**Data Vault Lab**

**Lab 1: Create Hub, Link, and Satellite Tables**

**Objective:**

Design and implement Hub, Link, and Satellite tables for a simple business scenario.

**Steps:**

1. **Identify Business Keys:**
   * Choose entities (e.g., Customer, Product).
   * Identify unique business keys for each entity.
2. **Create Hub Tables:**
   * Define columns: surrogate hash key, business key(s), load date, record source, and audit columns.
   * Generate hash keys from business keys (e.g., MD5 hash).
3. **Create Link Tables:**
   * Identify relationships between hubs (e.g., Customer-Product purchases).
   * Create link tables with composite hash keys from related hubs.
   * Include load date, record source, and audit columns.
4. **Create Satellite Tables:**
   * For each hub and link, identify descriptive attributes.
   * Create satellites to hold these attributes with load timestamp, record source, and hash diff key for change detection.
5. **Load Sample Data:**
   * Insert sample business keys and descriptive data.
   * Use SQL or ETL tools to calculate hash keys and load audit columns.

**Lab 2: Hash Key Generation and Collision Testing**

**Objective:**

Generate hash keys for business keys and test for collisions.

**Steps:**

1. **Choose Hash Algorithm:**
   * Use MD5, SHA-1, or SHA-256.
2. **Generate Hash Keys:**
   * Concatenate business key columns with delimiters.
   * Apply hash function to generate fixed-length key.
3. **Test Uniqueness:**
   * Load a large sample dataset.
   * Query for duplicate hash keys.
4. **Analyze Collisions:**
   * If duplicates found, review data and hash concatenation method.
   * Modify key concatenation (add separators, normalize case) and retest.

**Lab 3: Build PIT (Point-In-Time) Tables**

**Objective:**

Create PIT tables for simplifying historical queries.

**Steps:**

1. **Identify Satellites to Include:**
   * Select satellites related to a hub or link.
2. **Write SQL to Join Satellites:**
   * Join on hash key and load date, selecting the latest satellite record before or at each PIT date.
3. **Create PIT Table or View:**
   * Define columns for the hub/link key plus descriptive attributes from satellites.
   * Populate PIT table with a snapshot of descriptive data per business key at points in time.
4. **Test Queries:**
   * Query PIT table for data as of specific dates and verify accuracy.

**Lab 4: Implement Bridge Tables**

**Objective:**

Create bridge tables to manage many-to-many attribute relationships.

**Steps:**

1. **Identify Multi-Valued Attributes or Complex Relationships:**
   * E.g., Customer may have multiple contact methods.
2. **Create Bridge Table:**
   * Columns: surrogate key, related hub/link keys, attribute key, load date, record source.
3. **Load Data:**
   * Populate with combinations of keys representing multi-valued relationships.
4. **Test Drill-Across Queries:**
   * Write queries joining hubs, links, and bridge to fetch aggregated or filtered results.

**Lab 5: Raw Vault to Business Vault Pipeline**

**Objective:**

Build a basic ETL pipeline that moves data from Raw Vault to Business Vault.

**Steps:**

1. **Extract & Load Raw Vault:**
   * Load raw data into hubs, links, and satellites without transformations.
2. **Develop Automation Macros (or Scripts):**
   * Automate hash key generation and audit column addition.
3. **Create Business Vault Layers:**
   * Add business rules: calculated columns, conformed attributes.
   * Create PIT and bridge tables as required.
4. **Publish to Star Schema:**
   * Use views or tables for downstream reporting.

**Lab 6: Automation Macros Development**

**Objective:**

Write macros/scripts for repetitive Data Vault ETL tasks.

**Steps:**

1. **Create Hash Key Generator:**
   * Input: business keys.
   * Output: hash key using selected algorithm.
2. **Write Change Detection Macro:**
   * Compare current and previous hash diff keys to detect changed records.
3. **Create Audit Column Inserter:**
   * Automatically insert load date, record source, batch ID.
4. **Develop PIT and Bridge Table Builders:**
   * Automate creation and updating of these tables.
5. **Test Each Macro:**
   * Run macros on sample data and validate outputs.

**Lab 7: Zero-Copy Star Schema Publishing**

**Objective:**

Create star schema views on top of Data Vault tables without physical data duplication.

**Steps:**

1. **Identify Fact and Dimension Tables:**
   * Use links as facts and hubs + satellites as dimensions.
2. **Create Views for Dimensions:**
   * Combine hubs and latest satellite records.
3. **Create Fact View:**
   * Join link tables with dimension views.
4. **Test Query Performance:**
   * Run typical BI queries against views.
   * Compare performance with physically copied star schemas if possible.

**Lab 8: Manage Audit Columns**

**Objective:**

Add and utilize audit columns for data governance.

**Steps:**

1. **Define Audit Columns in Each Table:**
   * Load date/timestamp, record source, load end date, batch/run ID.
2. **Modify ETL to Populate Audit Columns:**
   * Ensure these columns are set correctly during data loads.
3. **Use Audit Data for Troubleshooting:**
   * Query data loads by date, source, or batch.
4. **Implement Soft Delete Using Load End Date:**
   * Mark satellite records as expired rather than deleting physically.