

RAW.SAMPLEU.DEPARTMENTcreate database if not exists analytics;

create database if not exists raw;

--create schemas

create schema if not exists analytics.sampleU;

create schema if not exists raw.sampleU;

create schema if not exists raw.conformed;

-- define file formats

create or replace file format RAW.PUBLIC.PARQUET

TYPE = parquet

REPLACE_INVALID_CHARACTERS = TRUE;

create or replace file format RAW.PUBLIC.JSONARRAY

TYPE = json

STRIP_OUTER_ARRAY = TRUE;

create or replace file format RAW.PUBLIC.JSON

TYPE = json

STRIP_OUTER_ARRAY = FALSE;

create or replace file format RAW.PUBLIC.CSVHEADER

TYPE = 'csv'

FIELD_DELIMITER = ','

SKIP_HEADER=1;

create or replace file format RAW.PUBLIC.CSV

TYPE = csv

FIELD_DELIMITER = ','

```
PARSE_HEADER = FALSE

SKIP_HEADER = 0;

-- create stages
-- varying file formats
CREATE or replace STAGE RAW.PUBLIC.externalworld_files
  URL = 'azure://externalworld.blob.core.windows.net/files/';

-- these are all parquet file formats
CREATE or replace STAGE RAW.PUBLIC.externalworld_database
  URL = 'azure://externalworld.blob.core.windows.net/database/'
  FILE_FORMAT = RAW.PUBLIC.PARQUET ;

-- stage the date dimension
CREATE or REPLACE TABLE raw.conformed.datedimension (
  datekey int
  ,date date
  ,datetime timestamp
  ,year int
  ,quarter int
  ,quartername varchar(2)
  ,month int
  ,monthname varchar(3)
  ,day int
  ,dayofweek int
  ,dayname varchar(3)
  ,weekday varchar(1)
  ,weekofyear int
```

```

,dayofyear int
) AS
WITH CTE_MY_DATE AS (
SELECT DATEADD(DAY, SEQ4(), '1900-01-01 00:00:00') AS MY_DATE
FROM TABLE(GENERATOR(ROWCOUNT=>365*30))
)
SELECT
REPLACE(TO_DATE(MY_DATE)::varchar, '-', '')::int as datekey,
TO_DATE(MY_DATE) as date
,TO_TIMESTAMP(MY_DATE) as datetime
,YEAR(MY_DATE) as year
,QUARTER(MY_DATE) as quarter
,CONCAT('Q', QUARTER(MY_DATE)::varchar) as quartername
,MONTH(MY_DATE) as month
,MONTHNAME(MY_DATE) as monthname
,DAY(MY_DATE) as day
,DAYOFWEEK(MY_DATE) as dayofweek
,DAYNAME(MY_DATE) as dayname
,case when DAYOFWEEK(MY_DATE) between 1 and 5 then 'Y' else 'N' end as weekday
,WEEKOFYEAR(MY_DATE) as weekofyear
,DAYOFYEAR(MY_DATE) as dayofyear
FROM CTE_MY_DATE
;

```

create or replace table RAW.sampleU.Course

```

(
CourseID int,
    Title varchar,
    Credits int,

```

```

        DepartmentID int
    );

copy into RAW.sampleU.Course
    from '@RAW.PUBLIC.externalworld_database/sampleu.course.parquet'
    MATCH_BY_COLUMN_NAME='CASE_INSENSITIVE';


create or replace table RAW.sampleU.CourseInstructor
(
    CourseID int,
    PersonID int
);

copy into RAW.sampleU.CourseInstructor
    FROM '@RAW.PUBLIC.externalworld_database/sampleu.courseinstructor.parquet'
    MATCH_BY_COLUMN_NAME='CASE_INSENSITIVE';


CREATE OR REPLACE TABLE RAW.sampleU.OfficeAssignment
(
    InstructorID int,
    Location varchar,
    TimestampVariant VARIANT
)
AS
SELECT
    $1:InstructorID::int,
    $1:Location::varchar,
    $1:Timestamp
FROM '@RAW.PUBLIC.externalworld_database/sampleu.officeassignment.parquet'

```

```
(FILE_FORMAT => RAW.PUBLIC.PARQUET);
```

```
create or replace table RAW.sampleU.Department
```

```
(
```

```
DepartmentID int,
```

```
    Name varchar,
```

```
    Budget DECIMAL(18, 2),
```

```
    StartDate string,
```

```
    Administrator int
```

```
);
```

```
copy into RAW.sampleU.Department
```

```
    FROM '@RAW.PUBLIC.externalworld_database/sampleu.department.parquet'
```

```
    MATCH_BY_COLUMN_NAME='CASE_INSENSITIVE';
```

```
create or replace table RAW.sampleU.OnlineCourse
```

```
(
```

```
CourseID int,
```

```
    URL varchar
```

```
);
```

```
copy into RAW.sampleU.OnlineCourse
```

```
    FROM '@RAW.PUBLIC.externalworld_database/sampleu.onlinecourse.parquet'
```

```
    MATCH_BY_COLUMN_NAME='CASE_INSENSITIVE';
```

```
create or replace table RAW.sampleU.OnsiteCourse
```

```
(
```

```
    CourseID int,
```

```
        Location varchar,  
        Days varchar,  
        Time string  
    );  
  
COPY INTO RAW.sampleU.OnsiteCourse  
FROM '@RAW.PUBLIC.externalworld_database/sampleu.onsitecourse.parquet'  
MATCH_BY_COLUMN_NAME='CASE_INSENSITIVE';  
  
update raw.sampleu.department set startdate = TO_DATE(startdate);
```

```
create or replace table RAW.sampleU.Person
```

```
(  
    PersonID int,  
    LastName varchar,  
    FirstName varchar,  
    HireDate string,  
    EnrollmentDate string,  
    Discriminator varchar  
);
```

```
copy into RAW.sampleU.Person
```

```
FROM '@RAW.PUBLIC.externalworld_database/sampleu.person.parquet'  
MATCH_BY_COLUMN_NAME='CASE_INSENSITIVE';
```

```
update raw.sampleu.person set HireDate = TO_DATE(startdate);  
update raw.sampleu.person set EnrollmentDate = TO_DATE(EnrollmentDate);
```

```
create or replace table RAW.sampleU.StudentGrade
```

```
(  
    EnrollmentID int,
```

```
        CourseID int,
        StudentID int,
        Grade decimal(3, 2)
    );
copy into RAW.sampleU.StudentGrade
    FROM '@RAW.PUBLIC.externalworld_database/sampleu.studentgrade.parquet'
    MATCH_BY_COLUMN_NAME='CASE_INSENSITIVE';
```

```
create or replace table RAW.sampleU.CourseEvaluation
(
    Course_ID int,
    Rating int,
    Evaluation varchar

);
```

```
COPY INTO "RAW"."SAMPLEU"."COURSEEVALUATION"
FROM '@"RAW"."SAMPLEU"."%COURSEEVALUATION"/__snowflake_temp_import_files__/'
FILES = ('course-evaluations1.json')
FILE_FORMAT = (
    TYPE=JSON,
    STRIP_OUTER_ARRAY=TRUE,
    REPLACE_INVALID_CHARACTERS=TRUE,
    DATE_FORMAT=AUTO,
    TIME_FORMAT=AUTO,
    TIMESTAMP_FORMAT=AUTO
```

)

MATCH_BY_COLUMN_NAME=CASE_INSENSITIVE

update raw.sampleu.department set startdate = TO_DATE(startdate);

ALTER TABLE RAW.sampleU.CourseEvaluation

ADD COLUMN Sentiment VARCHAR;