**Udemy DWH ETL Documentation (SSIS)**

**Project:** UdemyDWH  
**Tool:** SQL Server Integration Services (SSIS)  
**Environment:** Visual Studio 2019, SQL Server 2019  
**Author:** [Your Name]  
**Date:** [Today’s Date]

**1. Introduction**

This document describes the **complete Extract, Transform, and Load (ETL)** processes implemented using **SSIS** for the **UdemyDWH** project. The ETL loads, transforms, and prepares data from the operational OLTP systems into the Data Warehouse (DWH) following a star schema model.

The objective of this ETL is to ensure the Data Warehouse reflects accurate, timely, and historical data for reporting and analytical purposes.

Key aspects of the ETL include:

* Ensuring **data type consistency**.
* Managing **Slowly Changing Dimensions (SCD)** to preserve history.
* Implementing **audit metadata** (start/end dates, current record flags).
* Handling **errors and exceptions** during transformation.
* Preparing **Fact and Dimension tables** to support business intelligence.

**2. High Level Process Overview**

The SSIS ETL process follows these major steps:

1. Extract source data from transactional databases via OLE DB Source.
2. Apply necessary transformations: cleansing, type conversion, derived columns.
3. Manage SCD Type 2 for dimension tables.
4. Insert or update data in dimension and fact tables.
5. Handle failure cases and errors gracefully with logging.

**3. Detailed ETL Data Flow per Table**

**3.1 Dimension: DimUsers**

**Purpose:**  
Load student and instructor information into the DimUsers dimension table, maintaining historical changes (e.g., if a user's country, occupation, or social presence changes).

**Step-by-Step:**

| **Step** | **SSIS Component** | **Detailed Description** |
| --- | --- | --- |
| 1 | **OLE DB Source** | Connect to the OLTP database and execute a SELECT query to pull user profile data (fields: UserID, FullName, Bio, WalletAmount, Country, Email, SocialLinks, etc.). |
| 2 | **Data Conversion** | Some fields are cast into proper warehouse-compatible types, e.g.:  - FullName: NVARCHAR(255) → Unicode string  - WalletAmount: MONEY → DECIMAL(18,2)  - CreatedDate: datetime → DT\_DBTIMESTAMP |
| 3 | **Slowly Changing Dimension** | The core component where:  - **Business Key**: UserID\_BK is matched.  - **Changing Attributes**: Country, Occupation, WalletAmount, HasFacebook, HasInstagram, etc. - When changes are detected:  ➔ Current record’s EndDate is updated to current timestamp and IsDeleted = 1  ➔ New record inserted with updated attributes. |
| 4 | **Derived Column Transformation** | New columns are added:  - StartDate = GETDATE() (current ETL execution date)  - EndDate = NULL (because it is the current record)  - IsDeleted = 0 |
| 5 | **OLE DB Command (Historical Inserts Output)** | Executes an UPDATE SQL command to close previous active records by updating EndDate and setting IsDeleted = 1 for records where the user has changed. |
| 6 | **Union All** | Joins all new and changed rows into a single unified output to insert into the warehouse. |
| 7 | **Final Derived Column** | Ensure any null values are replaced (e.g., empty email addresses get a placeholder like 'unknown@domain.com'). |
| 8 | **OLE DB Destination** | Load data into DimUsers. Mapped precisely to columns. |

**Important Field Handling:**

| **Column** | **Special Treatment** |
| --- | --- |
| WalletAmount | Converted to 2 decimal places. |
| SocialLinks | Nulls replaced by 'N/A'. |
| Passwords / Sensitive info | Excluded from warehouse by SELECT projection. |

**3.2 Dimension: DimCourses**

**Purpose:**  
Load course information into DimCourses, managing price, language, title changes over time.

**Step-by-Step:**

| **Step** | **SSIS Component** | **Detailed Description** |
| --- | --- | --- |
| 1 | **OLE DB Source** | SELECT course data (CourseID, Title, Price, Level, Rating, Language). |
| 2 | **Data Conversion** | Ensure fields like Title (nvarchar) and Price (decimal) are in correct format. |
| 3 | **Slowly Changing Dimension** | - Business Key: CourseID\_BK - Change Attributes: Price, Level, Language, Rating. When detected, an old version is closed and a new one inserted (SCD2). |
| 4 | **Derived Column** | Add StartDate, EndDate, IsDeleted. |
| 5 | **OLE DB Destination** | Insert transformed data into DimCourses. |

**3.3 Sub Dimension: SDimQuiz**

**Purpose:**  
Track quizzes associated with courses, including their question types and numbers.

**Key Steps:**

* Extract quiz metadata.
* Track changes in number of MCQ vs. True/False questions.
* Apply SCD2 logic.
* Maintain audit fields.

**3.4 Sub Dimension: SubDimSection**

**Purpose:**  
Load and track sections/modules within courses.

| **Step** | **SSIS Component** | **Detailed Description** |
| --- | --- | --- |
| 1 | **OLE DB Source** | Pull sections data. |
| 2 | **SCD Transformation** | Detect changes in section title, description, content duration. |
| 3 | **Derived Column** | Add ETL audit columns. |
| 4 | **OLE DB Destination** | Insert into SubDimSection. |

**3.5 Sub Dimension: SubDimCrsReq**

**Purpose:**  
Load course pre-requisites and requirements.

| **Step** | **SSIS Component** | **Detailed Description** |
| --- | --- | --- |
| 1 | **OLE DB Source** | SELECT Requirements data. |
| 2 | **Direct Insert** | No SCD logic, simple overwrite. Insert into SubDimCrsReq. |

**3.6 Fact Table: FactOrder**

**Purpose:**  
Record every course purchase made by a user.

| **Step** | **SSIS Component** | **Detailed Description** |
| --- | --- | --- |
| 1 | **OLE DB Source** | Extract orders: (OrderID, UserID, CourseID, OrderDate, OrderAmount). |
| 2 | **Derived Column** | Create surrogate DateKey from OrderDate (e.g., 20250427). |
| 3 | **Lookups** | Validate foreign keys: - UserKey (DimUsers) - CourseKey (DimCourses) - DateKey (DimDate) |
| 4 | **OLE DB Destination** | Insert validated records into FactOrder. |

**3.7 Fact Table: FactEnrollment**

**Purpose:**  
Track enrollment activities including progress and grades.

| **Step** | **SSIS Component** | **Detailed Description** |
| --- | --- | --- |
| 1 | **OLE DB Source** | Extract enrollment (EnrollmentID, UserID, CourseID, EnrollmentDate, Progress%, Grade). |
| 2 | **Derived Column** | Create surrogate DateKey. |
| 3 | **Lookups** | Validate foreign key relationships. |
| 4 | **OLE DB Destination** | Load into FactEnrollment. |

**4. Error Handling Strategy**

* **Row-level Errors**: Logged using SSIS Event Handlers.
* **Data Type Errors**: Caught at Data Conversion stage and redirected to error outputs.
* **Business Rule Failures**: e.g., missing foreign keys → rows are quarantined into an error staging table.

**5. Logging and Monitoring**

* SSIS Logging enabled at Package and Data Flow levels.
* Logs stored in SQL Server tables.
* Captures: Start Time, End Time, Errors, Row Counts.

**6. Scheduling**

* Deployed to SSISDB Catalog.
* Scheduled daily via SQL Server Agent.
* Configured with retries and email notifications on failure