Graded assignment 3 - Hongyu (Ray)

August 1, 2019

[1]:
```
class Book:
    def __init__(self, *args):
        self.book_id, self.title, self.author, self.average_rating, self.isbn,␣
↪self.isbn13,\
        self.language_code, self.num_pages, self.ratings_count, self.
↪text_reviews_count = args

    def __str__(self):
        """
        Let student specify what an object should looks like
        """
        pass
```

---

[2]:
```
class GoodReadBookAPI:
    def __init__(self, csv_df, database):
        self.df = csv_df
        self.database = database # a list of all Book objects

    # -- Generator -- #
    def book_generator():
        """
        return one book object each time
        """
        pass

    # -- Object_Oriented -- #
    def search_by_author(author):
        pass
    def search_by_min_rating(min_rating):
        pass

    # -- pandas -- #
    def render_records(first_or_last, num):
        """
        return first or last amount(num) of records
```

```
        Output: pandas DataFrame
        """
        pass
    def sort_by(*args, ascending=True):
        """
        Input:  column names (position represent priority) and order
        Output: pandas DataFrame
        """
        pass
    def group_by(*args):
        """
        Input:  column names (position represent priority)
        Output: python dictionary
        """
        pass
    def group_count(*args):
        """
        Input:  column names (position represent priority)
        Output: pandas Series
        """
        pass
    def add_column_of_sum(start_column, end_column, new_column_name):
        """
        New column is made by adding the values from start_column to end_column

        Output: pandas DataFrame
        """
        pass
```