Where got time?

...

Real-time canteen information programme

By Ernest, Jeremy and Joshua

Program Information

```
# Program's Backbone (Skeleton)
class AppMainframe(tk.Tk):

    def __init__(self):
        tk.Tk.__init__(self)
        container = tk.Frame(self)
        container.pack(side="top", fill="both", expand=True)
        container.grid_rowconfigure(10, weight=10)
        container.grid_columnconfigure(10, weight=10)

        self.frames = {}

        for F in (WelcomePage, DtPage):
            frame = F(container, self)
```

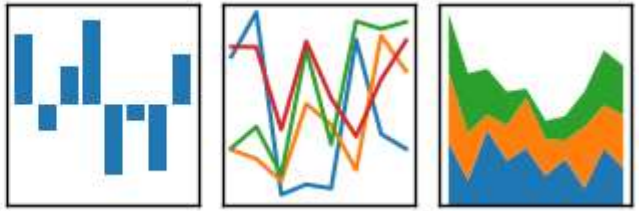
Pandas Library + Menu Sorting Function

```
# Program's CSV File
# Change the directory of the file
df = pd.read_csv('/Users/ernestang98/Desktop/Whole_Menu.csv')


# Menu Sorting Function
def menu_display_final(store_name=df['Store Name'], day=df['Availability (Day)'], time=df['Availability (Time)']):
    menu_display = df[
        (df['Store Name'] == store_name) & ((df['Availability (Day)'] == day) | (df['Availability (Day)'] ==
                                                                                 'Everyday')) & (
                (df['Availability (Time)'] == time) | (df['Availability (Time)'] == 'AM & PM'))]
    menu_display = menu_display[['Store Name', 'Food Item', 'Price ($)']].drop_duplicates()
    return menu_display
```

pandas

$$y_{it} = \beta' x_{it} + \mu_i + \epsilon_{it}$$



Shortest Queue Function

```python
# Shortest Queue Function
def queue_prob(hour):
    prob_BF_1 = [1, 2, 5, 1, 1, 3]
    prob_BF_2 = [2, 3, 5, 1, 2, 3]
    prob_LUN_1 = [3, 5, 1, 3, 2, 2]
    prob_LUN_2 = [5, 3, 1, 3, 2, 1]
    prob_DIN_1 = [3, 3, 2, 5, 3, 2]
    prob_DIN_2 = [3, 3, 2, 3, 5, 2]
    if hour == 8 or hour == 9:
        return prob_BF_1
    elif hour == 10 or hour == 11:
        return prob_BF_2
    elif hour == 12 or hour == 13:
        return prob_LUN_1
    elif hour == 14 or hour == 15:
        return prob_LUN_2
    elif hour == 16 or hour == 17:
        return prob_DIN_1
    elif hour == 18 or hour == 19 or hour == 20:
        return prob_DIN_2
    else:
        return 0
```

Different stores have different probability at different timing

```python
def shortest_queue():
    while True:
        store = ['Yong Tau Foo', 'Chicken Rice', 'Western Food', 'Mini Wok',
                 'Duck Rice', 'Indian']
        hour_input = hour
        prob_choice = queue_prob(hour_input)
        if prob_choice == 0:
            break
        else:
            shortest_queue = random.choices(store, prob_choice, k=1)
            return shortest_queue
            break

shortest_label = tk.Label(custom_menu,
                          text=('Shortest Queue: ' + str(
                              shortest_queue()).strip('[]').strip("'")),
                          fg="white", bg='#19221D', font=('Verdana', 12))
shortest_label.pack(side=tk.TOP)
```

Tagging the stores to the different probability
Using the 'random.choices()' method to return the predicted store with the shortest queue

Random Food Generator

```python
# Random Food Generator Function
def random_food(day1, time1, num1, frame1=None):
    menu1 = (menu_display_final(day=day1, time=time1))
    rand_fd = menu1.sample(n=int(num1), replace=True)

    def structured_menu(frame1):
        store_name_list = rand_fd['Store Name'].to_string(index=False)
        food_item_list = rand_fd['Food Item'].to_string(index=False)
        price_list = rand_fd['Price ($)'].to_string(index=False)
        label1 = tk.Label(frame1, text=store_name_list, fg="white", bg='#000000')
        label1.config(font=("Verdana", 14))
        label1.place(x=90, y=360, anchor='center')
        label2 = tk.Label(frame1, text=food_item_list, fg="white", bg='#000000')
        label2.config(font=("Verdana", 14))
        label2.place(x=240, y=360, anchor='center')
        label3 = tk.Label(frame1, text=price_list, fg='white", bg='#000000')
        label3.config(font=("Verdana", 14))
        label3.place(x=400, y=360, anchor='center')

    structured_menu(frame1)
    # return generated random food
    return rand_fd
```

Random Food Generator Function

Thank you