Course 2 Module 5 Programming Assignment

Assignment is to ETL MIMIC data into the OMOP CONDITION_OCCURRENCE table

Detailed instructions with Slide Notes

Assignment is to ETL MIMIC data into the OMOP CONDITION_OCCURRENCE table

ETL Steps

1. Understand source/target data models
2. Profile source tables
3. Create ETL mappings
4. Write transformation code
5. Execute transformation
6. Perform data quality assessment
7. Package documentation

Step 1: Understand source/target data models

CONDITION_OCCURRENCE is the TARGET OMOP table.

Read the OMOP documentation about the type of data stored in CONDITION_OCCURRENCE and for three fields below that are in that table:
- **person_id**
- **visit_occurrence_id**
- **condition_source_value**

Table Details: condition_occurrence

Schema	Details	Preview		
condition_occurrence_id	FLOAT	NULLABLE	int64
person_id	FLOAT	NULLABLE	int64
condition_concept_id	FLOAT	NULLABLE	int64
condition_start_date	STRING	NULLABLE	parse_date()
condition_start_datetime	STRING	NULLABLE	parse_datetime()
condition_end_date	STRING	NULLABLE	parse_date()
condition_end_datetime	STRING	NULLABLE	parse_datetime()
condition_type_concept_id	FLOAT	NULLABLE	int64
stop_reason	STRING	NULLABLE	Describe this field...
provider_id	FLOAT	NULLABLE	int64
visit_occurrence_id	FLOAT	NULLABLE	int64
visit_detail_id	FLOAT	NULLABLE	int64
condition_source_value	STRING	NULLABLE	Describe this field...
condition_source_concept_id	FLOAT	NULLABLE	int64
condition_status_source_value	STRING	NULLABLE	Describe this field...
condition_status_concept_id	FLOAT	NULLABLE	int64

Step 1: My answer

	A	B	C	D	E	F	G	H
1	CONDITION_OCCURENCE_TABLE			ADMISSIONS.csv	DIAGNOSES_ICD.csv	D_ICD_DIAGNOSES.csv	CAREGIVERS.csv	PROCEDUREEVENTS_MV.csv
2	condition_occurrence_id	INTEGER	NULLABLE					
3	person_id	INTEGER	NULLABLE	SUBJECT_ID	SUBJECT_ID			SUBJECT_ID
4	condition_concept_id	INTEGER	NULLABLE					
5	condition_start_date	DATE	NULLABLE	ADMITTIME				
6	condition_start_datetime	DATETIME	NULLABLE	ADMITTIME				
7	condition_end_date	DATE	NULLABLE	DISCHTIME				
8	condition_end_datetime	DATETIME	NULLABLE	DISCHTIME				
9	condition_type_concept_id	INTEGER	NULLABLE					
10	stop_reason	STRING	NULLABLE					
11	provider_id	INTEGER	NULLABLE				CGID	CGID
12	visit_occurrence_id	INTEGER	NULLABLE	HADM_ID				HADM_ID
13	visit_detail_id	INTEGER	NULLABLE					
14	condition_source_value	STRING	NULLABLE	DIAGNOSIS		LONG_TITLE		
15	condition_source_concept_id	INTEGER	NULLABLE		ICD9_CODE	ICD9_CODE		
16	condition_status_source_value	STRING	NULLABLE					
17	condition_status_concept_id	INTEGER	NULLABLE					

Step 2: Profile source table or tables

Using the White Rabbit profiling data from the 100 patient MIMIC database provided in the Assessment to comment on the distribution of the SUBJECT_ID field from one of the MIMIC tables selected in Step 1
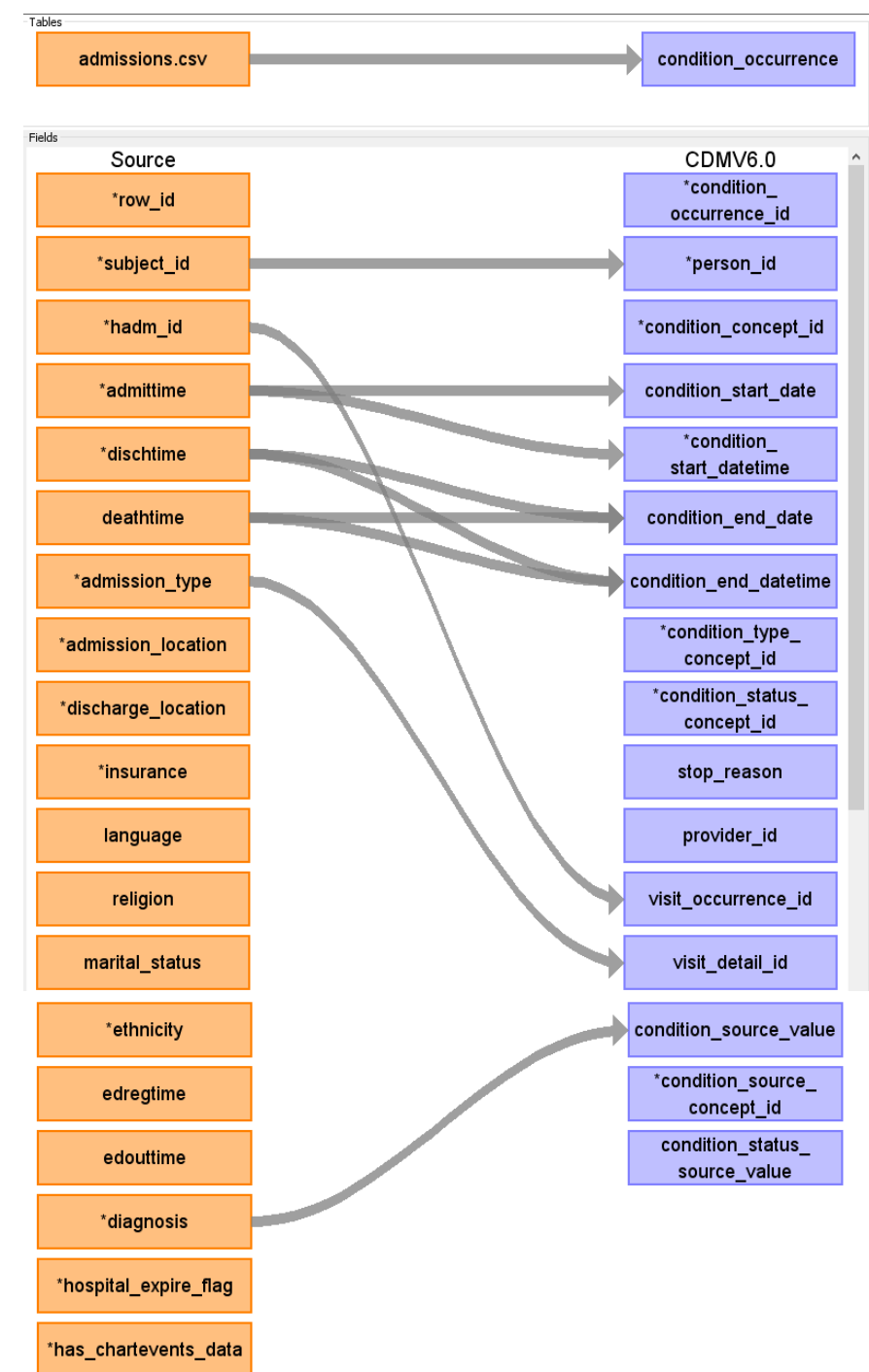
- MIMIC TableName
  - See rubric for the types of topics to include here

Step 3: My answer

subject_id map to person_id as both values are describing people

hadm_id map to visit_occurrence_id as both values related to hospital visit

diagnosis map to condition_source _value as the field described as 'Condition as it appears in the source data'

Step 4: My answer

```sql
WITH co_1 as (SELECT distinct ma.subject_id as person_id
              FROM `learnclinicaldatascience.mimic3_demo.ADMISSIONS` ma),
     co_2 as (SELECT co_1.person_id, ma.hadm_id as visit_occurence_id
              FROM co_1 JOIN `learnclinicaldatascience.mimic3_demo.ADMISSIONS` ma
              ON co_1.person_id = ma.SUBJECT_ID),
     co_3 as (SELECT co_2.person_id, co_2.visit_occurence_id, dicd.ICD9_CODE as condition_source_concept_id
              FROM co_2 JOIN `learnclinicaldatascience.mimic3_demo.DIAGNOSES_ICD` dicd
              ON co_2.visit_occurence_id = dicd.HADM_ID),
     CONDITION_OCCURRENCE as (SELECT co_3.person_id, co_3.visit_occurence_id, dict.LONG_TITLE as condition_source_value
              FROM co_3 JOIN `learnclinicaldatascience.mimic3_demo.D_ICD_DIAGNOSES` dict
              ON dict.ICD9_CODE = co_3.condition_source_concept_id)
SELECT * FROM CONDITION_OCCURRENCE
```

Step 5: Execute transformation code

Execute the ETL code from Step 4 but do not submit the output table.
Use the output table for Step 6.

There is no submission for this Step.

Step 6: My answer

Query results   📥 SAVE RESULTS   📊 EXPLORE DATA ▼

Query complete (1.1 sec elapsed, 927.3 KB processed)

Job information   **Results**   JSON   Execution details

Row	person_id	visit_occurence_id	condition_source_value
1	10043	168674	Pneumonia, organism unspecified
2	10043	168674	Intestinal infection due to Clostridium difficile
3	10043	168674	Thrombocytopenia, unspecified
4	10043	168674	Anemia of other chronic disease
5	10043	168674	Obstructive chronic bronchitis with (acute) exacerbation
6	10043	168674	Acute respiratory failure
7	10043	168674	Acute diastolic heart failure
8	10043	168674	Congestive heart failure, unspecified
9	10094	168074	Diabetes mellitus without mention of complication, type II or unspecified type, not stated as uncontrolled
10	10094	168074	Diseases of tricuspid valve
11	10094	168074	Benign neoplasm of pituitary gland and craniopharyngeal duct
12	10094	168074	Congestive heart failure, unspecified
13	10094	168074	Pneumonia, organism unspecified
14	10094	168074	Pressure ulcer, ankle

Rows per page: 100 ▼   1 - 100 of 1716   First page |<   <   >   >| Last page

Step 7: Package documentation

- Congratulations! The materials in the previous slides constitute a complete ETL package.

There is no submission for this Step.