|  |  |
| --- | --- |
| Adya Sinha and Hamna Qadir  Assignment Pair Practice Logs |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Assignment 2: Data Warehouse Design and ETL | | | |
| Date and Tasks | Main Practice Steps(By Driver) | Results(By Navigator) | Comments(By Both) |
| **1. Date**: 16/09/24  **Task**: Restoring Staging database | **Operation Main Steps:**  1. Download the DS7502Ass2-2024Staging file from Moodle.  2. Opened SQL server management and restored the staging database. | **Results:** | **Comments by Hamna and Adya:**  The staging database was successfully restored without any issues, setting a solid foundation for the next steps. |
| 2. **Date**: 17/09/24  **Task**: Creating a Data Mart Schema using Visio | **Operation Main Steps:**  1.Create a snowflake schema using Visio 2010.  2.Creat a fact table “BookingFacts” with measures (SalesAmount, Discount and GST)  3.Create Dimensions Tables (CustomerDim, AirportDim,AirlineDim, AgentDim, DateDim, CityDim, CountryDim, BranchDim)  4. Define the relationships and the hierarchies: -  **•** Airport → City → Country  **•** Customer → City → Country  **•** Agent → Branch → City → Country  **•** Day → Month → Quarter → Year  6. Define attributes for each of the tables, including the data types.  7. Define the Fact Table structure. 8. Specify the primary and foreign keys of each table. | **Results:**  Snowflake Schema for the GoTravelDM: | **Comments by Hamna and Adya:**  The snowflake schema was designed well, and all relationships, hierarchies, and attributes were defined accurately. |
| 3. **Date**: 18/09/24  **Task**: Creating the data mart using SQL Server. | **Operation Main Steps:**  1. Open SQL Server Management Studio  2. Write SQL statements to create the database and tables based on the Visio design: - CREATE DATABASE statement.  - CREATE TABLE statements for each table. - Define column data types matching the staging tables. - Add PRIMARY KEY and FOREIGN KEY constraints. - Execute the code to create the database names ‘GoTravelDM’.  - Expand your Database and right click on ‘Database Diagrams’.  - Select all the tables and click ‘Add’ to create the database diagram for the database and compare it with the one you made using Visio. | **Results:**  SQL script for GoTravelDM database:    Database diagram from SQL:  A screenshot of a computer  Description automatically generated | **Comments by Hamna and Adya:**  The SQL statements were executed successfully to create the GoTravelDM database and all the tables. The database diagram matched our Visio schema perfectly, confirming the accuracy of the SQL scripts and the overall design. |
| 4. **Date**: 19/0924  **Task**: Generating the Date Dimension using Analysis Services | **Operation Main Steps:**  1. Using Analysis Services to generate the Date dimension table: - Create a new Analysis Services project in Visual Studio. - Connect your server to the project.  - Right click ‘Dimensions’ in the object explorer to create a new dimension.  -Select ‘Generate a time table in the data source and click next’.  - Define the time period to be from 2005-01-01 to 2010-12-31.  - Select the ‘Fiscal Calendar’ and click next.  - Finally change the dimension name from ‘Time’ to ‘Date’.  - Right click solution and deploy it.  It should deploy successfully without any errors. | **Results:**      Successful deployment of the Date dimension:    A screenshot of a computer  Description automatically generated | **Comments by Hamna and Adya:**  The Date dimension was deployed successfully, with no errors during the process. |
| 5. **Date**: 23/09/24  **Task**: Importing Excel spreadsheets into staging database. | **Operation Main Steps:**  1. Importing the Airport and Airline Excel spreadsheets: - Open SQL Server.  - Right Click the ‘DS7502Ass2-2024Staging’ database, select Tasks, and the click on ‘Import Date’.  - Choose the data source as ‘Microsoft Excel’ and browse for the path where the file is saved in the C drive.  - Click next and choose the destination as ‘SQL Server Native Client 11.0’.  - Select the ‘Use windows Authentication’ and click next.  - Select ‘Copy data from 1 or more tables or view’ and click next.  - On the ‘Save and Run Package’ screen, click ‘Run immediately’.  - Click Finish preceding the Next button to then successfully execute it.  - After closing the wizard, the Excel file should appear under tables in the staging database.  - Follow the above steps to import the Airport.xlsx and Airline.xlsx into the staging database. | **Results:**          Successful import of the Airline and Airport Excel files: | **Comments by Hamna and Adya:**  Both the Airport and Airline Excel files were imported into the staging database smoothly, with all data appearing as expected |
| 6. **Date**: 27/09/24  **Task**: Creating a view for Customer data. | **Operation Main Steps:**  1. Create a view for Customer data: - Write a SQL query joining Customer and City tables. - Create a view using this query in the staging database.  2. Writ a SQL query which joins the Customer and City tables. 3. Execute the code to create a view using that query in the staging database. | **Results:**  SQL script for the Customer data view: | **Comments by Hamna and Adya:**  The view for Customer data was created successfully, with a correct SQL join of the Customer and City tables. |
| 7. **Date**: 03/10/2424  **Task**: Defining the data flow tasks. | **Operation Main Steps:**  1. Start Visual Studio 2019 using the **Run as Administrator** option open a new ‘Integration Services Project’ from the Templates.  2. Enter the name and click create.  3. Rename the ‘Package.dtsx’ to ‘GoTravelFactLoad.dtsx’.  4. Set up data source connections- - Create an ADO.NET connection to the staging database.  - Create an ADO.NET connection to the GoTravelDM database.  5. Create data flow tasks in the correct sequence: (Load Country, Load City, Load Branch, Load Agent Load Customer, Load Airport, Load Airline, Load Date, Load BookingFact).  6. Connect all of the data flow tasks by the blue arrow in order. | **Results:**  Running all the data flow tasks successfully : | **Comments by Hamna and Adya:**  All the data flow tasks were set up and sequenced correctly in Visual Studio. The ADO.NET connections to the staging and data mart databases worked as intended, and the data flows were logically organized to ensure smooth data transfer between systems. |
| 8. **Date**: 07/10/24  **Task**: Data flow tasks performing the required transformations. | **Operation Main Steps:**  1.For each of the data flow, a source component and the required transformations are added.  - Use the ADO NET Source (select the staging database and respective table) and ADO NET Destination (select the data mart database and respective table) for all the data flow tasks except ‘Load CityDim’, ‘Load AgentDim’, ‘Load BranchDim’, and ‘Load BookingFact Table’.  - In the ‘Load CityDim’, create two ADO NET Source tasks for the City and Airport staging tables, sort the data by CityName, merge the sorted data, remove duplicates with a Distinct task, and load into the City dimension table using ADO NET Destination.  - In the ‘Load AgentDim’, use an ADO NET Source with the Employee staging table, apply a Conditional Split to filter for employees with the title "Agent," and load the filtered data into the Agent dimension table using an ADO NET Destination.  - In the ‘Load BranchDim’, use an ADO NET Source with the Branch staging table, apply a Conditional Split to filter for "Branch Office" types, and load it into the Branch dimension table using an ADO NET Destination.  - In the ‘Load BookingFact Table’, use an ADO NET Source with the Customer staging table, add a Derived Column for GST calculation, and load the added columns into the BookingFact dimension table using an ADO NET Destination. | **Results:**  ADO NET Source and ADO NET Destination for data flow task:  A screenshot of a computer program  Description automatically generated  Using a Sort and Merge Join function in the City Dimension:    Using a Conditional Split function in the Agent dimension:  Using a Conditional Split in the Branch dimension:    Using a Derived Column in the Fact Booking table: | **Comments by Hamna and Adya:**  All transformations, including Sort, Merge, Conditional Split, and Derived Column, were applied accurately, and the data loaded as expected. The CityDim, AgentDim, and BranchDim transformations were handled with precision, and the BookingFact table was populated with accurate calculations for GST. |
| 9. **Date**:  **Task**: Testing and Execution. | **Operation Main Steps:**  1.Debugging each of the data flow  2. Then check the populated tables in the data mart database.  - When u double-click the ADO NET Destination and preview the ‘BookingFact’ table, you can see that the data mart has successfully been populated.  3. Executing the entire package successfully. | **Results:**  The data mart is successfully populated: | **Comments by Hamna and Adya:**  Testing and debugging confirmed that the data mart was populated successfully, with no issues during the execution of the package. Previewing the BookingFact table in SQL Server showed that the transformations and data loads were accurate |