Database Creation/CSV import Errors I encountered:

```
pop-os% psql -U postgres -d jugos_rodier < /home/srodier/repos/jugos_rodier/database_design/import_csvs.sql

ERROR:  syntax error at or near "USE"
LINE 1: USE jugos_rodier;
        ^
ERROR:  extra data after last expected column
CONTEXT:  COPY customer, line 2: "0, Addison, Riley,9816 Rosewood Drive,,93304,Bakersfield,CA,USA"
ERROR:  column "ingred_id" of relation "inventory" does not exist
ERROR:  relation "menu_item" does not exist
ERROR:  column "recipe_id" of relation "recipe" does not exist
ERROR:  column "trans_id" of relation "transaction_item" does not exist
ERROR:  column "trans_id" of relation "transaction" does not exist
```

(I fixed some of the above errors before deciding to document errors)

- **USE jugos_rodier;**
  - Use isn't a command for psql.
  - Should be changed to '\c jugos_rodier;', which is the command for selecting a database. - RESOLVED
- **Extra data after last expected column**
  - scrolled through customer csv in VS code and found that there was some trailing whitespace after any customer with first name Reagan or last name Yu – DID NOT RESOLVE
  - found that the import_csvs.sql file was missing the 'customer_id' column in its code.

```
COPY customer( first_name, last_name, address, sec_address, zipcode, city, state, country)
FROM '/home/srodier/repos/jugos_rodier/csvs/customer_info.csv'
DELIMITER ','
CSV HEADER;
```

  - RESOLVED (but then a new error appeared:)
  - **missing data for column "country"**

```
ERROR:  missing data for column "country"
CONTEXT:  COPY customer, line 140: "138,Raegan, Mayo,7349 Acorn Avenue,93308,Bakersfield,CA,USA"
```

    - This one was easy, found that the second comma was missing after address in the line specified. The second comma is for the second address. No information should be put there so that the field can be counted as NULL. Since it was missing, it took the zipcode as what was supposed to be the second address and kept moving down the line, incorrectly interpreting information. In the end, it thought that the country field was the state. I added comma via VS Code.
- Once customer_info data was successfully imported, psql output 'COPY 2500', which is the number of rows in this dataset.

- **Column "ingred_id" of relation "inventory" does not exist**
  - In csv file, the header was inventory_id. I changed this to "ingred_id" as it expects to see. - DID NOT RESOLVE
  - The import_csvs.sql file has listed ingred_id, however, the create_tables.sql file specifies that the column is inventory_id. I also undid the first change above so that the column ingred_id in the csv was renamed inventory_id so that all items matched. -RESOLVED
- **duplicate key value violates unique constraint "item_item_name_key":**
  - create_tables specifies that item_name should be unique values, but this is not the case in the csv because almost all items have duplicate rows, except different size of items. I must alter create_tables.sql in order to fix this. I will first drop all tables, remove unique constraint for item_name, and rerun create_tables.csv. - RESOLVED
- **insert or update on table "recipe" violates foreign key constraint "fk_inventory_id"**
  - My first thought was that recipe violated foreign key constraint because inventory_id has the UNIQUE constraint added to it. Some research revealed that the unique constraint is not inherited unless specified. - NOT RESOLVED

- Asking google about foreign key constraints revealed that the values in the foreign key MUST match the reference. I will check the table to make sure that they match!
- Found that inventory(inventory_id)s values are all a string 'inv' followed by an integer. recipe(inventory_id)s values are all a string 'ing' followed by an integer. I remember changing this in the csv because I changed from using the term 'ingredient' to 'inventory'. I will use python to make this quick change! - RESOLVED (and revealed new error:)
- **ERROR: insert or update on table "recipe" violates foreign key constraint "fk_item_id1"**
  DETAIL: Key (recipe_id)=(04fry01) is not present in table "item".
  - This is because I again modified the inventory csv but didn't modify other tables. I did this because the schema used to create recipe_id starts with a 2 digit number that represents item_category. Fries are in the side category, which didn't have its own unique 2 digit ID. Lesson Learned! If I have to touch something, I need to think about what that affects.
  - Items table:

  - Recipe table:

  - altered recipes table - RESOLVED
- **column "trans_id" of relation "transaction_item" does not exist**
  - create_tables.sql lists the column as transaction_id. I changed it to trans_id so that nothing else had to be modified. -RESOLVED (but new issue appeared)
- **column "trans_id" of relation "transaction" does not exist**
  - create_tables.sql lists the column as transaction_id. I changed it to trans_id so that nothing else had to be modified. I also made sure to modify foreign key constraint on transaction_item. - RESOLVED (but new issue appeared)
- **ERROR: insert or update on table "transaction_item" violates foreign key constraint "fk_transaction_id"**
  DETAIL: Key (trans_id)=(trans0) is not present in table "transaction"
  - Trans0 appears in both transaction and transaction_item. My first though is that this error occurred because transaction is the parent to transaction_item, but in the import_csvs.sql file, the data for transaction_item is imported before transaction. I will try reversing the order of the 2 imports. -RESOLVED

All csvs have been successfully imported!