**🧾 Full Procedure (Aligned with Your Flowchart and Our Conversation)**

**🔷 1. Start**

This marks the beginning of the database project.

**🔷 2. Create Entity-Relationship (ER) Model**

* Analyzed the dataset.
* Identified:
  + Strong entities: Drug, Disease, Product, ClinicalTrial, Researcher, Institution
  + Weak/multivalued attributes: side effects, interacts\_with
  + Relationships: Treats, HasSideEffect, TestedIn, Makes, LedBy, etc.
* Designed ER diagram in draw.io based on normalized structure.
* Discussed total/partial participation, weak vs strong entities, cardinality.

**🔷 3. Convert Excel to CSV**

* You prepared the raw Excel dataset with columns like:
  + Drug, Disease, Side Effects, Interactions, Trial Info, Researcher, Address
* Saved the Excel file as .csv for Python processing.

**🔷 4. Import Data from CSV Using Python**

* Used pandas and oracledb to load CSV:

python

CopyEdit

import pandas as pd

import oracledb

* Cleaned and mapped:
  + drug\_id, product\_id, country\_id using enumerate() with unique()
* Inserted rows into DRUGS\_FULL master table via SQL loop.

**🔷 5. Create Master Flat Table: DRUGS\_FULL**

* Contained all columns (wide schema): drugs, diseases, trials, researchers, side effects, conditions, etc.
* Temporarily acted as the central source for normalization.

**🔷 6. Create Schema**

* Created all normalized entity tables:
  + Drug, Disease, Product, SideEffect, Researcher, Institution, ClinicalTrial
* Created all relationship tables:
  + Treats, HasSideEffect, TestedIn, Makes, StudiedIn, etc.

**🔷 7. Insert Normalized Data**

* Used INSERT INTO ... SELECT DISTINCT ... FROM DRUGS\_FULL pattern.
* Handled multivalued attributes via UNION ALL (e.g. side effects, interacts\_with).
* Assigned foreign keys by matching drug\_name + drug\_category.

**🔷 8. Update Entity Tables**

* Cleaned and deduplicated data in:
  + Drug: ensured unique drug\_id per drug name/category
  + Product: linked to drug\_id
* Updated ClinicalTrial:
  + Added researcher\_name column
  + Added country\_id column or used institution for address-based mapping

**🔷 9. Handle Issues (Yes/No Loop)**

* Fixed:
  + ORA-00001 duplicate key violations
  + ORA-00942 missing tables
  + drug\_id mismatches between DRUGS\_FULL and normalized tables
* Ensured consistent ID generation in Python (no re-generation on rerun)

**🔷 10. Insert Normalized Data (Again)**

* Re-ran INSERT INTO for:
  + Treats, HasSideEffect, TestedIn, etc.
* Ensured joins used TRIM, LOWER, and drug\_name + category keys

**🔷 11. Update Table Structures**

* ALTER TABLE to:
  + Add country\_id and researcher\_name to ClinicalTrial
  + Add constraints (PKs, FKs)
* Added foreign key from ClinicalTrial.institute\_name → Institution

**🔷 12. Generate Answers for Query Questions**

You built and fixed advanced SQL queries for:

| **Query #** | **Description** |
| --- | --- |
| (a–d) | Side effects, drug interactions, disease treatments |
| (e) | Most common treatment for immunological but not hematological |
| (f) | Diseases treated by hydrocortisone but not etanercept |
| (g) | Side effects for asthma-treating drugs |
| (h) | Drugs in >3 trials with >30 participants |
| (i) | Drugs active in overlapping trials (peak activity) |
| (j) | Researchers for respiratory + cardiovascular drugs |
| (k) | Top 3 such researchers by trial count |
| (l) | Drug categories studied only in U.S.-based trials |

**🔷 13. Query Output Correct? (Yes/No Decision)**

* If output was wrong:
  + Rebuilt joins (esp. on drug\_id)
  + Re-inserted into HasSideEffect, Treats, etc.
* If correct:
  + Moved to next question and stored result

**🔷 14. Show Output Tables**

* Retrieved and displayed clean query results from normalized schema
* Verified against expected values from Excel (DRUGS\_FULL)

**🔷 15. Draw ER Diagram**

* Used flow understanding to finalize ER diagram (already created earlier)

**🔷 16. End**

* Flow is complete when:
  + Schema is finalized
  + Data is correctly normalized
  + All complex queries return verified output

**✅ Summary Flow:**

text

CopyEdit

Start

→ Create ERD

→ Convert Excel → CSV → Python Insert → DRUGS\_FULL

→ Create Tables

→ Normalize & Insert Data

→ Fix Issues → Update Tables → Query & Verify

→ Draw ERD + Show Output

→ End