**Step 1: Data Sources**

**Objective:** Load raw data from SQL Server tables into SSIS.

* **Tables:**
  + users → user information (signup\_date, country, user\_id)
  + employers → employer info (if needed for reporting)
  + demo\_sessions → demo attendance logs
  + engagement → user activity (logins\_last\_30\_days, messages\_sent, profile\_completed)

**Screenshot for Evidence:**

* SSIS **Data Flow Source** preview showing sample rows.
* Include one table’s preview (e.g., Engagement showing 0/1 counts or true/false).

**Step 2: Connections**

**Objective:** Connect SSIS to SQL Server database.

1. **Add OLE DB Connection Manager** → Server: OLUWATOSIN\SQLEXPRESS → Database: EntrovaDB → Windows Authentication
2. **Test Connection** → Ensure **Connection Successful**

**Screenshot for Evidence:**

* OLE DB Connection Manager configuration showing server, database, and test success.

**Step 3: Data Flow & Transformations**

**Objective:** Apply business logic and compute metrics.

**3.1 Users Table**

* **Derived Column:** parse signup\_date → signup\_date\_parsed; fill missing country:
* ISNULL(country) ? "Unknown" : country
* **Aggregate:** daily user counts grouped by signup\_date\_parsed
  + Output: daily\_users
  + Optional: cumulative users via Script Component or post-load SQL

**3.2 Demo Sessions Table**

* **Aggregate:** sum attended\_demo grouped by demo\_date → output total\_attended\_demo

**3.3 Engagement Table**

* **Derived Columns:** create binary flags for metrics:
* active\_user = logins\_last\_30\_days > 0 ? 1 : 0
* sent\_messages = messages\_sent > 0 ? 1 : 0
* profile\_done = profile\_completed > 0 ? 1 : 0
* **Lookup Transformation:** link user\_id to users table → bring in signup\_date
* **Aggregate:** sum metrics by signup\_date → output columns:
  + active\_user\_sum
  + messages\_sent\_sum
  + profile\_done\_sum

**Screenshot for Evidence:**

* Data Flow showing Derived Column and Aggregate transformations with example expressions.
* Highlight Engagement flags (active\_user, messages\_sent, profile\_done).

**Step 4: Destination / Loading**

**Objective:** Load transformed metrics into analytics schema tables.

* **OLE DB Destination for each table:**

| **Table** | **Destination** |
| --- | --- |
| Users | analytics.user\_cumulative\_signups |
| Demo Sessions | analytics.demo\_attendance\_summary |
| Engagement | analytics.engagement\_summary |
| Optional ML | analytics.ml\_demo\_dataset |

**Steps & Clicks:**

1. Drag OLE DB Destination → connect from last Aggregate
2. Double-click → select OLE DB Connection → Test Connection
3. Choose existing table or click **New** → SSIS generates CREATE TABLE script
4. Map Input Columns → Destination Columns
5. Data Access Mode → **Table or View – Fast Load**

**Screenshot for Evidence:**

* OLE DB Destination configuration showing mappings and Fast Load selection.

**Step 5: Logging & Validation**

**Objective:** Capture execution logs and row counts.

**5.1 Logging**

* Enable logging → Control Flow → SSIS → Logging
* Provider: SSIS log provider for Text files or SQL Server
* Events: OnError, OnWarning, OnInformation, OnPreExecute, OnPostExecute

**5.2 Row Counts**

* Drag **Row Count Transformation** between Aggregate → OLE DB Destination
* Assign **variables** to store counts (e.g., UserRowCount)
* Optional → Execute SQL Task → insert counts into staging.row\_counts table

**5.3 Data Viewer**

* Enable on green arrows → see live rows moving through transformations

**Screenshot for Evidence:**

* Logging configuration window
* Data Viewer showing live rows
* Sample log output with row counts and success status

**Step 6: Execution & Validation**

* Execute Package (F5) → monitor **Progress tab**
* Check **Execution Results** → all tasks succeeded
* Validate **Destination Tables in SQL Server** → row counts match expectations

**Screenshot for Evidence:**

* Execution Results tab showing all tasks succeeded
* Destination table in SSMS showing aggregated rows

**Step 7: Evidence-Ready Workflow Diagram**

[Users Source] [Demo Source] [Engagement Source]

| | |

v v v

[Derived Columns] [Aggregate] [Derived + Lookup]

| | |

v v v

[Aggregate] [Aggregate] [Aggregate by signup\_date]

| | |

---------------------------------------------

|

v

[OLE DB Destinations – analytics schema]

- user\_cumulative\_signups

- demo\_attendance\_summary

- engagement\_summary

- ml\_demo\_dataset (optional)

|

v

[Logging & Row Counts]

✅ **Outcome:**

* ETL package is fully automated
* Metrics aggregated and loaded into analytics schema
* Row counts and logs captured for Tech Nation evidence
* Screenshots ready: Control Flow, Data Flow, Destination mappings, Logging output, SQL Server tables