**Industry Project**

**Report**

**On**

**VERACITIZ SOLUTION**

|  |  |
| --- | --- |
| Developed By: - | Guided By:- |
| Aryan Mayurkumar Modi (20162121011) | Prof. Dharmesh Darji (Internal) |

**Submitted to**

**Faculty of Engineering and Technology**

**Institute of Computer Technology**

**Ganpat University**

****

**Year - 2024**

****

**CERTIFICATE**

This is to certify that the Summer Internship report submitted along with the Weekly basis task on Data Analysis has been carried out by Aryan Modi **at VERACITIZ SOLUTION** in partial fulfillment for the degree of Bachelor of Computer Science Engineering in BDA, 8th semester of Institute of Computer Technology, Ganpat University during the academic year 2023-24. The results/findings contained in this report have not been submitted in part or full to any other University / Institute for award of any other Degree/Diploma.

Prof. Dharmesh Darji Prof. Dharmesh Darji

Internal Guide Head , CSE Department

# ACKNOWLEDGEMENT

Summer Internship project is a golden opportunity for learning and self-development. I consider myself very lucky and honored to have so many wonderful people lead me through in completion of this project. First and foremost, I would like to thank Dr. Rohit Patel, Principal, ICT, and Prof. Dharmesh Darji, Head, ICT who gave us an opportunity to undertake this project. My grateful thanks to Prof. Dharmesh Darji & Mr. Jacky Patel for their guidance in Data Analysis, who despite being extraordinarily busy with academics, took time out to hear, guide and keep us on the correct path. We do not know where would have been without his/her help. CSE department monitored our progress and arranged all facilities to make life easier. We choose this moment to acknowledge their contribution gratefully.

**Aryan Modi (Enrollment No : 20162121011)**

# TABLE OF CONTENTS

|  |  |  |  |
| --- | --- | --- | --- |
| **Content** | | | **Page**  **Number** |
|  | Title Page | |  |
|  | College Certificate | | I |
|  | Acknowledgement | | II |
|  | Table of Contents | | III |
|  |  | |  |
| **1.** | **Overview of the Company** | | 1 |
|  | 1.1 | History | 1 |
|  | 1.2 | Different product / scope of work | 1 |
|  |  |  |  |
| **2** | **Week-1 Progress** | | 2 |
|  | 2.1 | Study about ETL | 2 |
|  | 2.2 | Extract(E) | 2 |
|  | 2.3 | Transform(T) | 3 |
|  | 2.4 | Load(L) | 3 |
|  | 2.5 | Screenshot/Code Snippet | 3 |
|  |  |  |  |
| **3** | **Week-2 Progress** | | 6 |
|  | 3.1 | SQL Server Management Studio | 6 |
|  | 3.2 | Query editor | 6 |
|  | 3.3 | Import and Export data | 6 |
|  | 3.4 | Database diagram | 6 |
|  | 3.5 | Screenshot/Code Snippet | 6 |
|  |  |  |  |
| **4** | **Week-3 Progress** | | 11 |
|  | 4.1 | SQL Server | 11 |
|  | 4.2 | Key component | 11 |
|  | 4.3 | Key concepts | 12 |
|  | 4.4 | Screenshot/Code Snippet | 13 |
|  |  | |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **5** | **Week-4 Progress** | | 17 |
|  | 5.1 | Lead and Lag Functions | 17 |
|  | 5.2 | Slowly Changing Dimension | 17 |
|  | 5.3 | Stored Procedure | 17 |
|  | 5.4 | Screenshot/Code Snippet | 18 |
| **6** | **Week-5 Progress** | | 22 |
|  | 6.1 | HackerRank Tasks | 22 |
|  | 6.2 | Screenshot/Code Snippet working on HackerRank | 22 |
|  |  |  |  |
| **7** | **Week-6 Progress** | | 25 |
|  | 7.1 | Improve Logic Building Task | 25 |
|  | 7.2 | Screenshot/Code Snippet | 26 |
|  |  |  |  |
| **8** | **Week-7 Progress** | | 29 |
|  | 8.1 | Improve Logic Building Task | 29 |
|  | 8.2 | Screenshot/Code Snippet | 30 |
|  |  |  |  |
| **9** | **Live Project** | | 35 |
|  | 9.1 | Loop SQL | 35 |
|  | 9.2 | JavaScript | 35 |
|  | 9.3 | Node JS | 35 |
|  | 9.4 | FIREBASE | 35 |
|  | 9.5 | Ngrok | 36 |
|  | 9.6 | Cognos Custom Control | 36 |
|  | 9.7 | Cognos | 37 |
|  | 9.8 | Screenshot/Code Snippet of project | 38 |
|  |  |  |  |
| **10** | **REFERENCES** | | 43 |
|  |  |  |  |

**CHAPTER-1**

## OVERVIEW OF THE COMPANY

### History

* Veracitiz Solutions is a performance management consulting firm and an IBM-exclusive Premier Business Partner with a long history of experience in developing and implementing sound planning and budgeting solutions. The company has been praised for its in-depth knowledge of the Cognos Enterprise Planning software suite and its ability to understand unique business models and industry-specific requirements. Veracitiz has a strong focus on harnessing emerging technologies for impactful business results and has a track record of accomplishing preset objectives on time and on budget. The company is also recognized as an IBM Gold Business Partner and has a presence in various industries, including retail. While specific details about the history of Veracitiz Solutions are not readily available in the provided search results, the company's expertise and reputation in the field of performance management and its long-standing partnership with IBM demonstrate its established presence in the industry. For more detailed information about the history of Veracitiz Solutions, it may be beneficial to directly contact the company or refer to official press releases and publications.

### Different product / scope of work

* IBM Planning Analytics : IBM Planning Analytics is an integrated planning solution that uses AI to automate planning, budgeting, and forecasting and drive more intelligent workflows.
* IBM Cognos Analytics : Cognos Analytics is an AI-fueled business intelligence platform that helps to visualize, analyze and share actionable insights about your data with anyone in your organization.
* IBM SPSS Statistics : IBM® SPSS® Statistics solves a wide range of business and research problems by providing rich statistical capabilities, ensuring high accuracy to drive quality decision-making
* IBM Decision Optimization : IBM ILOG CPLEX Optimization Studio is a data science solution that combines mathematical and AI techniques to help make complex decisions involving multiple decision-variables, constraints & trade-offs
* IBM Instana : With IBM® Observability by Instana, users can combine APM with automation capabilities and distributed tracing to deploy on premise or as a SaaS solution.
* IBM SevOne : IBM SevOne NPM solution helps you spot, address, and prevent network performance issues early with machine learning-powered analytics from a single source.

# CHAPTER-2

## WEEK-1 PROGRESS

### Study about ETL

ETL (Extract, Transform, Load) is a data integration process that combines data from multiple sources into a single, consistent data store, such as a data warehouse or data lake. It involves three key steps: extraction, transformation, and loading. During extraction, raw data is copied from source locations to a staging area. In the transformation step, the data is cleansed and organized to address specific business intelligence needs. Finally, the transformed data is loaded into the target system. ETL is essential for data analytics, machine learning, and business intelligence, and it has been used by organizations for decades. While ETL is a time-consuming batch operation, it significantly improves data quality and is particularly suitable for building smaller data repositories.

ETL is essential for data analytics, machine learning, and business intelligence, and it has been used by organizations for decades. While ETL is a time-consuming batch operation, it significantly improves data quality and is particularly suitable for building smaller data repositories. Modern ETL solutions must cope with the accelerating volume and speed of data, as well as the ability to ingest, enrich, and manage transactions from any source, whether on-premises or in the cloud. ETL is commonly used to collect and prepare marketing data, integrate IoT data, and support data warehousing and data lake initiatives. If you are interested in learning more about ETL, there are various online courses available from top universities and industry leaders, covering topics such as data management, data warehousing, data analysis, and data visualization

.

### Extract (E)

* In this phase, data is collected and extracted from various source systems, which can include databases, flat files, applications, APIs, or other data repositories.
* The goal is to gather raw data from these diverse sources, capturing relevant information for analysis.

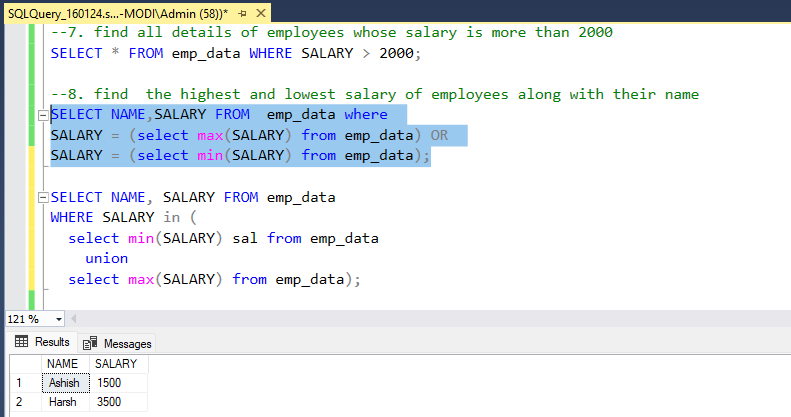
### Transform (T)

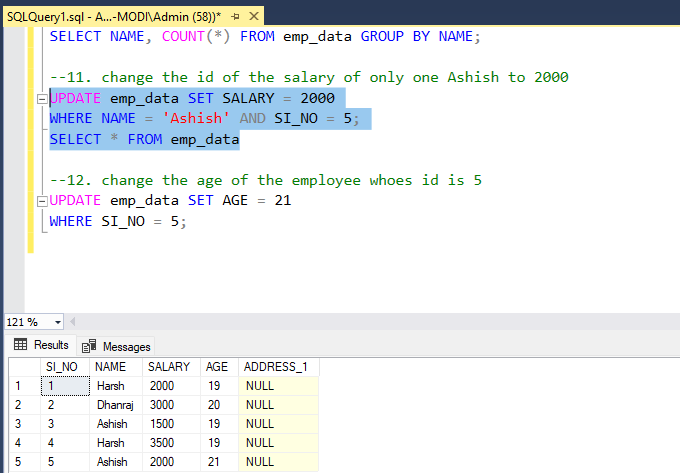
* Once data is extracted, it often needs to be transformed into a suitable format for analysis and storage. This involves cleaning, validating, and restructuring the data.
* Transformation processes may include data cleansing (fixing errors or inconsistencies), data normalization (standardizing formats), data enrichment (adding additional information), and more.
* The transformation phase ensures that the data is accurate, consistent, and ready for analysis.

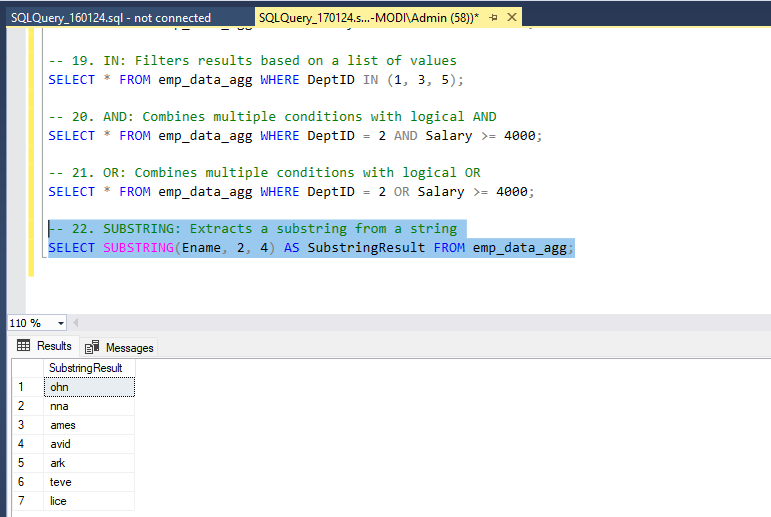
### Load (L)

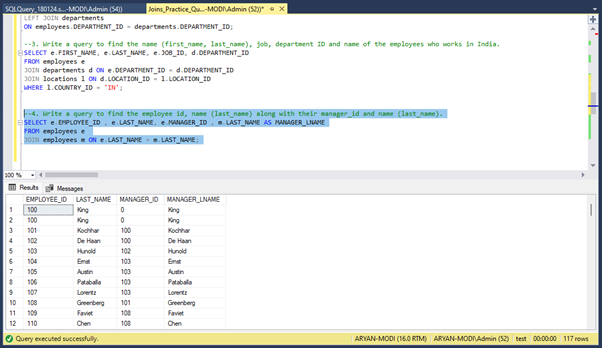
* After the data has been extracted and transformed, it is loaded into a target system, typically a data warehouse or a data mart.
* Loading involves putting the transformed data into a structured format within the target database, making it accessible for reporting and analysis.
* The load phase may involve batch processing, real-time streaming, or a combination of both, depending on the specific requirements of the data integration.
  1. **Screenshot/Code Snippet**

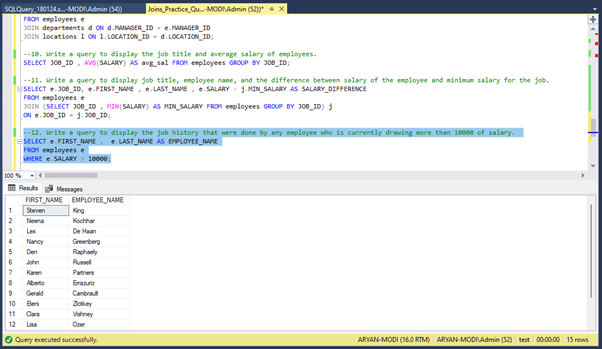
I have learn following concepts : brief introduction of ETL ,DATA ,Data Warehouse , SQL Concepts , ALTER , MODIFY , CHANGE , CONSTRAINT , RENAME , MIN-MAX , WHERE , AGGRIGATE functions and STRING functions , ORDER BY ,GROUP BY , JOINJS , DATE other task are mantion in the documents.











# CHAPTER-3

## WEEK-2 PROGRESS

### SQL Server Management Studio

It is a graphical user interface (GUI) tool developed by Microsoft for managing and interacting with Microsoft SQL Server. SQL Server is a relational database management system (RDBMS) that is widely used for storing and retrieving data.

### Query Editor

The Query Editor within SSMS allows users to write and execute Transact-SQL (T-SQL) queries against SQL Server databases. It provides syntax highlighting, code completion, and debugging capabilities.

### Import and Export Data

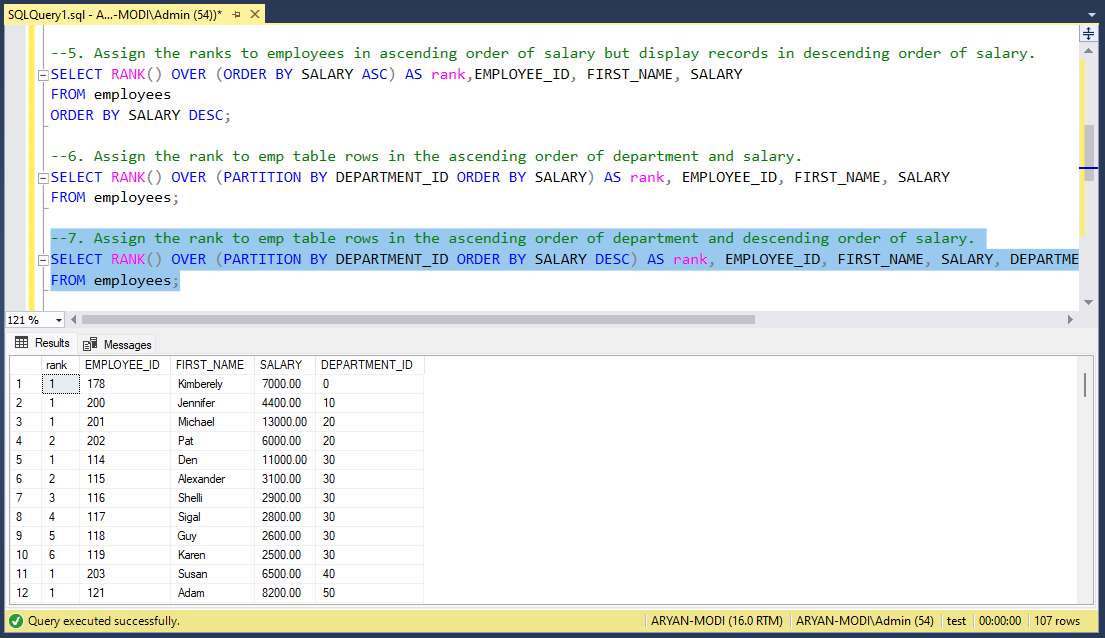
### SSMS provides wizards for importing and exporting data between SQL Server and other data sources. This is useful for tasks such as data migration and integration.

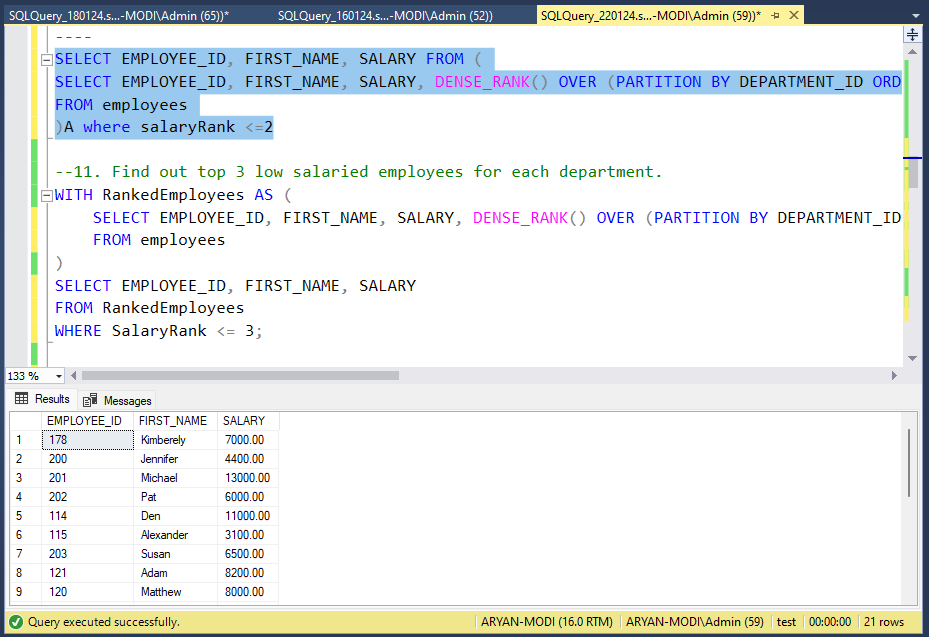
### Database Diagrams

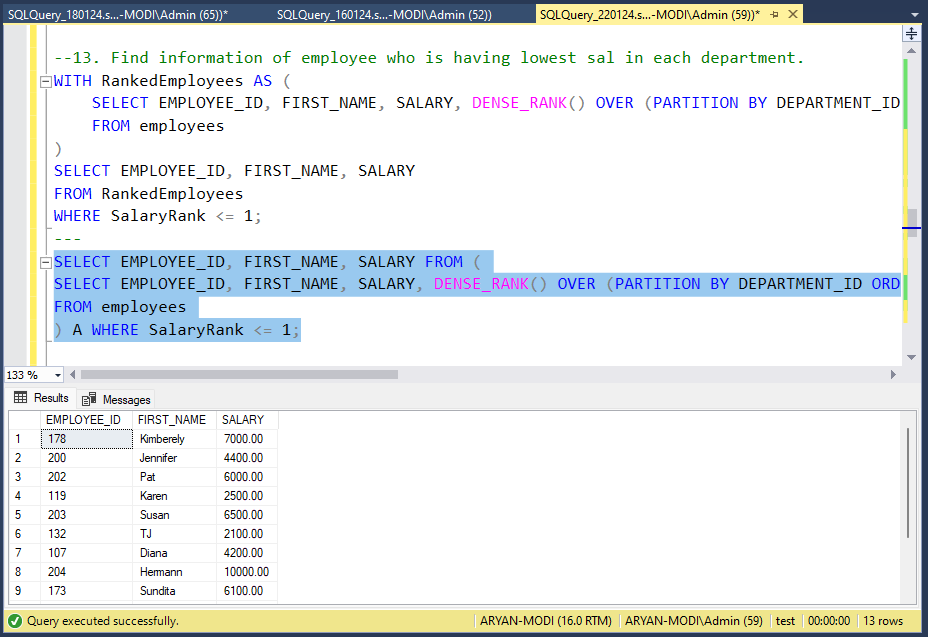
Users can create visual representations of database relationships using the Database Diagrams feature in SSMS.

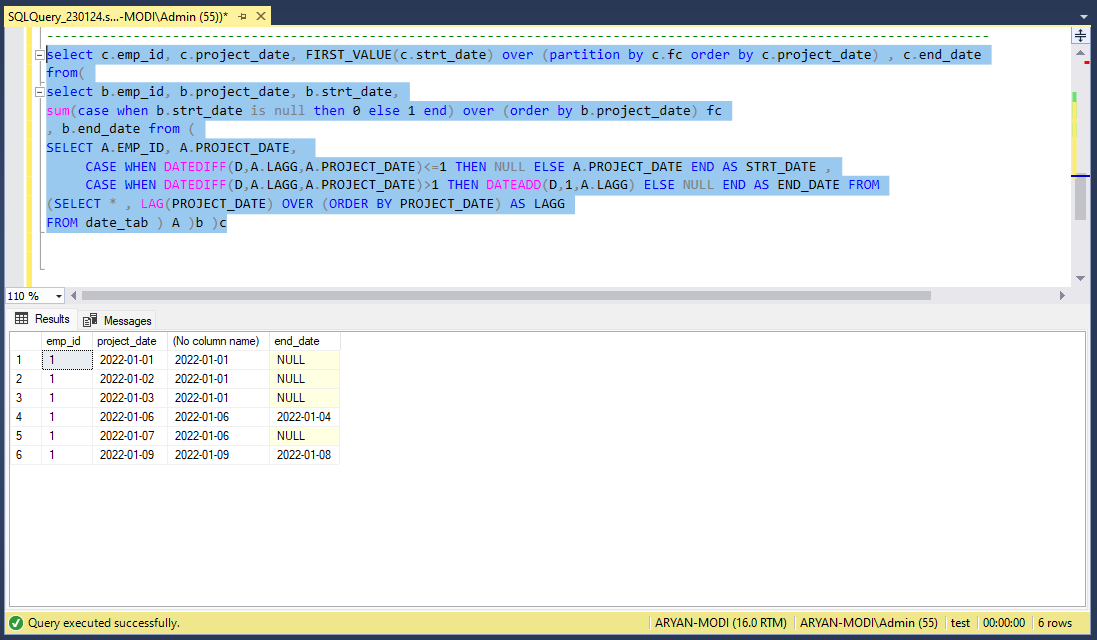
### Screenshot/Code Snippet

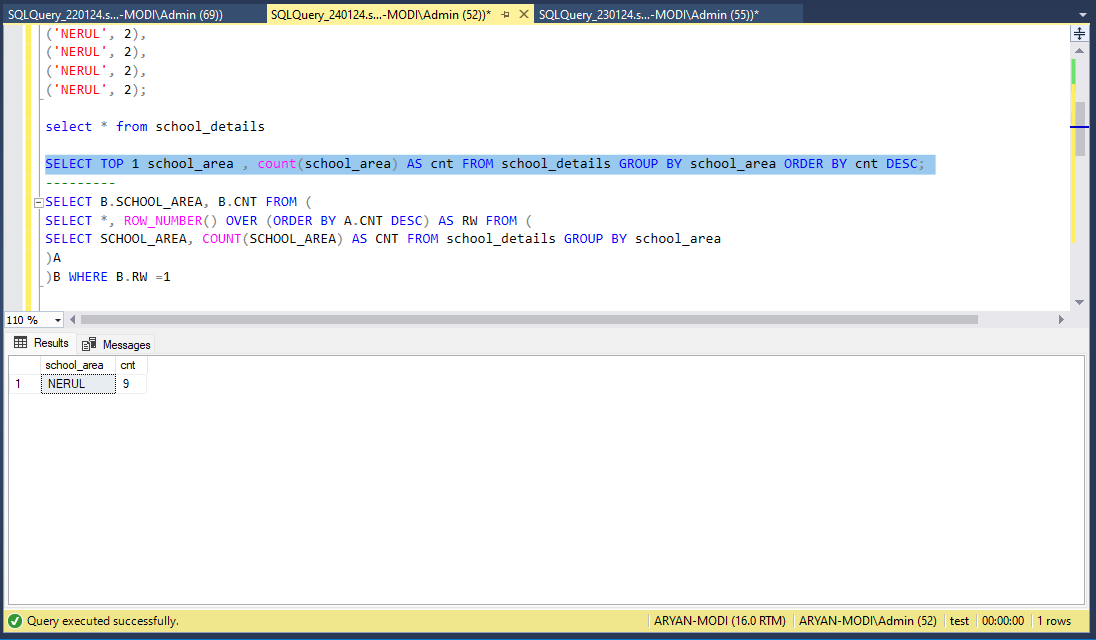
I have learn following concepts : RANK , DENSE RANK , Difference of both, Sub Quary, RANK and DENSE RANK with condition , LEAD & LAG FUNCTIONS , it's task ,STRING SPLIT functions , logic building tasks which are mentioned in Document.

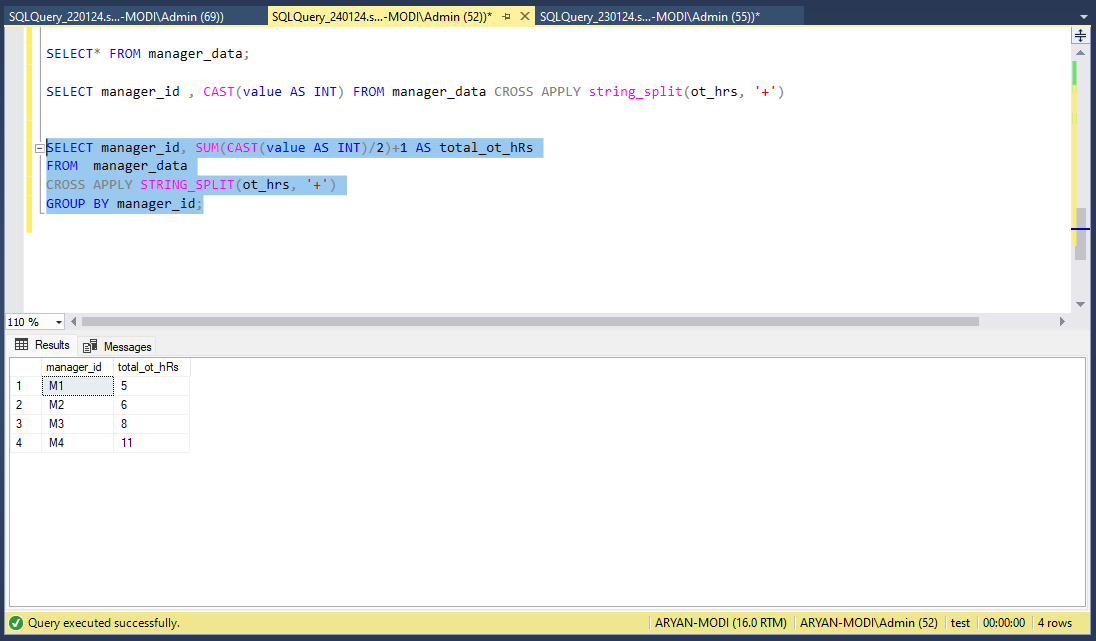
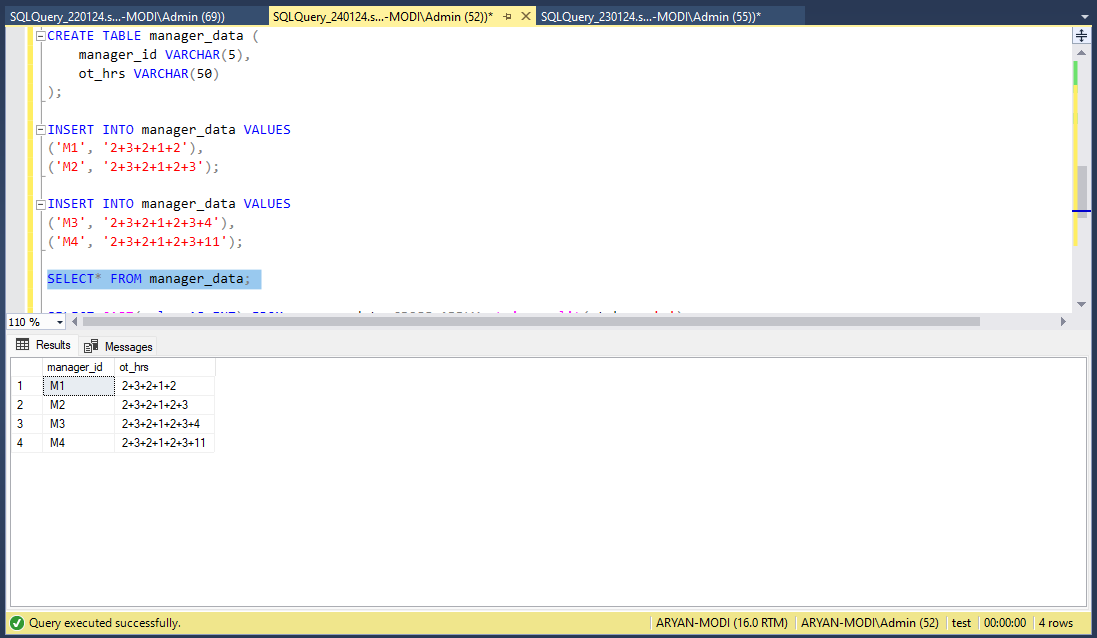












**CHAPTER-4**

**WEEK-3 PROGRESS**

### SQL SERVER

SQL Server is a relational database management system (RDBMS) developed by Microsoft. It is a software product that stores and retrieves data as requested by other software applications, running either on the same computer or on another computer across a network.

### Key Components

Database Engine: The core component that handles storage, processing, and security of data. It manages database objects such as tables, views, stored procedures, and triggers.

SQL Server Management Studio (SSMS): A graphical user interface (GUI) tool for managing and interacting with SQL Server. DBAs and developers use SSMS for tasks like writing queries, designing databases, and configuring server settings.

Transact-SQL (T-SQL): The SQL Server query language. It is an extension of SQL with additional programming constructs. T-SQL is used to interact with the SQL Server database engine.

Integration Services (SSIS): A tool for solving complex business problems by copying or downloading files, extracting and transforming data from different data sources, and loading data into one or more destinations.

Reporting Services (SSRS): A server-based reporting platform that provides a full range of ready-to-use tools and services to help you create, deploy, and manage reports for your organization.

Analysis Services (SSAS): Enables users to analyze and visualize data through online analytical processing (OLAP) and data mining.

### Key Concepts

Database: A container that holds a set of related tables, views, stored procedures, and other database objects. It's a way to organize and store data.

Table: A fundamental storage structure in SQL Server. It consists of rows and columns, where each column has a data type, and each row represents a record.

Query: A request for data or information from a database. SQL (Structured Query Language) is used to write queries to interact with SQL Server databases.

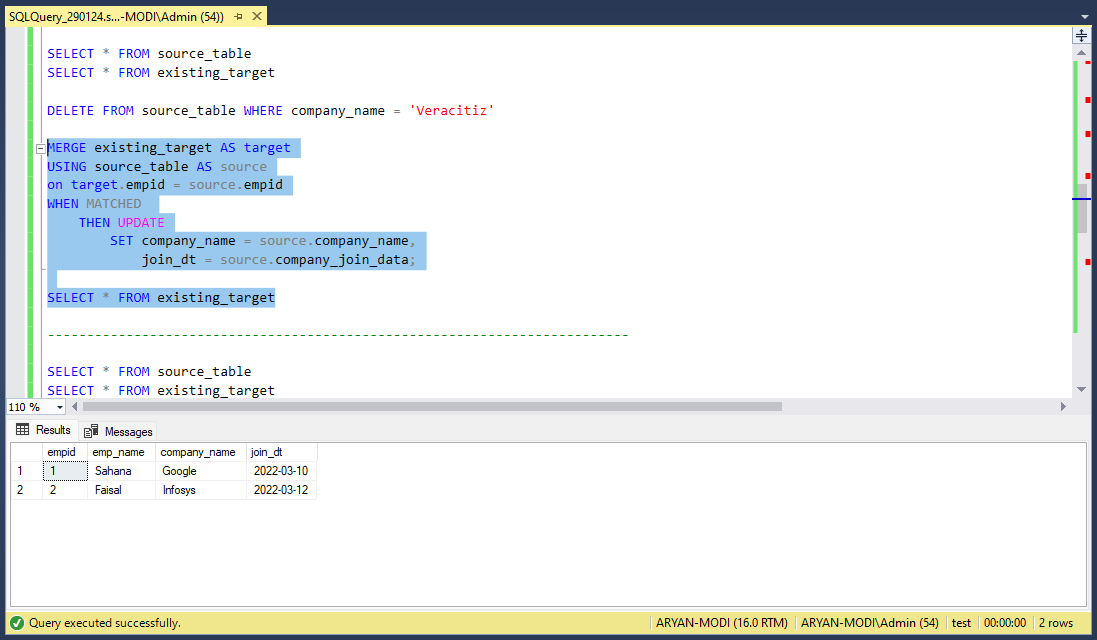
Primary Key: A unique identifier for a record in a table. It ensures data integrity by preventing duplicate or null values.

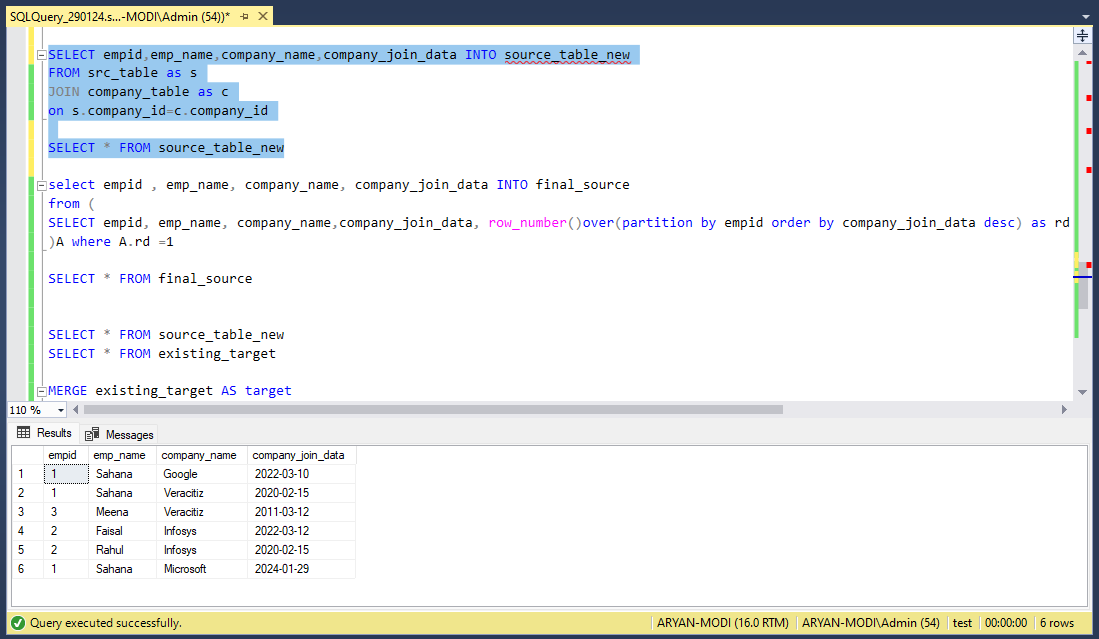
Foreign Key: A column or a set of columns in one table that refers to the primary key in another table. It establishes a link between the two tables.

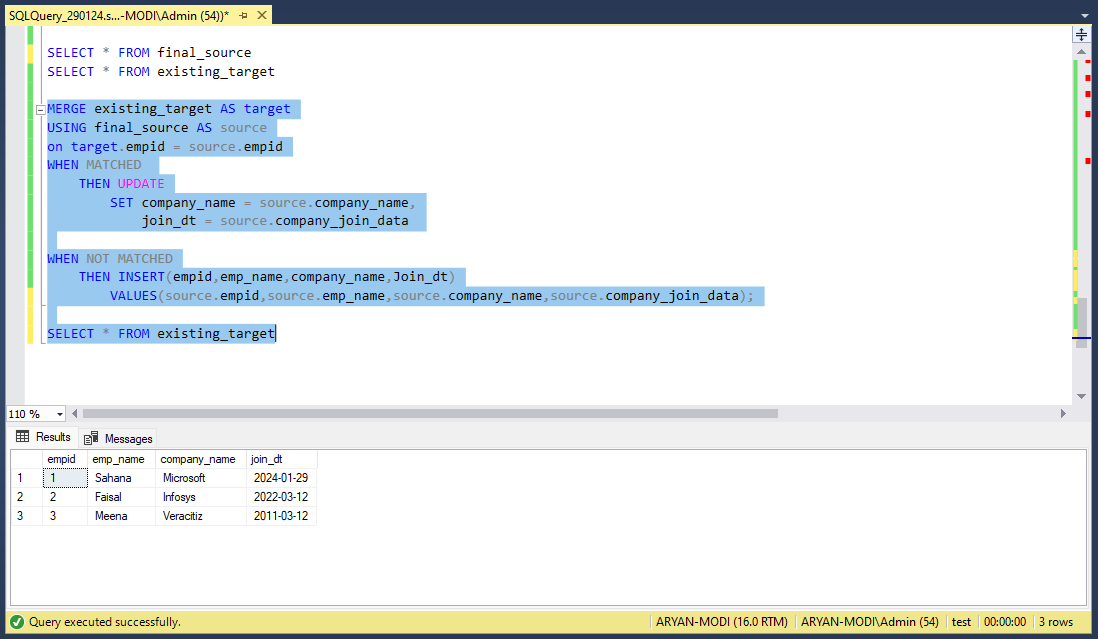
Index: A structure in a database that improves the speed of data retrieval operations on a database table.

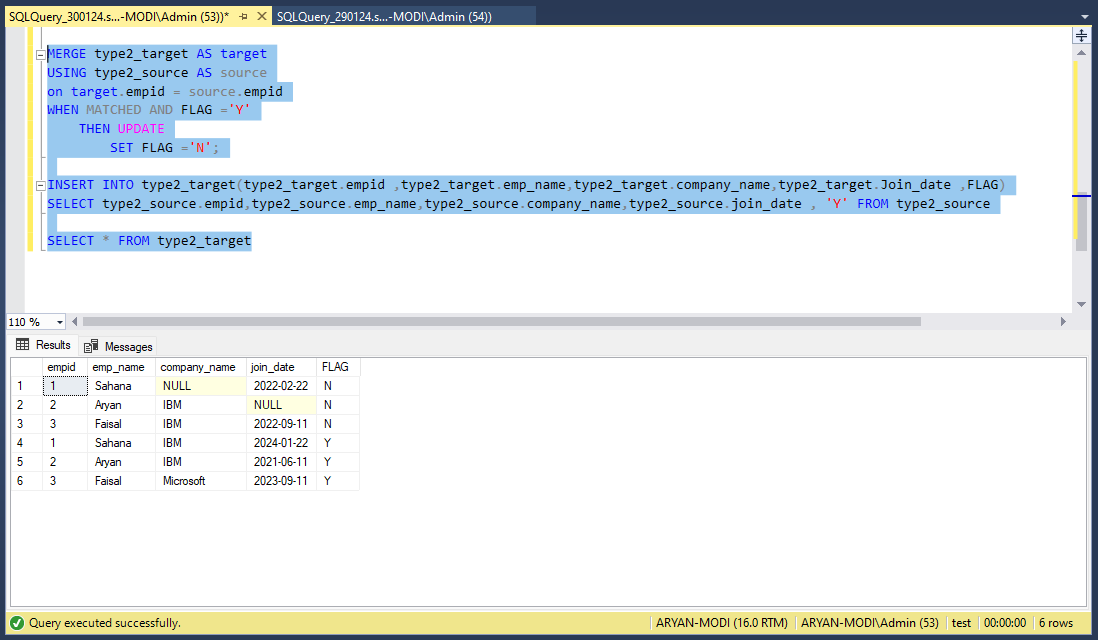
* 1. **Screenshot/Code Snippet**

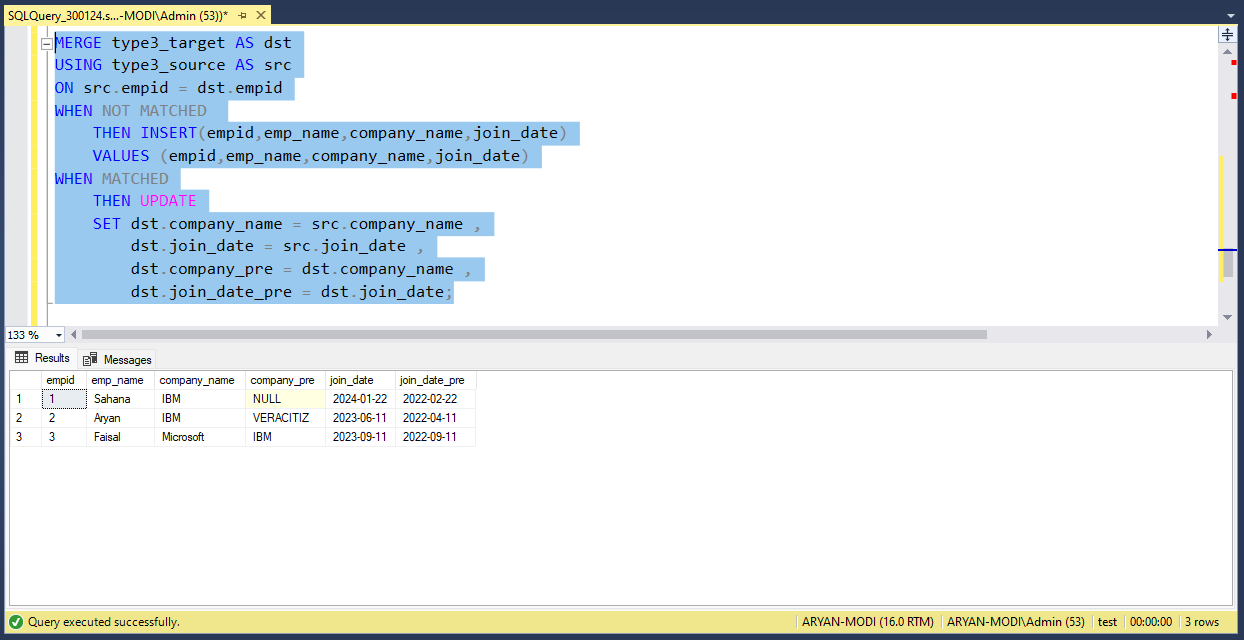
I have learn following concepts :SCD (Slowly Changing Dimension) and windows functions , Slowly Changing Dimension SCD type 1 examples with different scenarios , Slowly Changing Dimension SCD type 2 and SCD type 3 examples with different scenarios , DATABASE , DATA WAREHOUSE , STAR SCHEMA , SNOW FLAKE SCHEMA , GALAXY SCHEMA , DATA MART.

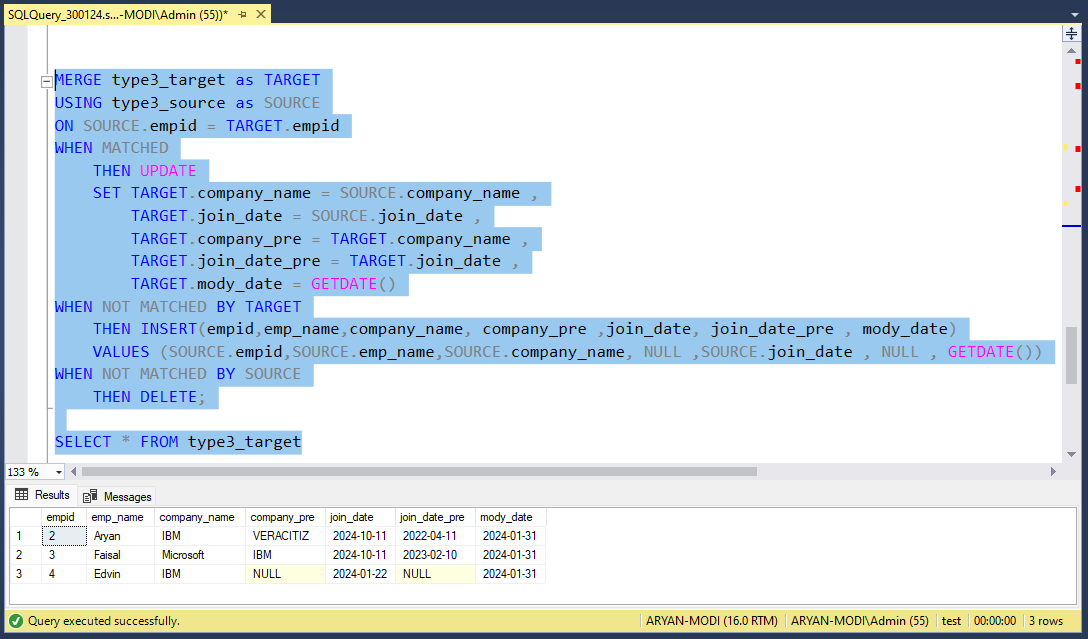


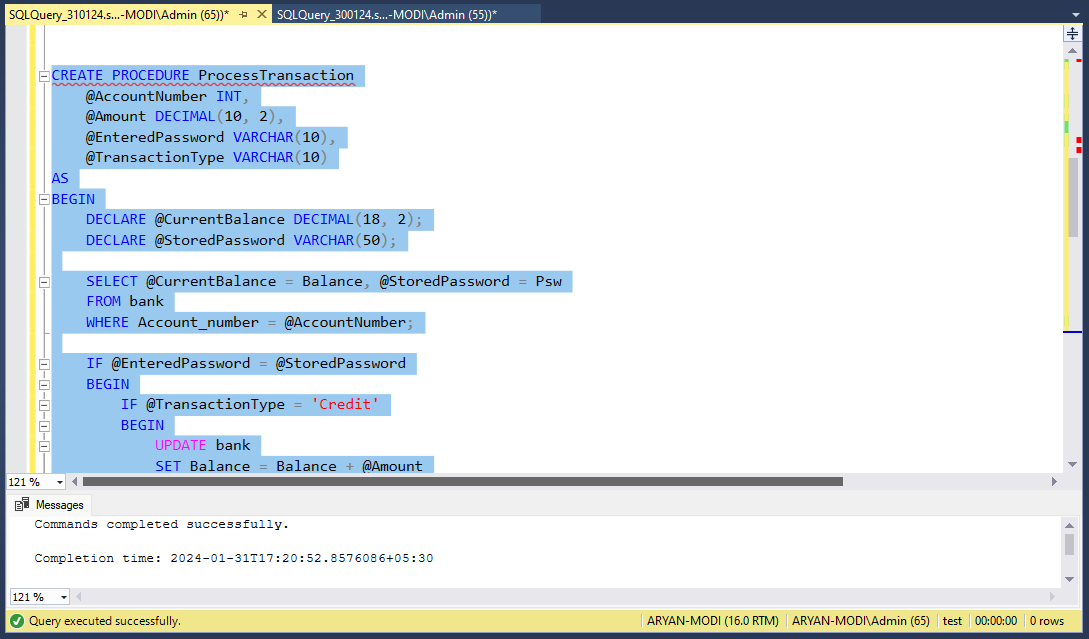


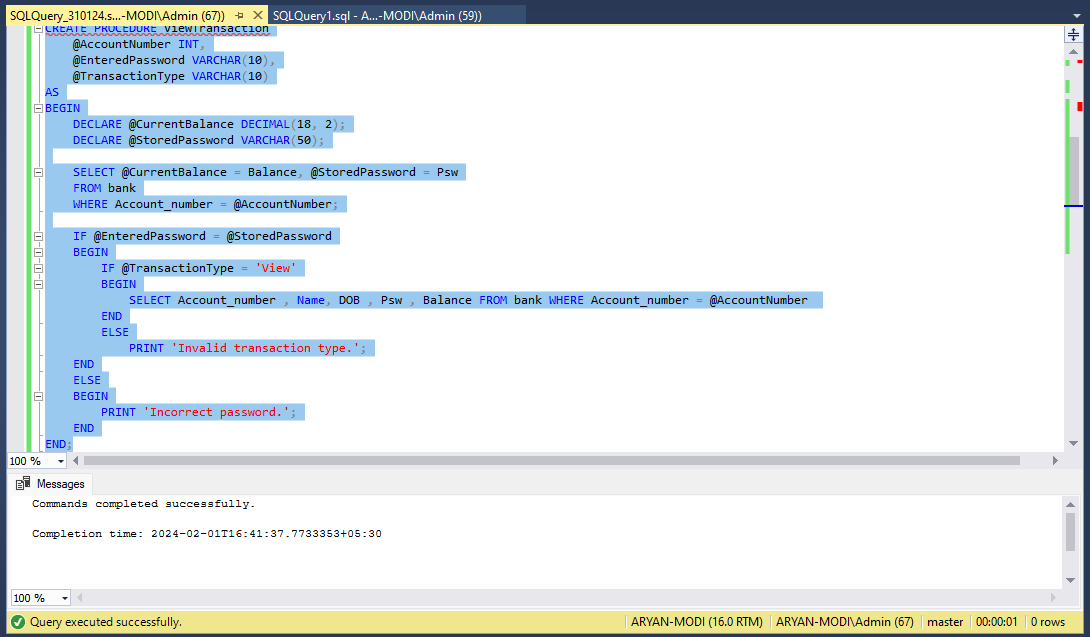












# CHAPTER-5

## WEEK-4 PROGRESS

### LEAD and LAG Functions

LEAD and LAG are window functions in SQL that allow you to access data from subsequent rows (LEAD) or preceding rows (LAG) within the result set.

SELECT

employee\_id,

salary,

LEAD(salary) OVER (ORDER BY employee\_id) AS next\_salary,

LAG(salary) OVER (ORDER BY employee\_id) AS prev\_salary

FROM

employee;

### Slowly Changing Dimensions (SCD)

SCD refers to the techniques used in data warehousing to manage changes in dimension attributes over time. There are three main types:

Type 1 (No History): Overwrite the existing data with the new values.

Type 2 (Add New Row): Create a new record with the updated information, maintaining historical records.

Type 3 (Add Columns): Add columns to the existing record to track changes.

### Stored Procedures

A stored procedure is a precompiled collection of one or more SQL statements that can be executed as a single unit. It is stored in the database and can be called from applications or other stored procedures.

CREATE PROCEDURE GetEmployeeDetails

@EmployeeID INT

AS

BEGIN

SELECT \*

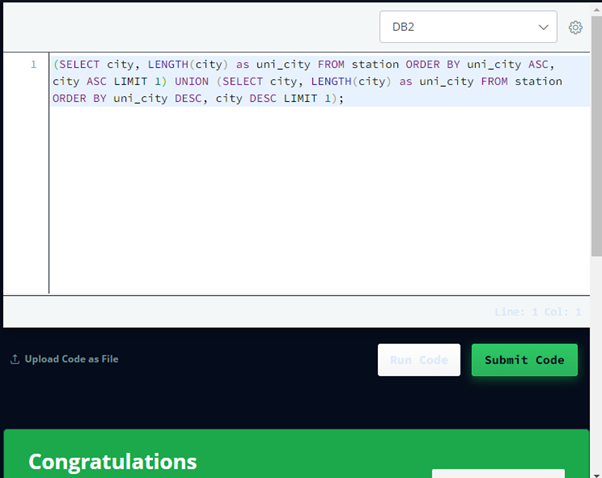
FROM Employees

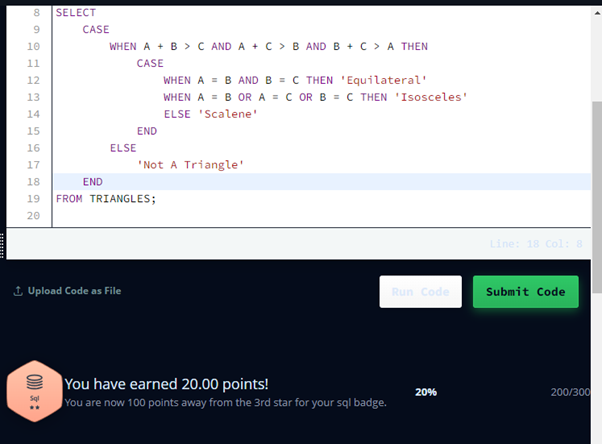
WHERE EmployeeID = @EmployeeID;

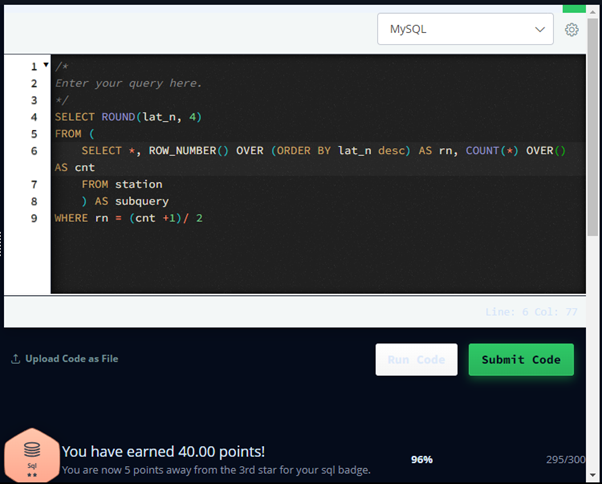
END

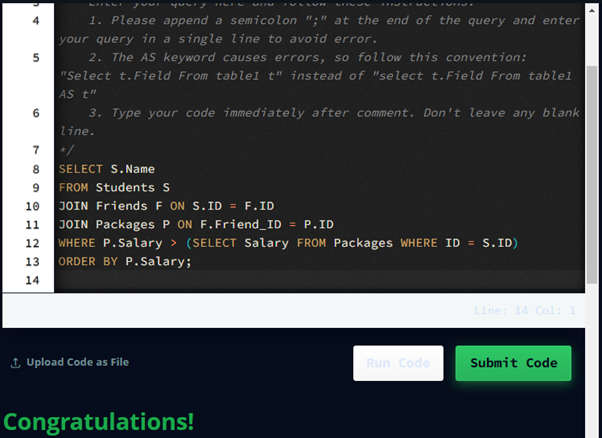
* 1. **Screenshot/Code Snippet working on HackerRank**

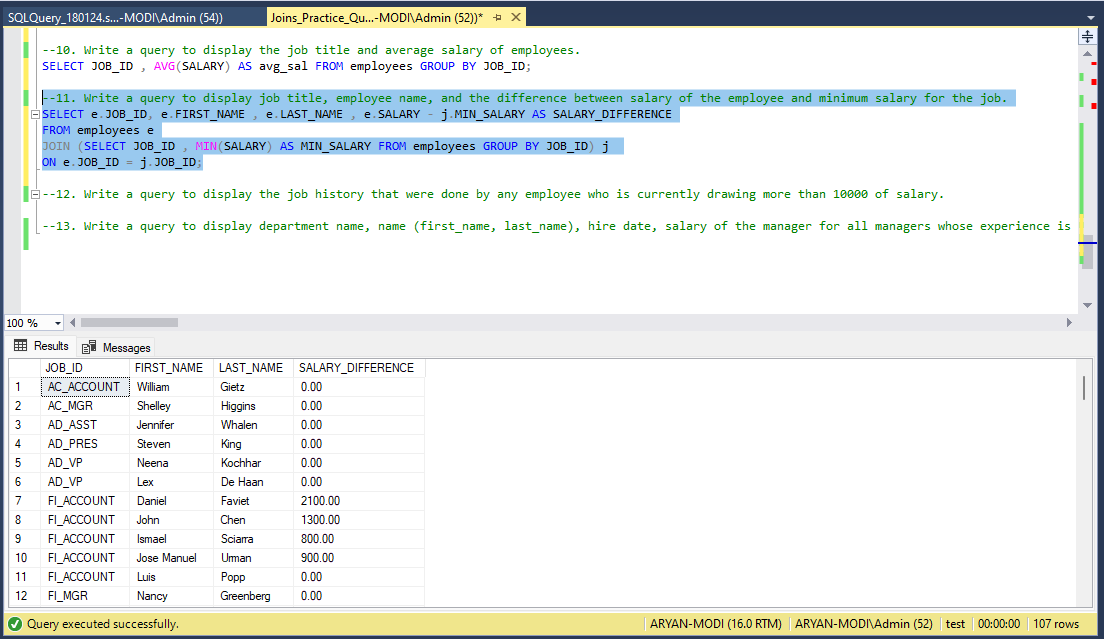
I have leant about SINGLE STORE installation process , how singlestore works , Shard Key , Data Skew , High Availability , type of Table and singlestore tools , SINGLE STORE installation process with cloud version and connect it with MySQL Workbanch , Indexing , Type of Indexing , Views , SPLIT and Store it in to new column Assignment and I have many Methods to implement dynamic way to store it , interval table with dates which are mentation in Document. In Extra , work on HackerRank problem solving tasks.

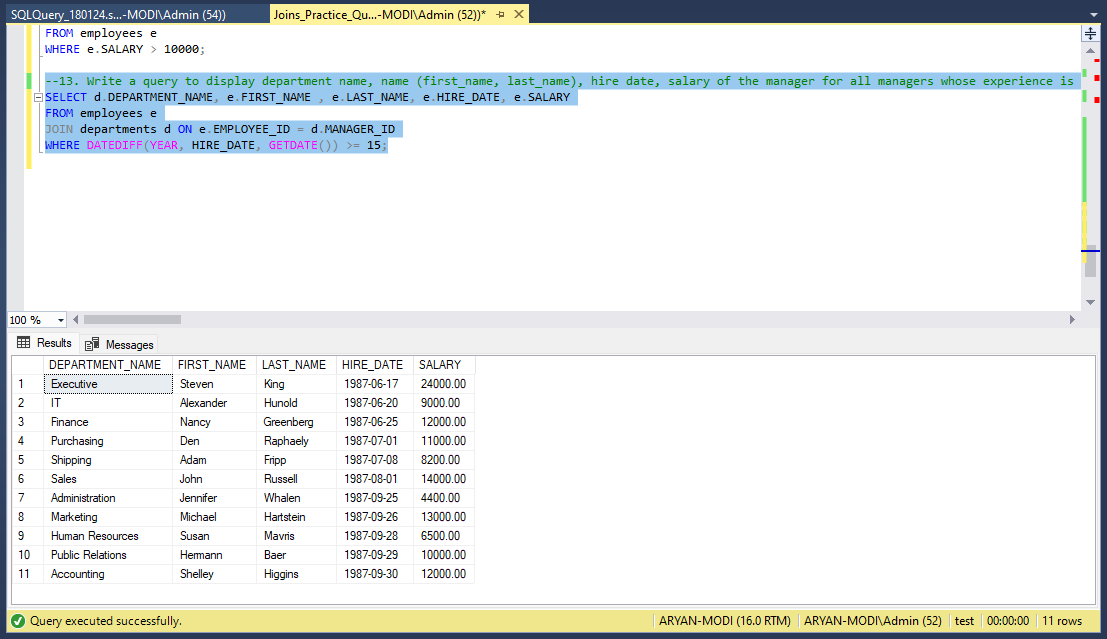












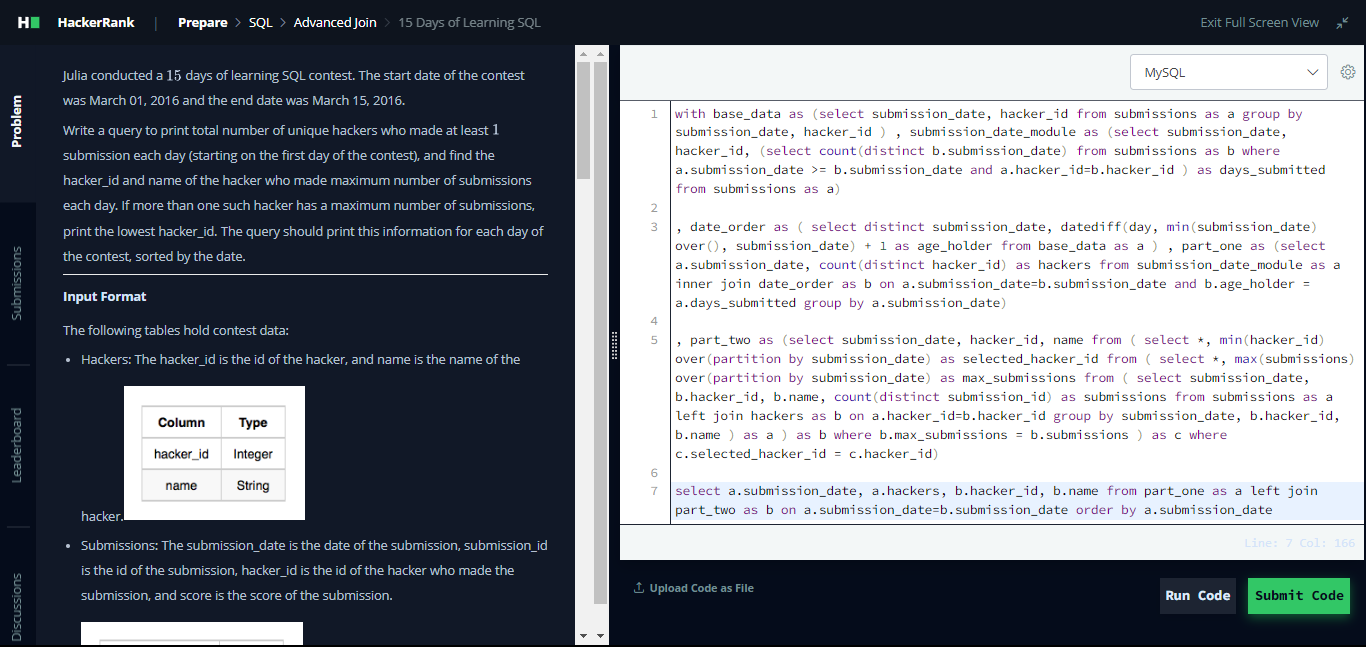
# CHAPTER-6

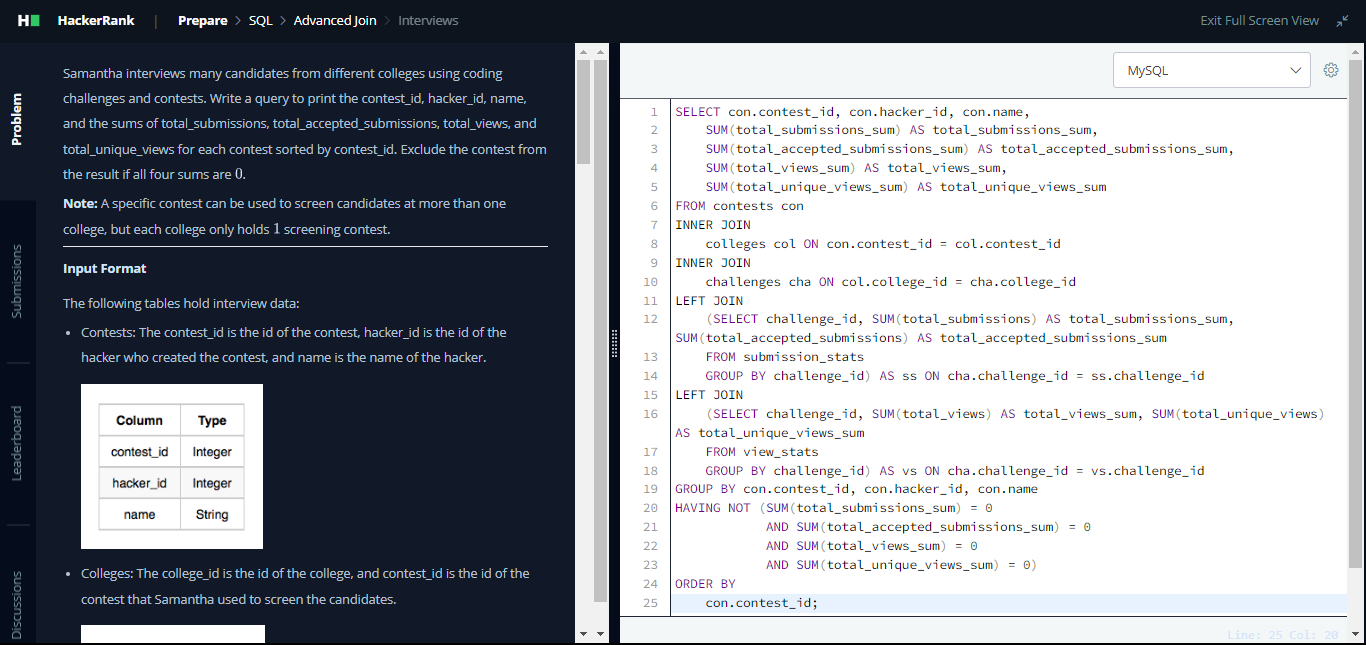
## WEEK-5 PROGRESS

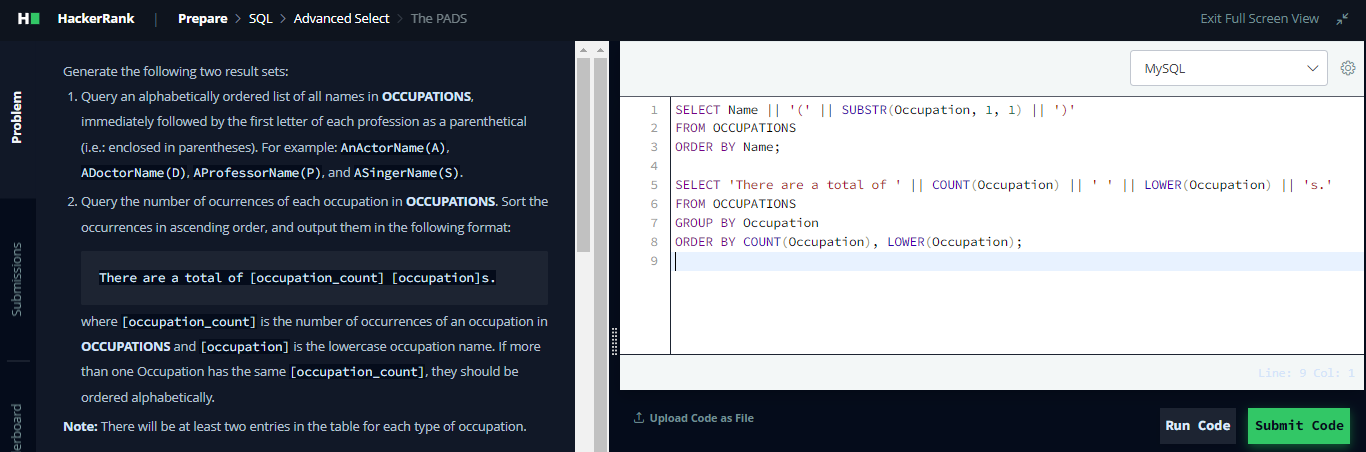
### HackerRank Tasks

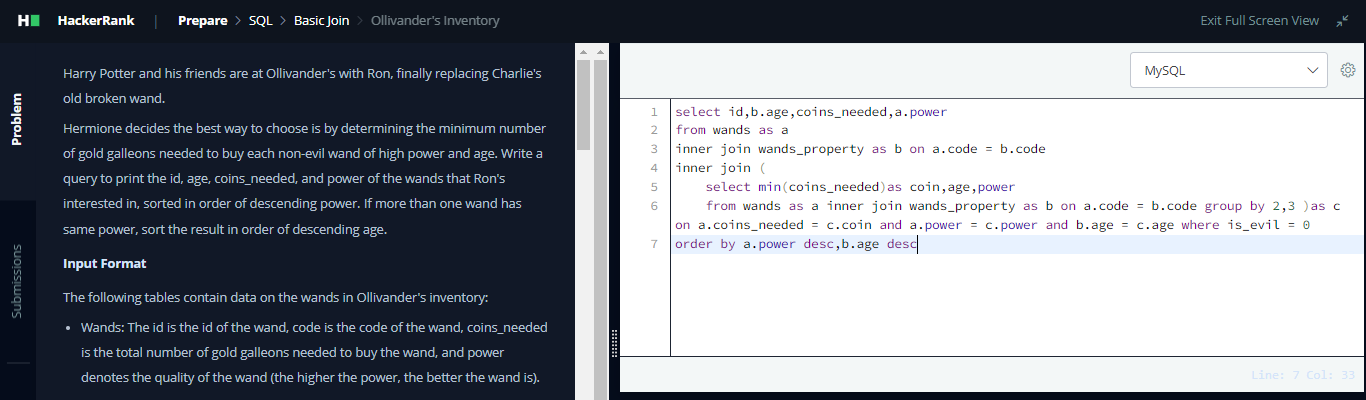
I have work on HackerRank problem solving tasks.

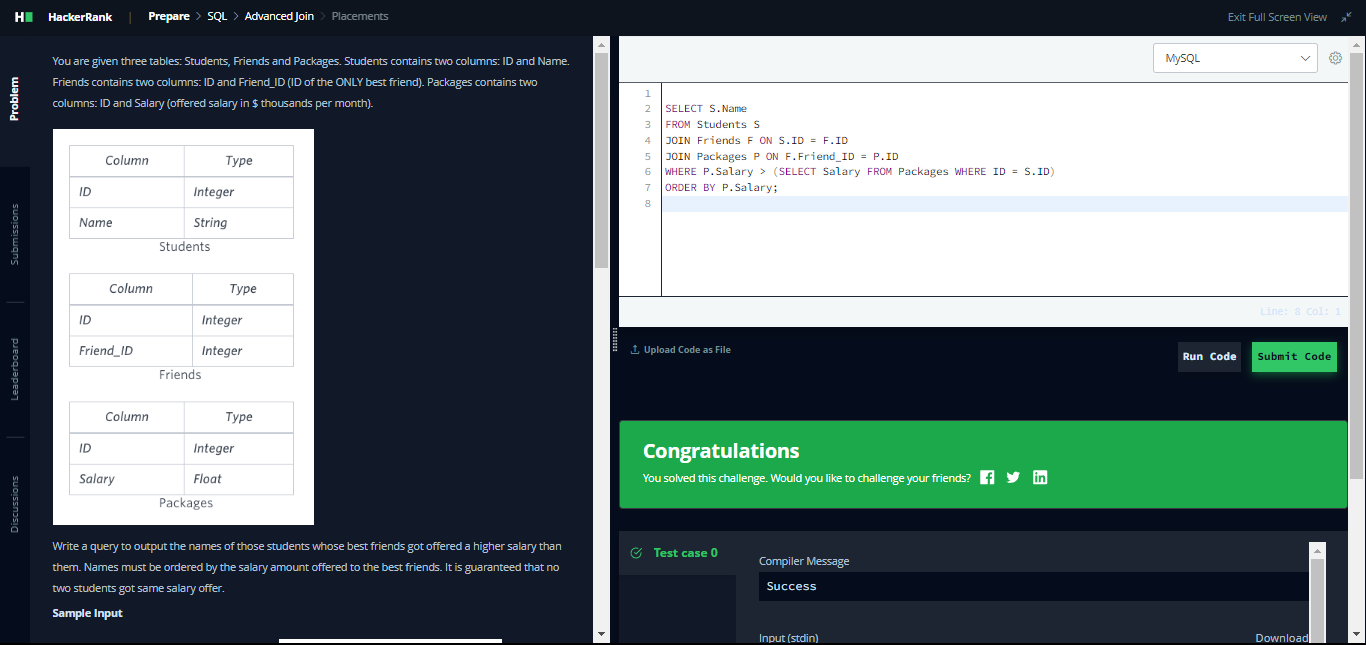
* 1. **Screenshot/Code Snippet working on HackerRank**

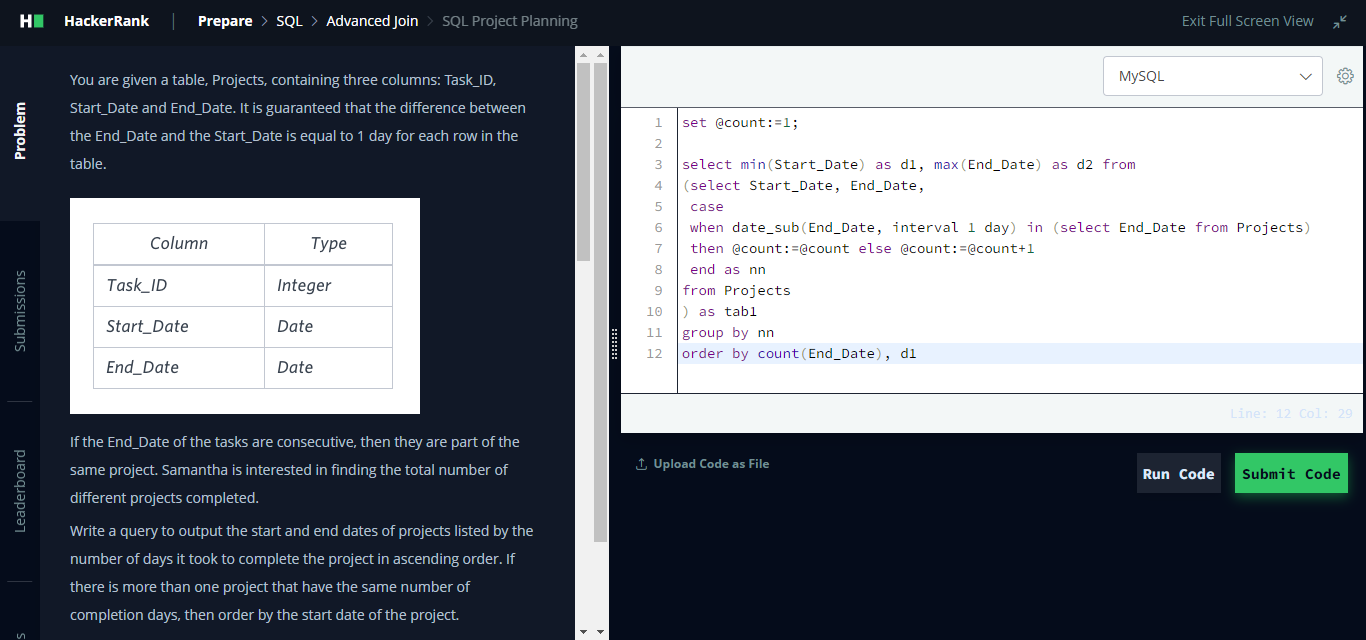












# CHAPTER-7

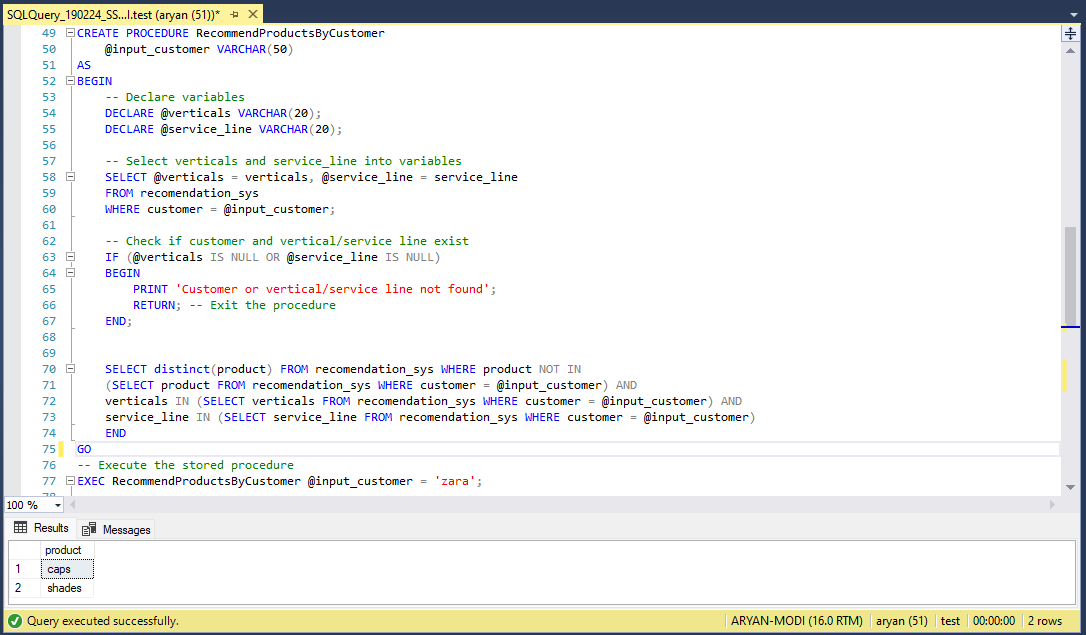
## WEEK-6 PROGRESS

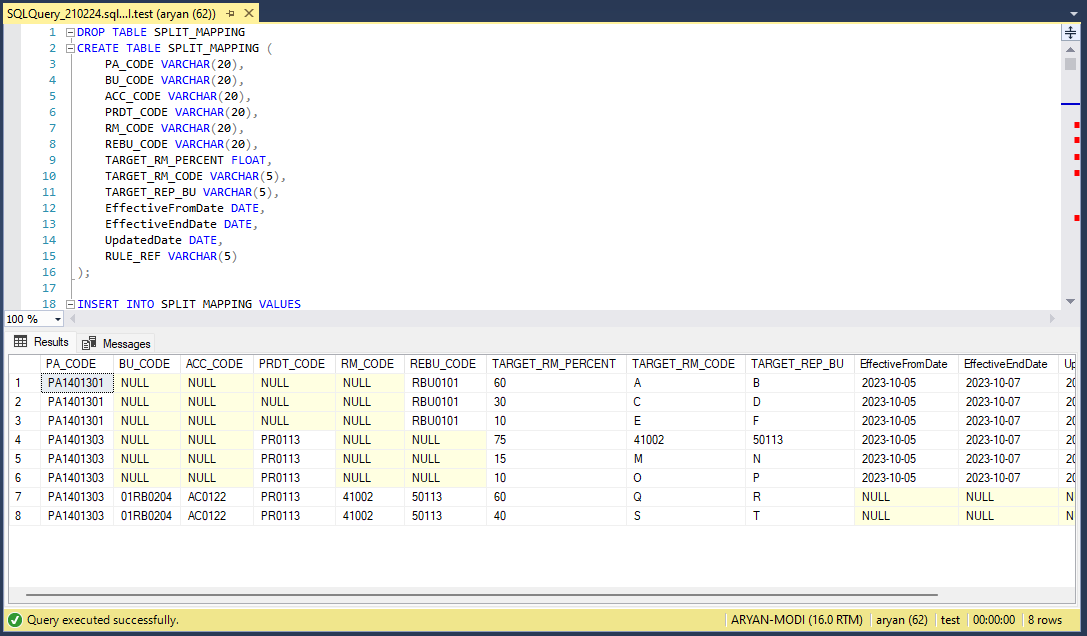
### Improve Logic Building skill

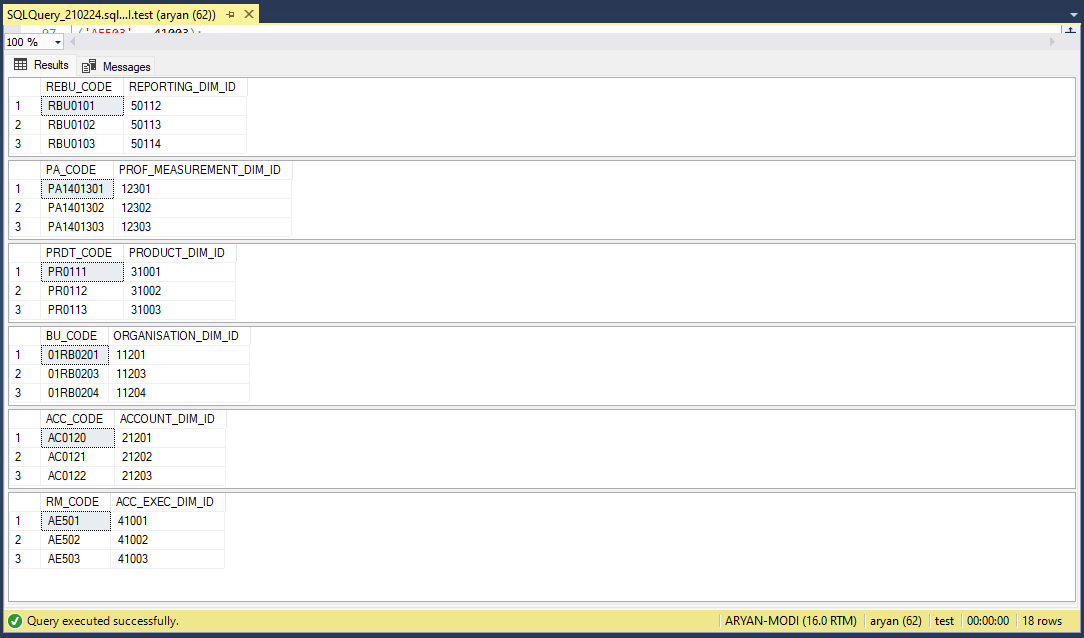
* 1. **Stored Procedures:** Practice creating and optimizing stored procedures for common tasks such as data manipulation, reporting, and business logic implementation.Explore advanced features of stored procedures such as parameterization, error handling, transactions, and dynamic SQL.
  2. **Subqueries**: Deepen your understanding of subqueries by practicing various types including scalar subqueries, correlated subqueries, nested subqueries, and common table expressions (CTEs).Work on tasks that require you to use subqueries for filtering, joining, and aggregating data from multiple tables.
  3. **Logic Building Tasks**: Challenge yourself with complex logic building tasks that involve multiple SQL operations such as filtering, sorting, grouping, and joining data to achieve specific outcomes.Practice breaking down complex problems into smaller, manageable steps and implementing the logic using SQL queries.
  4. **SingleStore Architecture**: Continue exploring SingleStore architecture and its features by working on real-world scenarios and use cases.Experiment with optimizing query performance in SingleStore by leveraging its distributed architecture, indexing strategies, and query optimization techniques.
  5. **Data Manipulation Tasks**: Work with real datasets such as cricket data or revenue data to perform various data manipulation tasks using SQL.Practice tasks like data cleaning, transformation, aggregation, and analysis to gain practical experience in handling real-world data scenarios.
  6. **Advanced SQL Functions**: Explore advanced SQL functions and features such as window functions, recursive queries, pivot tables, string manipulation functions, and date/time functions.Experiment with using these functions in combination with other SQL operations to solve complex problems efficiently.
  7. **Performance Tuning and Optimization**: Learn about performance tuning and optimization techniques in SQL, including indexing, query optimization, partitioning, and caching strategies.Practice identifying and resolving performance bottlenecks in SQL queries and procedures to improve overall system performance.
  8. **Continuous Learning and Practice**: Stay updated with the latest advancements and best practices in SQL by reading articles, tutorials, and documentation.Regularly participate in online coding platforms, forums, and communities to solve SQL challenges and share knowledge with peers.

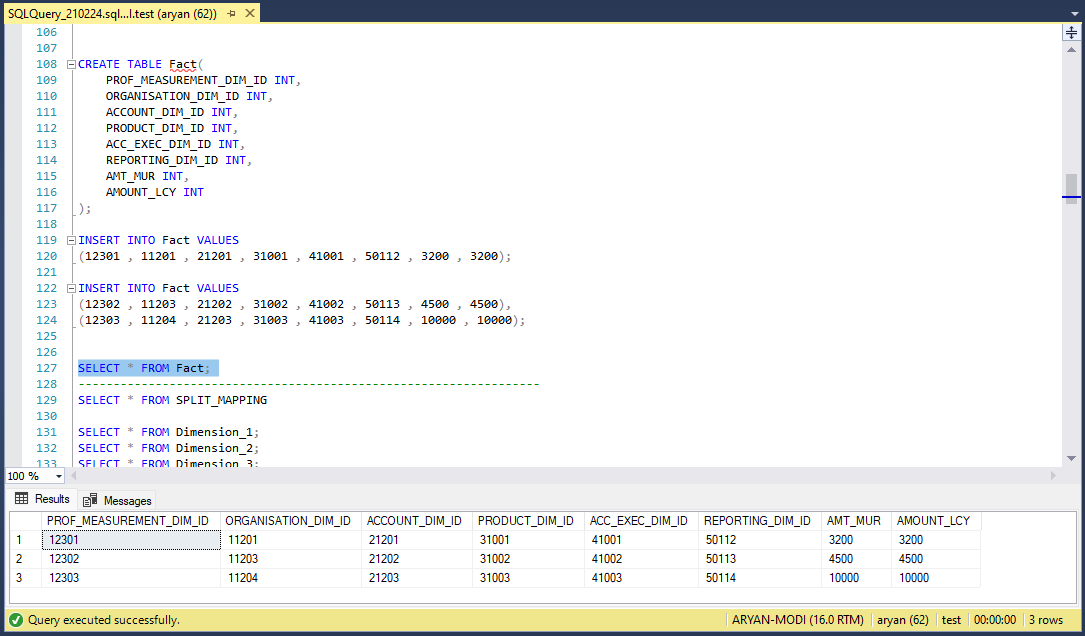
**7.2. Screenshot/Code Snippet working on Task**

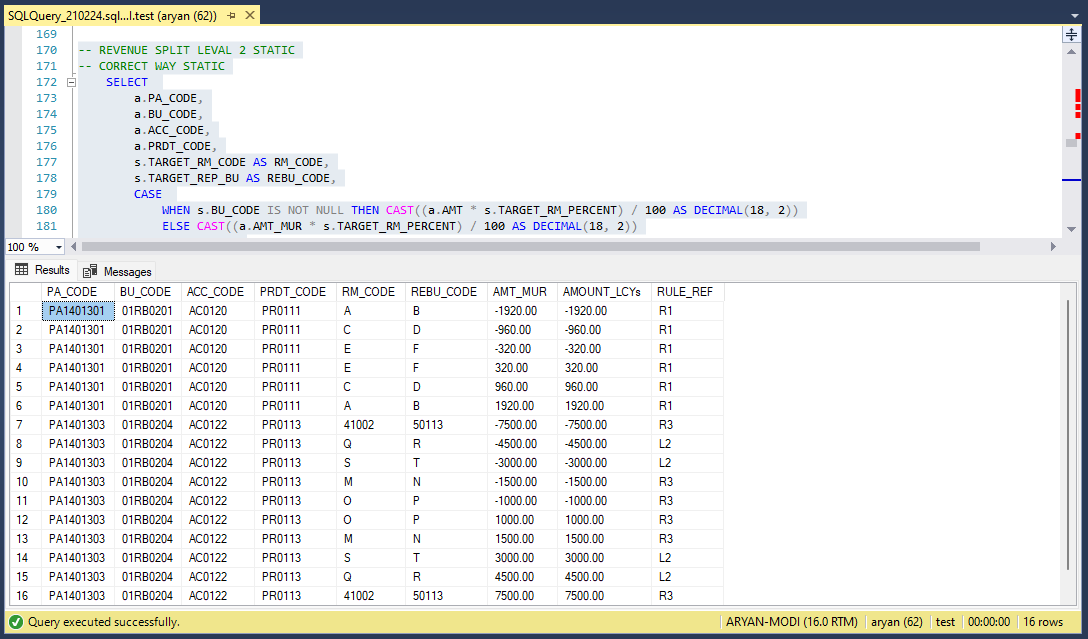
I have work on task regarding procedure , subquery , logic building task which mentation in document. I have leant about singlestore architecture , Demo: Recognizing Faces Using a SQL Database , SingleStore Architecture Overview and Product Demo , Demo: Recognizing Faces Using a SQL Database , Revenue Split Level 1 and learn about mapping and join ,cricket data and explore Singlestore and perform PIVOTE and SPLIT STRING functions , worked on SQL Logic Building which mention in document.

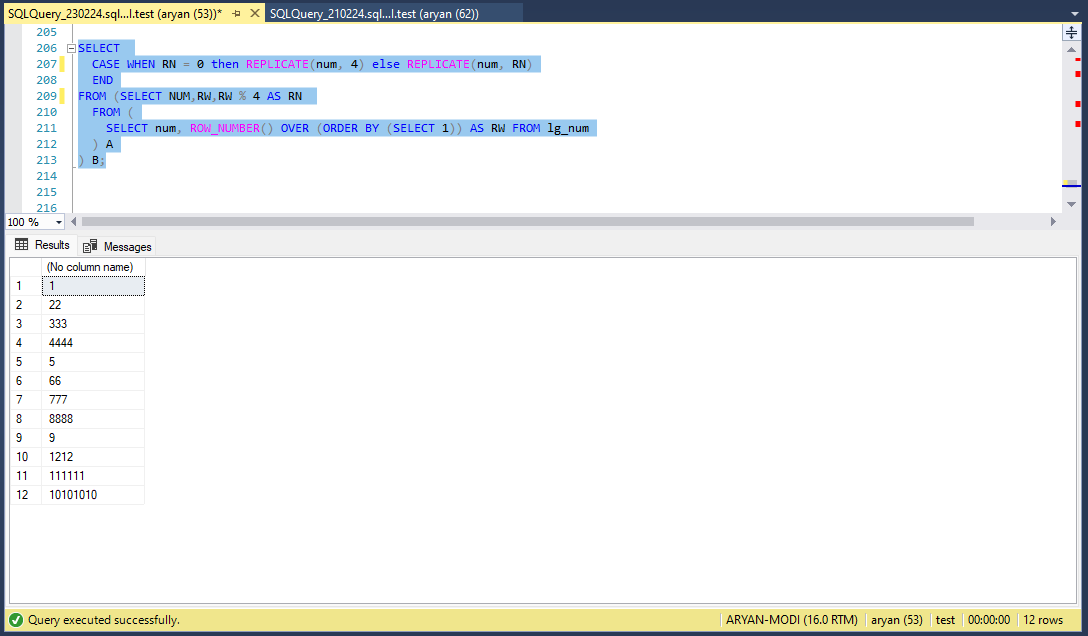












# CHAPTER-8

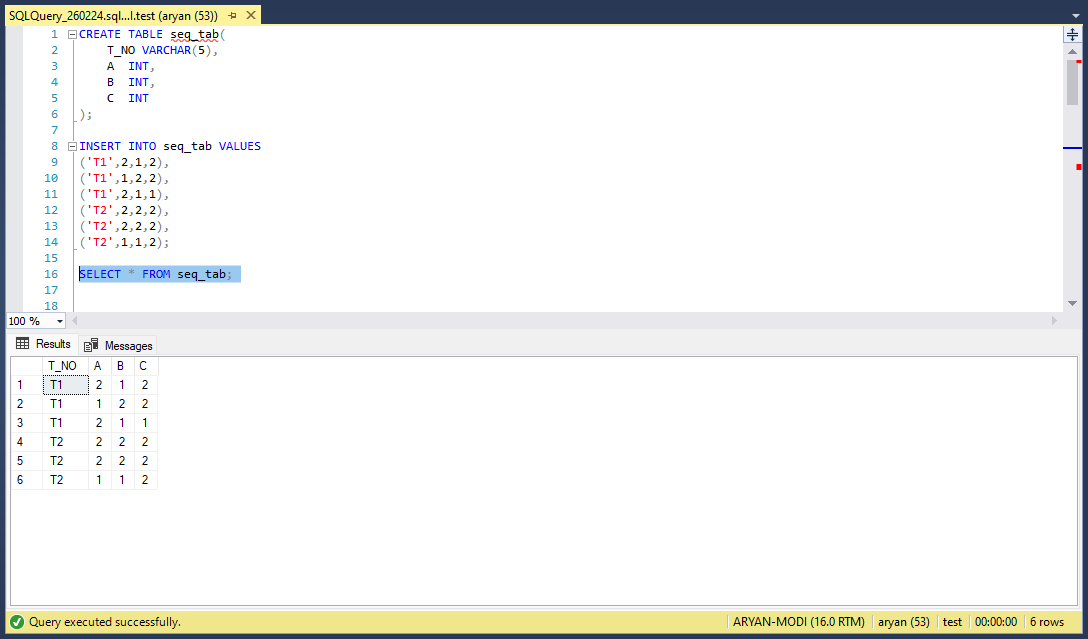
## WEEK-7 PROGRESS

### Improve Logic Building skill

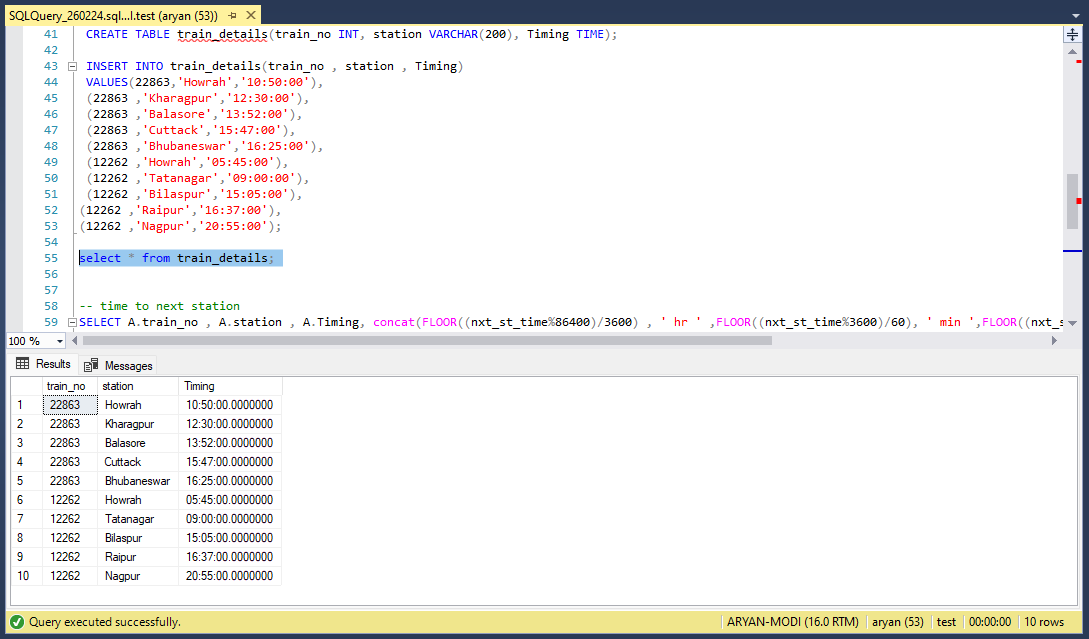
* 1. **Advanced Window Functions:-** Explore advanced window functions such as ROW\_NUMBER(), RANK(), DENSE\_RANK(), LEAD(), LAG(), NTILE(), and PERCENT\_RANK().Practice using window functions in combination with partitioning, ordering, and filtering to solve complex analytical problems.
  2. **Dynamic Approaches for Revenue Splitting:-** Experiment with dynamic SQL techniques to implement flexible revenue splitting logic that can adapt to changing business requirements or conditions.Explore the use of temporary tables, dynamic pivot queries, or CASE statements to dynamically calculate revenue splits based on various factors.
  3. **Team Hierarchy Split Data:-** Dive deeper into hierarchical data structures and practice manipulating hierarchical data using common table expressions (CTEs) or recursive queries.Explore different methods for representing and querying hierarchical data such as adjacency lists, nested sets, and path enumeration.
  4. **Transaction Management and Account SCD Data:-** Gain a deeper understanding of transaction management concepts in SQL, including ACID properties (Atomicity, Consistency, Isolation, Durability), transaction isolation levels, and locking mechanisms.Practice implementing transactional logic for handling complex operations involving multiple data updates and ensuring data integrity.Explore techniques for managing slowly changing dimensions (SCD) in a data warehouse environment, including Type 1, Type 2, and Type 3 SCD strategies.
  5. **Bank Data Population and CR/DR Details:-** Continue working with financial data and practice generating simulated bank transaction data for credit (CR) and debit (DR) details.Explore techniques for generating realistic transaction data such as random number generation, distribution sampling, and temporal patterns.Experiment with aggregating and analyzing bank transaction data to extract meaningful insights and perform financial analysis.
  6. **Optimization and Performance Tuning:**- Focus on optimizing SQL queries and procedures for performance by analyzing query execution plans, indexing strategies, and query optimization techniques.Practice identifying and resolving performance bottlenecks in SQL code to improve overall system efficiency and scalability.
  7. **Documentation and Best Practices:-** Document your SQL solutions, including logic, assumptions, and design decisions, to improve readability and maintainability.Follow best practices for SQL development, such as using meaningful naming conventions, writing clear and concise code, and properly commenting your code for future reference.

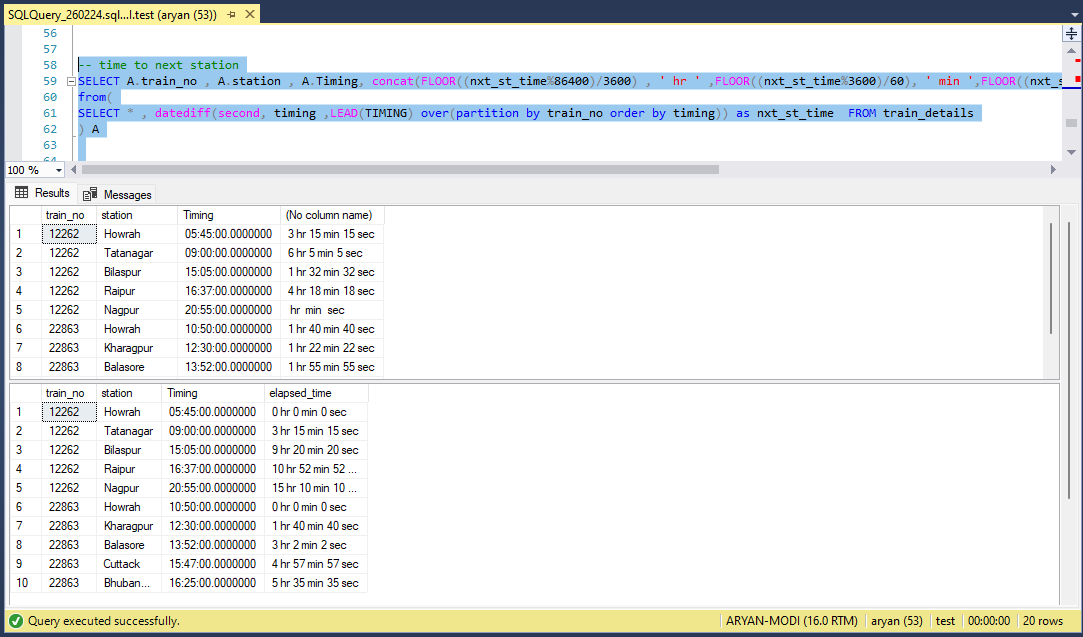
**8.2. Screenshot/Code Snippet working on Task**

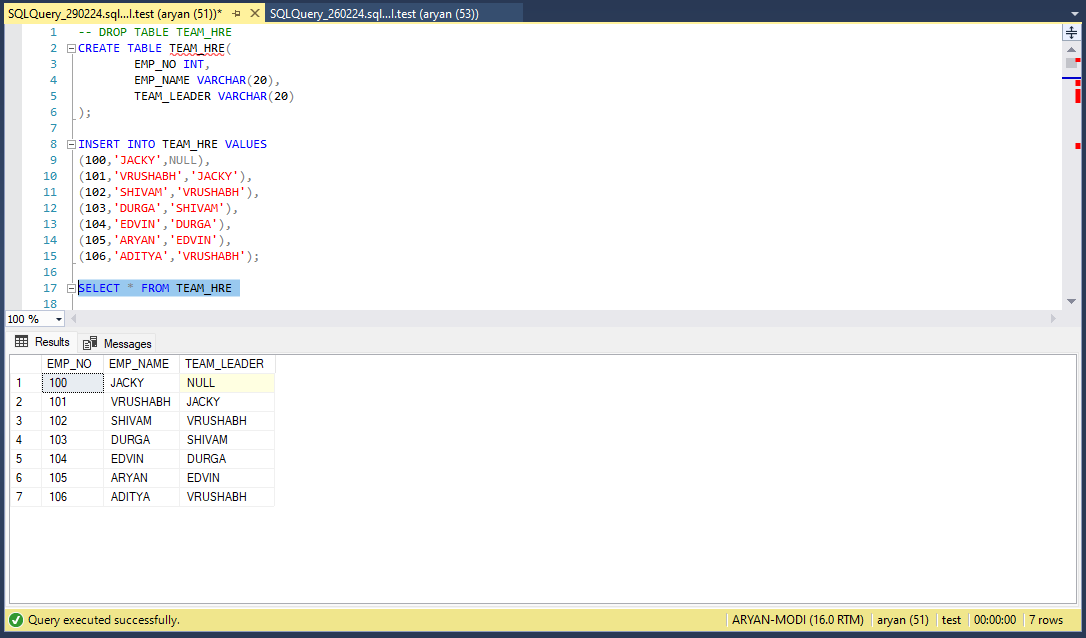
I have work on task regarding procedure , subquery , logic building task which mentation in document. I have leant about singlestore architecture , Demo: Recognizing Faces Using a SQL Database , SingleStore Architecture Overview and Product Demo , Demo: Recognizing Faces Using a SQL Database , Revenue Split Level 1 and learn about mapping and join ,cricket data and explore Singlestore and perform PIVOTE and SPLIT STRING functions , worked on SQL Logic Building which mention in document.

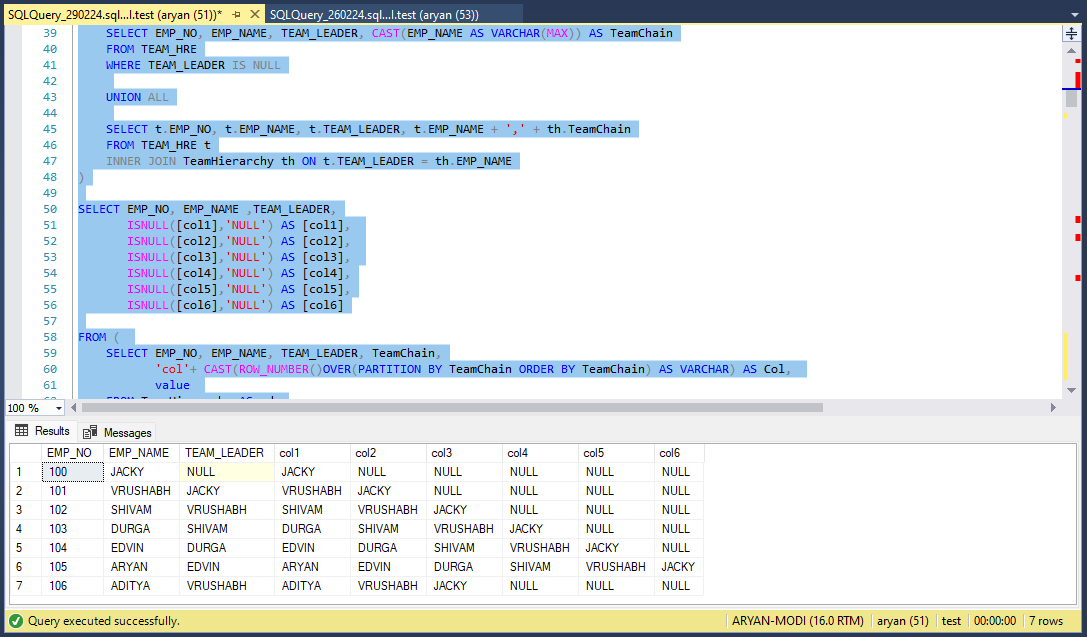


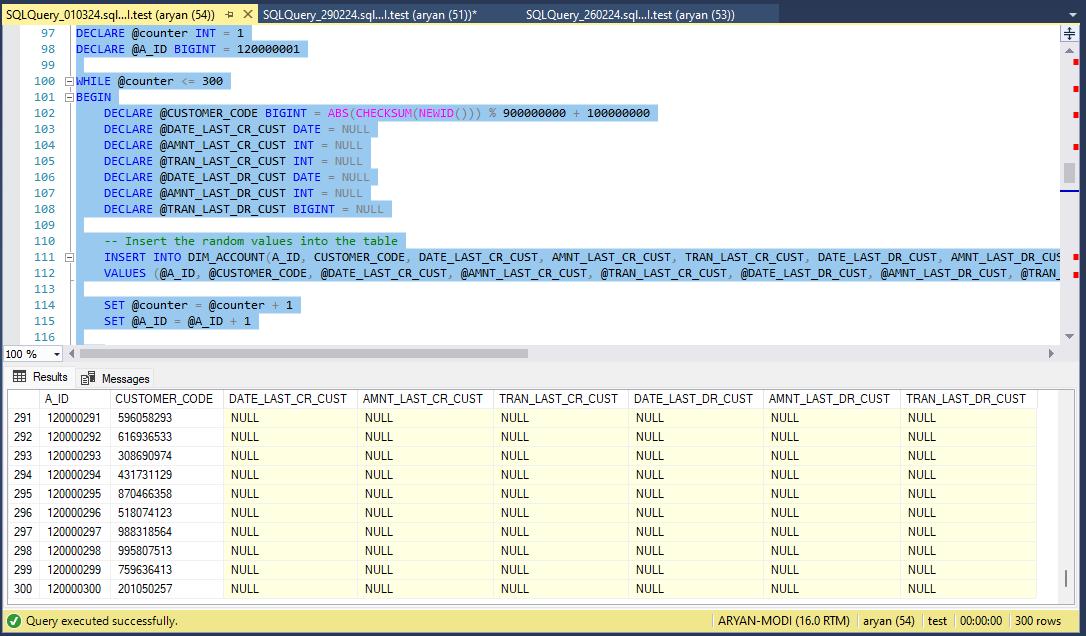


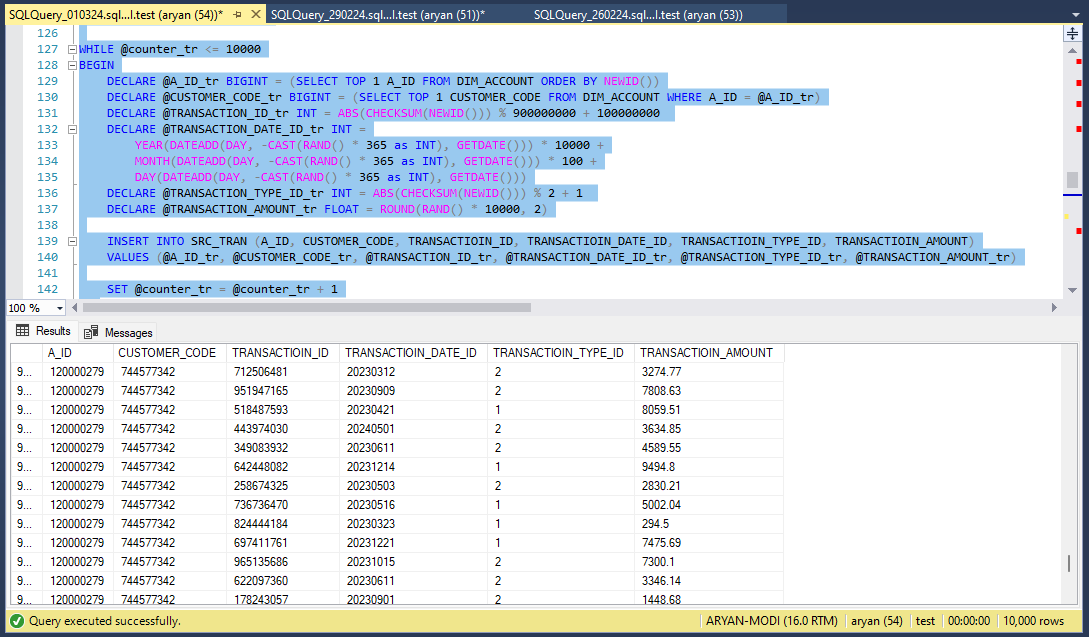


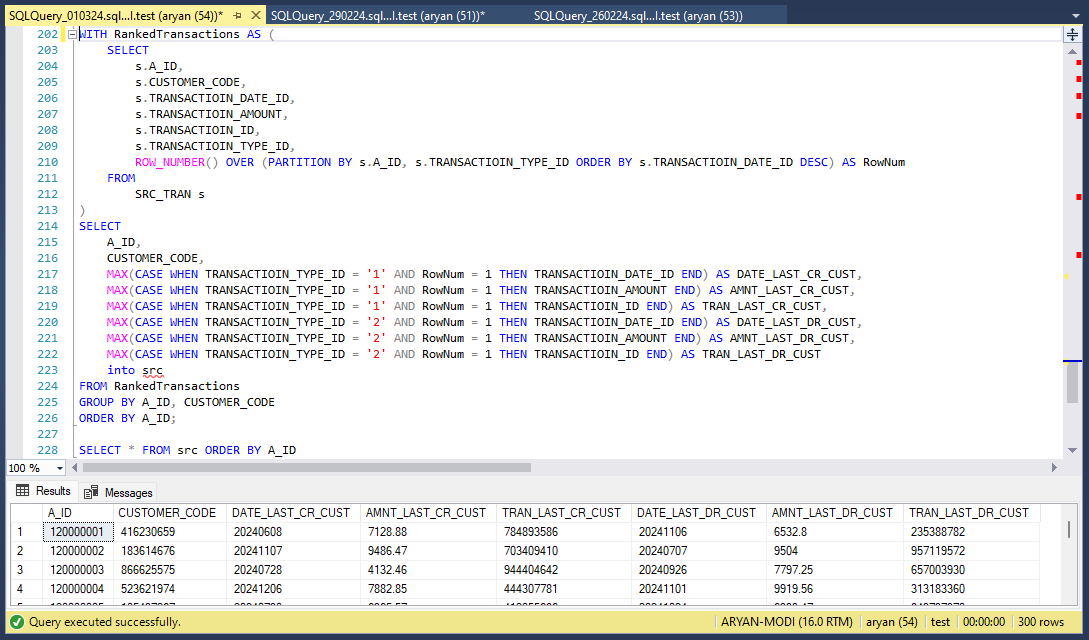












# CHAPTER-9

## LIVE PROJECT

### LOOP SQL

### In SQL Server, a loop is the technique where a set of SQL statements are executed repeatedly until a condition is met.

### SQL (Structured Query Language), loops are constructs used to iterate over a set of data or perform repetitive tasks.

### Unlike traditional programming languages like Java or Python, SQL doesn't have built-in looping constructs like "for" or "while" loops.

### However, there are methods available in SQL to achieve similar iterative functionality, such as cursors and recursive queries.

### JavaScript

### JavaScript is a versatile programming language commonly used for building interactive web applications.

### Event Handling:- JavaScript is frequently used to handle user interactions such as clicks, key presses, form submissions, etc. You can define event listeners that trigger functions when specific events occur on HTML elements.

### DOM Manipulation:- JavaScript allows you to dynamically update the content and styles of HTML elements on a webpage. You can manipulate the Document Object Model (DOM) using functions to add, remove, or modify elements.

### AJAX Requests:- JavaScript enables asynchronous communication with a server using AJAX (Asynchronous JavaScript and XML) requests. You can define functions to send HTTP requests to a server and handle the responses without reloading the entire page.

### Error Handling:- JavaScript allows you to handle runtime errors and exceptions gracefully using try-catch blocks. You can define functions to catch and handle errors to prevent them from crashing the entire application.

### node JS

### Node.js is widely used for building web servers due to its non-blocking, event-driven architecture. Frameworks like Express.js make it easy to create robust and scalable web applications.

### API Development:- Node js is ideal for building RESTful APIs to provide data and services to client-side applications. You can define routes, handle requests, and interact with databases using libraries like Mongoose or SQL databases.

### Real-time Applications:- Node js, along with frameworks like Socket.io, is used for building real-time web applications that require bidirectional communication between clients and servers.

### Microservices:- Node js is well-suited for building microservices architecture, where small, independent services communicate with each other to perform specific tasks. This allows for better scalability, maintainability, and deployment flexibility.

### FIREVASE

### Firebase is a comprehensive platform provided by Google for building web and mobile applications. It offers a variety of services that can be used individually or together to streamline the development process.

### Realtime Database:- Firebase Realtime Database is a NoSQL cloud database that stores and syncs data between your users in real-time. It's ideal for building collaborative and real-time applications such as chat apps, collaborative editing tools, etc.

### Hosting:- Firebase Hosting provides fast and secure hosting for your web app, including SSL encryption, CDN integration, and automatic deployment from a Git repository. It's ideal for hosting static websites, single-page apps, and progressive web apps.

### Cloud Functions:- Firebase Cloud Functions allows you to run server-side code in response to events triggered by Firebase features and HTTPS requests. You can use it to extend your app's functionality, integrate with third-party services, and automate tasks.

### Ngrok

### Ngrok is a service that creates secure tunnels to your localhost, exposing it to the internet. It's often used by developers during the development and testing phase of web applications or APIs. This allows them to share their work with others or test functionality across different devices without deploying it to a public server.

### Tunneling: Ngrok establishes secure tunnels from a public endpoint (e.g., ngrok.io) to a port on your local machine. This means you can expose a local server running on your machine to the internet without having to deploy it to a public server.

### Security: Ngrok tunnels are secure, utilizing TLS encryption for data transfer. This means that data transmitted between your local machine and the ngrok server is encrypted.

### Subdomains: Ngrok allows you to reserve a subdomain under ngrok.io for your account, making it easier to remember the URL for your tunnels.

### Inspecting traffic: Ngrok provides a web interface where you can inspect the HTTP traffic passing through the tunnel in real-time. This can be helpful for debugging and monitoring purposes.

### Authentication and custom domains: Ngrok offers features like HTTP basic authentication for added security, and you can also use custom domains if you’re on one of their paid plans.

### Pricing: Ngrok offers both free and paid plans. The free plan is limited in terms of features and concurrent connections, while the paid plans offer more features and higher limits.

### Cognos Custom Cntrol

### In IBM Cognos, Custom Controls are a way to extend the functionality of the report authoring environment by allowing developers to create custom user interface components. These components can be integrated into Cognos reports to enhance interactivity, visualization, or data manipulation capabilities beyond what's provided by default.

### Purpose: Custom Controls allow developers to create custom user interface elements or components that can be embedded within Cognos reports. These components can range from simple input fields to complex interactive visualizations.

### Development: Custom Controls are typically developed using web technologies such as HTML, CSS, and JavaScript. Developers can leverage libraries like jQuery or D3.js to create rich and interactive components.

### Integration: Once developed, Custom Controls can be integrated into Cognos reports using the Report Studio authoring tool. They are added to reports just like any other report item, such as tables or charts.

### Functionality: Custom Controls can provide a wide range of functionality, including custom input forms, interactive charts and graphs, data filters, and more. They can interact with Cognos data sources and respond to user actions within the report.

### Flexibility: Custom Controls offer flexibility in terms of design and functionality, allowing developers to tailor them to specific reporting requirements or user preferences.

### Deployment: Custom Controls need to be deployed to the Cognos server environment so that they can be used within reports. This typically involves uploading the control files to the appropriate directories on the Cognos server.

### Security: Like any custom code integrated into a software environment, Custom Controls should be developed and deployed with security considerations in mind to prevent vulnerabilities or exploits.

### Architecture: Custom Controls are typically built using a combination of HTML, CSS, and JavaScript. They leverage the Cognos JavaScript API (Cognos Mashup Services) to interact with the Cognos report environment. This API provides methods for accessing report data, parameters, and other properties, as well as for responding to user interactions.

### Types of Custom Controls:

### Input Controls: These controls allow users to input data or make selections that can affect the behavior of the report. Examples include custom dropdown lists, date pickers, sliders, etc.

### Visualization Controls: These controls are used to enhance the visual representation of data within the report. Examples include custom charts, graphs, maps, etc.

### Interactive Controls: These controls enable interactive features within the report, such as collapsible sections, expandable tables, tooltips, etc.

### Integration with Cognos Reports:

### Custom Controls are integrated into Cognos reports through the use of Report Studio, which is the report authoring tool in IBM Cognos.

### Developers can add Custom Controls to reports by dragging and dropping them onto the report canvas just like any other report item.

### Once added, Custom Controls can be configured to interact with report data, parameters, and other elements.

### Data Interaction:

### Custom Controls can interact with Cognos report data by fetching data from data sources, filtering data based on user input, and updating report content dynamically.

### They can also respond to events such as data updates, user interactions, or changes in report state.

### Extensibility:

### Custom Controls can be extended and customized further by integrating third-party libraries or frameworks.

### Developers can create reusable components or templates to streamline development and maintain consistency across reports.

### Documentation and Support:

### IBM provides documentation, tutorials, and forums to help developers learn how to create and integrate Custom Controls effectively.

### Developers can also find community-contributed resources, such as code samples and plugins, to accelerate development.

### Cognos

### IBM Cognos is a suite of business intelligence (BI) and performance management software products. It's designed to help businesses extract insights from their data and make informed decisions to improve performance and achieve their goals.

### Reporting: Cognos provides robust reporting capabilities, allowing users to create and distribute a wide range of reports, including operational reports, financial statements, dashboards, and scorecards. Reports can be highly customizable, with options for formatting, filtering, and interactive features.

### Analysis: With Cognos, users can perform multidimensional analysis of data to uncover trends, patterns, and correlations. It supports ad-hoc querying, slicing and dicing, and drill-down capabilities to explore data from different angles.

### Dashboarding: Cognos offers dashboarding functionality to visualize key performance indicators (KPIs) and metrics in a centralized and interactive manner. Users can create personalized dashboards with charts, graphs, and other visualizations to monitor performance and track progress towards goals.

### Data Integration: Cognos can integrate with a variety of data sources, including relational databases, data warehouses, OLAP cubes, spreadsheets, and cloud-based data sources. It supports both structured and unstructured data, allowing users to access and analyze data from multiple sources in a unified environment.

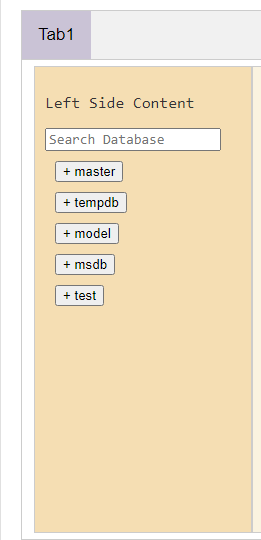
### Planning and Budgeting: Cognos includes capabilities for budgeting, planning, and forecasting, enabling organizations to create, manage, and analyze budgets and financial plans. It supports collaborative planning processes, workflow automation, and what-if scenario analysis.

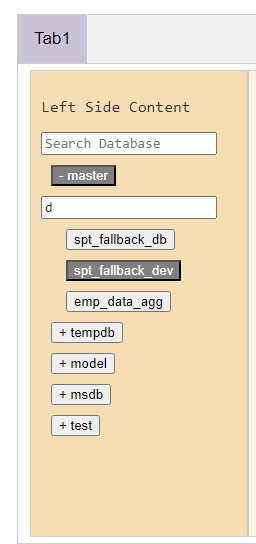
### Predictive Analytics: In addition to traditional BI and reporting features, Cognos offers advanced analytics capabilities, including predictive modeling, data mining, and statistical analysis. This allows users to uncover insights and make predictions based on historical data and predictive algorithms.

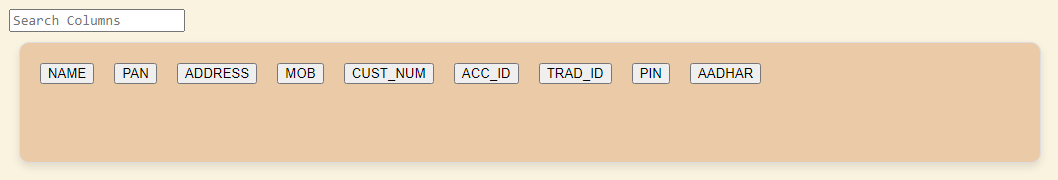
### Mobile Access: Cognos provides mobile access to reports, dashboards, and analytics, allowing users to access insights and make decisions on the go. It supports responsive design and native mobile apps for iOS and Android devices.

### Security and Governance: Cognos includes features for data security, access control, and governance to ensure that sensitive information is protected and regulatory compliance requirements are met. It supports role-based access control, encryption, auditing, and data masking.

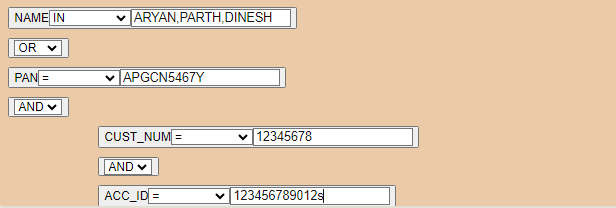
### Screenshot/Code Snippet working of project



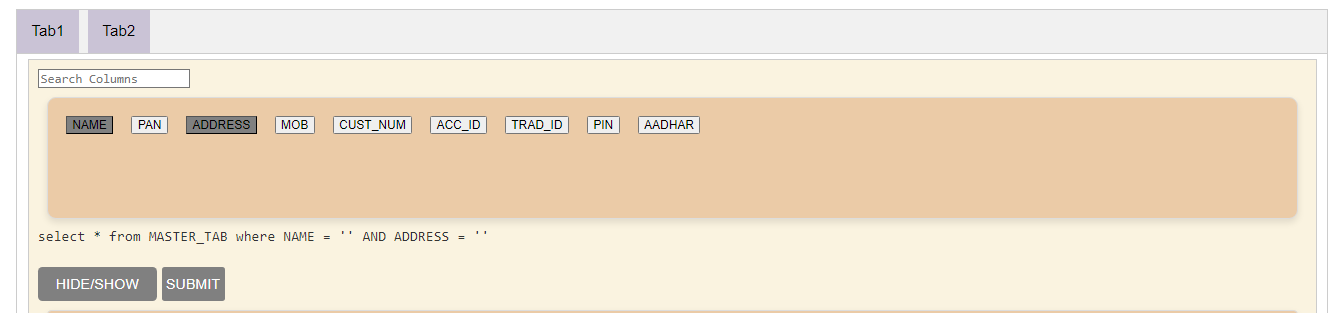


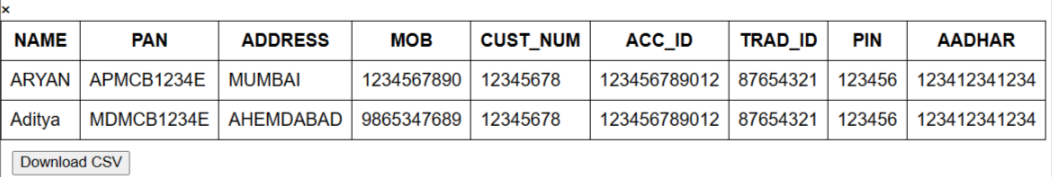


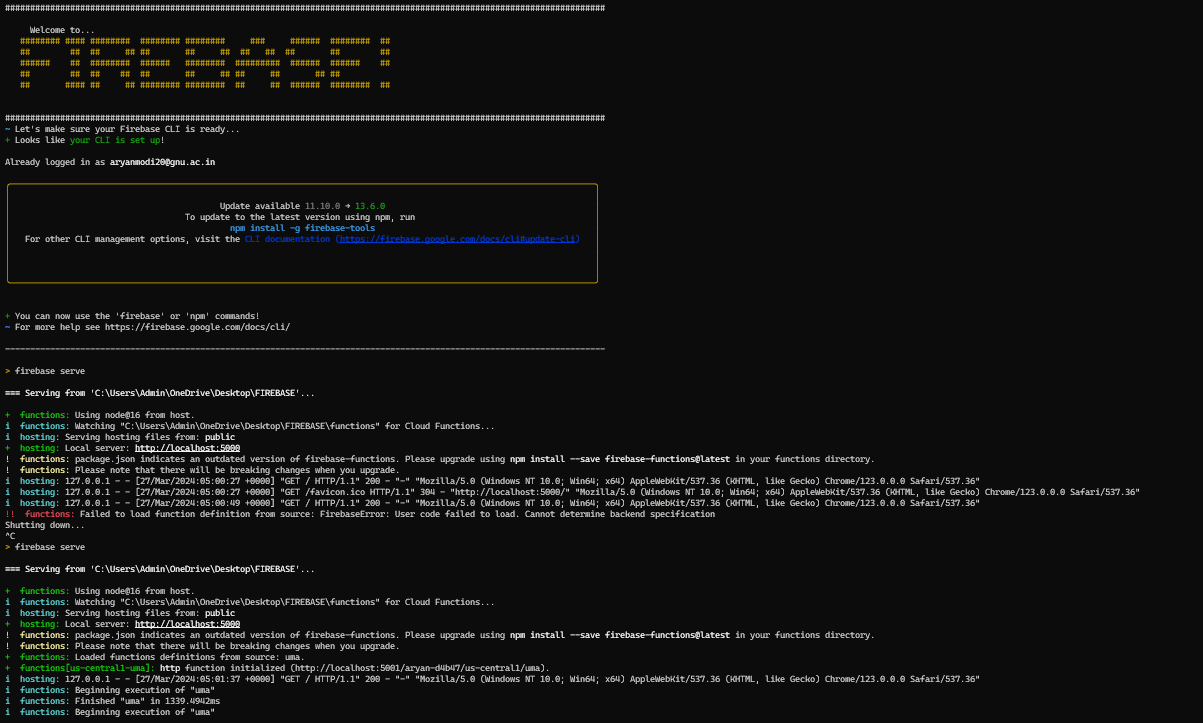


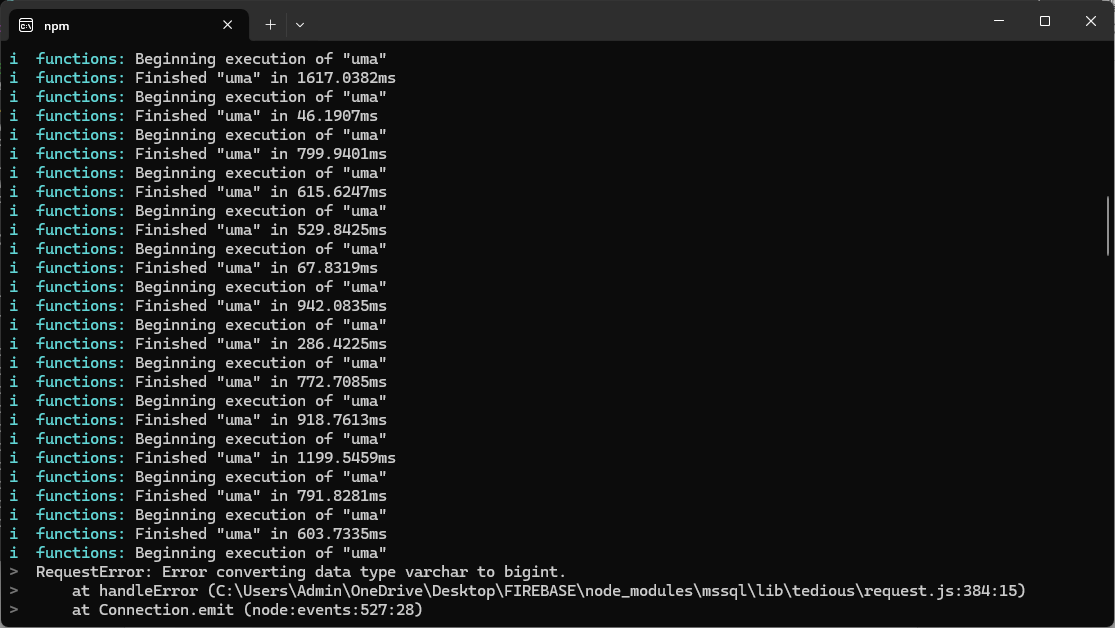


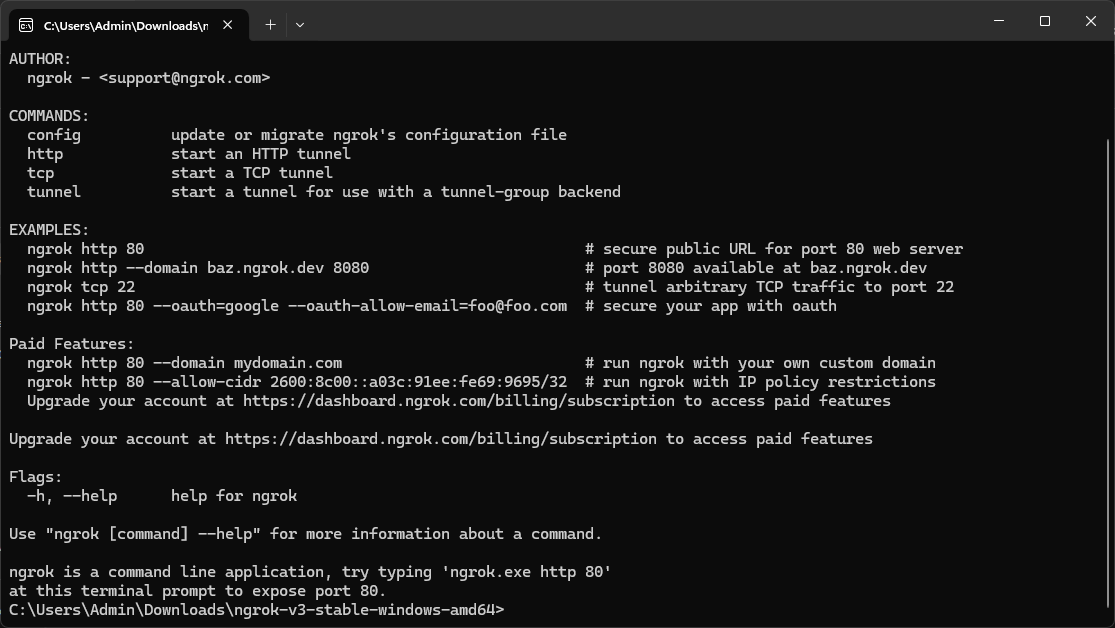


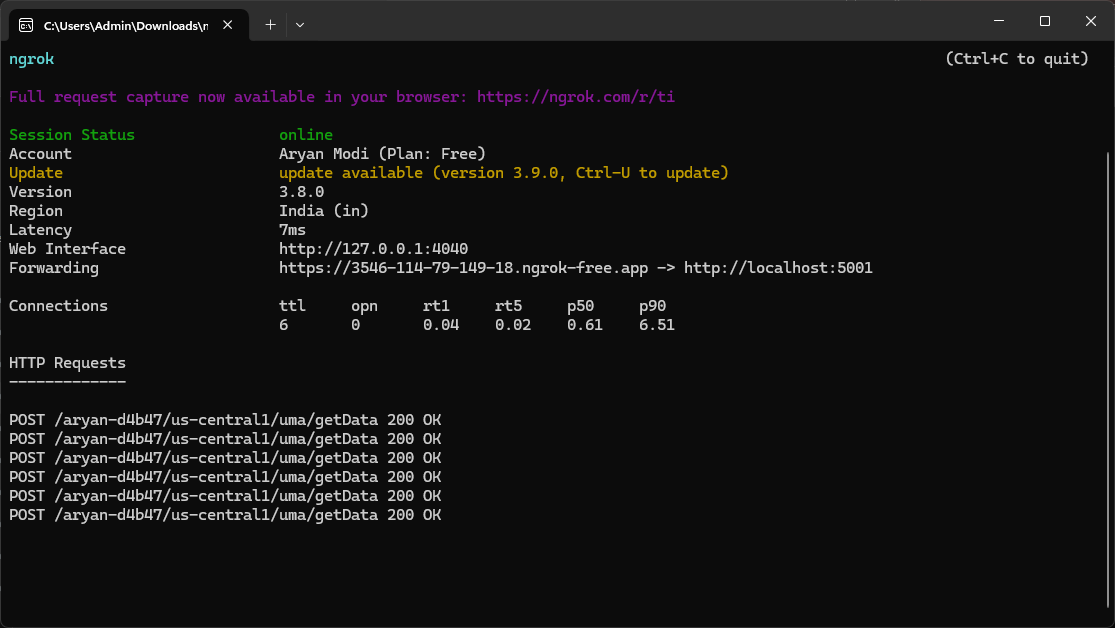












### REFERENCES

* 1. <https://stackoverflow.com/>
  2. <https://learn.microsoft.com/en-in/>
  3. <https://www.sqlservertutorial.net/>
  4. <https://www.tutorialspoint.com/index.htm>
  5. <https://hub.docker.com/>
  6. <https://portal.singlestore.com/organizations/5a2441ce-87e9-44e5-a6c1-83b759f33913/homepage>
  7. <https://medium.com/>
  8. <https://nodejs.org/en>
  9. <https://console.firebase.google.com/u/0/>