

```
#####
# Tasty Bytes is a fictitious, global food truck network, that is on a mission to serve unique food
# options with high quality items in a safe, convenient and cost effective way. In order to drive
# forward on their mission, Tasty Bytes is beginning to leverage the Snowflake Data Cloud.

# In this Python Worksheet, we will walk through the end to end process required to load a CSV file
# containing Menu specific data that is currently hosted in Blob Storage.

# Please click Run and see details below on what each step it doing. The final output will return
# a Dataframe
#####

#####
# Step 1 - To start, we must first import our Snowpark Package and a few Functions and Types
#####
## Note: You can add more packages by selecting them using the Packages control and then importing them.

import snowflake.snowpark as snowpark
from snowflake.snowpark.functions import col
from snowflake.snowpark.types import StructField, StructType, IntegerType, StringType, VariantType

#####
# Step 2 - Let's now define our Session and Create our Database, Schema and Blob Stage
#####

def main(session: snowpark.Session):
    # Use SQL to create our Tasty Bytes Database
    session.sql('CREATE OR REPLACE DATABASE tasty_bytes_sample_data;').collect()

    # Use SQL to create our Raw POS (Point-of-Sale) Schema
    session.sql('CREATE OR REPLACE SCHEMA tasty_bytes_sample_data.raw_pos;').collect()

    # Use SQL to create our Blob Stage
    session.sql('CREATE OR REPLACE STAGE tasty_bytes_sample_data.public.blob_stage url =
"s3://sfquickstarts/tastybytes/" file_format = (type = csv);').collect()

#####
# Step 3 - Now, we will define the Schema for our CSV file
#####

# Define our Menu Schema
menu_schema = StructType([StructField("menu_id",IntegerType()),\
                             StructField("menu_type_id",IntegerType()),\
                             StructField("menu_type",StringType()),\
                             StructField("truck_brand_name",StringType()),\
                             StructField("menu_item_id",IntegerType()),\
                             StructField("menu_item_name",StringType()),\
                             StructField("item_category",StringType()),\
                             StructField("item_subcategory",StringType()),\
                             StructField("cost_of_goods_usd",IntegerType()),\
                             StructField("sale_price_usd",IntegerType()),\
                             StructField("menu_item_health_metrics_obj",VariantType())])

#####
# Step 4 - Using the Schema, let's Create a Dataframe from the Menu file and Save it as a Table
#####

# Create a Dataframe from our Menu file from our Blob Stage
df_blob_stage_read = session.read.schema(menu_schema).csv('@tasty_bytes_sample_data.public.blob_stage/raw_pos/menu/')

# Save our Dataframe as a Menu table in our Tasty Bytes Database and Raw POS Schema
df_blob_stage_read.write.mode("overwrite").save_as_table("tasty_bytes_sample_data.raw_pos.menu")

#####
# Step 5 - With the table saved, let's create a new, filtered Dataframe and Return the results
#####

# Create a new Dataframe reading from our Menu table and filtering for the Freezing Point brand
df_menu_freezing_point = session.table("tasty_bytes_sample_data.raw_pos.menu").filter(col("truck_brand_name") ==
'Freezing Point')

# return our Dataframe
return df_menu_freezing_point
```