

# Shopper Data Generator

---

Audrey Jo, Carlo Mutuc, Evan Simpson

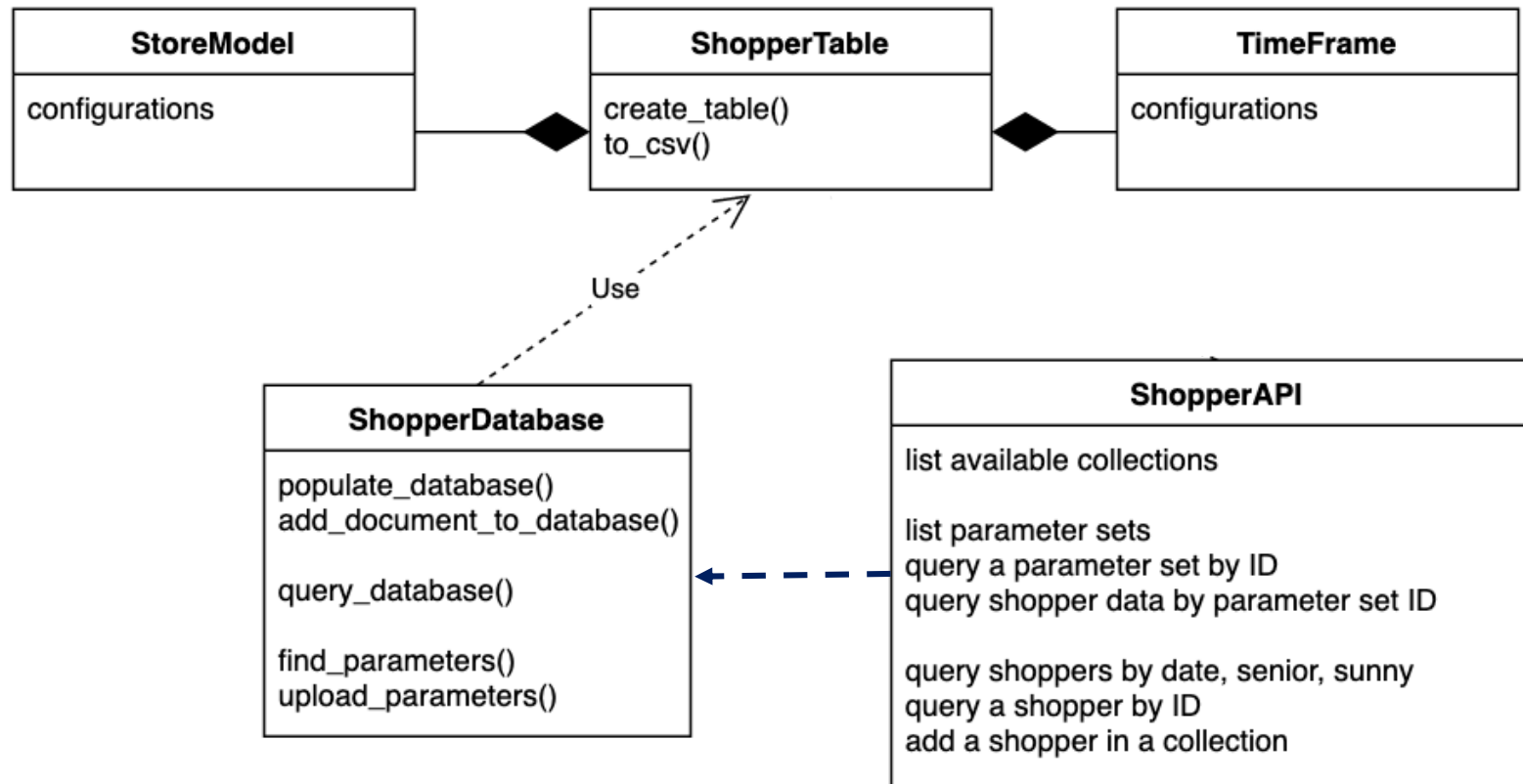
# Project Overview

---

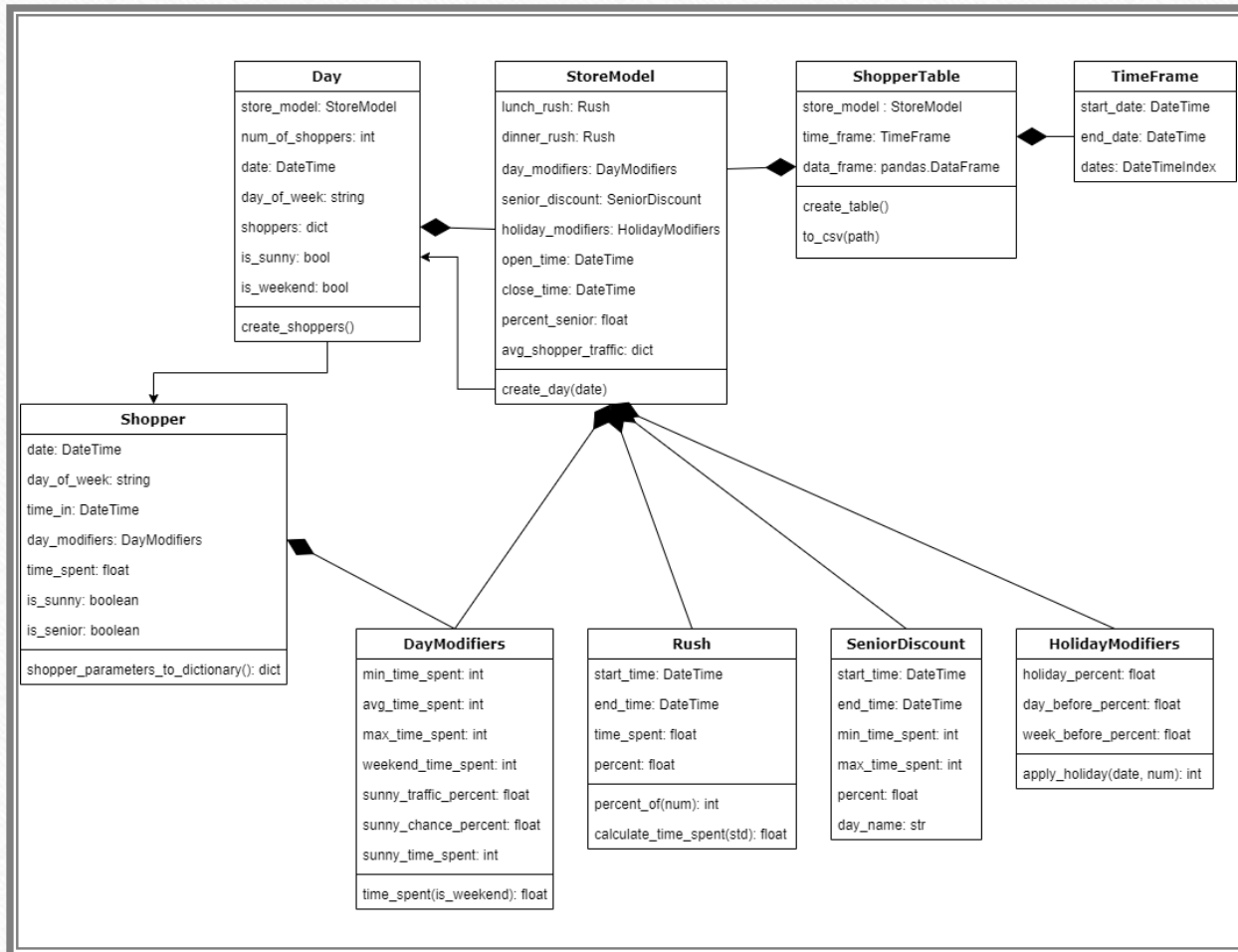
- Language used: Python 3.8
  - Libraries: pandas, pymongo, flask-restx
- 3 main classes, one class per sprint
  - ShopperTable: shopper data generator, create csv
    - Configurations package: holds all configurations for the data generator
  - ShopperDatabase: can populate/query a MongoDB collection
  - ShopperAPI: expose the database queries



# Project Design Overview



# Shopper Generator Design



- Classes to manage input data
- User input captured from command line
- Days create Shoppers using input parameters

## StoreModel

lunch\_rush: Rush

dinner\_rush: Rush

day\_modifiers: DayModifiers

senior\_discount: SeniorDiscount

holiday\_modifiers: HolidayModifiers

open\_time: DateTime

close\_time: DateTime

percent\_senior: float

avg\_shopper\_traffic: dict

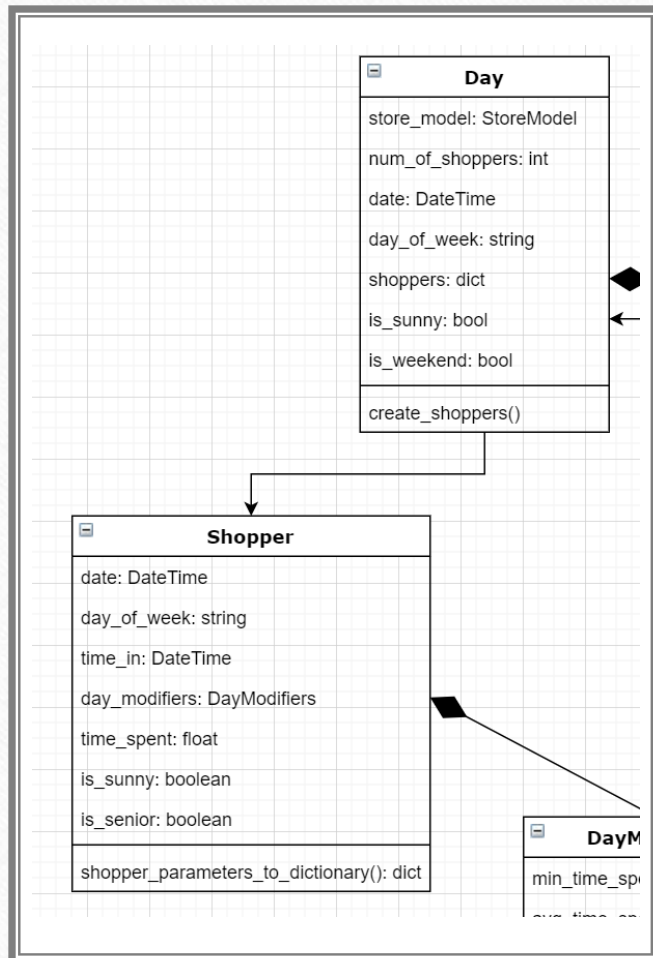
create\_day(date)

# StoreModel

- Holds all the input parameters that are required
- Has one method to create\_day given a date
  - Dates are taken from TimeFrame



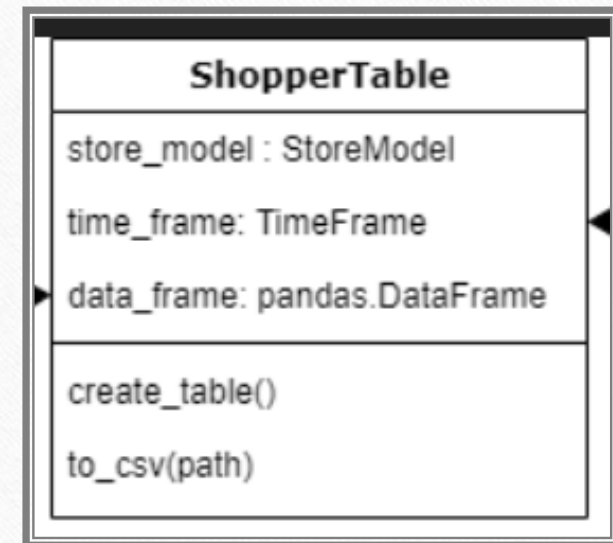
# Day & Shopper



- Days create Shoppers.
  - Create\_shoppers()
- Shopper data added to dictionary
  - Shopper\_parameters\_to\_dictionary

# ShopperTable

- Holds the StoreModel and TimeFrame
  - StoreModel has input parameters
  - TimeFrame has dates
- Create\_table method does heavy lifting
- Stores generated data as a pandas DataFrame which can be exported to csv



# Shopper Database Design

---

- One class: `ShopperDatabase()`
- Handles 3 main functionalities:
  - Populate a MongoDB collection with generated shopper data
  - Store configuration parameters used to generate the shopper data
  - Query the MongoDB collection for shopper data and configuration parameters



# Query Shopper Database

---

## Query shoppers

- `query(query_dict, sort_list, collection_name, limit)`
- `aggregate(agg_list, collection_name, limit)`
- `db_query(query_dict, result_dict, limit)`

## Query configuration parameters

- `find_parameters()`

# Shopper API Design

## **collections** Collections in the MongoDB database

**GET** **/collections** Returns a dictionary of all collections in the MongoDB database

## **parameters** Parameters used to generate the mock shopper data

**GET** **/parameters** Returns a dictionary of all the parameter documents in the database

**GET** **/parameters/{parameter\_id}** Returns a set of parameters given a parameter ID

**GET** **/parameters/{parameter\_id}/shoppers** Returns the shoppers generated by a specific set of parameters























## **shoppers** Shoppers generated from a set of parameters

**POST** **/shoppers/{collection\_name}** Creates a shopper document in the specified collection

**GET** **/shoppers/{collection\_name}** Returns a dictionary of all shoppers in the MongoDB database collection

**GET** **/shoppers/{collection\_name}/{shopper\_id}** Returns a dictionary with information about the shopper with the specified ID



# Code Metrics

- ▼  day.py : 3 refactors
  -  Too many instance attributes (11/7) (14:0) [too-many-instance-attributes]
  -  Too many local variables (22/15) (44:4) [too-many-locals]
  -  Too few public methods (1/2) (14:0) [too-few-public-methods]
- ▼  day\_modifiers.py : 2 refactors
  -  Too many arguments (8/5) (12:4) [too-many-arguments]
  -  Too few public methods (1/2) (7:0) [too-few-public-methods]
- ▼  holiday\_modifiers.py : 1 refactor
  -  Too few public methods (1/2) (9:0) [too-few-public-methods]
- ▼  main.py : 1 warning, 1 refactor
  -  Consider merging these comparisons with "in" to "user\_query in ('quit', 'q')" (244:11) [consider-using-in]
  -  Unused argument 'args' (255:14) [unused-argument]
- ▼  senior\_discount.py : 2 refactors
  -  Too many instance attributes (8/7) (7:0) [too-many-instance-attributes]
  -  Too many arguments (7/5) (12:4) [too-many-arguments]
- ▼  shopper.py : 3 refactors
  -  Too many instance attributes (8/7) (7:0) [too-many-instance-attributes]
  -  Too many arguments (6/5) (13:4) [too-many-arguments]
  -  Too few public methods (1/2) (7:0) [too-few-public-methods]
- ▼  shopper\_database.py : 6 refactors
  -  Similar lines in 2 files==main:28==test\_shopper\_database:19 parser.add\_argument('-sd', '--start-date', default='2020-01-01', type=str, help=
  -  Similar lines in 2 files==main:125==test\_shopper\_database:118def create\_config(args): """ Returns configuration objects initialized with data from pa
  -  Similar lines in 2 files==Configuration.rush:65==Configuration.senior\_discount:42 except AttributeError: self.\_start\_time = start\_time @prop
  -  Similar lines in 2 files==Configuration.rush:43==Configuration.senior\_discount:20 @property def start\_time(self): """Return the start time."""
  -  Similar lines in 2 files==Configuration.\_init\_:2==main:8from configuration.holiday\_modifiers import HolidayModifiersfrom configuration.rush import
  -  Similar lines in 2 files==Configuration.rush:82==Configuration.senior\_discount:59 a\_end\_time = datetime.strptime(end\_time, '%H:%M').time()
- ▼  store\_model.py : 2 refactors
  -  Too many instance attributes (11/7) (10:0) [too-many-instance-attributes]
  -  Too many arguments (10/5) (15:4) [too-many-arguments]



# Test Metrics

---

- ▼  test\_day\_modifiers.py : 1 warning
  - ⚠ Unused variable 'i' (48:12) [unused-variable]
- ▼  test\_shopper\_database.py : 1 warning
  - ⚠ Wildcard import configuration (8:0) [wildcard-import]

# Installation and DEMO

---

- Program is designed for Python 3.8
- `pip install -r requirements.txt` can be used to install requirements
- Project requirements:

flask~=1.1.2
flask-restx~=0.2.0
numpy~=1.18.1
pandas~=1.0.3
holidays~=0.10.2
python-dateutil~=2.8.1
pymongo~=3.9.0