339, 'Naveen Bansal', 'Private Equity', 'Series B', 'Ahmedabad'),
(340, 'Suman Khatri', 'Venture Capitalist', 'Series A', 'Bengaluru'),
(341, 'Aakash Nanda', 'Angel Investor', 'Pre-Seed', 'Hyderabad'),
(342, 'Pooja Lamba', 'Corporate Investor', 'Series A', 'Delhi'),
(343, 'Ajay Rao', 'Venture Capitalist', 'Seed', 'Mumbai'),
(344, 'Chitra Saxena', 'Private Equity', 'Series B', 'Kolkata'),
(345, 'Gaurav Meena', 'Corporate Investor', 'Seed', 'Bengaluru'),
(346, 'Anuja Pathak', 'Angel Investor', 'Series A', 'Hyderabad'),
(347, 'Ramesh Kale', 'Venture Capitalist', 'Seed', 'Chennai'),
(348, 'Divya Chopra', 'Private Equity', 'Series B', 'Delhi'),
(349, 'Sahil Ahuja', 'Corporate Investor', 'Series A', 'Ahmedabad'),
(350, 'Nidhi Bhatia', 'Angel Investor', 'Pre-Seed', 'Mumbai'),
(351, 'Rajat Bhardwaj', 'Venture Capitalist', 'Seed', 'Pune'),
(352, 'Kriti Mohan', 'Private Equity', 'Series B', 'Chennai'),
(353, 'Tejaswini R', 'Corporate Investor', 'Series A', 'Kolkata'),
(354, 'Shreyas D', 'Angel Investor', 'Seed', 'Bengaluru'),

(355, 'Arun K', 'Private Equity', 'Series B', 'Delhi'),
(356, 'Vidya Reddy', 'Corporate Investor', 'Series A', 'Mumbai'),
(357, 'Rahul Jain', 'Angel Investor', 'Pre-Seed', 'Pune'),
(358, 'Tanvi Arora', 'Venture Capitalist', 'Seed', 'Hyderabad'),
(359, 'Sameer Bhatt', 'Private Equity', 'Series B', 'Jaipur'),
(360, 'Kiran Rao', 'Corporate Investor', 'Series A', 'Bengaluru'),
(361, 'Mona Nanda', 'Angel Investor', 'Seed', 'Mumbai'),
(362, 'Rajat Sharma', 'Venture Capitalist', 'Series B', 'Delhi'),
(363, 'Nikita Singhania', 'Private Equity', 'Series A', 'Pune'),
(364, 'Gautam Shekhar', 'Corporate Investor', 'Seed', 'Hyderabad'),
(365, 'Arpita B', 'Angel Investor', 'Seed', 'Ahmedabad'),
(366, 'Ravi D', 'Venture Capitalist', 'Series B', 'Kolkata'),
(367, 'Ishaan M', 'Private Equity', 'Seed', 'Mumbai'),
(368, 'Ritika Y', 'Corporate Investor', 'Series A', 'Delhi'),
(369, 'Kabir T', 'Angel Investor', 'Pre-Seed', 'Chennai'),
(370, 'Simran N', 'Venture Capitalist', 'Series B', 'Hyderabad'),
(371, 'Aditya P', 'Private Equity', 'Seed', 'Bengaluru'),
(372, 'Preeti V', 'Corporate Investor', 'Series A', 'Pune'),
(373, 'Taranjeet C', 'Angel Investor', 'Seed', 'Jaipur'),
(374, 'Megha J', 'Venture Capitalist', 'Series A', 'Mumbai'),
(375, 'Nikhil L', 'Private Equity', 'Series B', 'Ahmedabad'),
(376, 'Alok G', 'Corporate Investor', 'Seed', 'Delhi'),
(377, 'Anuja K', 'Angel Investor', 'Seed', 'Bengaluru'),
(378, 'Divya R', 'Venture Capitalist', 'Series A', 'Hyderabad'),
(379, 'Ajay T', 'Private Equity', 'Series B', 'Pune'),
(380, 'Sonam M', 'Corporate Investor', 'Seed', 'Kolkata');

Investment Table

```

pgAdmin 4
File Object Tools Help
Object Explorer DBMS_G39/postgres@PostgreSQL_16*
Foreign Data Wrappers Languages Publications Schemas (2)
    SE
        Aggregates Collations Domains FTS Configurations FTS Dictionaries FTS Parsers FTS Templates Foreign Tables Functions Materialized Views Operators Procedures Sequences Tables Trigger Functions Types Views public Subscriptions
    postgres Login/Group Roles Tablespaces
Data Output Messages Notifications
Query History
INSERT INTO "SE".investment (investment_id, investor_id, startup_id, amount, date, stage) VALUES
(1000, 300, 1, '500000.00', '2023-01-15', 'Seed'),
(1001, 301, 1, '250000.00', '2023-02-20', 'Seed'),
(1002, 302, 2, '300000.00', '2023-03-18', 'Series A'),
(1003, 303, 2, '450000.00', '2023-04-12', 'Series A'),
(1004, 304, 3, '600000.00', '2023-05-08', 'Seed'),
(1005, 305, 3, '200000.00', '2023-05-28', 'Seed'),
(1006, 306, 4, '750000.00', '2023-06-15', 'Series B'),
(1007, 307, 4, '500000.00', '2023-07-10', 'Series B'),
(1008, 308, 5, '300000.00', '2023-08-05', 'Seed'),
(1009, 309, 5, '350000.00', '2023-08-25', 'Seed'),
(1010, 310, 6, '450000.00', '2023-09-14', 'Series A'),
(1011, 311, 6, '550000.00', '2023-10-10', 'Series A'),
Data Output Messages Notifications
INSERT 0 80
Query returned successfully in 141 msec.
Total rows: 80 of 80 Query complete 00:00:00.141
LN 1, Col 16
    ✓ Query returned successfully in 141 msec. X

```

```

INSERT INTO investment (investment_id, investor_id, startup_id, amount, date, stage) VALUES
(1000, 300, 1, '500000.00', '2023-01-15', 'Seed'),
(1001, 301, 1, '250000.00', '2023-02-20', 'Seed'),
(1002, 302, 2, '300000.00', '2023-03-18', 'Series A'),
(1003, 303, 2, '450000.00', '2023-04-12', 'Series A'),
(1004, 304, 3, '600000.00', '2023-05-08', 'Seed'),
(1005, 305, 3, '200000.00', '2023-05-28', 'Seed'),
(1006, 306, 4, '750000.00', '2023-06-15', 'Series B'),
(1007, 307, 4, '500000.00', '2023-07-10', 'Series B'),
(1008, 308, 5, '300000.00', '2023-08-05', 'Seed'),
(1009, 309, 5, '350000.00', '2023-08-25', 'Seed'),
(1010, 310, 6, '450000.00', '2023-09-14', 'Series A'),
(1011, 311, 6, '550000.00', '2023-10-10', 'Series A'),

```

(1010, 310, 6, '450000.00', '2023-09-14', 'Series A'),
(1011, 311, 6, '550000.00', '2023-10-10', 'Series A'),
(1012, 312, 7, '250000.00', '2023-11-05', 'Seed'),
(1013, 313, 7, '200000.00', '2023-11-20', 'Seed'),
(1014, 314, 8, '650000.00', '2023-12-05', 'Series B'),
(1015, 315, 8, '700000.00', '2024-01-01', 'Series B'),
(1016, 316, 9, '500000.00', '2024-01-15', 'Seed'),
(1017, 317, 9, '300000.00', '2024-02-05', 'Seed'),
(1018, 318, 10, '800000.00', '2024-02-25', 'Series A'),
(1019, 319, 10, '600000.00', '2024-03-10', 'Series A'),
(1020, 320, 11, '550000.00', '2024-03-20', 'Seed'),
(1021, 321, 12, '300000.00', '2024-04-05', 'Seed'),
(1022, 322, 12, '400000.00', '2024-04-25', 'Series A'),
(1023, 323, 13, '500000.00', '2024-05-10', 'Series B'),
(1024, 324, 13, '450000.00', '2024-05-30', 'Series B'),
(1025, 325, 14, '600000.00', '2024-06-15', 'Seed'),
(1026, 326, 15, '300000.00', '2024-06-28', 'Series A'),
(1027, 327, 16, '350000.00', '2024-07-10', 'Series A'),
(1028, 328, 17, '450000.00', '2024-07-25', 'Seed'),
(1029, 329, 18, '550000.00', '2024-08-10', 'Seed'),
(1030, 330, 19, '700000.00', '2024-08-20', 'Series B'),
(1031, 331, 20, '500000.00', '2024-09-01', 'Seed'),
(1032, 332, 20, '250000.00', '2024-09-15', 'Series A'),
(1033, 333, 21, '450000.00', '2024-09-30', 'Series A'),
(1034, 334, 22, '550000.00', '2024-10-10', 'Seed'),
(1035, 335, 23, '750000.00', '2024-10-20', 'Seed'),
(1036, 336, 23, '300000.00', '2024-11-01', 'Series A'),

(1037, 337, 24, '200000.00', '2024-11-15', 'Series B'),
(1038, 338, 25, '400000.00', '2024-11-30', 'Seed'),
(1039, 339, 26, '500000.00', '2024-12-10', 'Seed'),
(1040, 340, 26, '300000.00', '2024-12-20', 'Series A'),
(1041, 341, 27, '550000.00', '2024-12-30', 'Series B'),
(1042, 342, 28, '600000.00', '2025-01-10', 'Seed'),
(1043, 343, 28, '250000.00', '2025-01-20', 'Series A'),
(1044, 344, 29, '350000.00', '2025-01-30', 'Series A'),
(1045, 345, 30, '500000.00', '2025-02-10', 'Seed'),
(1046, 346, 30, '200000.00', '2025-02-20', 'Seed'),
(1047, 347, 31, '400000.00', '2025-03-01', 'Series A'),
(1048, 348, 32, '600000.00', '2025-03-15', 'Series B'),
(1049, 349, 33, '250000.00', '2025-03-28', 'Seed'),
(1050, 350, 34, '500000.00', '2025-04-10', 'Seed'),
(1051, 351, 34, '300000.00', '2025-04-25', 'Series A'),
(1052, 352, 35, '550000.00', '2025-05-10', 'Seed'),
(1053, 353, 36, '750000.00', '2025-05-25', 'Series A'),
(1054, 354, 37, '300000.00', '2025-06-10', 'Series B'),
(1055, 355, 38, '450000.00', '2025-06-30', 'Seed'),
(1056, 356, 39, '200000.00', '2025-07-15', 'Seed'),
(1057, 357, 40, '400000.00', '2025-07-25', 'Series A'),
(1058, 358, 40, '500000.00', '2025-08-10', 'Series A'),
(1059, 359, 41, '350000.00', '2025-08-20', 'Seed'),
(1060, 360, 42, '650000.00', '2025-08-30', 'Series B'),
(1061, 361, 43, '500000.00', '2025-09-10', 'Seed'),
(1062, 362, 44, '750000.00', '2025-09-20', 'Series A'),
(1063, 363, 45, '300000.00', '2025-09-30', 'Seed'),

(1064, 364, 46, '450000.00', '2025-10-10', 'Seed'),
(1065, 365, 47, '550000.00', '2025-10-20', 'Series A'),
(1066, 366, 48, '750000.00', '2025-10-30', 'Series B'),
(1067, 367, 49, '500000.00', '2025-11-10', 'Seed'),
(1068, 368, 50, '650000.00', '2025-11-20', 'Series A'),
(1069, 369, 50, '250000.00', '2025-11-30', 'Series B'),
(1070, 370, 51, '750000.00', '2025-12-10', 'Seed'),
(1071, 371, 52, '300000.00', '2025-12-20', 'Series A'),
(1072, 372, 53, '250000.00', '2025-12-30', 'Series A'),
(1073, 373, 54, '500000.00', '2026-01-10', 'Seed'),
(1074, 374, 55, '300000.00', '2026-01-20', 'Seed'),
(1075, 375, 56, '600000.00', '2026-02-10', 'Series B'),
(1076, 376, 57, '750000.00', '2026-02-20', 'Seed'),
(1077, 377, 58, '500000.00', '2026-02-28', 'Seed'),
(1078, 378, 59, '400000.00', '2026-03-10', 'Series A'),
(1079, 379, 60, '600000.00', '2026-03-20', 'Series B');

Mentor Table

The screenshot shows the pgAdmin 4 interface. On the left, the Object Explorer tree view is expanded to show the 'Schemas' node under 'SE'. The 'Query' tab in the main pane contains the following SQL code:

```

1 ✓ INSERT INTO "SE".Mentor (Mentor_ID, Name, Expertise, Affiliation, LinkedIn_Profile) VALUES
2 (500, 'Andrew', 'Product Development', 'Professional', 'https://linkedin.com/in/andrewmentor'),
3 (501, 'Brittany', 'Finance', 'Academic', 'https://linkedin.com/in/brittanymentor'),
4 (502, 'Colin', 'Operations', 'Organizational', 'https://linkedin.com/in/colinmentor'),
5 (503, 'Denise', 'Marketing Strategy', 'Professional', 'https://linkedin.com/in/denisementor'),
6 (504, 'Ethan', 'Business Development', 'Partnership', 'https://linkedin.com/in/ethanmentor'),
7 (505, 'Fiona', 'Human Resources', 'Academic', 'https://linkedin.com/in/fionamentor'),
8 (506, 'Gavin', 'Data Science', 'Professional', 'https://linkedin.com/in/gavinmentor'),
9 (507, 'Hannah', 'Legal Consulting', 'Organizational', 'https://linkedin.com/in/hannahmentor'),
10 (508, 'Isaac', 'Cybersecurity', 'Professional', 'https://linkedin.com/in/isaacmentor'),
11 (509, 'Julia', 'Software Engineering', 'Partnership', 'https://linkedin.com/in/juliamentor'),
12 (510, 'Kyle', 'Sales Strategy', 'Academic', 'https://linkedin.com/in/kylementor'),
13 (511, 'Lily', 'Cloud Computing', 'Professional', 'https://linkedin.com/in/lilymentor'),

```

Below the code, the status bar indicates 'Total rows: 80 of 80' and 'Query complete 00:00:00.123'. A green success message at the bottom right says 'Query returned successfully in 123 msec.' with a timestamp 'Lh 78, Col 88'.

```

INSERT INTO "SE".Mentor (Mentor_ID, Name, Expertise, Affiliation, LinkedIn_Profile) VALUES
(500, 'Andrew', 'Product Development', 'Professional', 'https://linkedin.com/in/andrewmentor'),
(501, 'Brittany', 'Finance', 'Academic', 'https://linkedin.com/in/brittanymentor'),
(502, 'Colin', 'Operations', 'Organizational', 'https://linkedin.com/in/colinmentor'),
(503, 'Denise', 'Marketing Strategy', 'Professional', 'https://linkedin.com/in/denisementor'),
(504, 'Ethan', 'Business Development', 'Partnership', 'https://linkedin.com/in/ethanmentor'),
(505, 'Fiona', 'Human Resources', 'Academic', 'https://linkedin.com/in/fionamentor'),
(506, 'Gavin', 'Data Science', 'Professional', 'https://linkedin.com/in/gavinmentor'),
(507, 'Hannah', 'Legal Consulting', 'Organizational', 'https://linkedin.com/in/hannahmentor'),
(508, 'Isaac', 'Cybersecurity', 'Professional', 'https://linkedin.com/in/isaacmentor'),
(509, 'Julia', 'Software Engineering', 'Partnership', 'https://linkedin.com/in/juliamentor'),
(510, 'Kyle', 'Sales Strategy', 'Academic', 'https://linkedin.com/in/kylementor'),
(511, 'Lily', 'Cloud Computing', 'Professional', 'https://linkedin.com/in/lilymentor'),
(512, 'Marcus', 'Artificial Intelligence', 'Organizational', 'https://linkedin.com/in/marcusmentor'),
(513, 'Nora', 'Investment Strategy', 'Professional', 'https://linkedin.com/in/noramenter'),
(514, 'Oscar', 'Supply Chain Management', 'Academic', 'https://linkedin.com/in/oscarmentor'),

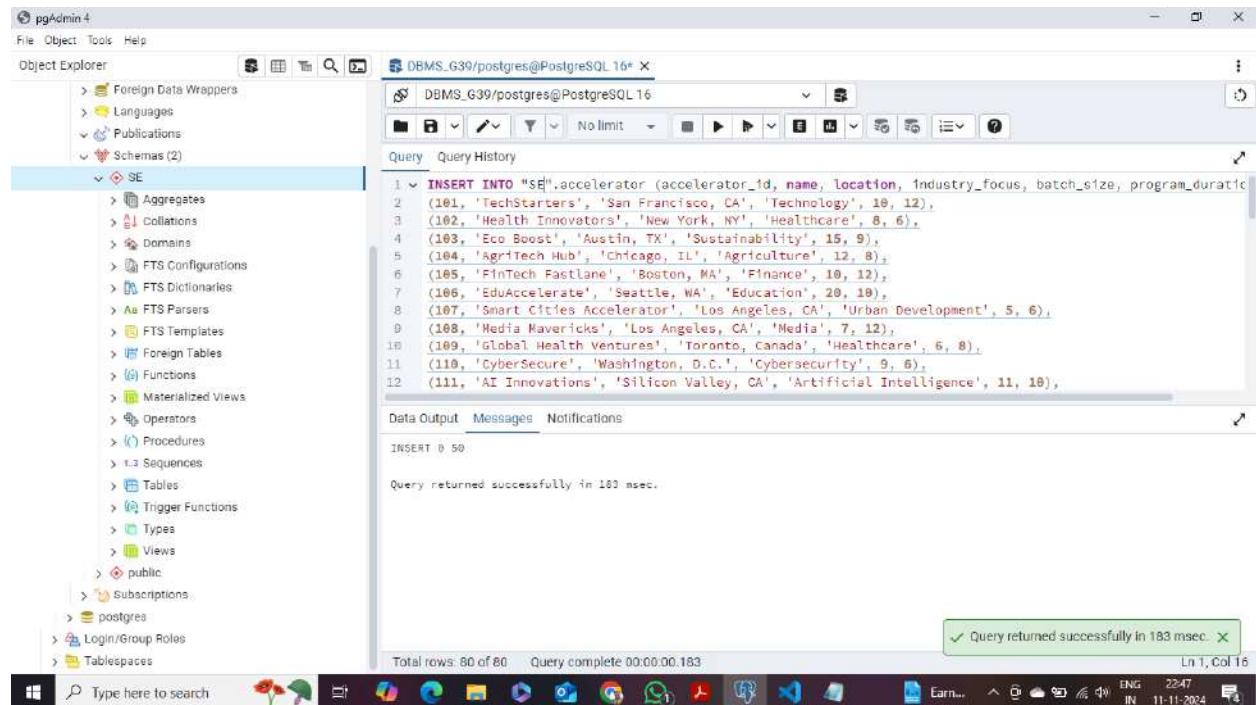
```

(515, 'Paula', 'Machine Learning', 'Partnership', 'https://linkedin.com/in/paulamentor'),
(516, 'Quentin', 'Product Design', 'Organizational', 'https://linkedin.com/in/quentinmentor'),
(517, 'Rachel', 'Corporate Strategy', 'Professional', 'https://linkedin.com/in/rachelmentor'),
(518, 'Steve', 'Healthcare', 'Academic', 'https://linkedin.com/in/stementor'),
(519, 'Tina', 'Renewable Energy', 'Partnership', 'https://linkedin.com/in/tinamentor'),
(520, 'Umar', 'E-commerce', 'Organizational', 'https://linkedin.com/in/umarmendor'),
(521, 'Violet', 'Marketing Research', 'Professional', 'https://linkedin.com/in/violetmentor'),
(522, 'Wesley', 'User Experience', 'Academic', 'https://linkedin.com/in/wesleymentor'),
(523, 'Xander', 'Blockchain', 'Partnership', 'https://linkedin.com/in/xandermentor'),
(524, 'Yvonne', 'Real Estate', 'Organizational', 'https://linkedin.com/in/yvonnementor'),
(525, 'Zachary', 'Ethics', 'Professional', 'https://linkedin.com/in/zacharymentor'),
(526, 'Alicea', 'Telecommunications', 'Academic', 'https://linkedin.com/in/aliceamentor'),
(527, 'Ben', 'Retail Strategy', 'Organizational', 'https://linkedin.com/in/benmentor'),
(528, 'Cathy', 'Content Strategy', 'Professional', 'https://linkedin.com/in/cathymentor'),
(529, 'Darius', 'Biotechnology', 'Academic', 'https://linkedin.com/in/dariusmentor'),
(530, 'Elena', 'Fashion', 'Partnership', 'https://linkedin.com/in/elenamentor'),
(531, 'Fred', 'Public Relations', 'Organizational', 'https://linkedin.com/in/fredmentor'),
(532, 'Gretchen', 'Venture Capital', 'Professional', 'https://linkedin.com/in/gretchenmentor'),
(533, 'Henry', 'Legal Affairs', 'Academic', 'https://linkedin.com/in/henrymentor'),
(534, 'Isabel', 'Accounting', 'Partnership', 'https://linkedin.com/in/isabelmentor'),
(535, 'Jon', 'Product Innovation', 'Organizational', 'https://linkedin.com/in/jonmentor'),
(536, 'Karen', 'Risk Management', 'Professional', 'https://linkedin.com/in/karenmentor'),
(537, 'Lance', 'Health Technology', 'Academic', 'https://linkedin.com/in/lancementor'),
(538, 'Marilyn', 'Agriculture', 'Partnership', 'https://linkedin.com/in/marilynmentor'),
(539, 'Nathan', 'Market Analysis', 'Organizational', 'https://linkedin.com/in/nathanmentor'),
(540, 'Olivia', 'Software Development', 'Professional', 'https://linkedin.com/in/oliviamentor'),
(541, 'Peter', 'Digital Marketing', 'Academic', 'https://linkedin.com/in/petermentor'),

(542, 'Queenie', 'Manufacturing', 'Partnership', 'https://linkedin.com/in/queeniementor'),
(543, 'Roy', 'Gaming Industry', 'Organizational', 'https://linkedin.com/in/roymentor'),
(544, 'Sarah', 'Social Media', 'Professional', 'https://linkedin.com/in/sarahmentor'),
(545, 'Travis', 'Human Capital', 'Academic', 'https://linkedin.com/in/travismentor'),
(546, 'Una', 'Transportation', 'Partnership', 'https://linkedin.com/in/unamentor'),
(547, 'Victor', 'Media', 'Organizational', 'https://linkedin.com/in/victormentor'),
(548, 'Willow', 'Tourism', 'Professional', 'https://linkedin.com/in/willowmentor'),
(549, 'Xavier', 'Robotics', 'Academic', 'https://linkedin.com/in/xaviermentor'),
(550, 'Yasmine', 'Education', 'Partnership', 'https://linkedin.com/in/yasminementor'),
(551, 'Zara', 'Medical Devices', 'Organizational', 'https://linkedin.com/in/zaramentor'),
(552, 'Arthur', 'Hospitality', 'Professional', 'https://linkedin.com/in/arthurmentor'),
(553, 'Bella', 'Customer Service', 'Academic', 'https://linkedin.com/in/bellamentor'),
(554, 'Caleb', 'Sports Management', 'Partnership', 'https://linkedin.com/in/calebmentor'),
(555, 'Diana', 'Creative Direction', 'Organizational', 'https://linkedin.com/in/dianamentor'),
(556, 'Eli', 'Cybersecurity', 'Professional', 'https://linkedin.com/in/elimentor'),
(557, 'Frida', 'Food & Beverage', 'Academic', 'https://linkedin.com/in/fridamentor'),
(558, 'Gabriel', 'Logistics', 'Partnership', 'https://linkedin.com/in/gabrielmentor'),
(559, 'Hugo', 'Mobile Technology', 'Organizational', 'https://linkedin.com/in/hugomentor'),
(560, 'Irene', 'Pharmaceuticals', 'Professional', 'https://linkedin.com/in/irenmentor'),
(561, 'Jack', 'Insurance', 'Academic', 'https://linkedin.com/in/jackmentor'),
(562, 'Kylie', 'Consumer Goods', 'Partnership', 'https://linkedin.com/in/kyliementor'),
(563, 'Leo', 'Telemedicine', 'Organizational', 'https://linkedin.com/in/leomentor'),
(564, 'Mona', 'Blockchain Strategy', 'Professional', 'https://linkedin.com/in/monamentor'),
(565, 'Nate', 'Mobile Applications', 'Academic', 'https://linkedin.com/in/natementor'),
(566, 'Olga', 'Event Management', 'Partnership', 'https://linkedin.com/in/olgamentor'),
(567, 'Paul', 'Climate Solutions', 'Organizational', 'https://linkedin.com/in/paulmentor'),
(568, 'Renee', 'AI Ethics', 'Professional', 'https://linkedin.com/in/renée_mentor'),

```
(569, 'Simon', 'Aerospace', 'Academic', 'https://linkedin.com/in/simonmentor'),
(570, 'Tara', 'FinTech', 'Partnership', 'https://linkedin.com/in/taramentor'),
(571, 'Ulrich', 'Cloud Solutions', 'Organizational', 'https://linkedin.com/in.ulrichmentor'),
(572, 'Vera', 'Quantum Computing', 'Professional', 'https://linkedin.com/in/veramentor'),
(573, 'Wayne', 'Policy Development', 'Academic', 'https://linkedin.com/in/waynementor'),
(574, 'Ximena', 'User Research', 'Partnership', 'https://linkedin.com/in/ximenamentor'),
(575, 'Yolanda', 'Smart City Planning', 'Organizational', 'https://linkedin.com/in/yolandamentor'),
(576, 'Zane', 'Ethical Hacking', 'Professional', 'https://linkedin.com/in/zanementor');
```

Accelerator Table



The screenshot shows the pgAdmin 4 interface with two panes. The left pane, 'Object Explorer', displays a tree view of database objects under the schema 'SE'. The right pane, 'Query', contains the following SQL code:

```
1 INSERT INTO "SE".accelerator (accelerator_id, name, location, industry_focus, batch_size, program_duration)
2 (101, 'TechStarters', 'San Francisco, CA', 'Technology', 10, 12),
3 (102, 'Health Innovators', 'New York, NY', 'Healthcare', 8, 6),
4 (103, 'Eco Boost', 'Austin, TX', 'Sustainability', 15, 9),
5 (104, 'AgriTech Hub', 'Chicago, IL', 'Agriculture', 12, 8),
6 (105, 'FinTech FastLane', 'Boston, MA', 'Finance', 10, 12),
7 (106, 'EduAccelerate', 'Seattle, WA', 'Education', 20, 18),
8 (107, 'Smart Cities Accelerator', 'Los Angeles, CA', 'Urban Development', 5, 6),
9 (108, 'Media Mavericks', 'Los Angeles, CA', 'Media', 7, 12),
10 (109, 'Global Health Ventures', 'Toronto, Canada', 'Healthcare', 6, 8),
11 (110, 'CyberSecure', 'Washington, D.C.', 'Cybersecurity', 9, 6),
12 (111, 'AI Innovations', 'Silicon Valley, CA', 'Artificial Intelligence', 11, 18),
```

The status bar at the bottom indicates 'Total rows: 80 of 80' and 'Query complete 00:00:00.183'. A message box in the bottom right corner says 'Query returned successfully in 183 msec.'

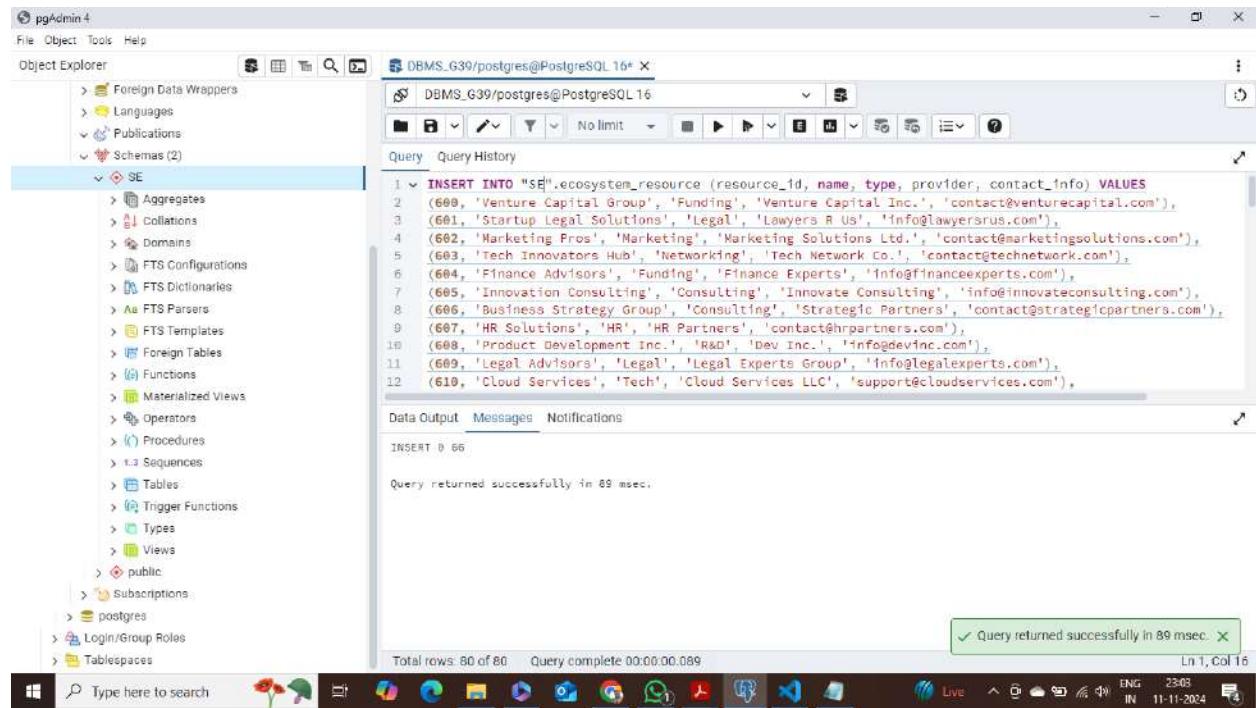
```
INSERT INTO accelerator (accelerator_id, name, location, industry_focus, batch_size, program_duration) VALUES
```

```
(101, 'TechStarters', 'San Francisco, CA', 'Technology', 10, 12),
(102, 'Health Innovators', 'New York, NY', 'Healthcare', 8, 6),
(103, 'Eco Boost', 'Austin, TX', 'Sustainability', 15, 9),
(104, 'AgriTech Hub', 'Chicago, IL', 'Agriculture', 12, 8),
```

- (105, 'FinTech Fastlane', 'Boston, MA', 'Finance', 10, 12),
(106, 'EduAccelerate', 'Seattle, WA', 'Education', 20, 10),
(107, 'Smart Cities Accelerator', 'Los Angeles, CA', 'Urban Development', 5, 6),
(108, 'Media Mavericks', 'Los Angeles, CA', 'Media', 7, 12),
(109, 'Global Health Ventures', 'Toronto, Canada', 'Healthcare', 6, 8),
(110, 'CyberSecure', 'Washington, D.C.', 'Cybersecurity', 9, 6),
(111, 'AI Innovations', 'Silicon Valley, CA', 'Artificial Intelligence', 11, 10),
(112, 'FoodTech Labs', 'San Diego, CA', 'Food Technology', 8, 8),
(113, 'IoT Incubator', 'New York, NY', 'Internet of Things', 10, 10),
(114, 'Space Startups', 'Houston, TX', 'Aerospace', 4, 12),
(115, 'E-commerce Launchpad', 'Miami, FL', 'E-commerce', 12, 8),
(116, 'Creative Labs', 'Los Angeles, CA', 'Media & Entertainment', 15, 9),
(117, 'Blockchain Foundry', 'Singapore', 'Blockchain', 5, 6),
(118, 'VR Incubator', 'San Francisco, CA', 'Virtual Reality', 7, 10),
(119, 'Digital Health Accelerator', 'Berlin, Germany', 'Healthcare', 6, 6),
(120, 'Impact Innovators', 'London, UK', 'Social Impact', 10, 12),
(121, 'TravelTech Hub', 'Barcelona, Spain', 'Travel & Tourism', 8, 10),
(122, 'Fintech Factory', 'Dublin, Ireland', 'Finance', 9, 8),
(123, 'CleanTech Accelerator', 'Vancouver, Canada', 'Clean Technology', 12, 9),
(124, 'Startup Garage', 'Austin, TX', 'Various', 15, 10),
(125, 'Fashion Forward', 'Paris, France', 'Fashion', 10, 6),
(126, 'Tech Quest', 'Tokyo, Japan', 'Technology', 11, 12),
(127, 'EduLab', 'Melbourne, Australia', 'Education', 20, 10),
(128, 'Wearable Tech Incubator', 'Los Angeles, CA', 'Wearables', 6, 8),
(129, 'Sports Tech Accelerator', 'New York, NY', 'Sports', 8, 10),
(130, 'Smart Mobility Hub', 'San Francisco, CA', 'Transportation', 7, 9),
(131, 'Robotics Lab', 'Bangalore, India', 'Robotics', 10, 6),

- (132, 'Telecom Innovators', 'Chicago, IL', 'Telecommunications', 12, 12),
(133, 'Gaming Accelerator', 'Seattle, WA', 'Gaming', 10, 8),
(134, 'Sustainable Living Accelerator', 'Toronto, Canada', 'Sustainability', 9, 10),
(135, 'Digital Media Hub', 'San Diego, CA', 'Media', 11, 8),
(136, 'BioTech Innovators', 'Boston, MA', 'Biotechnology', 5, 12),
(137, 'Smart Home Incubator', 'San Francisco, CA', 'Home Automation', 8, 6),
(138, 'AgriFood Accelerator', 'Atlanta, GA', 'Food Technology', 10, 10),
(139, 'Global Impact Accelerator', 'Amsterdam, Netherlands', 'Social Entrepreneurship', 15, 8),
(140, 'VR/AR Hub', 'Los Angeles, CA', 'Virtual/Augmented Reality', 7, 6),
(141, 'Investment Accelerator', 'New York, NY', 'Finance', 10, 10),
(142, 'Healthcare Innovations', 'Dublin, Ireland', 'Healthcare', 6, 12),
(143, 'Tech Frontier', 'Singapore', 'Technology', 12, 10),
(144, 'Elder Tech Hub', 'Sydney, Australia', 'Healthcare', 9, 8),
(145, 'Smart Fashion Accelerator', 'Paris, France', 'Fashion', 5, 6),
(146, 'Energy Innovators', 'Berlin, Germany', 'Energy', 10, 9),
(147, 'Pet Tech Incubator', 'Austin, TX', 'Pets', 8, 10),
(148, 'Youth Empowerment Hub', 'Mumbai, India', 'Social Impact', 20, 12),
(149, 'Artificial Intelligence Accelerator', 'Los Angeles, CA', 'AI', 15, 10),
(150, 'FinTech 101', 'New York, NY', 'Finance', 12, 8);

Ecosystem_Resource Table



```

INSERT INTO ecosystem_resource (resource_id, name, type, provider, contact_info) VALUES
(600, 'Venture Capital Group', 'Funding', 'Venture Capital Inc.', 'contact@venturecapital.com'),
(601, 'Startup Legal Solutions', 'Legal', 'Lawyers R Us', 'info@lawyersrus.com'),
(602, 'Marketing Pros', 'Marketing', 'Marketing Solutions Ltd.', 'contact@marketingsolutions.com'),
(603, 'Tech Innovators Hub', 'Networking', 'Tech Network Co.', 'contact@technetwork.com'),
(604, 'Finance Advisors', 'Funding', 'Finance Experts', 'info@financeexperts.com'),
(605, 'Innovation Consulting', 'Consulting', 'Innovate Consulting', 'info@innovateconsulting.com'),
(606, 'Business Strategy Group', 'Consulting', 'Strategic Partners', 'contact@strategicpartners.com'),
(607, 'HR Solutions', 'HR', 'HR Partners', 'contact@hrpartners.com'),
(608, 'Product Development Inc.', 'R&D', 'Dev Inc.', 'info@devinc.com'),
(609, 'Legal Advisors', 'Legal', 'Legal Experts Group', 'info@legalexperts.com'),
(610, 'Cloud Services', 'Tech', 'Cloud Services LLC', 'support@cloudservices.com'),
(611, 'Sales Accelerator', 'Sales', 'Sales Experts', 'contact@salsexperts.com'),
(612, 'Market Research Institute', 'Research', 'Research Group', 'info@researchgroup.com'),
(613, 'Funding Solutions', 'Funding', 'Funding Network', 'info@fundingnetwork.com'),

```

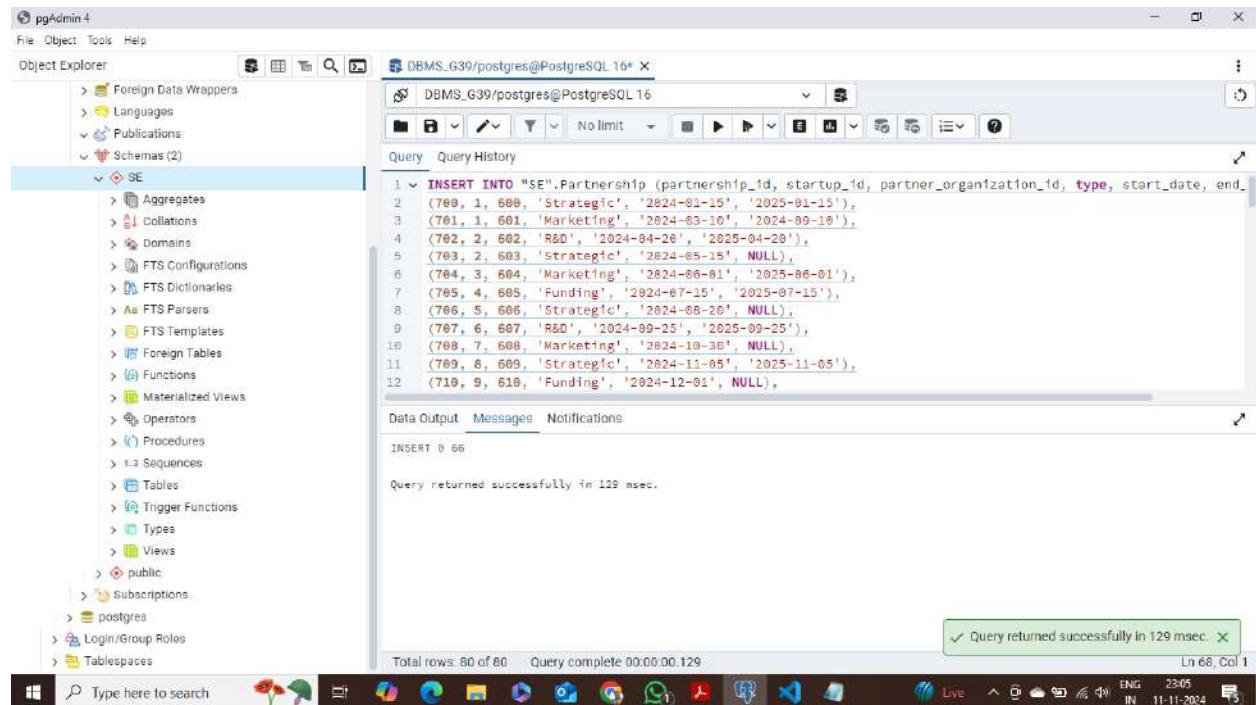
(614, 'Startup Advisors', 'Consulting', 'Startup Consultancy', 'contact@startupconsultancy.com'),
(615, 'Marketing Analytics Co.', 'Marketing', 'Analytics Group', 'info@analyticsgroup.com'),
(616, 'Design Studio', 'Design', 'Creative Design Co.', 'contact@creativedesign.com'),
(617, 'Investor Relations Group', 'Funding', 'Investor Connect', 'contact@investorconnect.com'),
(618, 'Tech Support Group', 'Tech', 'Support Services', 'support@supportservices.com'),
(619, 'Cybersecurity Solutions', 'Tech', 'Cyber Defense Co.', 'info@cyberdefense.com'),
(620, 'Intellectual Property Advisors', 'Legal', 'IP Experts', 'info@ipexperts.com'),
(621, 'Event Management Group', 'Marketing', 'Events Co.', 'info@eventsco.com'),
(622, 'Logistics Partners', 'Logistics', 'Logistics Group', 'info@logisticsgroup.com'),
(623, 'Corporate Training Group', 'Training', 'Training Solutions', 'contact@trainingsolutions.com'),
(624, 'Funding Network', 'Funding', 'Network Ventures', 'contact@networkventures.com'),
(625, 'Digital Marketing Agency', 'Marketing', 'Digital Agency', 'info@digitalagency.com'),
(626, 'Startup Growth Partners', 'Consulting', 'Growth Partners', 'info@growthpartners.com'),
(627, 'E-commerce Solutions', 'Tech', 'E-commerce Inc.', 'contact@ecommerce.com'),
(628, 'Social Media Agency', 'Marketing', 'Social Media Experts', 'info@socialmediaexperts.com'),
(629, 'Blockchain Innovations', 'Tech', 'Blockchain Co.', 'info@blockchainco.com'),
(630, 'Sustainability Consultants', 'Consulting', 'Sustainability Group',
'contact@sustainabilitygroup.com'),
(631, 'Data Analytics Firm', 'Research', 'Data Solutions', 'info@datasolutions.com'),
(632, 'Software Development Hub', 'Tech', 'Software Solutions', 'support@softwaresolutions.com'),
(633, 'Entrepreneurship Program', 'Education', 'Entrepreneur Group',
'info@entrepreneurgroup.com'),
(634, 'HealthTech Solutions', 'Health', 'Health Innovations', 'contact@healthinnovations.com'),
(635, 'Finance & Accounting Group', 'Finance', 'Finance Group', 'info@financegroup.com'),
(636, 'Crowdfunding Platform', 'Funding', 'Crowd Funders', 'contact@crowdfunders.com'),
(637, 'HR Consulting Firm', 'HR', 'HR Consulting Co.', 'info@hrconsultingco.com'),
(638, 'Product Management Group', 'Management', 'Product Experts',
'contact@productexperts.com'),

(639, 'Strategic Planning Firm', 'Consulting', 'Strategic Planners', 'info@strategicplanners.com'),
(640, 'Tech Support Solutions', 'Tech', 'Tech Support Co.', 'support@techsupportco.com'),
(641, 'Startup Community', 'Networking', 'Community Co.', 'contact@communityco.com'),
(642, 'Virtual Assistant Services', 'Support', 'Virtual Assistants', 'info@virtualassistants.com'),
(643, 'Startup Health Program', 'Health', 'Health Group', 'contact@healthgroup.com'),
(644, 'Investor Networking Group', 'Networking', 'Investor Network', 'info@investornetwork.com'),
(645, 'Online Learning Platform', 'Education', 'Learning Co.', 'support@learningco.com'),
(646, 'Advertising Agency', 'Marketing', 'Ad Solutions', 'info@adsolutions.com'),
(647, 'Social Impact Partners', 'Consulting', 'Impact Partners', 'contact@impactpartners.com'),
(648, 'App Development Studio', 'Tech', 'App Developers', 'info@appdevelopers.com'),
(649, 'Tech Innovations Inc.', 'Tech', 'Tech Innovators', 'contact@techinnovators.com'),
(650, 'E-commerce Growth Agency', 'Marketing', 'E-commerce Experts',
'info@ecommerceexperts.com'),
(651, 'Productivity Tools Company', 'Tech', 'Productivity Inc.', 'contact@productivityinc.com'),
(652, 'Investment Banking Group', 'Finance', 'Investment Bankers',
'info@investmentbankers.com'),
(653, 'Remote Work Solutions', 'Tech', 'Remote Work Co.', 'info@remoteworkco.com'),
(654, 'AI Consulting Firm', 'Consulting', 'AI Solutions', 'contact@aisolutions.com'),
(655, 'Non-profit Consulting Group', 'Consulting', 'Non-profit Experts',
'info@nonprofitexperts.com'),
(656, 'SaaS Development Company', 'Tech', 'SaaS Solutions', 'info@saassolutions.com'),
(657, 'Digital Transformation Agency', 'Consulting', 'Digital Transformations',
'contact@digitaltransformations.com'),
(658, 'Crisis Management Group', 'Consulting', 'Crisis Advisors', 'info@crisisadvisors.com'),
(659, 'Cybersecurity Consulting Firm', 'Tech', 'Cyber Solutions', 'info@cybersolutions.com'),
(660, 'FinTech Innovations', 'Tech', 'FinTech Co.', 'info@fintechco.com'),
(661, 'Content Marketing Group', 'Marketing', 'Content Experts', 'contact@contentexperts.com'),
(662, 'Global Trade Solutions', 'Logistics', 'Global Trade Group', 'info@globaltradegroup.com'),
(663, 'HR Management Solutions', 'HR', 'HR Management Co.', 'contact@hrmanagementco.com'),

```
(664, 'International Business Advisors', 'Consulting', 'International Advisors',
'info@internationaladvisors.com'),  

(665, 'Cloud Storage Solutions', 'Tech', 'Cloud Storage Co.', 'info@cloudstorageco.com');
```

Partnership Table



The screenshot shows the pgAdmin 4 interface. On the left, the Object Explorer tree view is expanded to show the 'Schemas' node, which contains a 'SE' schema. Inside 'SE', there are various database objects like Aggregates, Collations, Domains, FTS Configurations, FTS Dictionaries, FTS Parsers, FTS Templates, Foreign Tables, Functions, Materialized Views, Operators, Procedures, Sequences, Tables, Trigger Functions, Types, Views, and public. The 'Tables' node under 'SE' is selected. On the right, the main query editor window displays the following SQL code:

```
1 ✓ INSERT INTO "SE".Partnership (partnership_id, startup_id, partner_organization_id, type, start_date, end_
2 (700, 1, 600, 'Strategic', '2024-01-15', '2025-01-15'),
3 (701, 1, 601, 'Marketing', '2024-03-10', '2024-09-10'),
4 (702, 2, 602, 'R&D', '2024-04-20', '2025-04-20'),
5 (703, 2, 603, 'Strategic', '2024-05-15', NULL),
6 (704, 3, 604, 'Marketing', '2024-06-01', '2025-06-01'),
7 (705, 4, 605, 'Funding', '2024-07-15', '2025-07-15'),
8 (706, 5, 606, 'Strategic', '2024-08-20', NULL),
9 (707, 6, 607, 'R&D', '2024-09-25', '2025-09-25'),
10 (708, 7, 608, 'Marketing', '2024-10-30', NULL),
11 (709, 8, 609, 'Strategic', '2024-11-05', '2025-11-05'),
12 (710, 9, 610, 'Funding', '2024-12-01', NULL),
```

The status bar at the bottom of the pgAdmin window indicates 'Total rows: 80 of 80' and 'Query complete 00:00:00.129'. A green message box in the bottom right corner says 'Query returned successfully in 129 msec.'.

```
INSERT INTO "SE".Partnership (partnership_id, startup_id, partner_organization_id, type,
start_date, end_date) VALUES  

(700, 1, 600, 'Strategic', '2024-01-15', '2025-01-15'),  

(701, 1, 601, 'Marketing', '2024-03-10', '2024-09-10'),  

(702, 2, 602, 'R&D', '2024-04-20', '2025-04-20'),  

(703, 2, 603, 'Strategic', '2024-05-15', NULL),  

(704, 3, 604, 'Marketing', '2024-06-01', '2025-06-01'),  

(705, 4, 605, 'Funding', '2024-07-15', '2025-07-15'),  

(706, 5, 606, 'Strategic', '2024-08-20', NULL),  

(707, 6, 607, 'R&D', '2024-09-25', '2025-09-25'),  

(708, 7, 608, 'Marketing', '2024-10-30', NULL),  

(709, 8, 609, 'Strategic', '2024-11-05', '2025-11-05'),  

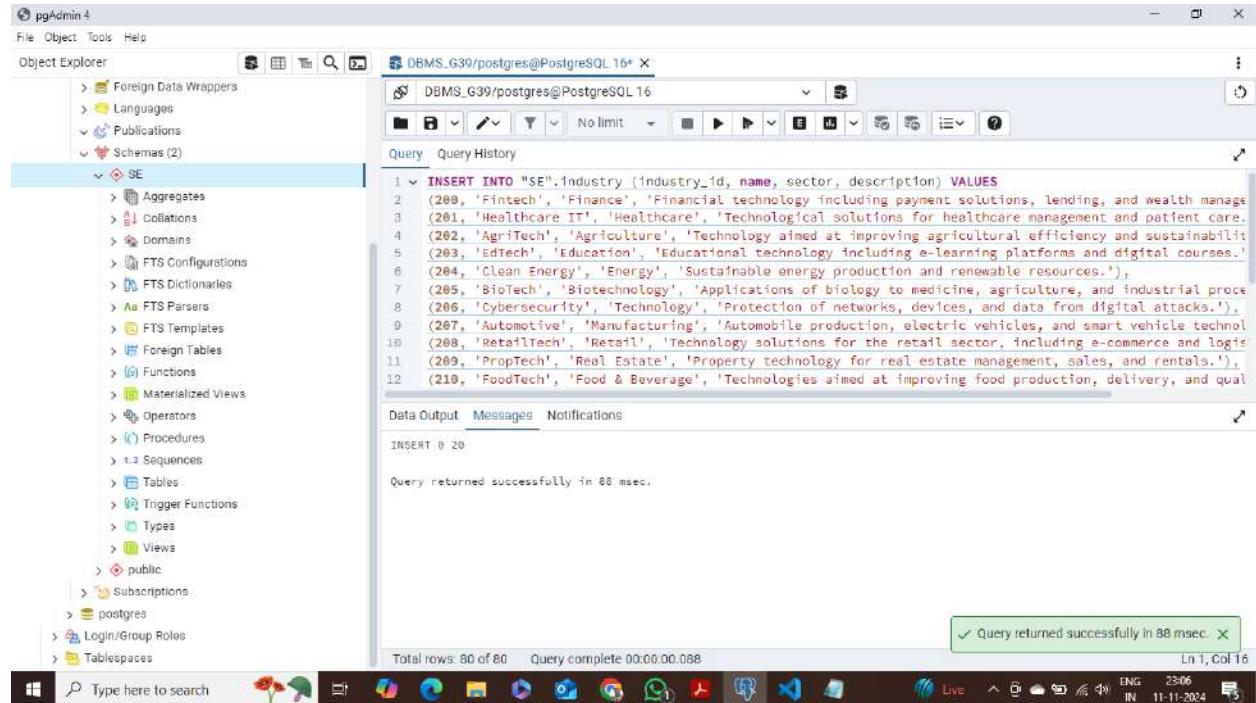
(710, 9, 610, 'Funding', '2024-12-01', NULL),
```

(709, 8, 609, 'Strategic', '2024-11-05', '2025-11-05'),
(710, 9, 610, 'Funding', '2024-12-01', NULL),
(711, 10, 611, 'R&D', '2024-01-10', '2025-01-10'),
(712, 11, 612, 'Marketing', '2024-02-15', NULL),
(713, 12, 613, 'Strategic', '2024-03-20', '2025-03-20'),
(714, 13, 614, 'Funding', '2024-04-25', NULL),
(715, 14, 615, 'R&D', '2024-05-30', '2025-05-30'),
(716, 15, 616, 'Marketing', '2024-06-05', NULL),
(717, 16, 617, 'Strategic', '2024-07-10', '2025-07-10'),
(718, 17, 618, 'Funding', '2024-08-15', NULL),
(719, 18, 619, 'R&D', '2024-09-20', '2025-09-20'),
(720, 19, 620, 'Marketing', '2024-10-25', NULL),
(721, 20, 621, 'Strategic', '2024-11-30', '2025-11-30'),
(722, 21, 622, 'Funding', '2024-12-05', NULL),
(723, 22, 623, 'R&D', '2024-01-15', '2025-01-15'),
(724, 23, 624, 'Marketing', '2024-02-20', NULL),
(725, 24, 625, 'Strategic', '2024-03-25', '2025-03-25'),
(726, 25, 626, 'Funding', '2024-04-30', NULL),
(727, 26, 627, 'R&D', '2024-05-05', '2025-05-05'),
(728, 27, 628, 'Marketing', '2024-06-10', NULL),
(729, 28, 629, 'Strategic', '2024-07-15', '2025-07-15'),
(730, 29, 630, 'Funding', '2024-08-20', NULL),
(731, 30, 631, 'R&D', '2024-09-25', '2025-09-25'),
(732, 31, 632, 'Marketing', '2024-10-30', NULL),
(733, 32, 633, 'Strategic', '2024-11-05', '2025-11-05'),
(734, 33, 634, 'Funding', '2024-12-01', NULL),
(735, 34, 635, 'R&D', '2024-01-10', '2025-01-10'),

(736, 35, 636, 'Marketing', '2024-02-15', NULL),
(737, 36, 637, 'Strategic', '2024-03-20', '2025-03-20'),
(738, 37, 638, 'Funding', '2024-04-25', NULL),
(739, 38, 639, 'R&D', '2024-05-30', '2025-05-30'),
(740, 39, 640, 'Marketing', '2024-06-05', NULL),
(741, 40, 641, 'Strategic', '2024-07-10', '2025-07-10'),
(742, 41, 642, 'Funding', '2024-08-15', NULL),
(743, 42, 643, 'R&D', '2024-09-20', '2025-09-20'),
(744, 43, 644, 'Marketing', '2024-10-25', NULL),
(745, 44, 645, 'Strategic', '2024-11-30', '2025-11-30'),
(746, 45, 646, 'Funding', '2024-12-05', NULL),
(747, 46, 647, 'R&D', '2024-01-15', '2025-01-15'),
(748, 47, 648, 'Marketing', '2024-02-20', NULL),
(749, 48, 649, 'Strategic', '2024-03-25', '2025-03-25'),
(750, 49, 650, 'Funding', '2024-04-30', NULL),
(751, 50, 651, 'R&D', '2024-05-05', '2025-05-05'),
(752, 51, 652, 'Marketing', '2024-06-10', NULL),
(753, 52, 653, 'Strategic', '2024-07-15', '2025-07-15'),
(754, 53, 654, 'Funding', '2024-08-20', NULL),
(755, 54, 655, 'R&D', '2024-09-25', '2025-09-25'),
(756, 55, 656, 'Marketing', '2024-10-30', NULL),
(757, 56, 657, 'Strategic', '2024-11-05', '2025-11-05'),
(758, 57, 658, 'Funding', '2024-12-01', NULL),
(759, 58, 659, 'R&D', '2024-01-10', '2025-01-10'),
(760, 59, 660, 'Marketing', '2024-02-15', NULL),
(761, 60, 661, 'Strategic', '2024-03-20', '2025-03-20'),
(762, 61, 662, 'Funding', '2024-04-25', NULL),

```
(763, 62, 663, 'R&D', '2024-05-30', '2025-05-30'),
(764, 63, 664, 'Marketing', '2024-06-05', NULL),
(765, 64, 665, 'Strategic', '2024-07-10', '2025-07-10');
```

Industry Table



The screenshot shows the pgAdmin 4 interface with the following details:

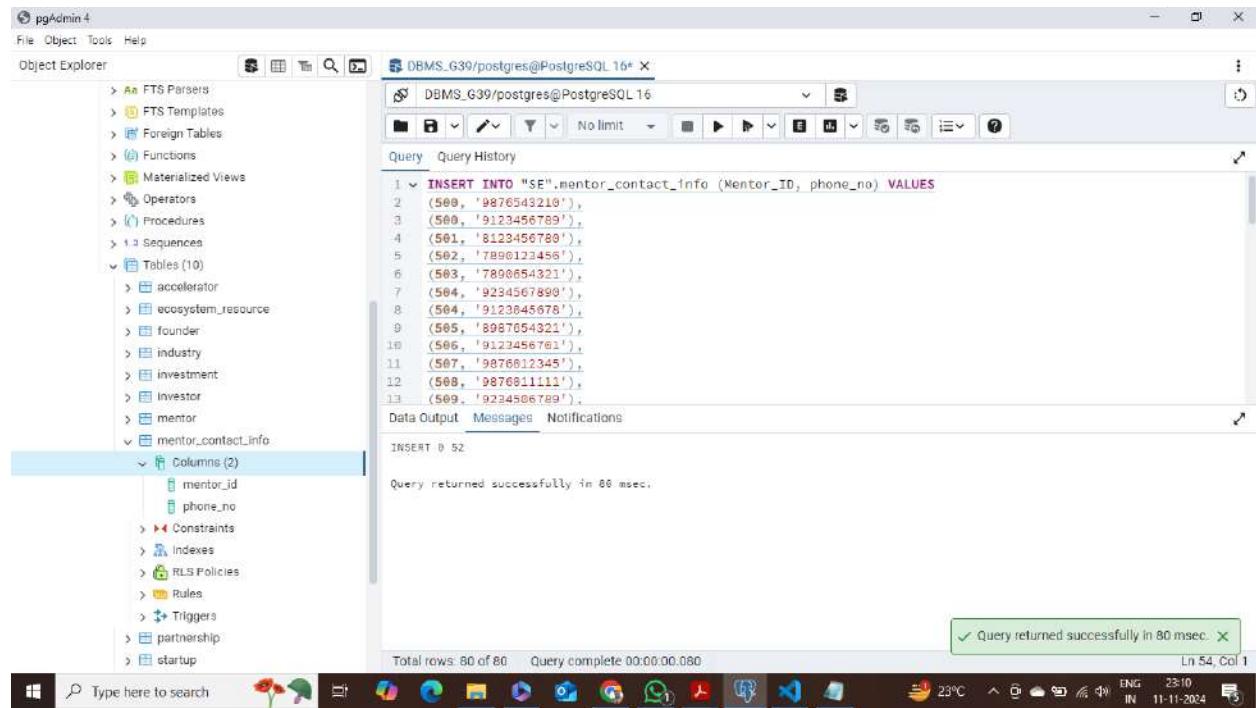
- Object Explorer:** Shows the database structure under the schema "SE".
- Query Editor:** Displays the SQL query used to insert data into the "industry" table.
- Data Output:** Shows the successful insertion of 80 rows.
- Messages:** Displays a success message indicating the query returned successfully.
- Notifications:** Shows a notification bar at the bottom right.

```
1 ✓ INSERT INTO "SE".industry (industry_id, name, sector, description) VALUES
2 (200, 'Fintech', 'Finance', 'Financial technology including payment solutions, lending, and wealth management.'),
3 (201, 'Healthcare IT', 'Healthcare', 'Technological solutions for healthcare management and patient care.'),
4 (202, 'AgriTech', 'Agriculture', 'Technology aimed at improving agricultural efficiency and sustainability.'),
5 (203, 'EdTech', 'Education', 'Educational technology including e-learning platforms and digital courses.'),
6 (204, 'Clean Energy', 'Energy', 'Sustainable energy production and renewable resources.'),
7 (205, 'BioTech', 'Biotechnology', 'Applications of biology to medicine, agriculture, and industrial processes.'),
8 (206, 'Cybersecurity', 'Technology', 'Protection of networks, devices, and data from digital attacks.'),
9 (207, 'Automotive', 'Manufacturing', 'Automobile production, electric vehicles, and smart vehicle technology.'),
10 (208, 'RetailTech', 'Retail', 'Technology solutions for the retail sector, including e-commerce and logistics.'),
11 (209, 'PropTech', 'Real Estate', 'Property technology for real estate management, sales, and rentals.'),
12 (210, 'FoodTech', 'Food & Beverage', 'Technologies aimed at improving food production, delivery, and quality control.')
```

```
INSERT INTO industry (industry_id, name, sector, description) VALUES
(200, 'Fintech', 'Finance', 'Financial technology including payment solutions, lending, and wealth management.'),
(201, 'Healthcare IT', 'Healthcare', 'Technological solutions for healthcare management and patient care.'),
(202, 'AgriTech', 'Agriculture', 'Technology aimed at improving agricultural efficiency and sustainability.'),
(203, 'EdTech', 'Education', 'Educational technology including e-learning platforms and digital courses.'),
(204, 'Clean Energy', 'Energy', 'Sustainable energy production and renewable resources.'),
```

- (205, 'BioTech', 'Biotechnology', 'Applications of biology to medicine, agriculture, and industrial processes.'),
- (206, 'Cybersecurity', 'Technology', 'Protection of networks, devices, and data from digital attacks.'),
- (207, 'Automotive', 'Manufacturing', 'Automobile production, electric vehicles, and smart vehicle technology.'),
- (208, 'RetailTech', 'Retail', 'Technology solutions for the retail sector, including e-commerce and logistics.'),
- (209, 'PropTech', 'Real Estate', 'Property technology for real estate management, sales, and rentals.'),
- (210, 'FoodTech', 'Food & Beverage', 'Technologies aimed at improving food production, delivery, and quality.'),
- (211, 'Gaming', 'Entertainment', 'Development and distribution of video games and interactive entertainment.'),
- (212, 'Logistics', 'Supply Chain', 'Technology and services for transportation and supply chain optimization.'),
- (213, 'Telecommunications', 'Communication', 'Communication technologies including mobile networks and internet services.'),
- (214, 'FashionTech', 'Fashion', 'Technological innovations in fashion design, production, and retail.'),
- (215, 'InsurTech', 'Insurance', 'Digital transformation in the insurance industry for products and services.'),
- (216, 'SpaceTech', 'Aerospace', 'Technologies and advancements in space exploration and satellite technology.'),
- (217, 'ConstructionTech', 'Construction', 'Innovations in construction processes, materials, and project management.'),
- (218, 'LegalTech', 'Legal', 'Technology solutions for legal processes, compliance, and case management.'),
- (219, 'MedTech', 'Medical Devices', 'Development of medical devices and diagnostic equipment.');

Mentor_Contact_Info Table



```
INSERT INTO "SE".mentor_contact_info (Mentor_ID, Contact_Details) VALUES
```

```
(500, '9876543210'),
```

```
(500, '9123456789'),
```

```
(501, '8123456780'),
```

```
(502, '7890123456'),
```

```
(503, '7890654321'),
```

```
(504, '9234567890'),
```

```
(504, '9123045678'),
```

```
(505, '8987654321'),
```

```
(506, '9123456701'),
```

```
(507, '9876012345'),
```

```
(508, '9876011111'),
```

```
(509, '9234506789'),
```

```
(510, '8123456799'),
```

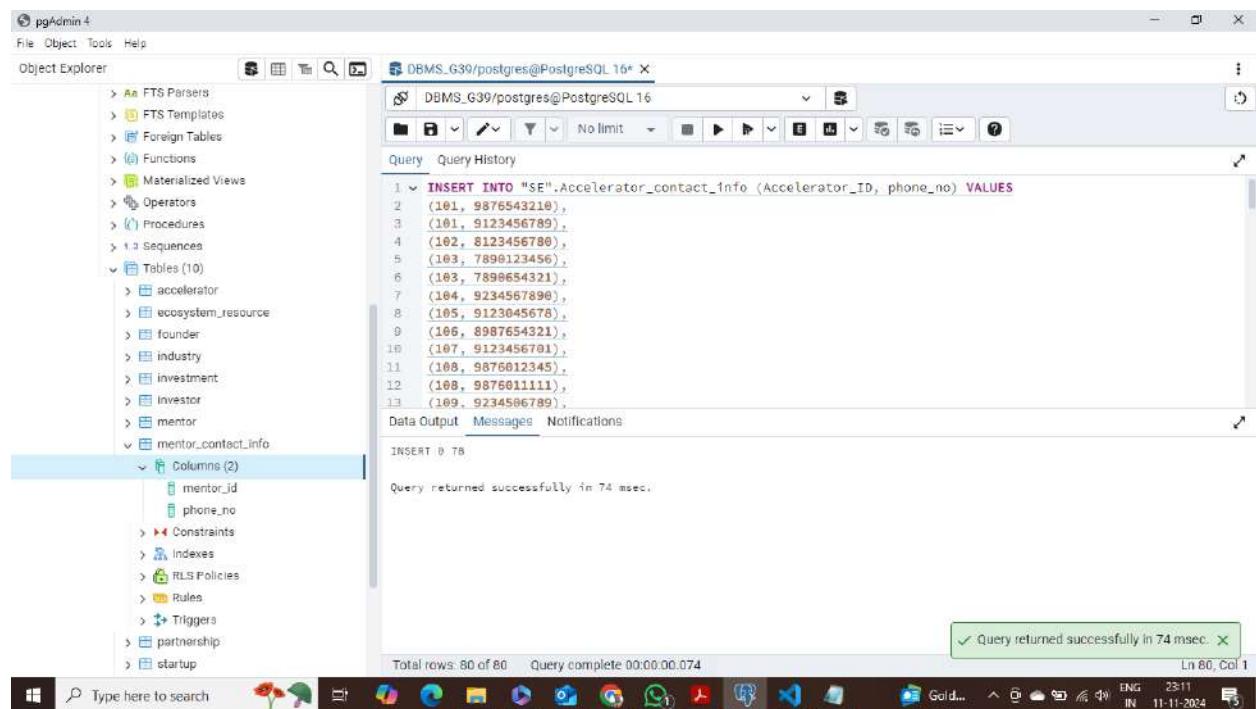
```
(511, '7890987654'),
```

```
(512, '7012345678'),
```

(513, '8123067895'),
(514, '7998654321'),
(514, '9123456023'),
(515, '9876543012'),
(516, '7890123098'),
(517, '9234567101'),
(518, '8012345765'),
(519, '9123345678'),
(520, '7890456789'),
(521, '8123456781'),
(522, '7012987654'),
(523, '9876543201'),
(524, '9234567801'),
(524, '8123098765'),
(525, '7890123097'),
(526, '9234567893'),
(527, '7012345676'),
(528, '8123409876'),
(529, '7890123987'),
(530, '9876543109'),
(531, '9234567809'),
(532, '8123456709'),
(533, '7890234567'),
(534, '7012345673'),
(535, '9876501234'),
(535, '9234567891'),
(536, '8123456782'),

```
(537, '7890987123'),
(538, '7012789563'),
(539, '9876554321'),
(540, '9234012345'),
(541, '8123901234'),
(542, '7890543210'),
(543, '7012876543'),
(544, '9876001234'),
(545, '9234098765'),
(545, '8123678901');
```

Accelerator_Contact_Info Table



The screenshot shows the pgAdmin 4 interface. In the Object Explorer, under the 'Tables (10)' section, the 'mentor_contact_info' table is selected. The 'Columns (2)' section shows two columns: 'mentor_id' and 'phone_no'. In the main query window, a query is being run:

```
1 ✓ INSERT INTO "SE".Accelerator_contact_info (Accelerator_ID, phone_no) VALUES
2   (101, 9876543210),
3   (101, 9123456789),
4   (102, 8123456780),
5   (103, 7890123456),
6   (103, 7890654321),
7   (104, 9234567896),
8   (105, 9123045678),
9   (106, 8987654321),
10  (107, 9123456781),
11  (108, 9876012345),
12  (108, 9876811111),
13  (109, 9234586789).
```

The status bar at the bottom indicates 'Total rows: 80 of 80' and 'Query complete 00:00:00.074'. A green message box in the bottom right corner says 'Query returned successfully in 74 msec.'

```
INSERT INTO "SE".Accelerator_contact_info (Accelerator_ID, phone_no) VALUES
(101, 9876543210),
```

(101, 9123456789),
(102, 8123456780),
(103, 7890123456),
(103, 7890654321),
(104, 9234567890),
(105, 9123045678),
(106, 8987654321),
(107, 9123456701),
(108, 9876012345),
(108, 9876011111),
(109, 9234506789),
(110, 8123456799),
(111, 7890987654),
(112, 7012345678),
(113, 8123067895),
(114, 7998654321),
(115, 9123456023),
(116, 9876543012),
(117, 7890123098),
(118, 9234567101),
(119, 8012345765),
(120, 9123345678),
(121, 7890456789),
(122, 8123456781),
(123, 7012987654),
(124, 9876543201),
(125, 9234567801),

(126, 8123098765),
(127, 7890123097),
(127, 9234567893),
(128, 7012345676),
(129, 8123409876),
(130, 7890123987),
(131, 9876543109),
(132, 9234567809),
(133, 8123456709),
(134, 7890234567),
(135, 7012345673),
(136, 9876501234),
(137, 9234567891),
(138, 8123456782),
(139, 7890987123),
(140, 7012789563),
(141, 9876554321),
(142, 9234012345),
(143, 8123901234),
(144, 7890543210),
(145, 7012876543),
(146, 9876001234),
(147, 9234098765),
(148, 8123678901),
(149, 7890345678),
(150, 7012009876),
(150, 9876112345),

(151, 9234123456),
(152, 8123789012),
(153, 7890564321),
(154, 7012456789),
(155, 9876023456),
(156, 9234234567),
(157, 8123456012),
(158, 7890891234),
(159, 7012987345),
(160, 9876123456),
(161, 9234345678),
(162, 8123567890),
(163, 7890789432),
(164, 7012345908),
(165, 9876345678),
(166, 9234560123),
(167, 8123678091),
(168, 7890132456),
(169, 7012901234),
(170, 9876432109),
(170, 9234056789),
(170, 8123456791),
(170, 7890123678);

Investor_Contact_Info Table

The screenshot shows the pgAdmin 4 interface. On the left, the Object Explorer tree view is expanded to show various database objects like FTS Parsers, Functions, Materialized Views, Operators, Procedures, Sequences, and Tables (10). The 'mentor_contact_info' table is selected, and its details are shown in the center pane. A query window titled 'DBMS_G39/postgres@PostgreSQL 16' contains the following SQL code:

```

1 v INSERT INTO "SE".investor_contact_info (Investor_ID, phone_no) VALUES
2 (300, '9876543210'),
3 (300, '9123456780'),
4 (301, '8765432109'),
5 (301, '9988776655'),
6 (302, '7654321098'),
7 (302, '9191919191'),
8 (303, '6543210987'),
9 (303, '8080808080'),
10 (304, '5432109876'),
11 (304, '7070707070'),
12 (305, '4321098765'),
13 (305, '6060606060'),
14 (306, '3210987654'),
15 (306, '5050505050'),
16 (307, '2109876543'),

```

The status bar at the bottom indicates 'Total rows: 80 of 80' and 'Query complete 00:00:00.176'. A message box in the bottom right corner says 'Query returned successfully in 176 msec.'.

```
INSERT INTO "SE".investor_contact_info (Investor_ID, phone_no) VALUES
```

```
(300, '9876543210'),
```

```
(300, '9123456780'),
```

```
(301, '8765432109'),
```

```
(301, '9988776655'),
```

```
(302, '7654321098'),
```

```
(302, '9191919191'),
```

```
(303, '6543210987'),
```

```
(303, '8080808080'),
```

```
(304, '5432109876'),
```

```
(304, '7070707070'),
```

```
(305, '4321098765'),
```

```
(305, '6060606060'),
```

```
(306, '3210987654'),
```

```
(306, '5050505050'),
```

```
(307, '2109876543'),
```

(307, '4040404040'),
(308, '1098765432'),
(308, '3030303030'),
(309, '0987654321'),
(309, '2020202020'),
(310, '9876501234'),
(310, '9191919292'),
(311, '8765401234'),
(311, '9292929292'),
(312, '7654301234'),
(312, '8383838383'),
(313, '6543201234'),
(313, '7474747474'),
(314, '5432101234'),
(314, '6565656565'),
(315, '4321001234'),
(315, '5656565656'),
(316, '3210901234'),
(316, '4747474747'),
(317, '2109801234'),
(317, '3838383838'),
(318, '1098701234'),
(318, '2929292929'),
(319, '0987601234'),
(319, '1010101010'),
(320, '9876509876'),
(320, '1111111111'),

(321, '8765409876'),
(321, '1212121212'),
(322, '7654309876'),
(322, '1313131313'),
(323, '6543209876'),
(323, '1414141414'),
(324, '5432109876'),
(324, '1515151515'),
(325, '4321009876'),
(325, '1616161616'),
(326, '3210909876'),
(326, '1717171717'),
(327, '2109809876'),
(327, '1818181818'),
(328, '1098709876'),
(328, '1919191919'),
(329, '0987609876'),
(329, '2020202020'),
(330, '9876512345'),
(330, '2121212121'),
(331, '8765412345'),
(331, '2222222222'),
(332, '7654312345'),
(332, '2323232323'),
(333, '6543212345'),
(333, '2424242424'),
(334, '5432112345'),

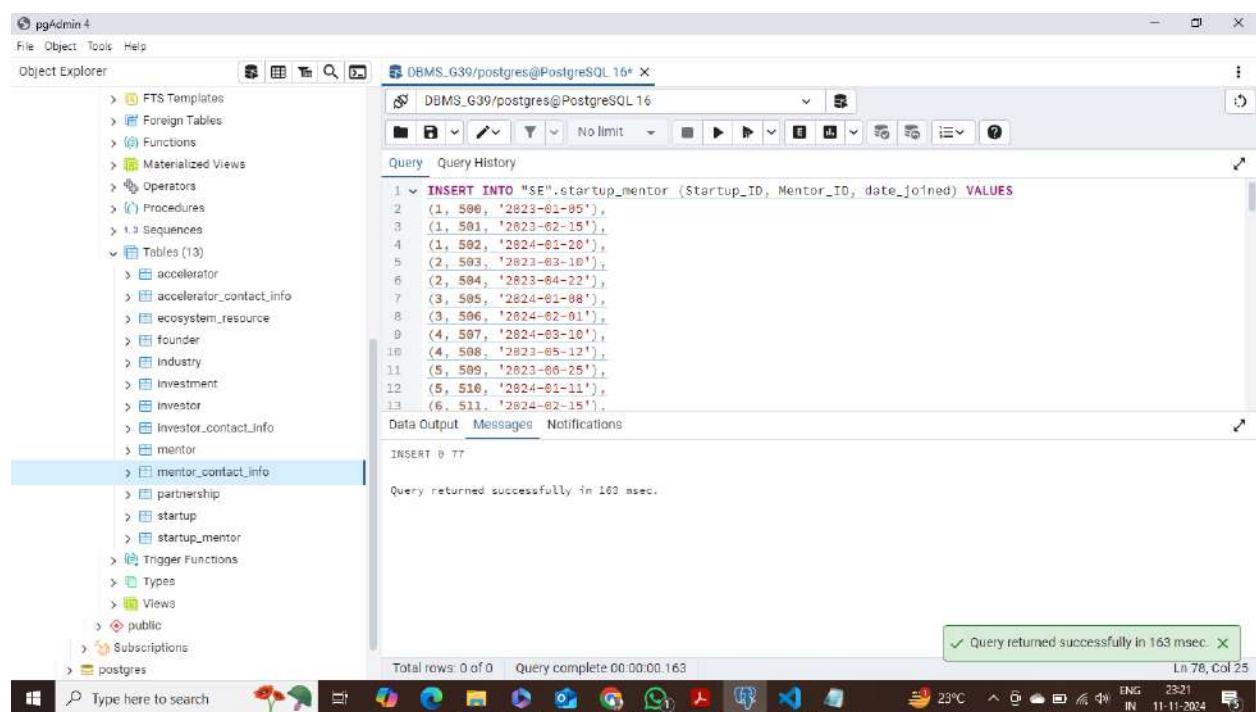
(334, '2525252525'),
(335, '4321112345'),
(335, '2626262626'),
(336, '3211012345'),
(336, '2727272727'),
(337, '2109912345'),
(337, '2828282828'),
(338, '1098712345'),
(338, '2929292929'),
(339, '0987612345'),
(339, '3030303030'),
(340, '9876523456'),
(340, '3131313131'),
(341, '8765423456'),
(341, '3232323232'),
(342, '7654323456'),
(342, '3333333333'),
(343, '6543223456'),
(343, '3434343434'),
(344, '5432123456'),
(344, '3535353535'),
(345, '4321123456'),
(345, '3636363636'),
(346, '3211023456'),
(346, '3737373737'),
(347, '2109923456'),
(347, '3838383838'),

(348, '1098723456'),
(348, '3939393939'),
(349, '0987623456'),
(349, '4040404040'),
(350, '9876534567'),
(350, '4141414141'),
(351, '8765434567'),
(351, '4242424242'),
(352, '7654334567'),
(352, '4343434343'),
(353, '6543234567'),
(353, '4444444444'),
(354, '5432134567'),
(354, '4545454545'),
(355, '4321134567'),
(355, '4646464646'),
(356, '3211034567'),
(356, '4747474747'),
(357, '2109934567'),
(357, '4848484848'),
(358, '1098734567'),
(358, '4949494949'),
(359, '0987634567'),
(359, '5050505050'),
(360, '9876545678'),
(360, '5151515151'),
(361, '8765445678'),

(361, '5252525252'),
(362, '7654345678'),
(362, '5353535353'),
(363, '6543245678'),
(363, '5454545454'),
(364, '5432145678'),
(364, '5555555555'),
(365, '4321145678'),
(365, '5656565656'),
(366, '3211045678'),
(366, '5757575757'),
(367, '2109945678'),
(367, '5858585858'),
(368, '1098745678'),
(368, '5959595959'),
(369, '0987645678'),
(369, '6060606060'),
(370, '9876556789'),
(370, '6161616161'),
(371, '8765456789'),
(371, '6262626262'),
(372, '7654356789'),
(372, '6363636363'),
(373, '6543256789'),
(373, '6464646464'),
(374, '5432156789'),
(374, '6565656565'),

```
(375, '4321156789'),
(375, '6666666666'),
(376, '3211056789'),
(376, '6767676767'),
(377, '2109956789'),
(377, '6868686868'),
(378, '1098756789'),
(378, '6969696969'),
(379, '0987656789'),
(379, '7070707070');
```

Startup_Mentor Table



The screenshot shows the pgAdmin 4 interface. The left pane, 'Object Explorer', displays a tree view of database objects. Under the 'Tables' node, 'Startup_Mentor' is selected. The right pane contains a query editor window titled 'DBMS_G39/postgres@PostgreSQL 16'. The query window shows the following SQL code:

```
1 v INSERT INTO "SE".startup_mentor (Startup_ID, Mentor_ID, date_joined) VALUES
2 (1, 500, '2023-01-05'),
3 (1, 501, '2023-02-15'),
4 (1, 502, '2024-01-20'),
5 (2, 503, '2023-03-10'),
6 (2, 504, '2023-04-22'),
7 (3, 505, '2024-01-08'),
8 (3, 506, '2024-02-01'),
9 (4, 507, '2024-03-10'),
10 (4, 508, '2023-05-12'),
11 (5, 509, '2023-06-25'),
12 (5, 510, '2024-01-11'),
13 (6, 511, '2024-02-15')
```

Below the query, the status bar indicates 'Query returned successfully in 163 msec.' and 'Total rows: 0 of 0 Query complete 00:00:00.163'. The bottom right corner of the pgAdmin window shows system information: 23°C, ENG IN, 11-11-2024, and a battery icon.

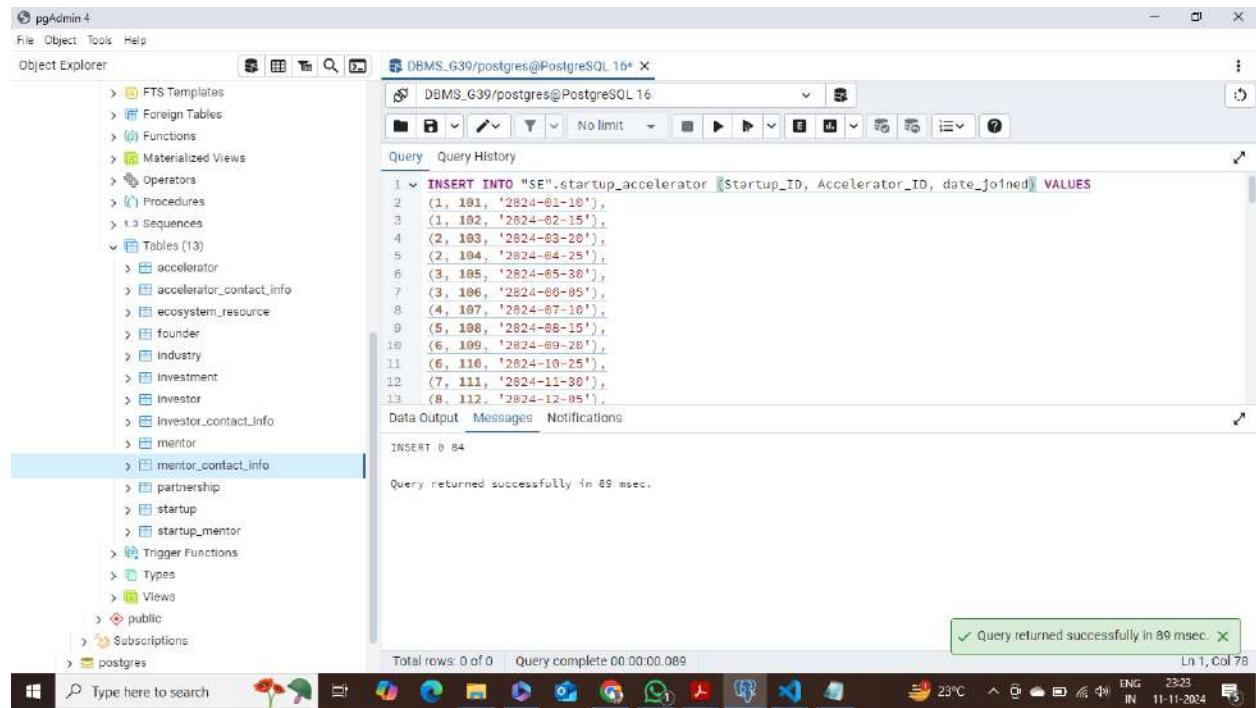
```
INSERT INTO "SE".startup_mentor (Startup_ID, Mentor_ID, Start_Date) VALUES
(1, 500, '2023-01-05'),
```

(1, 501, '2023-02-15'),
(1, 502, '2024-01-20'),
(2, 503, '2023-03-10'),
(2, 504, '2023-04-22'),
(3, 505, '2024-01-08'),
(3, 506, '2024-02-01'),
(4, 507, '2024-03-10'),
(4, 508, '2023-05-12'),
(5, 509, '2023-06-25'),
(5, 510, '2024-01-11'),
(6, 511, '2024-02-15'),
(6, 512, '2023-07-29'),
(7, 513, '2024-01-18'),
(7, 514, '2024-02-14'),
(8, 515, '2023-08-30'),
(8, 516, '2024-03-20'),
(9, 517, '2024-03-15'),
(9, 518, '2023-09-11'),
(10, 519, '2024-01-22'),
(10, 520, '2024-02-03'),
(11, 521, '2023-10-10'),
(11, 522, '2023-11-25'),
(12, 523, '2024-02-01'),
(12, 524, '2023-12-05'),
(13, 525, '2024-03-24'),
(13, 526, '2023-02-17'),
(14, 527, '2024-01-30'),

(14, 528, '2024-02-28'),
(15, 529, '2023-03-05'),
(15, 530, '2023-04-17'),
(16, 531, '2024-01-09'),
(16, 532, '2024-03-19'),
(17, 533, '2023-05-22'),
(17, 534, '2023-06-26'),
(18, 535, '2024-02-12'),
(18, 536, '2024-01-14'),
(19, 537, '2023-07-11'),
(19, 538, '2024-02-21'),
(20, 539, '2023-08-01'),
(20, 540, '2024-01-19'),
(21, 541, '2024-03-02'),
(21, 542, '2023-09-15'),
(22, 543, '2023-10-20'),
(22, 544, '2024-01-07'),
(23, 545, '2023-11-30'),
(23, 546, '2024-02-06'),
(24, 547, '2023-12-03'),
(24, 548, '2024-01-26'),
(25, 549, '2023-05-04'),
(25, 550, '2024-02-09'),
(26, 551, '2023-06-13'),
(26, 552, '2024-03-14'),
(27, 553, '2024-01-05'),
(27, 554, '2023-07-27'),

(28, 555, '2024-02-17'),
(28, 556, '2023-08-23'),
(29, 557, '2023-09-08'),
(29, 558, '2024-03-11'),
(30, 559, '2024-02-24'),
(30, 560, '2023-10-01'),
(31, 561, '2023-11-14'),
(31, 562, '2024-01-03'),
(32, 563, '2024-03-01'),
(32, 564, '2023-12-10'),
(33, 565, '2023-01-29'),
(33, 566, '2024-02-05'),
(34, 567, '2024-03-25'),
(34, 568, '2023-02-24'),
(35, 569, '2023-03-12'),
(35, 570, '2024-01-11'),
(36, 571, '2023-04-15'),
(36, 572, '2024-03-20'),
(37, 573, '2023-05-19'),
(37, 574, '2023-06-29'),
(38, 575, '2024-02-20'),
(38, 576, '2023-07-14');

Startup_Accelerator Table



```
INSERT INTO "SE".startup_accelerator (Startup_ID, Accelerator_ID, Start_Date) VALUES
```

```
(1, 101, '2024-01-10'),
(1, 102, '2024-02-15'),
(2, 103, '2024-03-20'),
(2, 104, '2024-04-25'),
(3, 105, '2024-05-30'),
(3, 106, '2024-06-05'),
(4, 107, '2024-07-10'),
(5, 108, '2024-08-15'),
(6, 109, '2024-09-20'),
(6, 110, '2024-10-25'),
(7, 111, '2024-11-30'),
(8, 112, '2024-12-05'),
(9, 113, '2025-01-10'),
(10, 114, '2025-02-15'),
(11, 115, '2025-03-20'),
```

(12, 116, '2025-04-25'),
(13, 117, '2025-05-30'),
(14, 118, '2025-06-05'),
(15, 119, '2025-07-10'),
(16, 120, '2025-08-15'),
(17, 121, '2025-09-20'),
(18, 122, '2025-10-25'),
(19, 123, '2025-11-30'),
(20, 124, '2025-12-05'),
(21, 125, '2026-01-10'),
(22, 126, '2026-02-15'),
(23, 127, '2026-03-20'),
(24, 128, '2026-04-25'),
(25, 129, '2026-05-30'),
(26, 130, '2026-06-05'),
(27, 131, '2026-07-10'),
(28, 132, '2026-08-15'),
(29, 133, '2026-09-20'),
(30, 134, '2026-10-25'),
(31, 135, '2026-11-30'),
(32, 136, '2026-12-05'),
(33, 137, '2027-01-10'),
(34, 138, '2027-02-15'),
(35, 139, '2027-03-20'),
(36, 140, '2027-04-25'),
(37, 141, '2027-05-30'),
(38, 142, '2027-06-05'),

(39, 143, '2027-07-10'),
(40, 144, '2027-08-15'),
(41, 145, '2027-09-20'),
(42, 146, '2027-10-25'),
(43, 147, '2027-11-30'),
(44, 148, '2027-12-05'),
(45, 149, '2028-01-10'),
(46, 150, '2028-02-15'),
(47, 101, '2028-03-10'),
(48, 102, '2028-04-15'),
(49, 103, '2028-05-20'),
(50, 104, '2028-06-25'),
(51, 105, '2028-07-30'),
(52, 106, '2028-08-05'),
(53, 107, '2028-09-10'),
(54, 108, '2028-10-15'),
(55, 109, '2028-11-20'),
(56, 110, '2028-12-25'),
(57, 111, '2029-01-10'),
(58, 112, '2029-02-15'),
(59, 113, '2029-03-20'),
(60, 114, '2029-04-25'),
(61, 115, '2029-05-30'),
(62, 116, '2029-06-05'),
(63, 117, '2029-07-10'),
(64, 118, '2029-08-15'),
(65, 119, '2029-09-20'),

```
(66, 120, '2029-10-25'),  
(67, 121, '2029-11-30'),  
(68, 122, '2029-12-05'),  
(69, 123, '2030-01-10'),  
(70, 124, '2030-02-15'),  
(71, 125, '2030-03-20'),  
(72, 126, '2030-04-25'),  
(73, 127, '2030-05-30'),  
(74, 128, '2030-06-05'),  
(75, 129, '2030-07-10'),  
(76, 130, '2030-08-15'),  
(77, 131, '2030-09-20'),  
(78, 132, '2030-10-25'),  
(79, 133, '2030-11-30'),  
(80, 134, '2030-12-05');
```

Data Snapshots

Startup Table

The screenshot shows the pgAdmin 4 interface with the 'Object Explorer' on the left and a 'Query' window on the right. The 'Object Explorer' lists various database objects, and the 'Tables (14)' section is expanded, with 'mentor_contact_info' selected. The 'Query' window contains the SQL command: 'select * from "SE".Startup;'. The results are displayed in a table with 14 columns: startup_id, name, industry, stage, founded_date, location, funding_ar, founder_id, fname, lname, role, linkedin_profile, and startup_id. There are 12 rows of data.

startup_id	name	industry	stage	founded_date	location	funding_ar	founder_id	fname	lname	role	linkedin_profile	startup_id
1	TechFlow	Technology	Seed	2023-01-15	San Francisco, CA	500	201	Alice	Smith	CEO	https://linkedin.com/in/alicesmith	1
2	HealthWave	Healthcare	Series A	2021-03-10	New York, NY	2000	202	Bob	Johnson	Co-founder	https://linkedin.com/in/bobjohnson	1
3	EcoSmart	Sustainability	Seed	2022-06-25	Austin, TX	300	203	Charlie	Brown	CTO	https://linkedin.com/in/charliebrown	2
4	AgriFuture	Agriculture	Series B	2020-09-05	Chicago, IL	1500	204	David	Davis	CEO	https://linkedin.com/in/daviddavis	3
5	FinTrend	Finance	Series A	2021-04-17	Boston, MA	1000	205	Eva	Garcia	Co-founder	https://linkedin.com/in/evagarcia	3
6	EduLeap	Education	Seed	2023-02-20	Seattle, WA	250	206	Frank	Miller	CEO	https://linkedin.com/in/frankmiller	4
7	SmartUrban	Urban Development	Series A	2022-05-30	Los Angeles, CA	800	207	Grace	Hernandez	CTO	https://linkedin.com/in/gracehernandez	5
8	MediaNest	Media	Seed	2023-01-08	Los Angeles, CA	400	208	Henry	Martinez	CEO	https://linkedin.com/in/henrymartinez	6
9	GlobalHealthTech	Healthcare	Series C	2019-07-15	Toronto, Canada	5000	209	Isabella	Wilson	Co-founder	https://linkedin.com/in/isabellawilson	7
10	CyberGuard	Cybersecurity	Series A	2021-08-12	Washington, D.C.	1200	210	Jack	Moore	CTO	https://linkedin.com/in/jackmoore	8
11	AI Innovations	Artificial Intelligence	Seed	2023-03-10	Silicon Valley, CA	600	211	Katherine	Taylor	CEO	https://linkedin.com/in/katherinetaylor	9
12	FoodieTech	Food Technology	Series A	2020-11-21	San Diego, CA	900	212	Liam	Anderson	Co-founder	https://linkedin.com/in/liamanderson	10

Founder Table

The screenshot shows the pgAdmin 4 interface with the 'Object Explorer' on the left and a 'Query' window on the right. The 'Object Explorer' lists various database objects, and the 'Tables (14)' section is expanded, with 'mentor_contact_info' selected. The 'Query' window contains the SQL command: 'select * from "SE".founder;'. The results are displayed in a table with 13 columns: founder_id, fname, lname, role, linkedin_profile, and startup_id. There are 13 rows of data.

founder_id	fname	lname	role	linkedin_profile	startup_id
1	Alice	Smith	CEO	https://linkedin.com/in/alicesmith	1
2	Bob	Johnson	Co-founder	https://linkedin.com/in/bobjohnson	1
3	Charlie	Brown	CTO	https://linkedin.com/in/charliebrown	2
4	David	Davis	CEO	https://linkedin.com/in/daviddavis	3
5	Eva	Garcia	Co-founder	https://linkedin.com/in/evagarcia	3
6	Frank	Miller	CEO	https://linkedin.com/in/frankmiller	4
7	Grace	Hernandez	CTO	https://linkedin.com/in/gracehernandez	5
8	Henry	Martinez	CEO	https://linkedin.com/in/henrymartinez	6
9	Isabella	Wilson	Co-founder	https://linkedin.com/in/isabellawilson	7
10	Jack	Moore	CTO	https://linkedin.com/in/jackmoore	8
11	Katherine	Taylor	CEO	https://linkedin.com/in/katherinetaylor	9
12	Liam	Anderson	Co-founder	https://linkedin.com/in/liamanderson	10
13	Mia	Thomas			

Investor Table

```
select * from "SE".investor;
```

investor_id	name	type	investment_stage_preference	location
1	300	Rajesh Sharma	Angel Investor	Seed
2	301	Priya Agarwal	Venture Capitalist	Series A
3	302	Anil Mehta	Private Equity	Series B
4	303	Sunita Rao	Venture Capitalist	Series A
5	304	Amitabh Gupta	Angel Investor	Seed
6	305	Kavita Patel	Corporate Investor	Series B
7	306	Ravilyer	Venture Capitalist	Series A
8	307	Swati Joshi	Angel Investor	Pre-Seed
9	308	Deepak Malhotra	Venture Capitalist	Seed
10	309	Neha Kapoor	Private Equity	Series B
11	310	Suresh Reddy	Corporate Investor	Series A
12	311	Vikas Singh	Angel Investor	Seed
13	312	Manisha Desai	Venture Capitalist	Delhi

Successfully run. Total query runtime: 116 msec. 81 rows affected.

Investment Table

```
select * from "SE".investment;
```

investment_id	investor_id	startup_id	amount	date	stage
1	1000	300	1	500000.00	2023-01-15
2	1001	301	1	250000.00	2023-02-20
3	1002	302	2	300000.00	2023-03-18
4	1003	303	2	450000.00	2023-04-12
5	1004	304	3	600000.00	2023-05-08
6	1005	305	3	200000.00	2023-05-28
7	1006	306	4	750000.00	2023-06-15
8	1007	307	4	500000.00	2023-07-10
9	1008	308	5	300000.00	2023-08-05
10	1009	309	5	350000.00	2023-08-25
11	1010	310	6	450000.00	2023-09-14
12	1011	311	6	550000.00	2023-10-10
13	1012	312	7	250000.00	2023-11-05

Successfully run. Total query runtime: 109 msec. 80 rows affected.

Mentor Table

The screenshot shows the pgAdmin 4 interface with the following details:

- Object Explorer:** Shows a tree view of database objects under "Tables (14)". The "mentor_contact_info" table is selected.
- Query Editor:** Contains the SQL query: `select * from "SE".mentor;`
- Data Output:** Displays the results of the query in a grid format. The columns are: mentor_id, name, expertise, affiliation, and linkedin_profile. The data includes 13 rows of mentor information, each with a unique LinkedIn profile URL.
- Status Bar:** Shows "Successfully run. Total query runtime: 113 msec. 77 rows affected." and the timestamp "Ln 1, Col 26".

Accelerator Table

The screenshot shows the pgAdmin 4 interface with the following details:

- Object Explorer:** Shows a tree view of database objects under "Tables (14)". The "mentor_contact_info" table is selected.
- Query Editor:** Contains the SQL query: `select * from "SE".accelerator;`
- Data Output:** Displays the results of the query in a grid format. The columns are: accelerator_id, name, location, industry_focus, batch_size, and program_duration. The data includes 13 rows of accelerator information.
- Status Bar:** Shows "Successfully run. Total query runtime: 370 msec. 50 rows affected." and the timestamp "Ln 1, Col 31".

Ecosystem_Resource Table

pgAdmin 4

File Object Tools Help

Object Explorer

- > FTS Configurations
- > FTS Dictionaries
- > A FTS Parsers
- > FTS Templates
- > Foreign Tables
- > Functions
- > Materialized Views
- > Operators
- > Procedures
- > Sequences
- > Tables (14)
 - > accelerator
 - > accelerator_contact_info
 - > ecosystem_resource
 - > founder
 - > Industry
 - > investment
 - > investor
 - > investor_contact_info
 - > mentor
 - > mentor_contact_info
 - > partnership
 - > startup
 - > startup_accelerator
 - > startup_mentor
 - > Trigger Functions
 - > Types

DBMS_G39/postgres@PostgreSQL_16

Query Query History

```
1 select * from "SE".ecosystem_resource;
```

Data Output Messages Notifications

resource_id	[PK] Integer	name	character varying (255)	type	character varying (50)	provider	character varying (100)	contact_info	character varying (255)
1	600	Venture Capital Group		Funding		Venture Capital Inc.		contact@venturecapital.com	
2	601	Startup Legal Solutions		Legal		Lawyers R Us		info@lawyersrus.com	
3	602	Marketing Pros		Marketing		Marketing Solutions Ltd.		contact@marketingolutions.com	
4	603	Tech Innovators Hub		Networking		Tech Network Co.		contact@technetwork.com	
5	604	Finance Advisors		Funding		Finance Experts		info@financeexperts.com	
6	605	Innovation Consulting		Consulting		Innovate Consulting		info@innovateconsulting.com	
7	606	Business Strategy Group		Consulting		Strategic Partners		contact@strategicpartners.com	
8	607	HR Solutions		HR		HR Partners		contact@hrpartners.com	
9	608	Product Development Inc.		R&D		Dev Inc.		info@devinc.com	
10	609	Legal Advisors		Legal		Legal Experts Group		info@legalexperts.com	
11	610	Cloud Services		Tech		Cloud Services LLC		support@cloudservices.com	
12	611	Sales Accelerator		Sales		Sales Experts		contact@salsexperts.com	
13	612	Market Research Institute		Research					

Total rows: 66 of 66 Query complete 00:00:00.117 Ln 1, Col 38

Successfully run. Total query runtime: 117 msec. 66 rows affected. X

Partnership Table

pgAdmin 4

File Object Tools Help

Object Explorer

- > FTS Configurations
- > FTS Dictionaries
- > A FTS Parsers
- > FTS Templates
- > Foreign Tables
- > Functions
- > Materialized Views
- > Operators
- > Procedures
- > Sequences
- > Tables (14)
 - > accelerator
 - > accelerator_contact_info
 - > ecosystem_resource
 - > founder
 - > Industry
 - > investment
 - > investor
 - > investor_contact_info
 - > mentor
 - > mentor_contact_info
 - > partnership
 - > startup
 - > startup_accelerator
 - > startup_mentor
 - > Trigger Functions
 - > Types

DBMS_G39/postgres@PostgreSQL_16

Query Query History

```
1 select * from "SE".partnership;
```

Data Output Messages Notifications

partnership_id	[PK] Integer	startup_id	integer	partner_organization_id	integer	type	character varying (50)	start_date	date	end_date	date
1	700	1		600		Strategic		2024-01-15		2025-01-15	
2	701	1		601		Marketing		2024-03-10		2024-09-10	
3	702	2		602		R&D		2024-04-20		2025-04-20	
4	703	2		603		Strategic		2024-05-15	[null]		
5	704	3		604		Marketing		2024-06-01		2025-06-01	
6	705	4		605		Funding		2024-07-15		2025-07-15	
7	706	5		606		Strategic		2024-08-20	[null]		
8	707	6		607		R&D		2024-09-25		2025-09-25	
9	708	7		608		Marketing		2024-10-30	[null]		
10	709	8		609		Strategic		2024-11-05		2025-11-05	
11	710	9		610		Funding		2024-12-01	[null]		
12	711	10		611		R&D		2024-01-10		2025-01-10	
13	712	11		612		Market					

Total rows: 66 of 66 Query complete 00:00:00.133 Ln 1, Col 31

Successfully run. Total query runtime: 133 msec. 66 rows affected. X

Industry Table

The screenshot shows the pgAdmin 4 interface with the following details:

- Object Explorer:** Shows a tree view of database objects including FTS Configurations, FTS Dictionaries, FTS Parsers, FTS Templates, Foreign Tables, Functions, Materialized Views, Operators, Procedures, Sequences, and Tables (14).
- Query Editor:** Displays the SQL query: `select * from "SE".industry;`
- Data Output:** Shows the results of the query in a table format.

industry_id	name	sector	description
1	200	Fintech	Financial technology including payment solutions, lending, and wealth management.
2	201	Healthcare IT	Technological solutions for healthcare management and patient care.
3	202	AgriTech	Technology aimed at improving agricultural efficiency and sustainability.
4	203	EdTech	Educational technology including e-learning platforms and digital courses.
5	204	Clean Energy	Sustainable energy production and renewable resources.
6	205	BioTech	Applications of biology to medicine, agriculture, and industrial processes.
7	206	Cybersecurity	Protection of networks, devices, and data from digital attacks.
8	207	Automotive	Automobile production, electric vehicles, and smart vehicle technology.
9	208	RetailTech	Technology solutions for the retail sector, including e-commerce and logistics.
10	209	PropTech	Property technology for real estate management, sales, and rentals.
11	210	FoodTech	Technologies aimed at improving food production, delivery, and quality.
12	211	Gaming	Development and distribution of video games and interactive entertainment.
13	212	Logistics	Supply Chain

Message bar: Successfully run. Total query runtime: 256 msec. 20 rows affected. Ln 1, Col 28

Mentor_Contact_Info Table

The screenshot shows the pgAdmin 4 interface with the following details:

- Object Explorer:** Shows a tree view of database objects including FTS Configurations, FTS Dictionaries, FTS Parsers, FTS Templates, Foreign Tables, Functions, Materialized Views, Operators, Procedures, Sequences, and Tables (14).
- Query Editor:** Displays the SQL query: `select * from "SE".mentor_contact_info;`
- Data Output:** Shows the results of the query in a table format.

mentor_id	phone_no
1	500 9876543210
2	500 9123456789
3	501 8123456780
4	502 7890123456
5	503 7890654321
6	504 9234567890
7	504 9123045678
8	505 8987654321
9	505 9123456780
10	507 9876012345
11	508 9876011111
12	509 9234567890
13	510 8123456799

Message bar: Successfully run. Total query runtime: 182 msec. 52 rows affected. Ln 1, Col 39

Accelerator_Contact_Info Table

The screenshot shows the pgAdmin 4 interface. The left pane is the Object Explorer, displaying various database objects like FTS Configurations, Functions, and Tables. The right pane is the Query Editor, showing the result of a query:

```
select * from "SE".accelerator_contact_info;
```

The Data Output tab shows the results of the query:

	accelerator_id	phone_no
1	101	9876543210
2	101	9123456789
3	102	8123456789
4	103	7890123456
5	103	7890054321
6	104	9234567890
7	105	9123045678
8	106	8987654321
9	107	9123456781
10	108	9876012345
11	108	9876011111
12	109	9234567899
13	110	8123456789

Total rows: 78 of 78 Query complete 00:00:00.193 Ln 1, Col 44

A green message bar at the bottom right says: Successfully run. Total query runtime: 193 msec. 78 rows affected.

Investor_Contact_Info Table

The screenshot shows the pgAdmin 4 interface. The left pane is the Object Explorer, displaying various database objects like FTS Configurations, Functions, and Tables. The right pane is the Query Editor, showing the results of a query:

```
select * from "SE".investor_contact_info;
```

The Data Output tab shows the results of the query:

	investor_id	phone_no
1	300	9876543210
2	300	9123456780
3	301	8765432109
4	301	9988776555
5	302	7654321098
6	302	9191919191
7	303	6543210987
8	303	8080800000
9	304	5432109876
10	304	7070707070
11	305	4321098765
12	305	6090606060
13	306	3210987654

Total rows: 160 of 160 Query complete 00:00:00.155 Ln 1, Col 41

A green message bar at the bottom right says: Successfully run. Total query runtime: 155 msec. 160 rows affected.

Startup_Mentor Table

```
select * from "SE".startup_mentor;
```

startup_id	mentor_id	date_joined
1	1	2023-01-05
2	1	2023-02-15
3	1	2024-01-20
4	2	2023-03-10
5	2	2023-04-22
6	3	2024-01-08
7	3	2024-02-01
8	4	2024-03-10
9	4	2023-05-12
10	5	2023-06-25
11	5	2024-01-11
12	6	2024-02-15
13	6	2023-07-29

Successfully run. Total query runtime: 171 msec. 77 rows affected.

Startup_Accelerator Table

```
select * from "SE".startup_accelerator;
```

startup_id	accelerator_id	date_joined
1	101	2024-01-10
2	102	2024-02-15
3	103	2024-03-20
4	104	2024-04-25
5	105	2024-05-30
6	106	2024-06-05
7	107	2024-07-10
8	108	2024-08-15
9	109	2024-09-20
10	110	2024-10-25
11	111	2024-11-30
12	112	2024-12-05
13	113	2025-01-10

Successfully run. Total query runtime: 106 msec. 84 rows affected.

3. SQL Queries

1. Select all startups.

```
SELECT * FROM "SE".startup;
```

The screenshot shows the PgAdmin 4 interface with the following details:

- Object Explorer:** Shows various database objects like FTS Configurations, Functions, Sequences, and 14 Tables.
- Query Editor:** Contains the SQL query: `SELECT * FROM "SE".startup;`
- Data Output:** Displays the results of the query in a tabular format. The columns are: startup_id [PK] integer, name character varying (255), industry character varying (100), stage character varying (50), founded_date date, location character varying (100), and funding_usd numeric (1).
- Table Data:** The results show 11 rows of startup information.

startup_id [PK] integer	name character varying (255)	industry character varying (100)	stage character varying (50)	founded_date date	location character varying (100)	funding_usd numeric (1)
1	TechFlow	Technology	Seed	2023-01-15	San Francisco, CA	50C
2	HealthWave	Healthcare	Series A	2021-03-10	New York, NY	200C
3	EcoSmart	Sustainability	Seed	2022-06-25	Austin, TX	30C
4	AgriFuture	Agriculture	Series B	2020-05-05	Chicago, IL	150C
5	FinTrend	Finance	Series A	2021-04-17	Boston, MA	100C
6	EduLeap	Education	Seed	2023-02-20	Seattle, WA	25C
7	SmartUrban	Urban Development	Series A	2022-05-30	Los Angeles, CA	80C
8	MediaNest	Media	Seed	2023-01-08	Los Angeles, CA	40C
9	GlobalHealthTech	Healthcare	Series C	2019-07-15	Toronto, Canada	500C
10	CyberGuard	Cybersecurity	Series A	2021-08-12	Washington, D.C.	120C
11	AI Innovations	Artificial Intelligence	Seed	2023-03-10	Silicon Valley, CA	60C

2. Get details of a specific startup by ID.

```
SELECT * FROM "SE".startup WHERE Startup_ID = 25;
```

The screenshot shows the pgAdmin 4 interface. The left sidebar is titled 'Object Explorer' and lists various database objects under 'Tables (14)'. The 'mentor_contact_info' table is currently selected. The main pane contains a query editor with the following SQL code:

```
1 SELECT * FROM "SE".startup WHERE Startup_ID = 25;
2
```

Below the query editor is a 'Data Output' tab showing the results of the query:

startup_id	[PK] integer	name	character varying (255)	industry	character varying (100)	stage	character varying (50)	founded_date	date	location	character varying (100)	funding_amount	numeric (15,2)
1	25	FashionTech		Fashion		Series A		2021-04-12		Paris, France		600000	

A green status bar at the bottom right indicates: 'Successfully run. Total query runtime: 110 msec. 1 rows affected.' and 'Ln 2, Col 1'.

3. Get all investors.

```
SELECT * FROM "SE".investor;
```

The screenshot shows the pgAdmin 4 interface. The left sidebar is titled 'Object Explorer' and lists various database objects under 'Tables (14)'. The 'mentor_contact_info' table is currently selected. The main pane contains a query editor with the following SQL code:

```
1 SELECT * FROM "SE".investor;
```

Below the query editor is a 'Data Output' tab showing the results of the query:

investor_id	[PK] integer	name	character varying (255)	type	character varying (50)	investment_stage_preference	character varying (50)	location	character varying (100)
1	300	Rajesh Sharma		Angel Investor		Seed		Mumbai	
2	301	Priya Agarwal		Venture Capitalist		Series A		Bengaluru	
3	302	Anil Mehta		Private Equity		Series B		Delhi	
4	303	Sunita Rao		Venture Capitalist		Series A		Hyderabad	
5	304	Amitabh Gupta		Angel Investor		Seed		Pune	
6	305	Keval Patel		Corporate Investor		Series B		Ahmedabad	
7	306	Ravi Iyer		Venture Capitalist		Series A		Chennai	
8	307	Swati Joshi		Angel Investor		Pre-Seed		Jaipur	
9	308	Deepak Malhotra		Venture Capitalist		Seed		Kolkata	
10	309	Neha Kapoor		Private Equity		Series B		Mumbai	
11	310	Suresh Reddy		Corporate Investor		Series A		Chennai	
12	311	Vikas Singh		Angel Investor		Pre-Seed		Jaipur	

A green status bar at the bottom right indicates: 'Successfully run. Total query runtime: 112 msec. 81 rows affected.' and 'Ln 1, Col 29'.

4. List all founder in a particular role.

```
SELECT * FROM "SE".founder WHERE role = 'CEO';
```

The screenshot shows the PgAdmin 4 interface. The left pane is the Object Explorer, displaying a tree structure of database objects. The right pane contains a query editor window titled 'DBMS_G39/postgres@PostgreSQL_16'. The query is:

```
1: SELECT * FROM "SE".founder WHERE role = 'CEO';
2:
3:
```

The results are displayed in a Data Output tab, showing a table with columns: founder_id, fname, lname, role, linkedin_profile, and startup_id. The data is as follows:

founder_id	fname	lname	role	linkedin_profile	startup_id
201	Alice	Smith	CEO	https://linkedin.com/in/alicesmith	1
204	David	Davis	CEO	https://linkedin.com/in/daviddavis	3
205	Frank	Miller	CEO	https://linkedin.com/in/frankmiller	4
208	Henry	Martinez	CEO	https://linkedin.com/in/henrymartinez	6
211	Katherine	Taylor	CEO	https://linkedin.com/in/katherinetaylor	9
214	Noah	Jackson	CEO	https://linkedin.com/in/noahjackson	12
217	Quinn	Clark	CEO	https://linkedin.com/in/quinnclark	15
220	Tina	Walker	CEO	https://linkedin.com/in/tinawalter	18
223	Will	King	CEO	https://linkedin.com/in/willking	21
225	Zane	Adams	CEO	https://linkedin.com/in/zaneadams	24
229	Cindy	Mitchell	CEO	https://linkedin.com/in/cindymitchell	27
232	Felix	Turner	CEO	https://linkedin.com/in/felixturner	30

Total rows: 28 of 28 Query complete 00:00:00.116 Ln 1, Col 1

5. Get all founders of startups.

```
SELECT * FROM "SE".founder;
```

The screenshot shows the pgAdmin 4 interface with the database 'DBMS_G39/postgres@PostgreSQL_16'. The Object Explorer on the left lists various database objects, and the main pane displays the results of the SQL query:

```
1. SELECT * FROM "SE".founder;
```

The results table shows 12 rows of data:

founder_id	fname	lname	role	linkedin_profile	startup_id
1	Alice	Smith	CEO	https://linkedin.com/in/alicesmith	1
2	Bob	Johnson	Co-founder	https://linkedin.com/in/bobjohnson	1
3	Charlie	Brown	CTO	https://linkedin.com/in/charliebrown	2
4	David	Davis	CEO	https://linkedin.com/in/daviddavis	3
5	Eva	Garcia	Co-founder	https://linkedin.com/in/evagarcia	3
6	Frank	Miller	CEO	https://linkedin.com/in/frankmiller	4
7	Grace	Hernandez	CTO	https://linkedin.com/in/gracehernandez	5
8	Henry	Martinez	CEO	https://linkedin.com/in/henrymartinez	6
9	Isabella	Wilson	Co-founder	https://linkedin.com/in/isabellwilson	7
10	Jack	Moore	CTO	https://linkedin.com/in/jackmoore	8
11	Katherine	Taylor	CEO	https://linkedin.com/in/katherinetaylor	9
12	Liam	Anderson	Co-founder	https://linkedin.com/in/liamanderson	10

Message bar: Successfully run. Total query runtime: 119 msec. 80 rows affected.

6. Get all accelerators.

```
SELECT * FROM "SE".accelerator;
```

The screenshot shows the pgAdmin 4 interface with the database 'DBMS_G39/postgres@PostgreSQL_16'. The Object Explorer on the left lists various database objects, and the main pane displays the results of the SQL query:

```
1. SELECT * FROM "SE".accelerator;
```

The results table shows 12 rows of data:

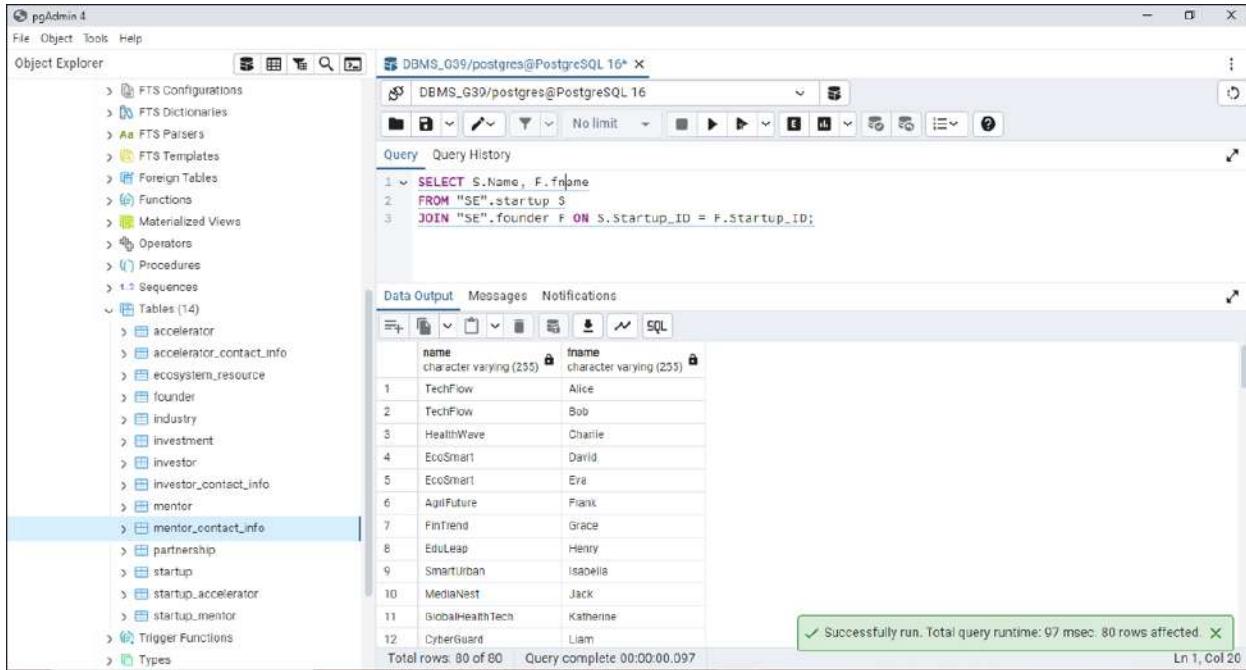
accelerator_id	name	location	industry_focus	batch_size	program_duration
1	TechStarters	San Francisco, CA	Technology	10	12
2	Health Innovators	New York, NY	Healthcare	8	6
3	Eco Boost	Austin, TX	Sustainability	15	9
4	AgrTech Hub	Chicago, IL	Agriculture	12	8
5	FinTech Fastlane	Boston, MA	Finance	10	12
6	EduAccelerate	Seattle, WA	Education	20	10
7	Smart Cities Accelerator	Los Angeles, CA	Urban Development	5	6
8	Media Mavericks	Los Angeles, CA	Media	7	12
9	Global Health Ventures	Toronto, Canada	Healthcare	6	8
10	CyberSecure	Washington, D.C.	Cybersecurity	9	6
11	AI Innovations	Silicon Valley, CA	Artificial Intelligence	10	10
12	FoodTech Labs	San Diego, CA	Food Technology	15	10

Message bar: Successfully run. Total query runtime: 114 msec. 50 rows affected.

7. Select startup names and their founders.

```
SELECT S.Name, F.Name
```

```
FROM "SE".startup S
JOIN "SE".founder F ON S.Startup_ID = F.Startup_ID;
```



The screenshot shows the pgAdmin 4 interface with the following details:

- Object Explorer:** Shows a tree view of database objects including FTS Configurations, FTS Dictionaries, FTS Parsers, FTS Templates, Foreign Tables, Functions, Materialized Views, Operators, Procedures, Sequences, and Tables (14). The 'mentor_contact_info' table is currently selected.
- Query Editor:** Displays the SQL query:


```
1 ~ SELECT S.Name, F.fname
2   FROM "SE".startup S
3     JOIN "SE".founder F ON S.Startup_ID = F.Startup_ID;
```
- Data Output:** Shows the results of the query as a table with columns 'name' and 'fname'. The data is as follows:

	name	fname
1	TechFlow	Alice
2	TechFlow	Bob
3	HealthWave	Charlie
4	EcoSmart	David
5	EcoSmart	Eva
6	AgilFuture	Frank
7	FinTrend	Grace
8	EduLeap	Henry
9	SmartUrban	Isabella
10	MediaNest	Jack
11	GlobalHealthTech	Katherine
12	CyberGuard	Liam
- Status Bar:** Shows a green message: "Successfully run. Total query runtime: 97 msec. 80 rows affected." and "Ln 1, Col 20".

8. Count startup in each stage.

```
SELECT Industry, COUNT(*) AS Num_Startups
FROM "SE".Startup
GROUP BY Industry
ORDER BY Num_Startups DESC;
```

The screenshot shows the pgAdmin 4 interface. On the left, the Object Explorer pane lists various database objects like FTS Configurations, Functions, and Tables. The Tables section shows 14 entries, with 'mentor_contact_info' selected. The main pane displays a query window with the following SQL code:

```

1 ~ SELECT Industry, COUNT(*) AS Num_Startups
2   FROM "SE".Startup
3   GROUP BY Industry
4   ORDER BY Num_Startups DESC;
5

```

The results are shown in a Data Output grid:

Industry	num_startups
Healthcare	8
Finance	7
Technology	5
Education	5
Clean Technology	4
Agriculture	3
Sustainability	3
Artificial Intelligence	3
Energy	3
Travel & Tourism	3
Media	3
Social Impact	2

Total rows: 36 of 36 Query complete 00:00:00.222 Ln 5, Col 1

9. Find Startups with Funding Above 10000.

```

SELECT Name, Funding_Amount
FROM "SE".Startup
WHERE Funding_Amount > 10000;j

```

The screenshot shows the pgAdmin 4 interface. On the left is the Object Explorer pane, which lists various database objects like FTS Configurations, FTS Dictionaries, FTS Parsers, FTS Templates, Foreign Tables, Functions, Materialized Views, Operators, Procedures, Sequences, and Tables (14). The 'Tables (14)' node is expanded, and the 'mentor_contact_info' table is selected. The main pane displays a SQL query window with the following code:

```

1 ✓ SELECT Name, Funding_Amount
2   FROM "SE".Startup
3   WHERE Funding_Amount > 10000;
4

```

Below the query window is a Data Output tab showing the results of the query. The results are as follows:

	name	funding_amount
1	TechFlow	500000.00
2	HealthWave	2000000.00
3	EcoSmart	300000.00
4	AgriFuture	1500000.00
5	FinTrend	1000000.00
6	EduLeap	250000.00
7	SmartUrban	800000.00
8	MediKest	400000.00
9	GlobalHealthTech	5000000.00
10	CyberGuard	1200000.00
11	AI Innovations	600000.00
12	FoodieTech	900000.00

Total rows: 80 of 80 Query complete 00:00:00.120

A green success message at the bottom right of the results pane says: "Successfully run. Total query runtime: 120 msec. 80 rows affected. Ln 3, Col 23".

10. Get Startups Founded in a Specific Year.

```

SELECT Name, Founded_Date

FROM "SE".Startup

WHERE EXTRACT(YEAR FROM Founded_Date) = 2023;

```

```

SELECT Name, Founded_Date
FROM "SE".Startup
WHERE EXTRACT(YEAR FROM Founded_Date) = 2023;

```

	name	Founded Date
1	TechFlow	2023-01-15
2	EduApp	2023-02-20
3	MediaNest	2023-01-08
4	AI Innovations	2023-03-10
5	IoT Connect	2023-04-05
6	CreativeMinds	2023-02-15
7	VR Experiences	2023-03-28
8	Impact Ventures	2023-05-01
9	Startup Hub	2023-01-20
10	TechNest	2023-02-22
11	Wearable Innovations	2023-04-15
12	Mobility Solutions	2023-03-12

11. Join Startup and Investment to Get Funding Details.

```

SELECT S.Name AS Startup_Name, I.Amount AS Investment_Amount, I.Date AS
Investment_Date
FROM "SE".Startup S
JOIN "SE".Investment I ON S.Startup_ID = I.Startup_ID;

```

```

SELECT S.Name AS Startup_Name, I.Amount AS Investment_Amount, I.Date AS Investment_Date
FROM "SE".Startup S
JOIN "SE".Investment I ON S.Startup_ID = I.Startup_ID;

```

	startup_name	investment_amount	investment_date
1	TechFlow	500000.00	2023-01-15
2	TechFlow	250000.00	2023-02-20
3	HealthWave	300000.00	2023-03-18
4	HealthWave	450000.00	2023-04-12
5	EcoSmart	600000.00	2023-05-08
6	EcoSmart	200000.00	2023-05-28
7	AgriFuture	750000.00	2023-06-15
8	AgriFuture	500000.00	2023-07-10
9	FinTrend	300000.00	2023-08-05
10	FinTrend	350000.00	2023-08-25
11	EduLeap	450000.00	2023-09-14
12	EduLeap	550000.00	2023-10-10

12. Get Investors Who Invested in a Specific Startup.

```

SELECT I.Name AS Investor_Name, I.Type AS Investor_Type, I.Investment_Stage_Preference
FROM "SE".Investor I
JOIN "SE".Investment Inv ON I.Investor_ID = Inv.Investor_ID
WHERE Inv.Startup_ID = 11;

```

```

SELECT I.Name AS Investor_Name, I.Type AS Investor_Type, I.Investment_Stage_Preference
FROM "SE".Investor I
JOIN "SE".Investment Inv ON I.Investor_ID = Inv.Investor_ID
WHERE Inv.Startup_ID = 11;
  
```

Investor_name	investor_type	investment_stage_preference
Lakshmi Pillai	Angel Investor	Seed

13. Count Startups in Each Industry.

```

SELECT Industry, COUNT(*) AS Num_Startups
FROM "SE".Startup
GROUP BY Industry;
  
```

The screenshot shows the pgAdmin 4 interface. In the Object Explorer, under the 'Tables (14)' section, the 'mentor_contact_info' table is selected. In the main window, a query is run:

```

1. SELECT Industry, COUNT(*) AS Num_Startups
2. FROM "SE".Startup
3. GROUP BY Industry;

```

The Data Output tab displays the results:

Industry	num_startups
Wellness	1
Sustainability	3
Various	1
Education	5
Home Automation	1
Wearables	1
Agriculture	3
Sports	1
Healthcare	8
Media & Entertainment	1
Media	3
Virtual Reality	1

Total rows: 36 of 36 Query complete 00:00:00.182

Message bar: Successfully run. Total query runtime: 162 msec. 36 rows affected.

14. Total Investment Amount by Investor.

```

SELECT I.Name, SUM(Inv.Amount) AS Total_Investment
FROM "SE".Investor I
JOIN "SE".Investment Inv ON I.Investor_ID = Inv.Investor_ID
GROUP BY I.Investor_ID;

```

```

SELECT I.Name, SUM(Inv.Amount) AS Total_Investment
FROM "SE".Investor I
JOIN "SE".Investment Inv ON I.Investor_ID = Inv.Investor_ID
GROUP BY I.Investor_ID;

```

	name	total_investment
1	Rajat Bhardwaj	300000.00
2	Ramesh Kale	400000.00
3	Anuja Pathak	200000.00
4	Kavita Patel	200000.00
5	Simran N	750000.00
6	Mona Nanda	500000.00
7	Nitin Kapoor	300000.00
8	Amitab Gupta	600000.00
9	Nidhi Bhatia	500000.00
10	Sunita Rao	450000.00
11	Vikram Pandey	450000.00
12	Mohit Goel	600000.00

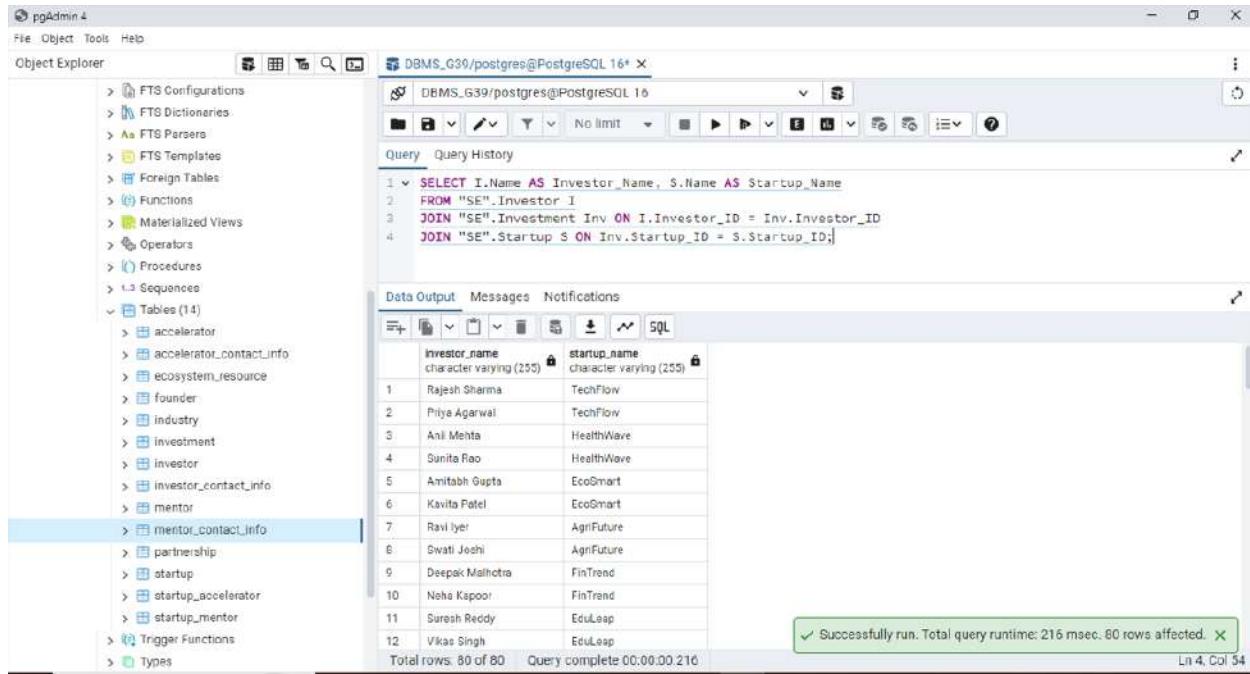
Total rows: 80 of 80 Query complete 00:00:00.105 Ltr 4, Col 24

15. List Investors and the Startups They Have Invested In.

```

SELECT I.Name AS Investor_Name, S.Name AS Startup_Name
FROM "SE".Investor I
JOIN "SE".Investment Inv ON I.Investor_ID = Inv.Investor_ID
JOIN "SE".Startup S ON Inv.Startup_ID = S.Startup_ID;

```



```

SELECT I.Name AS Investor_Name, S.Name AS Startup_Name
FROM "SE".Investor I
JOIN "SE".Investment Inv ON I.Investor_ID = Inv.Investor_ID
JOIN "SE".Startup S ON Inv.Startup_ID = S.Startup_ID;

```

	Investor_name	startup_name
1	Rajesh Sharma	TechFlow
2	Priya Agarwal	TechFlow
3	Anil Mehta	HealthWave
4	Sunita Rao	HealthWave
5	Amitabh Gupta	EcoSmart
6	Kavita Patel	EcoSmart
7	Ravi Iyer	AgriFuture
8	Swati Joshi	AgriFuture
9	Deepak Malhotra	FinTrend
10	Neha Kapoor	FinTrend
11	Suresh Reddy	EduLeap
12	Vikas Singh	EduLeap

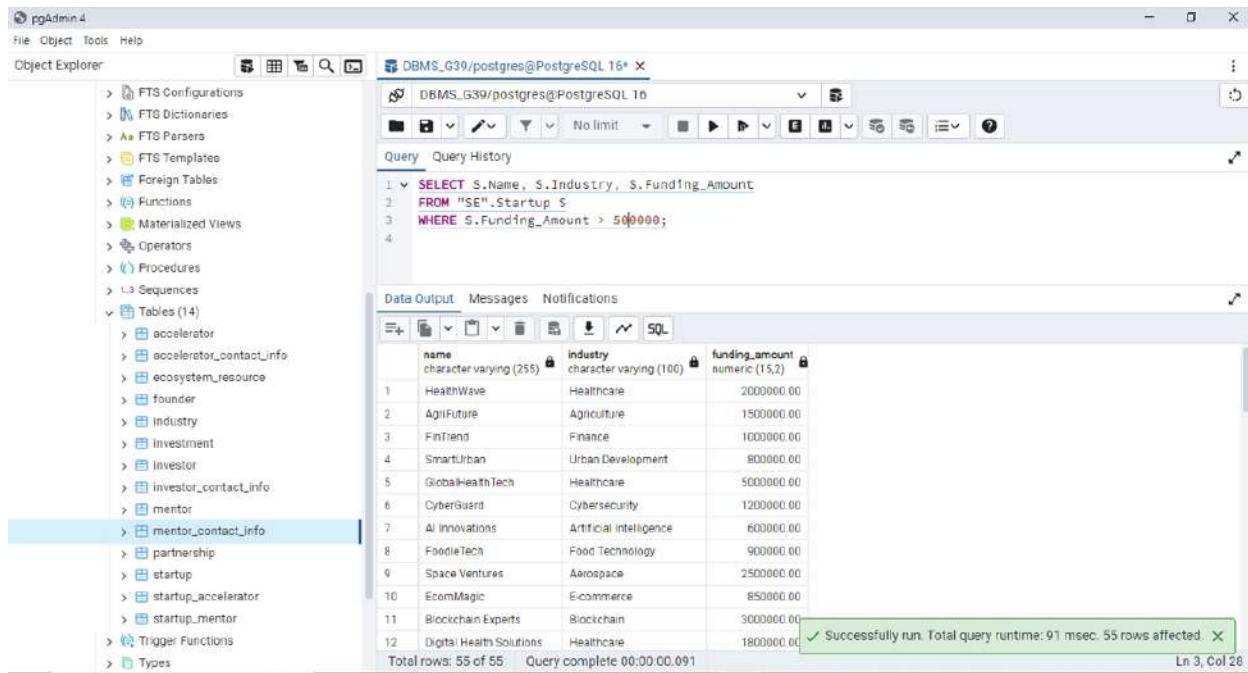
Successfully run. Total query runtime: 216 msec, 80 rows affected.

16. Find Startups and Their Industry Focus with Funding Above 500000.

```

SELECT S.Name, S.Industry, S.Funding_Amount
FROM "SE".Startup S
WHERE S.Funding_Amount > 500000;

```



```

SELECT S.Name, S.Industry, S.Funding_Amount
FROM "SE".Startup S
WHERE S.Funding_Amount > 500000;

```

	name	industry	funding_amount
1	HealthWave	Healthcare	2000000.00
2	AgriFuture	Agriculture	1500000.00
3	FinTenn	Finance	1000000.00
4	SmartUrban	Urban Development	800000.00
5	GlobalHealthTech	Healthcare	5000000.00
6	CyberGuard	Cybersecurity	1200000.00
7	AI Innovations	Artificial Intelligence	600000.00
8	FoodieTech	Food Technology	900000.00
9	Space Ventures	Aerospace	2500000.00
10	EcomMagis	E-commerce	850000.00
11	Blockchain Experts	Blockchain	3000000.00
12	Digital Health Solutions	Healthcare	1800000.00

Total rows: 55 of 55 Query complete 00:00:00.091 ✓ Successfully run. Total query runtime: 91 msec. 55 rows affected. Ln 3, Col 28

17. Get Startups That Received Funding After a Specific Date.

```

SELECT Name
FROM "SE".Startup
WHERE Startup_ID IN (SELECT DISTINCT Startup_ID FROM "SE".Investment WHERE Date > '2023-01-01');

```

```

SELECT Name
FROM "SE".Startup
WHERE Startup_ID IN (SELECT DISTINCT Startup_ID FROM "SE".Investment WHERE Date > '2023-01-01');

```

name
TechFlow
HealthWave
EcoSmart
AgriFuture
FinTrend
EduLeap
SmartUrban
MediaNest
GlobalHealthTech
CyberGuard
AI Innovations
FoodieTech

Successfully run. Total query runtime: 109 msec. 60 rows affected.

18. List Accelerators and Their Batch Sizes.

```
SELECT Name, Batch_Size FROM "SE".Accelerator;
```

```

SELECT Name, Batch_Size FROM "SE".Accelerator;

```

name	batch_size
TechStarters	10
Health Innovators	8
Eco Boost	15
AgriTech Hub	12
FinTech Fastlane	10
EduAccelerate	20
Smart Cities Accelerator	5
Media Mavericks	7
Global Health Ventures	6
CyberSecure	9
AI Innovations	11
FoodTech Labs	8

19. Find Investors Who Invested More Than the Average Investment Amount.

```
SELECT I.Name, SUM(Inv.Amount) AS Total_Investment
FROM "SE".Investor I
JOIN "SE".Investment Inv ON I.Investor_ID = Inv.Investor_ID
GROUP BY I.Investor_ID
HAVING SUM(Inv.Amount) > (SELECT AVG(Amount) FROM "SE".Investment);
```

The screenshot shows the pgAdmin 4 interface with the following details:

- Object Explorer:** Shows a tree view of database objects including FTS Configurations, FTS Dictionaries, FTS Parsers, FTS Templates, Foreign Tables, Functions, Materialized Views, Operators, Procedures, Sequences, and Tables (14).
- Query Editor:** Contains the SQL query provided above.
- Data Output:** Displays the results of the query, which are 12 rows of investor names and their total investments.
- Messages:** Shows a success message: "Successfully run. Total query runtime: 99 msec. 40 rows affected."

	name	total_investment
1	Simran N	750000.00
2	Mona Nanda	500000.00
3	Amitabh Gupta	600000.00
4	Nidhi Bhatia	500000.00
5	Mohit Goel	600000.00
6	Askaash Nanda	550000.00
7	Anrita Ghoch	650000.00
8	Roshni Tiwari	550000.00
9	Shivani Rath	700000.00
10	Gaurav Meena	500000.00
11	Taranjeet C	500000.00
12	Kriti Mohan	550000.00

20. Find the Latest Investment in Each Startup.

```
SELECT S.Name AS Startup_Name, I.Amount AS Latest_Investment_Amount, I.Date AS
Investment_Date
FROM "SE".Startup S
JOIN "SE".Investment I ON S.Startup_ID = I.Startup_ID
WHERE I.Date IN (SELECT MAX(Date) FROM "SE".Investment WHERE Startup_ID =
S.Startup_ID)
ORDER BY S.Name;
```

```

SELECT S.Name AS Startup_Name, I.Amount AS Latest_Investment_Amount, I.Date AS Investment_Date
FROM "SE".Startup S
JOIN "SE".Investment I ON S.Startup_ID = I.Startup_ID
WHERE I.Date IN (SELECT MAX(Date) FROM "SE".Investment WHERE Startup_ID = S.Startup_ID)
ORDER BY S.Name;

```

The screenshot shows the pgAdmin 4 interface with the following details:

- Object Explorer:** Shows a tree view of database objects including FTS Configurations, Dictionaries, Parsers, Templates, Foreign Tables, Functions, Materialized Views, Operators, Procedures, Sequences, and Tables (14).
- Query Editor:** Contains the SQL query provided above.
- Data Output:** A table showing the results of the query. The columns are startup_name, latest_investment_amount, and investment_date. The data consists of 12 rows:

	startup_name	latest_investment_amount	investment_date
1	AgriFuture	500000.00	2022-07-10
2	AgrTech Hub	450000.00	2025-06-30
3	AI Accelerator	500000.00	2025-11-10
4	AI Innovations	550000.00	2024-03-20
5	BioInnovations	750000.00	2025-05-25
6	Blockchain Experts	450000.00	2024-07-25
7	CleanAir Tech	400000.00	2026-03-10
8	CleanEnergy Solutions	300000.00	2024-11-01
9	CreativeMinds	350000.00	2024-07-10
10	CyberGuard	600000.00	2024-03-10
11	Digital Health Solutions	700000.00	2024-08-20
12	EcomMagic	300000.00	2024-06-28

- Messages:** Shows a success message: "Successfully run. Total query runtime: 85 msec. 60 rows affected."
- Notifications:** Shows "Ln 5, Col 17".

21. List Mentors and the Startups They Are Mentoring.

```

SELECT M.Name AS Mentor_Name, S.Name AS Startup_Name
FROM "SE".Mentor M
JOIN "SE".Startup_Mentor SM ON M.Mentor_ID = SM.Mentor_ID
JOIN "SE".Startup S ON SM.Startup_ID = S.Startup_ID;

```

The screenshot shows the pgAdmin 4 interface. The left pane is the Object Explorer, displaying various database objects like FTS Configurations, Functions, and Tables. The right pane is the Query Editor, showing a query that joins three tables to get mentor names and startup names. Below the query is a Data Output grid showing 12 rows of data. A status bar at the bottom indicates the query was successfully run with a runtime of 88 msec and 77 rows affected.

```

SELECT M.Name AS Mentor_Name, S.Name AS Startup_Name
FROM "SE".Mentor M
JOIN "SE".Startup_Mentor SM ON M.Mentor_ID = SM.Mentor_ID
JOIN "SE".Startup S ON SM.Startup_ID = S.Startup_ID;

```

	mentor_name	startup_name
1	Andrew	TechFlow
2	Brittany	TechFlow
3	Colin	TechFlow
4	Denise	HealthWave
5	Ethan	HealthWave
6	Fiona	EcoSmart
7	Gavin	EcoSmart
8	Hannah	AgriFuture
9	Isaac	AgriFuture
10	Julia	FinTrend
11	Kyle	FinTrend
12	Lily	EduLeap

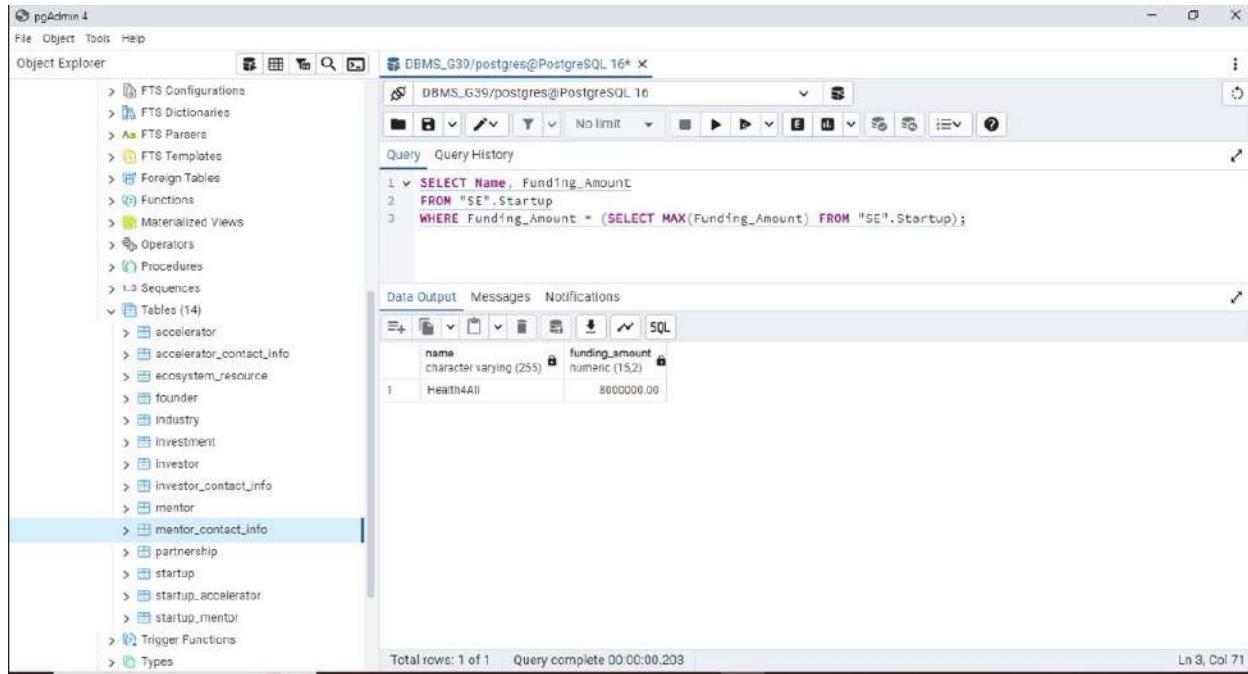
Total rows: 77 of 77 Query complete 00:00:00.086 Ln 4, Col 53

22. Find Startups with the Highest Funding Amount.

```

SELECT Name, Funding_Amount
FROM "SE".Startup
WHERE Funding_Amount = (SELECT MAX(Funding_Amount) FROM "SE".Startup);

```



The screenshot shows the pgAdmin 4 interface. The left pane is the Object Explorer, displaying various database objects like FTS Configurations, Functions, and Tables. The right pane is the Query Editor, showing a query to select the startup with the highest funding from the 'Startup' table in the 'SE' schema.

```

SELECT Name, Funding_Amount
FROM "SE".Startup
WHERE Funding_Amount = (SELECT MAX(Funding_Amount) FROM "SE".Startup);
    
```

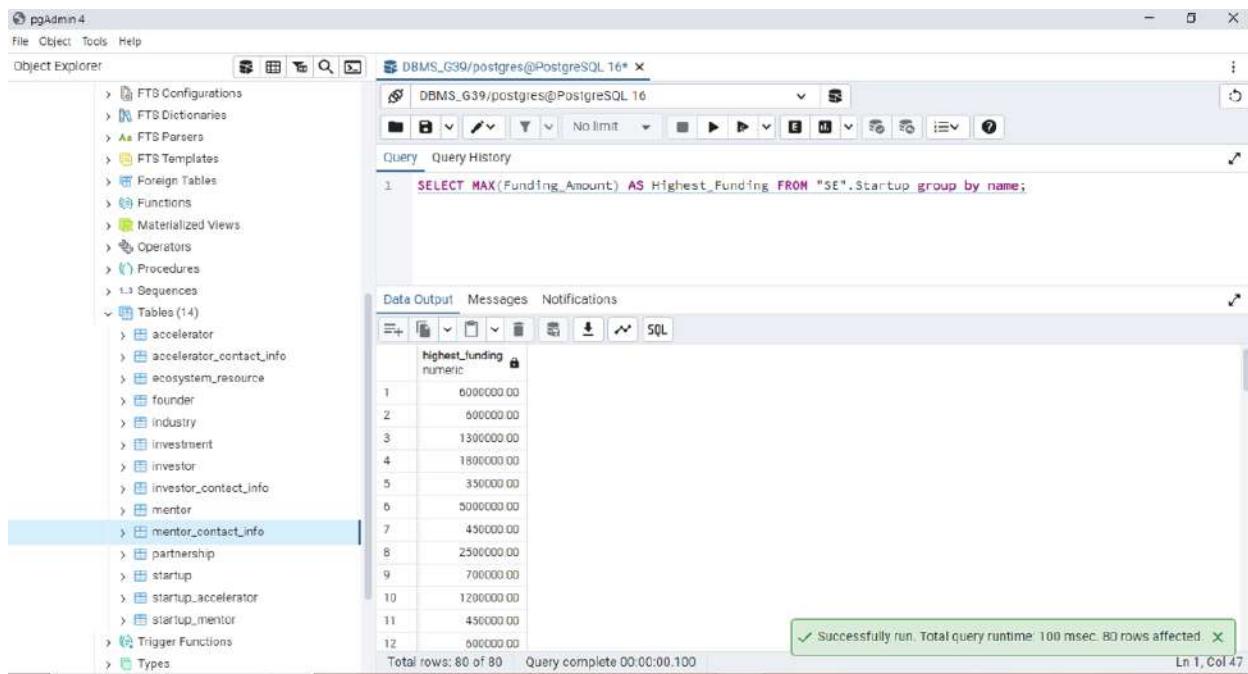
The results table shows one row:

name	funding_amount
Health4All	8000000.00

Total rows: 1 of 1 Query complete 00:00:00.203 Ln 3, Col 71

23. Find the Startup with the Highest Funding.

```
SELECT MAX(Funding_Amount) AS Highest_Funding FROM "SE".Startup Group by name;
```



The screenshot shows the pgAdmin 4 interface. The left pane is the Object Explorer, displaying various database objects like FTS Configurations, Functions, and Tables. The right pane is the Query Editor, showing a query to find the highest funding amount for each startup and group the results by startup name.

```

SELECT MAX(Funding_Amount) AS Highest_Funding FROM "SE".Startup group by name;
    
```

The results table shows 12 rows of startup names and their respective highest funding amounts:

Highest_funding
6000000.00
6000000.00
13000000.00
18000000.00
3500000.00
5000000.00
4500000.00
2500000.00
7000000.00
12000000.00
4560000.00
6000000.00

Total rows: 80 of 80 Query complete 00:00:00.100 ✓ Successfully run. Total query runtime: 100 msec. 80 rows affected. Ln 1, Col 47

24. List Founders and Their LinkedIn Profiles.

```
SELECT fName, lName, LinkedIn_Profile FROM "SE".Founder;
```

The screenshot shows the pgAdmin 4 interface. On the left is the Object Explorer pane, which lists various database objects like FTS Configurations, Functions, Procedures, Sequences, and 14 Tables. The 'Tables (14)' section is expanded, and the 'founder' table is selected. The main pane contains a SQL query window with the command:

```
1. SELECT fName, lName, LinkedIn_Profile FROM "SE".Founder;
```

Below the query window is a Data Output tab showing the results of the query. The results are presented in a table with columns: fName, lName, and linkedin_profile. The data consists of 12 rows:

	fName	lName	linkedin_profile
1	Alice	Smith	https://linkedin.com/in/alicesmith
2	Bob	Johnson	https://linkedin.com/in/bobjohnson
3	Charlie	Brown	https://linkedin.com/in/charliebrown
4	David	Davis	https://linkedin.com/in/daviddavis
5	Eva	Garcia	https://linkedin.com/in/evagarcia
6	Frank	Miller	https://linkedin.com/in/frankmiller
7	Grace	Hernandez	https://linkedin.com/in/gracehernandez
8	Henry	Martinez	https://linkedin.com/in/henrymartinez
9	Isabella	Wilson	https://linkedin.com/in/isabellawilson
10	Jack	Moore	https://linkedin.com/in/jackmoore
11	Katherine	Taylor	https://linkedin.com/in/katherinetaylor
12	Liam	Anderson	https://linkedin.com/in/liamanderson

At the bottom of the pgAdmin window, a message box indicates: "Successfully run. Total query runtime: 106 msec. 80 rows affected." and "Ln 1, Col 57".

25. Count Startups by Location.

```
SELECT Location, COUNT(*) AS Num_Startups
FROM "SE".Startup
GROUP BY Location;
```

```

SELECT Location, COUNT(*) AS Num_Startups
FROM "SE".Startup
GROUP BY Location;
  
```

Location	num_startups
San Diego, CA	3
San Francisco, CA	8
London, UK	2
Tokyo, Japan	1
Silicon Valley, CA	2
Vancouver, Canada	1
New York, NY	10
Amsterdam, Netherlands	1
Houston, TX	1
Atlanta, GA	1
Paris, France	3
Barcelona, Spain	2

Successfully run. Total query runtime: 100 msec. 27 rows affected.

26. Find Investors in a Specific Location.

```

SELECT Name, Type, Investment_Stage_Preference
FROM "SE".Investor
WHERE Location = 'Mumbai';
  
```

The screenshot shows the pgAdmin 4 interface. On the left, the Object Explorer pane displays various database objects like FTS Configurations, FTS Dictionaries, FTS Parsers, FTS Templates, Foreign Tables, Functions, Materialized Views, Operators, Procedures, Sequences, and 14 Tables. The 'Tables' node is expanded, showing tables such as accelerator, ecosystem_resource, founder, industry, investment, investor, investor_contact_info, mentor, mentor_contact_info, partnership, startup, startup_accelerator, startup_mentor, and Trigger Functions. The 'Tables' node itself is selected.

In the center, the main window shows a query editor with the following SQL code:

```

1 v SELECT Name, Type, Investment_Stage_Preference
2   FROM "SE".Investor
3 WHERE Location = 'Mumbai';

```

The Data Output tab displays the results of the query:

	name	type	investment_stage_preference
1	Rajesh Sharma	Angel Investor	Seed
2	Neha Kapoor	Private Equity	Series B
3	Siddharth Kulkarni	Angel Investor	Seed
4	Rajiv Sinha	Venture Capitalist	Series A
5	Komal Thakur	Corporate Investor	Series A
6	Ajay Rao	Venture Capitalist	Seed
7	Nidhi Bhatia	Angel Investor	Pre-Seed
8	Vidya Reddy	Corporate Investor	Series A
9	Mona Nanda	Angel Investor	Seed
10	Ishaan M	Private Equity	Seed
11	Megha J	Venture Capitalist	Series A

At the bottom of the pgAdmin window, a message box indicates: "Successfully run. Total query runtime: 105 msec. 11 rows affected." and "Ln 3, Col 25".

27. Find Startups and Their Founders in a Specific Location.

```

SELECT S.Name AS Startup_Name, F.fName, F.lName
FROM "SE".Startup S
JOIN "SE".Founder F ON S.Startup_ID = F.Startup_ID
WHERE S.Location = 'Los Angeles, CA';

```

```

1 v SELECT S.Name AS Startup_Name, F.FName, F.lName
2   FROM "SE".Startup S
3   JOIN "SE".Founder F ON S.Startup_ID = F.Startup_ID
4   WHERE S.Location = 'Los Angeles, CA';
5
6 -- select * from "SE".startup;

```

	startup_name	fname	lname
1	SmartUrban	Isabella	Wilson
2	MediaNest	Jack	Moore
3	CreativeMinds	Rita	Lewis
4	VR Experiences	Tina	Walker
5	Wearable Innovations	Daniel	Perez
6	VR/AR Innovations	Paula	Rogers
7	AI Accelerator	Yasmine	Fleming
8	SmartEnergy	Diana	Cook
9	SmartEnergy	Edward	Ford
10	FoodInnovate	Uma	George
11	CyberDefend	Yara	Fernandez
12	CyberDefend	Zoe	Bradley

Total rows: 12 of 12 Query complete 00:00:00.139 Ln 4, Col 33

28. Find All Startups Receiving Investment in a Specific Year.

```

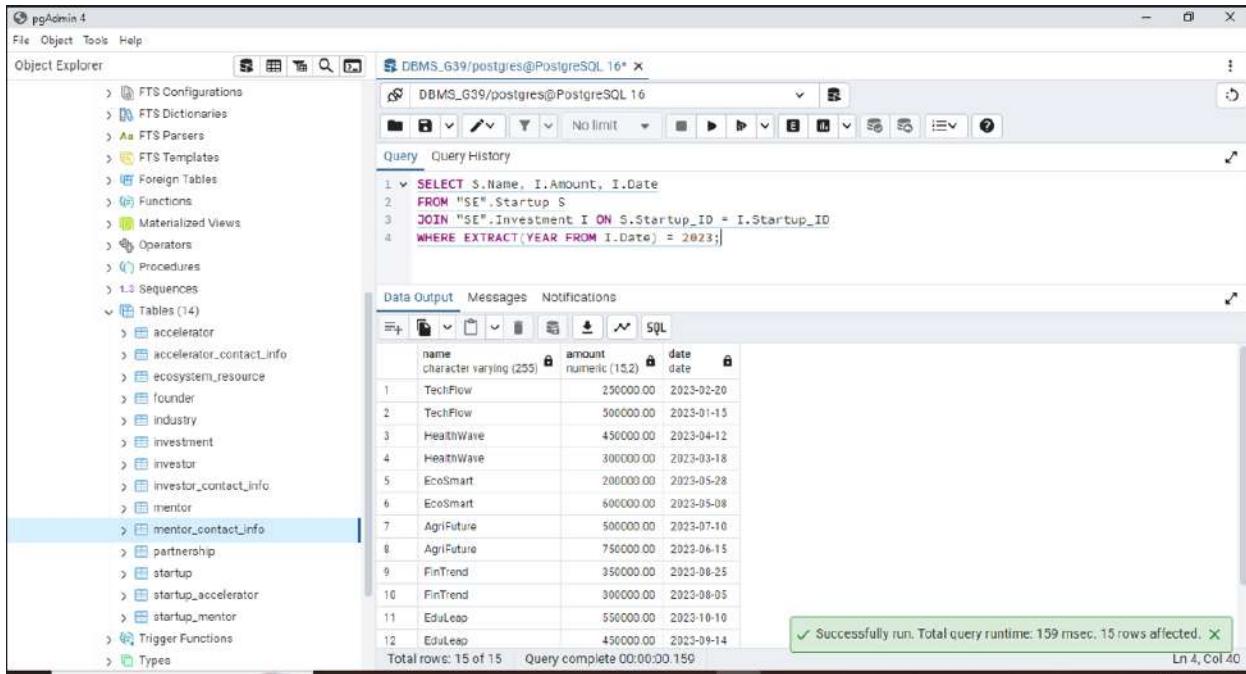
SELECT S.Name, I.Amount, I.Date

FROM "SE".Startup S

JOIN "SE".Investment I ON S.Startup_ID = I.Startup_ID

WHERE EXTRACT(YEAR FROM I.Date) = 2023;

```



```

SELECT S.Name, I.Amount, I.Date
FROM "SE".Startup S
JOIN "SE".Investment I ON S.Startup_ID = I.Startup_ID
WHERE EXTRACT(YEAR FROM I.Date) = 2023;

```

	name	amount	date
1	TechFlow	250000.00	2023-02-20
2	TechFlow	500000.00	2023-01-15
3	HealthWave	450000.00	2023-04-12
4	HealthWave	300000.00	2023-03-18
5	EcoSmart	200000.00	2023-05-28
6	EcoSmart	600000.00	2023-05-08
7	AgriFuture	500000.00	2023-07-10
8	AgriFuture	750000.00	2023-06-15
9	FinTrend	350000.00	2023-08-25
10	FinTrend	300000.00	2023-08-05
11	EduLeap	550000.00	2023-10-10
12	EduLeap	450000.00	2023-09-14

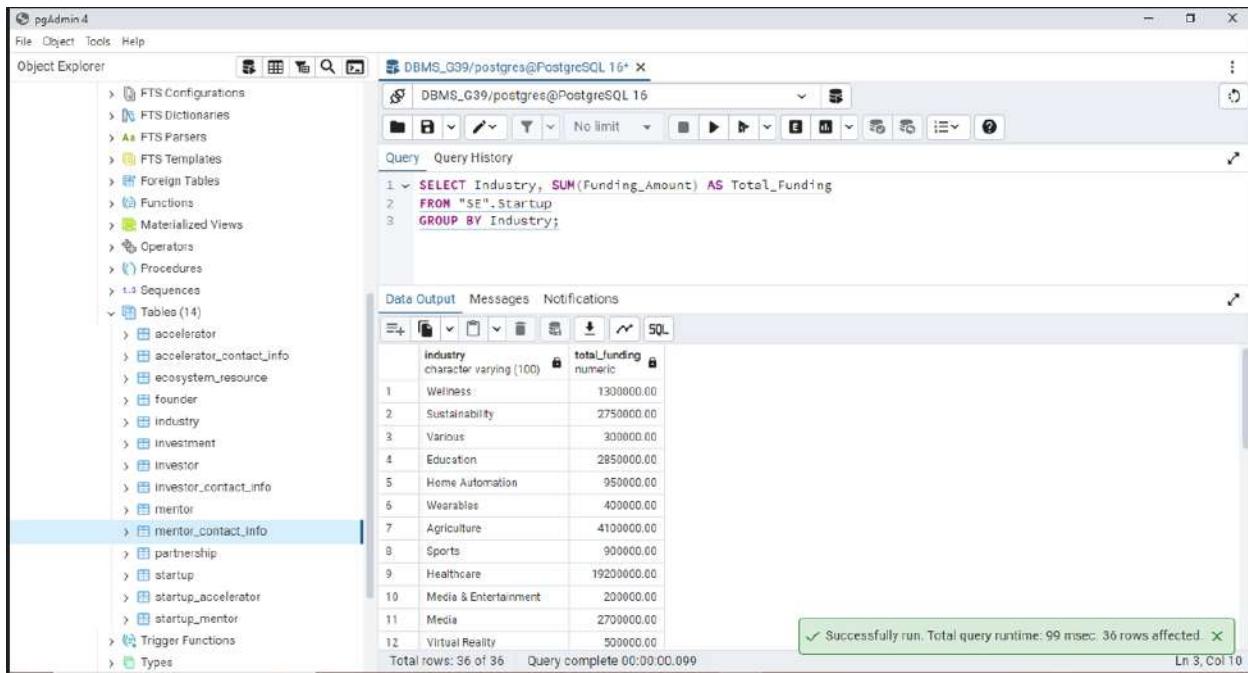
Successfully run. Total query runtime: 159 msec. 15 rows affected.

29. Get the Total Funding Amount in Each Industry.

```

SELECT Industry, SUM(Funding_Amount) AS Total_Funding
FROM "SE".Startup
GROUP BY Industry;

```



The screenshot shows the pgAdmin 4 interface. The left pane is the Object Explorer, displaying various database objects like FTS Configurations, Functions, and Tables. The right pane is the Query Editor, showing a SQL query and its results.

```
SELECT Industry, SUM(Funding_Amount) AS Total_Funding
FROM "SE".Startup
GROUP BY Industry;
```

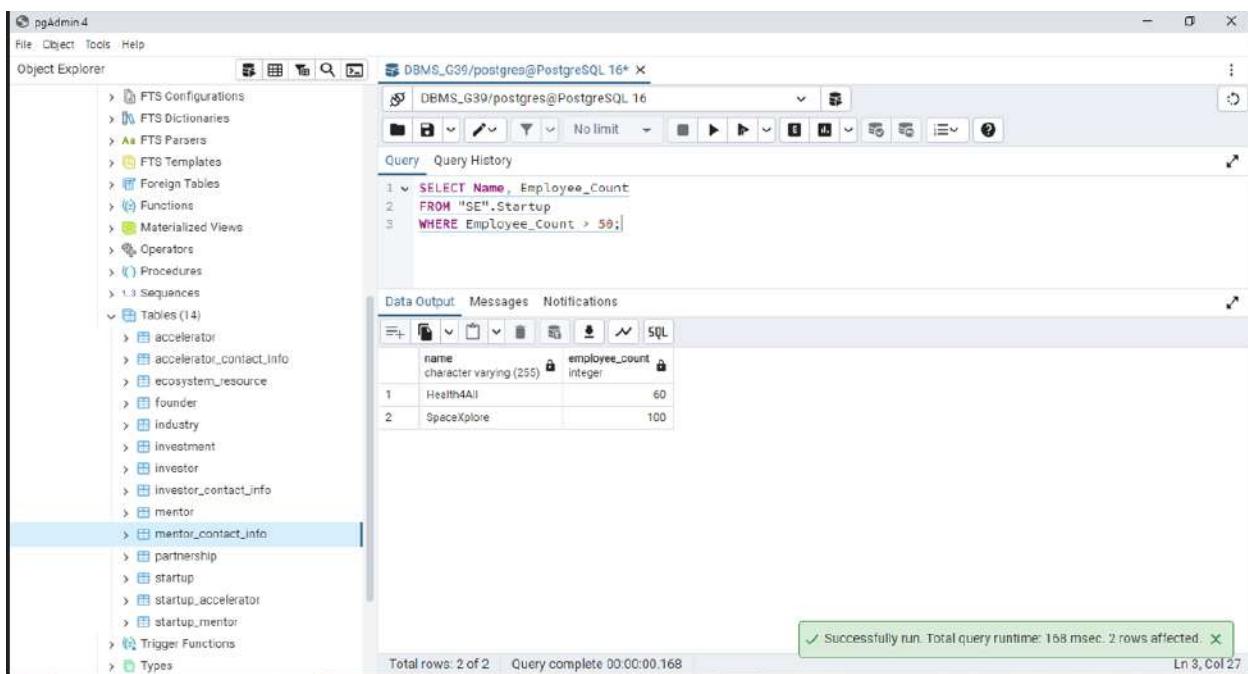
The Data Output tab shows the results of the query:

Industry	Total_Funding
Wellness	1300000.00
Sustainability	2750000.00
Various	300000.00
Education	2850000.00
Home Automation	950000.00
Wearables	400000.00
Agriculture	4100000.00
Sports	900000.00
Healthcare	1920000.00
Media & Entertainment	200000.00
Media	2700000.00
Virtual Reality	500000.00

Message bar: Successfully run. Total query runtime: 99 msec. 36 rows affected.

30. List Startups with More Than 50 Employees.

```
SELECT Name, Employee_Count
FROM "SE".Startup
WHERE Employee_Count > 50;
```



The screenshot shows the pgAdmin 4 interface. The left pane is the Object Explorer, displaying various database objects like FTS Configurations, Functions, and Tables. The right pane is the Query Editor, showing a SQL query and its results.

```
SELECT Name, Employee_Count
FROM "SE".Startup
WHERE Employee_Count > 50;
```

The Data Output tab shows the results of the query:

Name	Employee_Count
Health4All	60
SpaceXplore	100

Message bar: Successfully run. Total query runtime: 168 msec. 2 rows affected.

31. Find the Most Recent Investment for a Startup.

```
SELECT S.Name, MAX(I.Date) AS Latest_Investment_Date
FROM "SE".Startup S
JOIN "SE".Investment I ON S.Startup_ID = I.Startup_ID
GROUP BY S.Startup_ID;
```

The screenshot shows the pgAdmin 4 interface with the following details:

- Object Explorer:** Shows a tree view of database objects including FTS Configurations, FTS Dictionaries, FTS Parsers, FTS Templates, Functions, Materialized Views, Operators, Procedures, Sequences, and Tables (14).
- Query Editor:** Contains the SQL query:


```
1 v SELECT S.Name, MAX(I.Date) AS Latest_Investment_Date
2   FROM "SE".Startup S
3   JOIN "SE".Investment I ON S.Startup_ID = I.Startup_ID
4   GROUP BY S.Startup_ID;
```
- Data Output:** Displays the results of the query in a table format:

	name	latest_investment_date
1	GreenFuture	2026-01-10
2	SportsConnect	2025-01-30
3	Agrifuture	2023-07-10
4	SustainableLiving	2023-04-25
5	InnovateTech	2025-12-10
6	HealthTech Solutions	2025-12-20
7	CyberGuard	2024-03-10
8	MediaSpark	2023-05-10
9	SmartFashion	2025-09-30
10	EduLeap	2023-10-10
11	Impact Accelerator	2025-07-15
12	BioInnovations	2025-05-25
- Messages:** A green message box indicates the query was successfully run: "Successfully run. Total query runtime: 91 msec, 60 rows affected."
- Status Bar:** Shows "L14, Col 23".

32. Find the Number of Startups for Each Investor.

```
SELECT I.Name AS Investor_Name, COUNT(*) AS Num_Investments
FROM "SE".Investor I
JOIN "SE".Investment Inv ON I.Investor_ID = Inv.Investor_ID
GROUP BY I.Investor_ID;
```

```

SELECT I.Name AS Investor_Name, COUNT(+) AS Num_Investments
FROM "SE".Investor I
JOIN "SE".Investment Inv ON I.Investor_ID = Inv.Investor_ID
GROUP BY I.Investor_ID;

```

Investor_name	num_investments
Rajat Bhardwaj	1
Ramesh Kale	1
Anuja Pathak	1
Kavita Patel	1
Simran N	1
Mona Nanda	1
Nitin Kapoor	1
Amitabh Gupta	1
Nidhi Bhatia	1
Sunita Rao	1
Vikram Pandey	1
Monit Goel	1

Successfully run. Total query runtime: 192 msec. 80 rows affected.

33. Get the Number of Mentors for Each Startup.

```

SELECT S.Name, COUNT(M.Mentor_ID) AS Num_Mentors
FROM "SE".Startup S
JOIN "SE".Startup_Mentor SM ON S.Startup_ID = SM.Startup_ID
JOIN "SE".Mentor M ON SM.Mentor_ID = M.Mentor_ID
GROUP BY S.Startup_ID;

```

The screenshot shows the pgAdmin 4 interface with the following details:

- Object Explorer:** Shows a tree view of database objects under "Tables (14)".
- Query Tab:** Contains the following SQL code:


```

1 v SELECT S.Name, COUNT(M.Mentor_ID) AS Num_Mentors
2   FROM "SE".Startup S
3   JOIN "SE".Startup_Mentor SM ON S.Startup_ID = SM.Startup_ID
4   JOIN "SE".Mentor M ON SM.Mentor_ID = M.Mentor_ID
5   GROUP BY S.Startup_ID;
      
```
- Data Output Tab:** Displays the results of the query as a table:

	name	num_mentors
1	SportsConnect	2
2	AgriFuture	2
3	SustainableLiving	2
4	CyberGuard	2
5	MediaSpark	2
6	EduLeap	2
7	BioInnovations	2
8	Robotics Innovations	2
9	Space Ventures	2
10	FinBuddy	2
11	IoT Connect	2
12	HealthWave	2
- Messages Tab:** Shows a green message: "Successfully run. Total query runtime: 96 msec. 38 rows affected."

34. List Accelerators and Their Program Durations.

```
SELECT Name, Program_Duration FROM "SE".Accelerator;
```

The screenshot shows the pgAdmin 4 interface with the following details:

- Object Explorer:** Shows a tree view of database objects under "Tables (14)".
- Query Tab:** Contains the following SQL code:


```

1 SELECT Name, Program_Duration FROM "SE".Accelerator;
2
      
```
- Data Output Tab:** Displays the results of the query as a table:

	name	program_duration
1	TechStarters	12
2	Health Innovators	6
3	Eco Boost	9
4	AgriTech Hub	8
5	FinTech Fastlane	12
6	EduAccelerate	10
7	Smart Cities Accelerator	6
8	Media Mavericks	12
9	Global Health Ventures	8
10	CyberSecure	6
11	AI Innovations	10
12	FoodTech Labs	8
- Messages Tab:** Shows a green message: "Successfully run. Total query runtime: 107 msec. 50 rows affected."

35. Find Startups with Investments in Multiple Stages.

```
SELECT S.Name, COUNT(DISTINCT I.Stage) AS Num_Stages
FROM "SE".Startup S
JOIN "SE".Investment I ON S.Startup_ID = I.Startup_ID
GROUP BY S.Startup_ID
HAVING COUNT(DISTINCT I.Stage) > 1;
```

The screenshot shows the pgAdmin 4 interface with the following details:

- Object Explorer:** Shows a tree view of database objects including FTS Configurations, FTS Dictionaries, FTS Parsers, FTS Templates, Foreign Tables, Functions, Materialized Views, Operators, Procedures, Sequences, and Tables (14). The 'mentor_contact_info' table is currently selected.
- Query Editor:** Displays the SQL query:


```
1 SELECT S.Name, COUNT(DISTINCT I.Stage) AS Num_Stages
2 FROM "SE".Startup S
3 JOIN "SE".Investment I ON S.Startup_ID = I.Startup_ID
4 GROUP BY S.Startup_ID
5 HAVING COUNT(DISTINCT I.Stage) > 1;
```
- Data Output:** Shows the results of the query in a table format:

	name	num_stages
1	FoodieTech	2
2	Impact Ventures	2
3	CleanEnergy Solutions	2
4	TechNest	2
5	Wearable Innovations	2
6	SustainableLiving	2
7	FinTech Pioneers	2
- Status Bar:** Shows the message "Successfully run. Total query runtime: 110 msec. 7 rows affected." and the location "Ln 5, Col 36".

36. Find Startups with the Maximum Number of Founders.

```
SELECT S.Name, COUNT(F.Founder_ID) AS Num_Founders
FROM "SE".Startup S
JOIN "SE".Founder F ON S.Startup_ID = F.Startup_ID
GROUP BY S.Startup_ID
ORDER BY Num_Founders DESC
LIMIT 1;
```

```

SELECT S.Name, COUNT(F.Founder_ID) AS Num_Founders
FROM "SE".Startup S
JOIN "SE".Founder F ON S.Startup_ID = F.Startup_ID
GROUP BY S.Startup_ID
ORDER BY Num_Founders DESC
LIMIT 1;
  
```

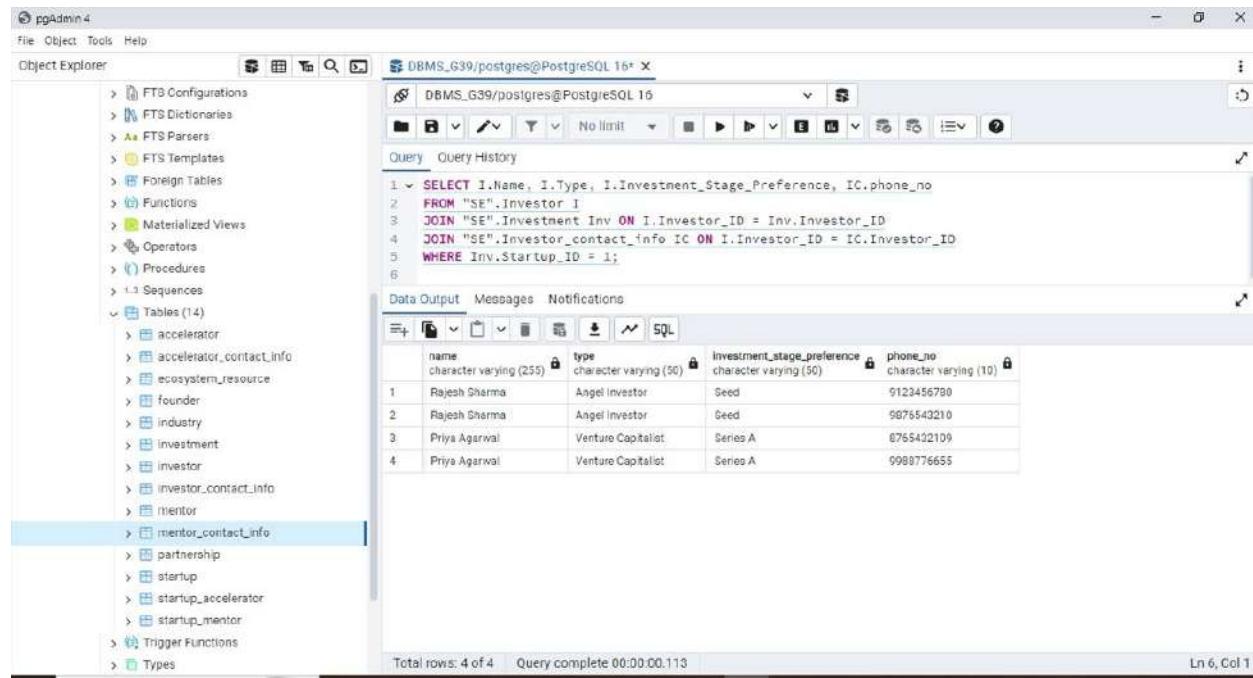
The screenshot shows the pgAdmin 4 interface with the following details:

- Object Explorer:** Shows a tree view of database objects including FTS Configurations, FTS Dictionaries, FTS Parsers, FTS Templates, Foreign Tables, Functions, Materialized Views, Operators, Procedures, Sequences, and 14 Tables.
- Query Editor:** Contains the SQL query shown above.
- Data Output:** Displays the results of the query in a table format.
- Messages:** Shows a green message bar indicating the query was successfully run.
- Status Bar:** Shows "Total rows: 1 of 1" and "Query complete 00:00:00.098".

37. Get Investors Who Have Invested in a Startup and Their Contact Info.

```

SELECT I.Name, I.Type, I.Investment_Stage_Preference, IC.phone_no
FROM "SE".Investor I
JOIN "SE".Investment Inv ON I.Investor_ID = Inv.Investor_ID
JOIN "SE".Investor_contact_info IC ON I.Investor_ID = IC.Investor_ID
WHERE Inv.Startup_ID = 1;
  
```



```

1 v SELECT I.Name, I.Type, I.Investment_Stage_Preference, IC.phone_no
2   FROM "SE".Investor I
3   JOIN "SE".Investment Inv ON I.Investor_ID = Inv.Investor_ID
4   JOIN "SE".Investor_contact_info IC ON I.Investor_ID = IC.Investor_ID
5   WHERE Inv.Startup_ID = 1;
6

```

	name	type	investment_stage_preference	phone_no
1	Rajesh Sharma	Angel Investor	Seed	9123456780
2	Rajesh Sharma	Angel Investor	Seed	9876543210
3	Priya Agarwal	Venture Capitalist	Series A	8765432109
4	Priya Agarwal	Venture Capitalist	Series A	9988776655

38. Find Startups with the Highest Number of Mentors.

```

SELECT S.Name, COUNT(M.Mentor_ID) AS Num_Mentors
FROM "SE".Startup S
JOIN "SE".Startup_Mentor SM ON S.Startup_ID = SM.Startup_ID
JOIN "SE".Mentor M ON SM.Mentor_ID = M.Mentor_ID
GROUP BY S.Startup_ID;

```

```

SELECT S.Name, COUNT(M.Mentor_ID) AS Num_Mentors
FROM "SE".Startup S
JOIN "SE".Startup_Mentor SM ON S.Startup_ID = SM.Startup_ID
JOIN "SE".Mentor M ON SM.Mentor_ID = M.Mentor_ID
GROUP BY S.Startup_ID;
  
```

name	num_mentors
SportsConnect	2
AgriFuture	2
SustainableLiving	2
CyberGuard	2
MediaSpark	2
EduLeap	2
BioInnovations	2
Robotics Innovations	2
Space Ventures	2
FinBuddy	2
IoT Connect	2
HealthWave	2

Successfully run. Total query runtime: 96 msec; 38 rows affected.

39. Find Startups with the Most Investment in a Specific Year.

```

SELECT S.Name, SUM(I.Amount) AS Total_Investment
FROM "SE".Startup S
JOIN "SE".Investment I ON S.Startup_ID = I.Startup_ID
WHERE EXTRACT(YEAR FROM I.Date) = 2023
GROUP BY S.Startup_ID
ORDER BY Total_Investment DESC
LIMIT 1;
  
```

The screenshot shows the pgAdmin 4 interface with the following details:

- Object Explorer:** Shows a tree view of database objects under "Tables (14)". One item, "mentor_contact_info", is selected.
- Query Editor:** Contains the following SQL query:


```
2 FROM "SE".Startup S
3 JOIN "SE".Investment I ON S.Startup_ID = I.Startup_ID
4 WHERE EXTRACT(YEAR FROM I.Date) = 2023
5 GROUP BY S.Startup_ID
6 ORDER BY Total_Investment DESC
7 LIMIT 1;
```
- Data Output:** Displays the result of the query in a table:

name	total_investment
AgriFuture	1250000.00
- Messages:** A green message box indicates: "Successfully run. Total query runtime: 189 msec. 1 rows affected." and "Ln 7, Col 9".

40. Find Startups with More Employees Than the Average Employee Count.

```
SELECT Name, Employee_Count
FROM "SE".Startup
WHERE Employee_Count > (SELECT AVG(Employee_Count) FROM "SE".Startup);
```

The screenshot shows the pgAdmin 4 interface with the following details:

- Object Explorer:** Shows a tree view of database objects under "Tables (14)". One item, "mentor_contact_info", is selected.
- Query Editor:** Contains the following SQL query:


```
1 SELECT Name, Employee_Count
2 FROM "SE".Startup
3 WHERE Employee_Count > (SELECT AVG(Employee_Count) FROM "SE".Startup);
4
```
- Data Output:** Displays the result of the query in a table:

name	employee_count
HealthWave	15
AgriFuture	20
GlobalHealthTech	25
Space Ventures	15
Blockchain Experts	20
Robotics Innovations	15
MediaSpark	15
Impact Accelerator	20
AI Accelerator	20
HealthAll	60
GoGreen	25
CleanAir Tech	40
- Messages:** A green message box indicates: "Total rows: 12 of 23 Query complete 00:00:00.098" and "Ln 3, Col 60".

Chapter 5: Interface Implementation

1. Setup JDBC and Basic GUI

Startup Table

DatabaseManager.java

```
import java.sql.*;  
  
public class DatabaseManager {
```

```
private static final String URL = "jdbc:postgresql://localhost:5432/DBMS_G39";
private static final String USER = "postgres";
private static final String PASSWORD = "";

public Connection connect() throws SQLException {
    return DriverManager.getConnection(URL, USER, PASSWORD);
}

public void insertStartup(String name, String industry, String stage, String founded_date, String location, int funding_amount, int employee_count) {
    String insertSQL = "INSERT INTO \"SE\".startup (name, industry, stage, founded_date, location, funding_amount, employee_count) VALUES (?, ?, ?, ?, ?, ?, ?)";
    try (Connection connection = connect()) {
        PreparedStatement pstmt = connection.prepareStatement(insertSQL) {
            pstmt.setString(1, name);
            pstmt.setString(2, industry);
            pstmt.setString(3, stage);
            pstmt.setString(4, founded_date);
            pstmt.setString(5, location);
            pstmt.setInt(6, funding_amount);
            pstmt.setInt(7, employee_count);
            pstmt.executeUpdate();
        } catch (SQLException e) {
            e.printStackTrace();
        }
    }
}

public ResultSet readStartup() {
```

```
String selectSQL = "SELECT * FROM \"SE\".startup";  
try {  
    Connection connection = connect();  
    PreparedStatement pstmt = connection.prepareStatement(selectSQL);  
    return pstmt.executeQuery();  
} catch (SQLException e) {  
    e.printStackTrace();  
}  
return null;  
  
}  
  
public void updateStartup(int startup_id, String name, String industry, String stage, String founded_date, String location, int funding_amount, int employee_count) {  
    String updateSQL = "UPDATE \"SE\".startup SET name = ?, industry = ?, stage = ?, founded_date = ?, location = ?, funding_amount = ?, employee_count = ? WHERE startup_id = ?";  
    try (Connection connection = connect()) {  
        PreparedStatement pstmt = connection.prepareStatement(updateSQL) {  
            pstmt.setString(1, name);  
            pstmt.setString(2, industry);  
            pstmt.setString(3, stage);  
            pstmt.setString(4, founded_date);  
            pstmt.setString(5, location);  
            pstmt.setInt(6, funding_amount);  
            pstmt.setInt(7, employee_count);  
            pstmt.setInt(8, startup_id);  
            pstmt.executeUpdate();  
        } catch (SQLException e) {  
            e.printStackTrace();  
        }  
    }  
}
```

```
}

}

public void deleteStartup(int startup_id) {
    String deleteSQL = "DELETE FROM \"SE\".startup WHERE startup_id = ?";
    try (Connection connection = connect()) {
        PreparedStatement pstmt = connection.prepareStatement(deleteSQL) {
            pstmt.setInt(1, startup_id);
            pstmt.executeUpdate();
        } catch (SQLException e) {
            e.printStackTrace();
        }
    }
}
```

StartupGUI.java

```
import javax.swing.*;
import javax.swing.table.DefaultTableModel;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.sql.ResultSet;
import java.sql.SQLException;
public class StartupGUI extends JFrame {
    private JTextField nameField, industryField, stageField, dateField, locationField,
    funding_amountField, employee_countField, idField;
```

```
private JButton addButton, updateButton, deleteButton, loadButton;  
private JTable startupTable;  
private DefaultTableModel tableModel;  
private DatabaseManager dbManager;  
  
public StartupGUI() {  
    dbManager = new DatabaseManager();  
    setTitle("Startup Management");  
    setLayout(new BorderLayout());  
  
    JPanel inputPanel = new JPanel(new GridLayout(4, 1));  
    inputPanel.add(new JLabel("ID:"));  
    idField = new JTextField();  
    inputPanel.add(idField);  
  
    inputPanel.add(new JLabel("Name:"));  
    nameField = new JTextField();  
    inputPanel.add(nameField);  
  
    inputPanel.add(new JLabel("Industry:"));  
    industryField = new JTextField();  
    inputPanel.add(industryField);  
  
    inputPanel.add(new JLabel("Stage:"));  
    stageField = new JTextField();  
    inputPanel.add(stageField);
```

```
inputPanel.add(new JLabel("Date:"));
dateField = new JTextField();
inputPanel.add(dateField);

inputPanel.add(new JLabel("Location:"));
locationField = new JTextField();
inputPanel.add(locationField);

inputPanel.add(new JLabel("Funding Amount:"));
funding_amountField = new JTextField();
inputPanel.add(funding_amountField);

inputPanel.add(new JLabel("Employee Count:"));
employee_countField = new JTextField();
inputPanel.add(employee_countField);

JPanel buttonPanel = new JPanel();
addButton = new JButton("Add");
updateButton = new JButton("Update");
deleteButton = new JButton("Delete");
loadButton = new JButton("Load");
buttonPanel.add(addButton);
buttonPanel.add(updateButton);
buttonPanel.add(deleteButton);
buttonPanel.add(loadButton);

tableModel = new DefaultTableModel(new String[]{"ID", "Name", "Industry", "Stage", "Date",
"Location", "FundAmount", "EmployeeCount"}, 0);
```

```
startupTable = new JTable(tableModel);

JScrollPane scrollPane = new JScrollPane(startupTable);

add(inputPanel, BorderLayout.NORTH);

add(scrollPane, BorderLayout.CENTER);

add(buttonPanel, BorderLayout.SOUTH);

addButton.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        String name = nameField.getText();

        String industry= industryField.getText();

        String stage= stageField.getText();

        String date= dateField.getText();

        String location= locationField.getText();

        int funding_amount = Integer.parseInt(funding_amountField.getText());

        int employee_count = Integer.parseInt(employee_countField.getText());

        dbManager.insertStartup(name, industry, stage, date, location ,funding_amount ,employee_count);

        loadStartup();

    }

});

updateButton.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        int startup_id = Integer.parseInt(idField.getText());
```

```
String name = nameField.getText();
String industry= industryField.getText();
String stage= stageField.getText();
String founded_date = dateField.getText();
String location= locationField.getText();
int funding_amount = Integer.parseInt(funding_amountField.getText());
int employee_count = Integer.parseInt(employee_countField.getText());
dbManager.updateStartup(startup_id,name,industry,stage,founded_date,
location ,funding_amount, employee_count);
loadStartup();
}
});

deleteButton.addActionListener(new ActionListener() {
@Override
public void actionPerformed(ActionEvent e) {
int startup_id = Integer.parseInt(idField.getText());
dbManager.deleteStartup(startup_id);
loadStartup();
}
});

loadButton.addActionListener(new ActionListener() {
@Override
public void actionPerformed(ActionEvent e) {
loadStartup();
}
});
});
```

```
setSize(600, 400);

setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

setVisible(true);

}

private void loadStartup() {

try {

ResultSet resultSet = dbManager.readStartup();

tableModel.setRowCount(0); // Clear existing data

while (resultSet != null && resultSet.next()) {

int id = resultSet.getInt("startup_id");

String name = resultSet.getString("name");

String industry= resultSet.getString("industry");

String stage = resultSet.getString("stage");

String date= resultSet.getString("founded_date");

String location= resultSet.getString("location");

int fundAmt =resultSet.getInt("funding_amount");

int employee_count= resultSet.getInt("employee_count");



tableModel.addRow(new Object[]{id, name, industry,stage,date, location ,fundAmt ,employee_count});

}

} catch (SQLException e) {

e.printStackTrace();

}

}

public static void main(String[] args) {
```

```
    new StartupGUI();  
}  
}  
}
```

Accelerator Table

DatabaseManager1.java

```
import java.sql.*;  
  
public class DatabaseManager1 {  
  
    private static final String URL = "jdbc:postgresql://localhost:5432/DBMS_G39";  
  
    private static final String USER = "postgres";  
  
    private static final String PASSWORD = "";  
  
    public Connection connect() throws SQLException {  
  
        return DriverManager.getConnection(URL, USER, PASSWORD);  
    }  
  
    public void insertAccelerator(String name, String location, String industry_focus, int batch_size, int program_duration){  
  
        String insertSQL = "INSERT INTO \"SE\".accelerator (name, location, industry_focus, batch_size, program_duration) VALUES (?, ?, ?, ?, ?)";  
  
        try (Connection connection = connect());  
        PreparedStatement pstmt = connection.prepareStatement(insertSQL)){  
            pstmt.setString(1, name);  
            pstmt.setString(2, location);  
            pstmt.setString(3, industry_focus);  
            pstmt.setInt(4, batch_size);  
        }  
    }  
}
```

```
        pstmt.setInt(5, program_duration);

        pstmt.executeUpdate();

    } catch (SQLException e) {
        e.printStackTrace();
    }
}

public ResultSet readAccelerator() {
    String selectSQL = "SELECT * FROM \"SE\".accelerator";
    try {
        Connection connection = connect();
        PreparedStatement pstmt = connection.prepareStatement(selectSQL);
        return pstmt.executeQuery();
    } catch (SQLException e) {
        e.printStackTrace();
    }
    return null;
}

public void updateAccelerator(int accelerator_id, String name, String location, String industry_focus, int batch_size, int program_duration ) {
    String updateSQL = "UPDATE \"SE\".accelerator SET name = ?, location = ?, industry_focus = ?, batch_size = ?, program_duration = ? WHERE accelerator_id = ?";
    try (Connection connection = connect()) {
        PreparedStatement pstmt = connection.prepareStatement(updateSQL) {
            pstmt.setString(1, name);
            pstmt.setString(2, location);
```

```
        pstmt.setString(3, industry_focus);
        pstmt.setInt(4, batch_size);
        pstmt.setInt(5, program_duration);
        pstmt.setInt(6, accelerator_id);
        pstmt.executeUpdate();
    } catch (SQLException e) {
        e.printStackTrace();
    }
}

public void deleteAccelerator(int accelerator_id) {
    String deleteSQL = "DELETE FROM \"SE\".accelerator WHERE accelerator_id = ?";
    try (Connection connection = connect()) {
        PreparedStatement pstmt = connection.prepareStatement(deleteSQL) {
            pstmt.setInt(1, accelerator_id);
            pstmt.executeUpdate();
        } catch (SQLException e) {
            e.printStackTrace();
        }
    }
}
```

AcceleratorGUI.java

```
import javax.swing.*;
import javax.swing.table.DefaultTableModel;
import java.awt.*;
import java.awt.event.ActionEvent;
```

```
import java.awt.event.ActionListener;
import java.sql.ResultSet;
import java.sql.SQLException;
public class AcceleratorGUI extends JFrame {
    private JTextField nameField, locationField, industry_focusField, batch_sizeField,
program_durationField, idField;
    private JButton addButton, updateButton, deleteButton, loadButton;
    private JTable acceleratorTable;
    private DefaultTableModel tableModel1;
    private DatabaseManager1 dbManager;

    public AcceleratorGUI() {
        dbManager = new DatabaseManager1();
        setTitle("Accelerator Management");
        setLayout(new BorderLayout());
        JPanel inputPanel = new JPanel(new GridLayout(5, 2));
        inputPanel.add(new JLabel("ID:"));
        idField = new JTextField();
        inputPanel.add(idField);
        inputPanel.add(new JLabel("Name:"));
        nameField = new JTextField();
        inputPanel.add(nameField);
        inputPanel.add(new JLabel("Location:"));
        locationField = new JTextField();
        inputPanel.add(locationField);
```

```
inputPanel.add(new JLabel("Industry Focus :"));

industry_focusField = new JTextField();

inputPanel.add(industry_focusField);

inputPanel.add(new JLabel("Batch Size:"));

batch_sizeField = new JTextField();

inputPanel.add(batch_sizeField);

inputPanel.add(new JLabel("Program Duration:"));

program_durationField = new JTextField();

inputPanel.add(program_durationField);

JPanel buttonPanel = new JPanel();

addButton = new JButton("Add");

updateButton = new JButton("Update");

deleteButton = new JButton("Delete");

loadButton = new JButton("Load");

buttonPanel.add(addButton);

buttonPanel.add(updateButton);

buttonPanel.add(deleteButton);

buttonPanel.add(loadButton);

tableModel1 = new DefaultTableModel(new String[]{"ID", "Name", "Location", "Industry Focus",
"Batch Size", "Program Duration"}, 0);

acceleratorTable = new JTable(tableModel1);

JSScrollPane scrollPane = new JScrollPane(acceleratorTable);
```

```
add(inputPanel, BorderLayout.NORTH);
add(scrollPane, BorderLayout.CENTER);
add(buttonPanel, BorderLayout.SOUTH);

addButton.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent e) {
        String name = nameField.getText();
        String location = locationField.getText();
        String industry_focus = industry_focusField.getText();
        int batch_size = Integer.parseInt(batch_sizeField.getText());
        int program_duration = Integer.parseInt(program_durationField.getText());
        dbManager.insertAccelerator(name, location, industry_focus, batch_size , program_duration);
        loadAccelerator();
    }
});

updateButton.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent e) {
        int accelerator_id = Integer.parseInt(idField.getText());
        String name = nameField.getText();
        String location = locationField.getText();
        String industry_focus = industry_focusField.getText();
        int batch_size = Integer.parseInt(batch_sizeField.getText());
        int program_duration = Integer.parseInt(program_durationField.getText());
        dbManager.updateAccelerator(accelerator_id, name, location, industry_focus, batch_size,
program_duration);
    }
});
```

```
        loadAccelerator();  
    }  
});  
  
deleteButton.addActionListener(new ActionListener() {  
    @Override  
    public void actionPerformed(ActionEvent e) {  
        int accelerator_id = Integer.parseInt(idField.getText());  
        dbManager.deleteAccelerator(accelerator_id);  
        loadAccelerator();  
    }  
});  
  
loadButton.addActionListener(new ActionListener() {  
    @Override  
    public void actionPerformed(ActionEvent e) {  
        loadAccelerator();  
    }  
});  
  
setSize(600, 400);  
setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);  
setVisible(true);  
}  
  
private void loadAccelerator() {  
    try {
```

```
ResultSet resultSet = dbManager.readAccelerator();

tableModel1.setRowCount(0); // Clear existing data

while (resultSet != null && resultSet.next()) {

    int id = resultSet.getInt("accelerator_id");

    String name = resultSet.getString("name");

    String location = resultSet.getString("location");

    String industry_focus = resultSet.getString("industry_focus");

    int batch_size = resultSet.getInt("batch_size");

    int program_duration = resultSet.getInt("program_duration");



    tableModel1.addRow(new Object[]{id, name, location, industry_focus, batch_size,
program_duration});

}

} catch (SQLException e) {

    e.printStackTrace();

}

}

public static void main(String[] args) {

    new AcceleratorGUI();

}

}
```

Mentor Table

DatabaseManager2.java

```
import java.sql.*;

public class DatabaseManager2 {
```

```
private static final String URL = "jdbc:postgresql://localhost:5432/DBMS_G39";
private static final String USER = "postgres";
private static final String PASSWORD = "";
public Connection connect() throws SQLException {
    return DriverManager.getConnection(URL, USER, PASSWORD);
}

public void insertMentor(String name, String expertise, String affiliation, String linkedin_profile){
    String insertSQL = "INSERT INTO \"SE\".mentor (name, expertise, affiliation, linkedin_profile)
VALUES (?, ?, ?, ?);"
    try (Connection connection = connect()) {
        PreparedStatement pstmt = connection.prepareStatement(insertSQL) {
            pstmt.setString(1, name);
            pstmt.setString(2, expertise);
            pstmt.setString(3, affiliation);
            pstmt.setString(4, linkedin_profile);
            pstmt.executeUpdate();
        } catch (SQLException e) {
            e.printStackTrace();
        }
    }
}

public ResultSet readMentor() {
    String selectSQL = "SELECT * FROM \"SE\".mentor";
    try {
        Connection connection = connect();
        PreparedStatement pstmt = connection.prepareStatement(selectSQL);
        return pstmt.executeQuery();
    }
}
```

```
        } catch (SQLException e) {
            e.printStackTrace();
        }
        return null;
    }

    public void updateMentor(int mentor_id, String name, String expertise, String affiliation, String linkedin_profile) {
        String updateSQL = "UPDATE \"SE\".mentor SET name = ?, expertise = ?, affiliation = ?, linkedin_profile = ? WHERE mentor_id = ?";
        try (Connection connection = connect()) {
            PreparedStatement pstmt = connection.prepareStatement(updateSQL) {
                pstmt.setString(1, name);
                pstmt.setString(2, name);
                pstmt.setString(3, expertise);
                pstmt.setString(4, affiliation);
                pstmt.setInt(5, mentor_id);
                pstmt.executeUpdate();
            } catch (SQLException e) {
                e.printStackTrace();
            }
        }
    }

    public void deleteMentor(int mentor_id) {
        String deleteSQL = "DELETE FROM \"SE\".mentor WHERE mentor_id = ?";
        try (Connection connection = connect()) {
            PreparedStatement pstmt = connection.prepareStatement(deleteSQL) {
                pstmt.setInt(1, mentor_id);
            }
        }
    }
}
```

```
        pstmt.executeUpdate();

    } catch (SQLException e) {
        e.printStackTrace();
    }
}
```

MentorGUI.java

```
import javax.swing.*;
import javax.swing.table.DefaultTableModel;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.sql.ResultSet;
import java.sql.SQLException;
public class MentorGUI extends JFrame {

    private JTextField nameField, expertiseField, affiliationField, linkedin_profileField, idField;
    private JButton addButton, updateButton, deleteButton, loadButton;
    private JTable mentorTable;
    private DefaultTableModel tableModel1;
    private DatabaseManager2 dbManager;
    public MentorGUI() {
        dbManager = new DatabaseManager2();
        setTitle("Mentor Management");
        setLayout(new BorderLayout());
    }
}
```

```
JPanel inputPanel = new JPanel(new GridLayout(5, 2));  
inputPanel.add(new JLabel("ID:"));  
idField = new JTextField();  
inputPanel.add(idField);  
  
inputPanel.add(new JLabel("Name:"));  
nameField = new JTextField();  
inputPanel.add(nameField);  
  
inputPanel.add(new JLabel("Expertise:"));  
expertiseField = new JTextField();  
inputPanel.add(expertiseField);  
  
inputPanel.add(new JLabel("Affiliation:"));  
affiliationField = new JTextField();  
inputPanel.add(affiliationField);  
  
inputPanel.add(new JLabel("Linkedin Profile :"));  
linkedin_profileField = new JTextField();  
inputPanel.add(linkedin_profileField);  
  
JPanel buttonPanel = new JPanel();  
addButton = new JButton("Add");  
updateButton = new JButton("Update");  
deleteButton = new JButton("Delete");  
loadButton = new JButton("Load");  
buttonPanel.add(addButton);
```

```
buttonPanel.add(updateButton);
buttonPanel.add(deleteButton);
buttonPanel.add(loadButton);

tableModel1 = new DefaultTableModel(new String[]{"ID", "Name", "Expertise", "Affiliation",
"Linkedin Profile"}, 0);
mentorTable = new JTable(tableModel1);
JScrollPane scrollPane = new JScrollPane(mentorTable);

add(inputPanel, BorderLayout.NORTH);
add(scrollPane, BorderLayout.CENTER);
add(buttonPanel, BorderLayout.SOUTH);

 addButton.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent e) {
        String name = nameField.getText();
        String expertise = expertiseField.getText();
        String affiliation = affiliationField.getText();
        String linkedin_profile = linkedin_profileField.getText();
        dbManager.insertMentor(name, expertise, affiliation, linkedin_profile);
        loadMentor();
    }
});

updateButton.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent e) {
```

```
int mentor_id = Integer.parseInt(idField.getText());
String name = nameField.getText();
String expertise = expertiseField.getText();
String affiliation = affiliationField.getText();
String linkedin_profile = linkedin_profileField.getText();
dbManager.updateMentor(mentor_id, name, expertise, affiliation, linkedin_profile);
loadMentor();
}

});

deleteButton.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent e) {
        int mentor_id = Integer.parseInt(idField.getText());
        dbManager.deleteMentor(mentor_id);
        loadMentor();
    }
});

loadButton.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent e) {
        loadMentor();
    }
});

setSize(600, 400);
```

```
setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
setVisible(true);
}

private void loadMentor() {
    try {
        ResultSet resultSet = dbManager.readMentor();
        tableModel1.setRowCount(0); // Clear existing data
        while (resultSet != null && resultSet.next()) {
            int id = resultSet.getInt("mentor_id");
            String name = resultSet.getString("name");
            String expertise = resultSet.getString("expertise");
            String affiliation = resultSet.getString("affiliation");
            String linkedin_profile = resultSet.getString("linkedin_profile");

            tableModel1.addRow(new Object[]{id, name, expertise, affiliation, linkedin_profile});
        }
    } catch (SQLException e) {
        e.printStackTrace();
    }
}

public static void main(String[] args) {
    new MentorGUI();
}
```

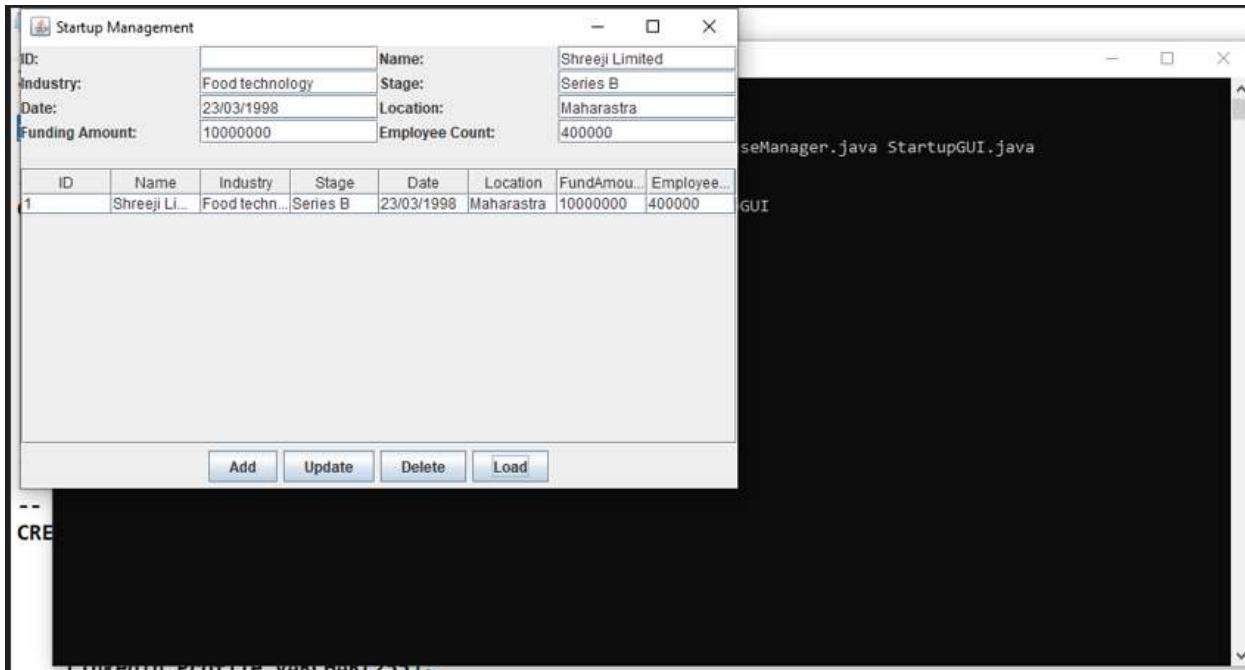
2. CRUD Operations in GUI

Startup Table

GUI

The screenshot shows a Windows application window titled "Startup Management". At the top, there are input fields for "ID", "Name", "Industry", "Stage", "Date", "Location", "Funding Amount", and "Employee Count". Below these fields is a table with columns: ID, Name, Industry, Stage, Date, Location, FundAmou..., and Employee... (partially visible). At the bottom of the window are four buttons: Add, Update, Delete, and Load.

Add Startup



Update Startup

ID:	10	Name:	Shreeji Limited				
Industry:	CyberGuard	Stage:	Seed				
Date:	2020/04/23	Location:	Mumbai				
Funding Amount:	10000	Employee Count:	400000				
ID	Name	Industry	Stage	Date	Location	FundAmount	EmployeeCount
1	Shreeji Limited	Food technology	Series B	23/03/1998	Maharashtra	10000000	400000
2	HealthWave	Healthcare	Series A	2021-03-10	New York, NY	2000000	15
3	EcoSmart	Sustainability	Seed	2022-06-25	Austin, TX	300000	4
4	Agrifuturs	Agriculture	Series B	2020-09-05	Chicago, IL	1500000	20
5	FinTrend	Finance	Series A	2021-04-17	Boston, MA	1000000	10
6	EduLop	Education	Seed	2023-02-20	Seattle, WA	250000	6
7	SmartUrban	Urban Development	Series A	2022-05-30	Los Angeles, CA	800000	8
8	MediaNext	Media	Seed	2023-01-08	Los Angeles, CA	400000	3
9	GlobalHealthTech	Healthcare	Series C	2019-07-15	Toronto, Canada	5000000	25
10	CyberGuard	Cybersecurity	Series A	2021-08-12	Washington, D.C.	1200000	12
11	AI Innovations	Artificial Intelligence	Seed	2023-03-10	Silicon Valley, CA	600000	7
12	FoodieTech	Food Technology	Series A	2020-11-21	San Diego, CA	900000	9
13	IoT Connect	Internet of Things	Seed	2023-04-05	New York, NY	350000	5
14	Space Ventures	Aerospace	Series B	2019-10-30	Houston, TX	2500000	15
15	EcomMagic	E-commerce	Series A	2021-05-02	Miami, FL	850000	10
16	Creativeminds	Media & Entertainment	Seed	2023-02-15	Los Angeles, CA	200000	4
17	Blockchain Expert	Blockchain	Series A	2022-06-17	Singapore	3000000	20
18	VR Experiences	Virtual Reality	Seed	2023-03-28	Los Angeles, CA	500000	6
19	Digital Health Solutions	Healthcare	Series B	2020-08-14	Berlin, Germany	1800000	14
20	Impact Ventures	Social impact	Seed	2023-05-01	London, UK	400000	5
21	TravelGenie	Travel & Tourism	Series A	2021-07-22	Barcelona, Spain	700000	8
22	FinBuddy	Finance	Seed	2022-12-05	Dublin, Ireland	250000	3
23	CleanEnergy Solutions	Clean Technology	Series A	2020-09-10	Vancouver, Canada	1300000	11
24	Startup Hub	Various	Seed	2023-01-20	Austin, TX	300000	6
25	FashionTech	Fashion	Series A	2021-04-12	Paris, France	600000	5
26	TechTest	Technology	Seed	2023-02-22	Tokyo, Japan	750000	9
27	EduTech Innovations	Education	Series B	2021-05-30	Melbourne, Australia	1200000	12
28	Wearable Innovations	Wearables	Seed	2023-04-15	Los Angeles, CA	400000	4
29	SportsConnect	Sports	Series A	2022-11-02	New York, NY	900000	10
30	Mobility Solutions	Transportation	Seed	2023-03-12	San Francisco, CA	600000	7
31	Robotics Innovations	Robotics	Series A	2021-07-18	Bangalore, India	1500000	15
32	Telecom Innovations	Telecommunications	Seed	2022-10-05	Chicago, IL	900000	8
33	GameTech	Gaming	Series A	2021-09-30	Seattle, WA	1100000	9
34	SustainableLiving	Sustainability	Seed	2023-01-11	Toronto, Canada	450000	6

Delete Startup

Startup Management

ID:	Name:	Industry:	Stage:	Date:	Location:	FundAmount:	EmployeeCount:
1							
2	HealthWave	Healthcare	Series A	2021-03-10	New York, NY	200000	15
3	EcoSmart	Sustainability	Seed	2022-05-25	Austin, TX	300000	4
4	AgriFuture	Agriculture	Series B	2020-09-05	Chicago, IL	150000	20
5	FinTrend	Finance	Series A	2021-04-17	Boston, MA	100000	10
6	EduLeap	Education	Seed	2023-02-20	Seattle, WA	250000	6
7	SmartUrban	Urban Development	Series A	2022-05-30	Los Angeles, CA	800000	8
8	MediaNest	Media	Seed	2023-01-08	Los Angeles, CA	400000	3
9	GlobalHealthTech	Healthcare	Series C	2019-07-15	Toronto, Canada	500000	25
11	AI Innovations	Artificial Intelligence	Seed	2023-03-10	Silicon Valley, CA	600000	7
12	FoodieTech	Food Technology	Series A	2020-11-21	San Diego, CA	900000	9
13	IoT Connect	Internet of Things	Seed	2023-04-05	New York, NY	350000	5
14	Space Ventures	Aerospace	Series B	2019-10-30	Houston, TX	2500000	15
15	Ecomagic	E-commerce	Series A	2021-05-02	Miami, FL	850000	10
16	CreativeBlends	Media & Entertainment	Seed	2023-02-15	Los Angeles, CA	200000	4
17	Blockchain Experts	Blockchain	Series A	2022-09-17	Singapore	3000000	20
18	VR Experiences	Virtual Reality	Seed	2023-03-28	Los Angeles, CA	500000	6
19	DigitalHealth Solutions	Healthcare	Series B	2020-08-14	Berlin, Germany	180000	14
20	Impact Ventures	Social Impact	Seed	2023-05-01	London, UK	400000	5
21	TravelGenie	Travel & Tourism	Series A	2021-07-22	Barcelona, Spain	700000	9
22	FinBuddy	Finance	Seed	2022-12-05	Dublin, Ireland	250000	3
23	CleanEnergy Solutions	Clean Technology	Series A	2020-09-10	Vancouver, Canada	120000	11
24	Startup Hub	Various	Seed	2023-01-20	Austin, TX	300000	6
25	FashionTech	Fashion	Series A	2021-04-12	Paris, France	600000	5
26	TechNest	Technology	Seed	2023-02-22	Tokyo, Japan	750000	9
27	EduTech Innovations	Education	Series B	2021-05-30	Melbourne, Australia	1200000	12
28	Wearable Innovations	Wearables	Seed	2023-04-15	Los Angeles, CA	400000	4
29	SportsConnect	Sports	Series A	2022-11-02	New York, NY	900000	10
30	Mobility Solutions	Transportation	Seed	2023-03-12	San Francisco, CA	600000	7
31	Robotics Innovations	Robotics	Series A	2021-07-18	Bangalore, India	1500000	15
32	Telecom Innovations	Telecommunications	Seed	2022-10-05	Chicago, IL	800000	8
33	GameTech	Gaming	Series A	2021-09-30	Seattle, WA	1100000	9
34	SustainableLiving	Sustainability	Seed	2023-01-11	Toronto, Canada	450000	6
35	MediaSpark	Media	Series B	2020-12-01	San Diego, CA	200000	15
36	BioInnovations	Biotechnology	Seed	2023-05-10	Boston, MA	300000	4

Add **Update** **Delete** **Load**

Load Startup

Startup Management

ID:	Name:	Industry:	Stage:	Date:	Location:	FundAmount:	EmployeeCount:
1							
2	HealthWave	Healthcare	Series A	2021-03-10	New York, NY	200000	15
3	EcoSmart	Sustainability	Seed	2022-05-25	Austin, TX	300000	4
4	AgriFuture	Agriculture	Series B	2020-09-05	Chicago, IL	150000	20
5	FinTrend	Finance	Series A	2021-04-17	Boston, MA	100000	10
6	EduLeap	Education	Seed	2023-02-20	Seattle, WA	250000	6
7	SmartUrban	Urban Development	Series A	2022-05-30	Los Angeles, CA	800000	8
8	MediaNest	Media	Seed	2023-01-08	Los Angeles, CA	400000	3
9	GlobalHealthTech	Healthcare	Series C	2019-07-15	Toronto, Canada	500000	25
11	AI Innovations	Artificial Intelligence	Seed	2023-03-10	Silicon Valley, CA	600000	7
12	FoodieTech	Food Technology	Series A	2020-11-21	San Diego, CA	900000	9
13	IoT Connect	Internet of Things	Seed	2023-04-05	New York, NY	350000	5
14	Space Ventures	Aerospace	Series B	2019-10-30	Houston, TX	2500000	15
15	Ecomagic	E-commerce	Series A	2021-05-02	Miami, FL	850000	10
16	CreativeBlends	Media & Entertainment	Seed	2023-02-15	Los Angeles, CA	200000	4
17	Blockchain Experts	Blockchain	Series A	2022-09-17	Singapore	3000000	20
18	VR Experiences	Virtual Reality	Seed	2023-03-28	Los Angeles, CA	500000	6
19	DigitalHealth Solutions	Healthcare	Series B	2020-08-14	Berlin, Germany	1800000	14
20	Impact Ventures	Social Impact	Seed	2023-05-01	London, UK	400000	5
21	TravelGenie	Travel & Tourism	Series A	2021-07-22	Barcelona, Spain	700000	8
22	FinBuddy	Finance	Seed	2022-12-05	Dublin, Ireland	250000	3
23	CleanEnergy Solutions	Clean Technology	Series A	2020-09-10	Vancouver, Canada	1300000	11
24	Startup Hub	Various	Seed	2023-01-20	Austin, TX	300000	6
25	FashionTech	Fashion	Series A	2021-04-12	Paris, France	600000	5
26	TechNest	Technology	Seed	2023-02-22	Tokyo, Japan	750000	9
27	EduTech Innovations	Education	Series B	2021-05-30	Melbourne, Australia	1200000	12
28	Wearable Innovations	Wearables	Seed	2023-04-15	Los Angeles, CA	400000	4
29	SportsConnect	Sports	Series A	2022-11-02	New York, NY	900000	10
30	Mobility Solutions	Transportation	Seed	2023-03-12	San Francisco, CA	600000	7
31	Robotics Innovations	Robotics	Series A	2021-07-18	Bangalore, India	1500000	15
32	Telecom Innovations	Telecommunications	Seed	2022-10-05	Chicago, IL	800000	8
33	GameTech	Gaming	Series A	2021-09-30	Seattle, WA	1100000	9
34	SustainableLiving	Sustainability	Seed	2023-01-11	Toronto, Canada	450000	6
35	MediaSpark	Media	Series B	2020-12-01	San Diego, CA	2000000	15
36	BioInnovations	Biotechnology	Seed	2023-05-10	Boston, MA	300000	4

Add **Update** **Delete** **Load**

Accelerator Table

 Accelerator Management

ID:	<input type="text"/>	Name:						
	<input type="text"/>							
Industry Focus :	<input type="text"/>	Batch Size:						
	<input type="text"/>							
<table border="1"> <thead> <tr> <th>ID</th> <th>Name</th> <th>Location</th> <th>Industry Focus</th> <th>Batch Size</th> <th>Program Durat...</th> </tr> </thead> </table>			ID	Name	Location	Industry Focus	Batch Size	Program Durat...
ID	Name	Location	Industry Focus	Batch Size	Program Durat...			
<input type="button" value="Add"/> <input type="button" value="Update"/> <input type="button" value="Delete"/> <input type="button" value="Load"/>								

Add Accelerator

 Accelerator Management

ID:	<input type="text"/>	Name:												
TechStarters	<input type="text"/>	San Francisco												
Industry Focus :	<input type="text"/>	Batch Size:												
10	<input type="text"/>	12												
<table border="1"> <thead> <tr> <th>ID</th> <th>Name</th> <th>Location</th> <th>Industry Focus</th> <th>Batch Size</th> <th>Program Duration</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>TechStarters</td> <td>San Francisco</td> <td>CA</td> <td>10</td> <td>12</td> </tr> </tbody> </table>			ID	Name	Location	Industry Focus	Batch Size	Program Duration	1	TechStarters	San Francisco	CA	10	12
ID	Name	Location	Industry Focus	Batch Size	Program Duration									
1	TechStarters	San Francisco	CA	10	12									
<input type="button" value="Add"/> <input type="button" value="Update"/> <input type="button" value="Delete"/> <input type="button" value="Load"/>														

Update Accelerator

ID	Name	Location	Industry Focus	Batch Size	Program Duration
140	VR/AR Hub	Ahmedabad	Virtual/Augmented Reality	5	5
7	Industry Focus :				
117	Blockchain Foundry	Singapore	Blockchain	5	6
118	VR Incubator	San Francisco, CA	Virtual Reality	7	10
119	Digital Health Accelerator	Berlin, Germany	Healthcare	5	6
120	Impact Innovators	London, UK	Social Impact	10	12
121	TravelTech Hub	Barcelona, Spain	Travel & Tourism	8	10
122	Fintech Factory	Dublin, Ireland	Finance	9	8
123	CleanTech Accelerator	Vancouver, Canada	Clean Technology	12	9
124	Startup Garage	Austin, TX	Various	15	10
125	Fashion Forward	Paris, France	Fashion	10	6
126	Tech Quest	Tokyo, Japan	Technology	11	12
127	EduLab	Melbourne, Australia	Education	20	10
128	Wearable Tech Incubator	Los Angeles, CA	Wearables	5	8
129	Sports Tech Accelerator	New York, NY	Sports	8	10
130	Smart Mobility Hub	San Francisco, CA	Transportation	7	9
131	Robotics Lab	Bangalore, India	Robotics	10	6
132	Telecom Innovators	Chicago, IL	Telecommunications	12	12
133	Gaming Accelerator	Seattle, WA	Gaming	10	8
134	Sustainable Living Accelerator	Toronto, Canada	Sustainability	9	10
135	Digital Media Hub	San Diego, CA	Media	11	8
136	BioTech Innovators	Boston, MA	Biotechnology	5	12
137	Smart Home Incubator	San Francisco, CA	Home Automation	0	6
138	AgriFood Accelerator	Atlanta, GA	Food Technology	10	10
139	Global Impact Accelerator	Amsterdam, Netherlands	Social Entrepreneurship	15	8
141	Investment Accelerator	New York, NY	Finance	10	10
142	Healthcare Innovations	Dublin, Ireland	Healthcare	5	12
143	Tech Frontier	Singapore	Technology	12	10
144	Elder Tech Hub	Sydney, Australia	Healthcare	9	8
145	SmartFashion Accelerator	Paris, France	Fashion	5	6
146	Energy Innovators	Berlin, Germany	Energy	10	9
147	Pet Tech Incubator	Austin, TX	Pets	8	10
148	Youth Empowerment Hub	Mumbai, India	Social Impact	20	12
149	Artificial Intelligence Accelerator	Los Angeles, CA	AI	15	10
150	FinTech 101	New York, NY	Finance	12	8
140	VR/AR Hub	Ahmedabad	Virtual/Augmented Reality	7	5

Add Update Delete Load

Delete Accelerator

ID	Name	Location	Industry Focus	Batch Size	Program Duration
115	Industry Focus :				
117	Blockchain Foundry	Singapore	Blockchain	5	6
118	VR Incubator	San Francisco, CA	Virtual Reality	7	10
119	Digital Health Accelerator	Berlin, Germany	Healthcare	5	6
120	Impact Innovators	London, UK	Social Impact	10	12
121	TravelTech Hub	Barcelona, Spain	Travel & Tourism	8	10
122	Fintech Factory	Dublin, Ireland	Finance	9	8
123	CleanTech Accelerator	Vancouver, Canada	Clean Technology	12	9
124	Startup Garage	Austin, TX	Various	15	10
125	Fashion Forward	Paris, France	Fashion	10	6
126	Tech Quest	Tokyo, Japan	Technology	11	12
127	EduLab	Melbourne, Australia	Education	20	10
128	Wearable Tech Incubator	Los Angeles, CA	Wearables	5	8
129	Sports Tech Accelerator	New York, NY	Sports	8	10
130	Smart Mobility Hub	San Francisco, CA	Transportation	7	9
131	Robotics Lab	Bangalore, India	Robotics	10	6
132	Telecom Innovators	Chicago, IL	Telecommunications	12	12
133	Gaming Accelerator	Seattle, WA	Gaming	10	8
134	Sustainable Living Accelerator	Toronto, Canada	Sustainability	9	10
135	Digital Media Hub	San Diego, CA	Media	11	8
136	BioTech Innovators	Boston, MA	Biotechnology	5	12
137	Smart Home Incubator	San Francisco, CA	Home Automation	8	6
138	AgriFood Accelerator	Atlanta, GA	Food Technology	10	10
139	Global Impact Accelerator	Amsterdam, Netherlands	Social Entrepreneurship	15	8
140	VR/AR Hub	Los Angeles, CA	Virtual/Augmented Reality	7	5
141	Investment Accelerator	New York, NY	Finance	10	10
142	Healthcare Innovations	Dublin, Ireland	Healthcare	5	12
143	Tech Frontier	Singapore	Technology	12	10
144	Elder Tech Hub	Sydney, Australia	Healthcare	9	8
145	SmartFashion Accelerator	Paris, France	Fashion	5	6
146	Energy Innovators	Berlin, Germany	Energy	10	9
147	Pet Tech Incubator	Austin, TX	Pets	8	10
148	Youth Empowerment Hub	Mumbai, India	Social Impact	20	12
149	Artificial Intelligence Accelerator	Los Angeles, CA	AI	15	10
150	FinTech 101	New York, NY	Finance	12	8

Add Update Delete Load

Load Accelerator

Accelerator Management

ID:	Location:	Name:			
TechStarters	CA	Salt Francisco			
Industry Focus :		Batch Size:			
10		12			
ID	Name	Location	Industry Focus	Batch Size	Program Duration
1	TechStarters	San Francisco	CA	10	12
102	Health Innovators	New York, NY	Healthcare	8	5
103	Eco Boost	Austin, TX	Sustainability	15	9
104	AgriTech Hub	Chicago, IL	Agriculture	12	8
105	FinTech Fastlane	Boston, MA	Finance	10	12
106	EduAccelerate	Seattle, WA	Education	20	10
107	Smart Cities Accelerator	Los Angeles, CA	Urban Development	5	5
108	Media Mavericks	Los Angeles, CA	Media	7	12
109	Global Health Ventures	Toronto, Canada	Healthcare	6	8
110	CyberSecure	Washington, D.C.	Cybersecurity	9	5
111	H Innovations	Silicon Valley, CA	Artificial Intelligence	11	10
112	FoodTech Lab	San Diego, CA	Food Technology	9	8
113	IoT Incubator	New York, NY	Internet of Things	10	10
114	Space Startups	Houston, TX	Aerospace	4	12
115	E-commerce Launchpad	Miami, FL	E-commerce	12	8
116	Creative Labs	Los Angeles, CA	Media & Entertainment	15	9
117	Blockchain Foundry	Singapore	Blockchain	5	5
118	VR Incubator	San Francisco, CA	Virtual Reality	7	10
119	Digital Health Accelerator	Berlin, Germany	Healthcare	6	9
120	Impact Innovators	London, UK	Social Impact	10	12
121	TravelTech Hub	Barcelona, Spain	Travel & Tourism	8	10
122	Fintech Factory	Dublin, Ireland	Finance	9	8
123	CleanTech Accelerator	Vancouver, Canada	Clean Technology	12	9
124	Startup Garage	Austin, TX	Various	15	10
125	Fashion Forward	Paris, France	Fashion	10	5
126	Tech Quest	Tokyo, Japan	Technology	11	12
127	EduLab	Melbourne, Australia	Education	20	10
128	Wearable Tech Incubator	Los Angeles, CA	Wearables	6	8
129	Sports Tech Accelerator	New York, NY	Sports	8	10
130	Smart Mobility Hub	San Francisco, CA	Transportation	7	9
131	Robotics Lab	Bangalore, India	Robotics	10	5
132	Telecom Innovators	Chicago, IL	Telecommunications	12	12
133	Gaming Accelerator	Seattle, WA	Gaming	10	8
134	Sustainable Living Accelerator	Toronto, Canada	Sustainability	9	10

Add Update Delete Load

Mentor Table**GUI**

Mentor Management

ID:				
Name:				
Expertise:				
Affiliation:				
LinkedIn Profile :				
ID	Name	Expertise	Affiliation	LinkedIn Profile

Add Update Delete Load

Add Mentor

Mentor Management

ID:	Andrew Product Development Professional https://linkedin.com/in/andrewmentor			
Name:				
Expertise:				
Affiliation:				
LinkedIn Profile :				
ID	Name	Expertise	Affiliation	LinkedIn Profile
1	Andrew	Product Development	Professional	https://linkedin.com/in/andrewmentor

[Add](#) [Update](#) [Delete](#) [Load](#)

Update Mentor

Mentor Management

ID:	515 Jloni Public Expertise Academic https://linkedin.com/in/jlonimentor			
Name:				
Expertise:				
Affiliation:				
LinkedIn Profile :				
ID	Name	Expertise	Affiliation	LinkedIn Profile
1	Andrew	Product Development	Professional	https://linkedin.com/in/andrewmentor
501	Brittany	Finance	Academic	https://linkedin.com/in/brittanymentor
503	Denise	Marketing Strategy	Professional	https://linkedin.com/in/denisenmentor
504	Ethan	Business Development	Partnership	https://linkedin.com/in/ethanmentor
505	Fiona	Human Resources	Academic	https://linkedin.com/in/fionammentor
506	Gavin	Data Science	Professional	https://linkedin.com/in/gavimentor
507	Hannah	Legal Consulting	Organizational	https://linkedin.com/in/hannahmentor
508	Isaac	Cybersecurity	Professional	https://linkedin.com/in/isaacmentor
509	Julia	Software Engineering	Partnership	https://linkedin.com/in/juliammentor
510	Kyle	Sales Strategy	Academic	https://linkedin.com/in/kylementor
511	Lily	Cloud Computing	Professional	https://linkedin.com/in/lillymentor
512	Marcus	Artificial Intelligence	Organizational	https://linkedin.com/in/marcusmentor
513	Nora	Investment Strategy	Professional	https://linkedin.com/in/norammentor
514	Oscar	Supply Chain Management	Academic	https://linkedin.com/in/oscarmentor
515	Paula	Machine Learning	Partnership	https://linkedin.com/in/paulammentor
516	Quentin	Product Design	Organizational	https://linkedin.com/in/quentinmentor
517	Rachel	Corporate Strategy	Professional	https://linkedin.com/in/rachelmmentor
518	Steve	Healthcare	Academic	https://linkedin.com/in/stevementor
519	Tina	Renewable Energy	Partnership	https://linkedin.com/in/tinammentor
520	Umar	E-commerce	Organizational	https://linkedin.com/in/umarmentor
521	Violet	Marketing Research	Professional	https://linkedin.com/in/violetmentor
522	Wesley	User Experience	Academic	https://linkedin.com/in/wesleymentor
523	Xander	Blockchain	Partnership	https://linkedin.com/in/xandermentor
524	Yvonne	Real Estate	Organizational	https://linkedin.com/in/yvonnementor
525	Zachari	Ethics	Professional	https://linkedin.com/in/zacharimentor
526	Alicea	Telecommunications	Academic	https://linkedin.com/in/aliceammentor
527	Ben	Retail Strategy	Organizational	https://linkedin.com/in/benmentor
528	Cathy	Content Strategy	Professional	https://linkedin.com/in/cathymentor
529	Darius	Biotechnology	Academic	https://linkedin.com/in/dariusmentor
530	Elena	Fashion	Partnership	https://linkedin.com/in/elenamentor
531	Fred	Public Relations	Organizational	https://linkedin.com/in/fredmentor
532	Gretchen	Venture Capital	Professional	https://linkedin.com/in/gretchenmentor
533	Henry	Legal Affairs	Academic	https://linkedin.com/in/henrymentor
534	Isabel	Accounting	Partnership	https://linkedin.com/in/isabellmentor

[Add](#) [Update](#) [Delete](#) [Load](#)

Delete Mentor

Mentor Management					
ID:	Name:	Expertise:	Affiliation:	LinkedIn Profile:	
501	Andrew	Product Development	Professional	https://linkedin.com/in/andrewmentor	
502	Brittany	Finance	Academic	https://linkedin.com/in/brittanymentor	
503	Denise	Marketing Strategy	Professional	https://linkedin.com/in/denisenmentor	
504	Ethan	Business Development	Partnership	https://linkedin.com/in/ethanmentor	
505	Fiona	Human Resources	Academic	https://linkedin.com/in/fionamonitor	
506	Gavin	Data Science	Professional	https://linkedin.com/in/gavimentor	
507	Hannah	Legal Consulting	Organizational	https://linkedin.com/in/hannahmentor	
508	Isaac	Cybersecurity	Professional	https://linkedin.com/in/isaacmentor	
509	Julia	Software Engineering	Partnership	https://linkedin.com/in/juliamentor	
510	Kyle	Sales Strategy	Academic	https://linkedin.com/in/kylementor	
511	Lily	Cloud Computing	Professional	https://linkedin.com/in/lilymentor	
512	Marcus	Artificial Intelligence	Organizational	https://linkedin.com/in/marcusmentor	
513	Nora	Investment Strategy	Professional	https://linkedin.com/in/noramonitor	
514	Oscar	Supply Chain Management	Academic	https://linkedin.com/in/oscarmentor	
515	Fiona	Machine Learning	Partnership	https://linkedin.com/in/fionamonitor	
516	Quentin	Product Design	Organizational	https://linkedin.com/in/quentinmentor	
517	Rachel	Corporate Strategy	Professional	https://linkedin.com/in/rachelmentor	
518	Steve	Healthcare	Academic	https://linkedin.com/in/stevementor	
519	Tina	Renewable Energy	Partnership	https://linkedin.com/in/tinamonitor	
520	Umar	E-commerce	Organizational	https://linkedin.com/in/umarmonitor	
521	Miles	Marketing Research	Professional	https://linkedin.com/in/milesmentor	
522	Wesley	User Experience	Academic	https://linkedin.com/in/weseymentor	
523	Xander	Blockchain	Partnership	https://linkedin.com/in/xandermentor	
524	Yvonne	Real Estate	Organizational	https://linkedin.com/in/yvonnementor	
525	Zachary	Ethics	Professional	https://linkedin.com/in/zacharymentor	
526	Alicea	Telecommunications	Academic	https://linkedin.com/in/aliceamentor	
527	Ben	Retail Strategy	Organizational	https://linkedin.com/in/benmentor	
528	Cathy	Content Strategy	Professional	https://linkedin.com/in/cathymentor	
529	Darius	Biotechnology	Academic	https://linkedin.com/in/dariusmentor	
530	Elena	Fashion	Partnership	https://linkedin.com/in/elenamentor	
531	Fred	Public Relations	Organizational	https://linkedin.com/in/fredmentor	
532	Gretchen	Venture Capital	Professional	https://linkedin.com/in/gretchenmentor	
533	Henry	Legal Affairs	Academic	https://linkedin.com/in/henrymentor	
534	Isabel	Accounting	Partnership	https://linkedin.com/in/isabelmentor	

Add Update Delete Load

Load Mentor

Mentor Management					
ID:	Name:	Expertise:	Affiliation:	LinkedIn Profile:	
501	Andrew	Product Development	Professional	https://linkedin.com/in/andrewmentor	
502	Brittany	Finance	Academic	https://linkedin.com/in/brittanymentor	
503	Denise	Marketing Strategy	Professional	https://linkedin.com/in/denisenmentor	
504	Ethan	Business Development	Partnership	https://linkedin.com/in/ethanmentor	
505	Fiona	Human Resources	Academic	https://linkedin.com/in/fionamonitor	
506	Gavin	Data Science	Professional	https://linkedin.com/in/gavimentor	
507	Hannah	Legal Consulting	Organizational	https://linkedin.com/in/hannahmentor	
508	Isaac	Cybersecurity	Professional	https://linkedin.com/in/isaacmentor	
509	Julia	Software Engineering	Partnership	https://linkedin.com/in/juliamentor	
510	Kyle	Sales Strategy	Academic	https://linkedin.com/in/kylementor	
511	Lily	Cloud Computing	Professional	https://linkedin.com/in/lilymentor	
512	Marcus	Artificial Intelligence	Organizational	https://linkedin.com/in/marcusmentor	
513	Nora	Investment Strategy	Professional	https://linkedin.com/in/noramonitor	
514	Oscar	Supply Chain Management	Academic	https://linkedin.com/in/oscarmentor	
515	Fiona	Machine Learning	Partnership	https://linkedin.com/in/fionamonitor	
516	Quentin	Product Design	Organizational	https://linkedin.com/in/quentinmentor	
517	Rachel	Corporate Strategy	Professional	https://linkedin.com/in/rachelmentor	
518	Steve	Healthcare	Academic	https://linkedin.com/in/stevementor	
519	Tina	Renewable Energy	Partnership	https://linkedin.com/in/tinamonitor	
520	Umar	E-commerce	Organizational	https://linkedin.com/in/umarmonitor	
521	Miles	Marketing Research	Professional	https://linkedin.com/in/milesmentor	
522	Wesley	User Experience	Academic	https://linkedin.com/in/weseymentor	
523	Xander	Blockchain	Partnership	https://linkedin.com/in/xandermentor	
524	Yvonne	Real Estate	Organizational	https://linkedin.com/in/yvonnementor	
525	Zachary	Ethics	Professional	https://linkedin.com/in/zacharymentor	
526	Alicea	Telecommunications	Academic	https://linkedin.com/in/aliceamentor	
527	Ben	Retail Strategy	Organizational	https://linkedin.com/in/benmentor	
528	Cathy	Content Strategy	Professional	https://linkedin.com/in/cathymentor	
529	Darius	Biotechnology	Academic	https://linkedin.com/in/dariusmentor	
530	Elena	Fashion	Partnership	https://linkedin.com/in/elenamentor	
531	Fred	Public Relations	Organizational	https://linkedin.com/in/fredmentor	
532	Gretchen	Venture Capital	Professional	https://linkedin.com/in/gretchenmentor	
533	Henry	Legal Affairs	Academic	https://linkedin.com/in/henrymentor	
534	Isabel	Accounting	Partnership	https://linkedin.com/in/isabelmentor	

Add Update Delete Load

Chapter 6: Technical Issues and Solution

1. Technical Issues

List all technical challenges encountered during project development.

Issue 1: ERD Constraints and Relationship Mapping

- **Description:** While using pgAdmin's GUI interface, defining certain relationships and constraints within the ERD was occasionally challenging, as the interface had limitations in representing complex many-to-many relationships directly.

Issue 2: Data Export for Backup

- **Description:** The pgAdmin GUI didn't offer straightforward options to export all database objects and data in a single operation, making it challenging to generate a consistent backup in a single .sql file.

2. Solution

○ **Describe the approach taken to resolve each issue.**

- - **Approach:** We worked around these limitations by defining complex relationships and constraints directly through SQL statements rather than relying solely on the GUI. This ensured that our ERD accurately represented the intended structure, including all necessary primary keys, foreign keys, and constraints.
 - **Solution for Issue 2 (Data Export for Backup)**
 - **Approach:** To address the export limitation, we utilized pgAdmin's "Backup" and "Restore" functions, exporting our database as a .backup file for PostgreSQL. This allowed for a straightforward and consistent export of our database objects and data, which could be restored as needed to prevent any data loss.