AB0403 Presentation HDB Sales Scenario 5

Done by: Quek Minn, Emma (U2410898A)

Which storey remains most popular among Singapore residences by flat type across the years.

01

Filter Criterion

Found in the main code

```python
#MAIN PROGRAM
print("""Welcome to my program!
In this program, you are able to check which storey remains most popular among Singapore residences by flat type across the years.
The flat models available to check are:""")
models = ["2-room","Adjoined flat","Apartment","DBSS","Improved","Maisonette","Model A","Model A-Maisonette","Model A2",
          "Multi Generation","New Generation","Premium Apartment","Simplified","Standard","Terrace","Type S1","Type S2"]
for model in models:
    print(model)
while True:
    filter_criterion = input(str("\nEnter flat model you would like to check (or press 'Enter' to exit the program):"))
    if filter_criterion == "":
        print("Thank you for using my program!")
        break
    elif filter_criterion not in models:
        print("There is no such model. Try again.")
    else:
        result = sort_data(filter_criterion)
        print_table(filter_criterion, result)
        insight = insights(result)
```

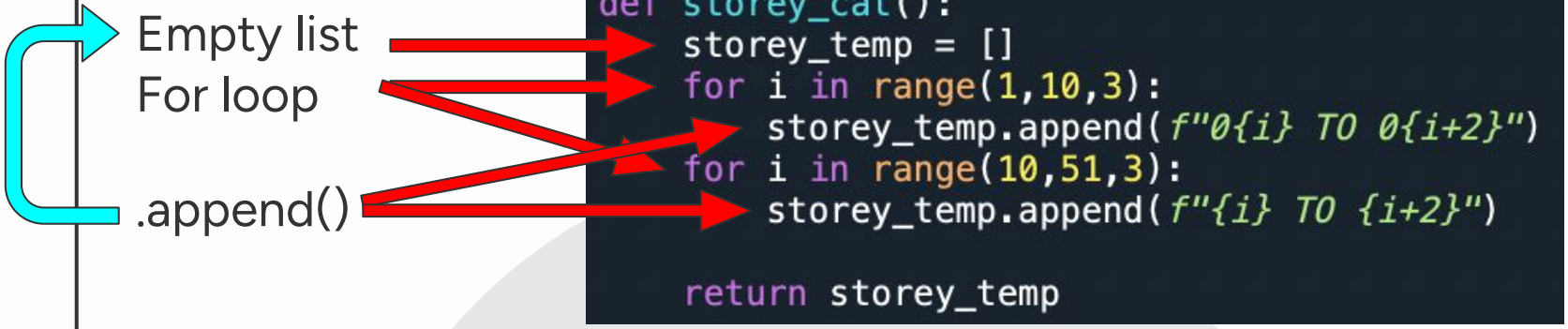
```
Welcome to my program!
In this program, you are able to check which storey remains most popular among Singapore residences by
flat type across the years.
The flat models available to check are:
2-room
Adjoined flat
Apartment
DBSS
Improved
Maisonette
Model A
Model A-Maisonette
Model A2
Multi Generation
New Generation
Premium Apartment
Simplified
Standard
Terrace
Type S1
Type S2

Enter flat model you would like to check (or press 'Enter' to exit the program):
```

02

Table Tabulation

Several functions were defined & called in main code

Step 1: Define storeys



Step 2: Filter data from CSV

Empty dictionary

Define year and storey

Empty dictionary formatting

Import & read csv

Indexing to match lines in CSV

Value added in dictionary

```
#FILTER OUT RELEVANT DATA FROM CSV
def sort_data(flat_model):
    data = {}
    year = [2020, 2021, 2022, 2023, 2024]
    storey = storey_cat()

    for y in year:
        data[y] = {}
        for s in storey:
            data[y][s] = 0

    import csv
    filename = "HDB_Sales.csv"
    with open(filename, "r") as file_pointer:
        csv_pointer = csv.reader(file_pointer)
        next(csv_pointer)

        for line in csv_pointer:
            s = line[3]
            y = int(line[-3])
            if line[5] == flat_model and y in data and s in storey:
                data[y][s] += 1
    return data
```

Step 3: Print table

Header

For loop

Print storey

Extract value in dictionary

```
#TABLE FORMATTING
def print_table(flat_model, data):
    print(f"""
{"_"*72}
|{"Popularity for " + flat_model:^70}|
|{"_"*70}|
|{'Storey':<15}|{'2020':>10}|{'2021':>10}|{'2022':>10}|{'2023':>10}|{'2024':>10}|
|{"_"*70}| """)

    for key in data[2020].keys():
        print_row = f"|{key:<15}|"
        for value in data.keys():
            print_row += f"{str(data[value][key]):>10}|"
        print(print_row)

    print(f"|{"_"*70}|\n")
```

Example Output

Popularity for Improved					
Storey	2020	2021	2022	2023	2024
01 TO 03	991	1185	1060	944	1156
04 TO 06	1303	1582	1435	1328	1500
07 TO 09	1187	1494	1464	1352	1456
10 TO 12	1134	1347	1243	1130	1156
13 TO 15	652	810	743	657	713
16 TO 18	305	393	356	279	299
19 TO 21	130	189	142	130	163
22 TO 24	94	116	88	70	76
25 TO 27	40	54	42	33	37
28 TO 30	19	25	30	16	14
31 TO 33	6	4	9	6	7
34 TO 36	2	7	6	5	6
37 TO 39	2	4	1	6	3
40 TO 42	0	4	1	1	2
43 TO 45	0	0	0	2	0
46 TO 48	0	0	1	0	1
49 TO 51	0	0	0	0	0

03

Generate Insights

Using max()

```python
#INSIGHTS: WHICH STOREY HAS THE MOST FLATS PURCHASED FOR EACH YEAR & OVERALL
def insights(data):
    year = [2020, 2021, 2022, 2023, 2024]
    storey = storey_cat()
    storey_max = {}
    for st in storey:
        storey_max[st] = 0

    for yr in year:
        highest_yearly = max(data[yr].values())
        if highest_yearly == 0:
            print(f"There were no such flats purchased in {yr}.")
            continue
        storey_check = []
        for st in storey:
            if highest_yearly == data[yr][st]:
                storey_check.append(st)
        storey_print = f"{storey_check[0]}"
        if len(storey_check) > 1:
            for i in range(1,len(storey_check)):
                storey_print += f", {storey_check[i]}"
        print(f"The most popular storey in {yr} is {storey_print} with total of {highest_yearly} flats purchased.")
        for st in storey_check:
            storey_max[st] += 1

    highest = max(storey_max.values())
    highest_check = []
    for st in storey:
        if highest == storey_max[st]:
            highest_check.append(st)
    highest_print = f"{highest_check[0]}"
    if len(highest_check) > 1:
        for j in range(1,len(highest_check)):
            highest_print += f", {highest_check[j]}"
    print(f"The most popular storey throughout the years is {highest_print}.")
```

Part 1: Per year basis - checks for highest value every year

Part 2: Overall - checks for which storey most repeated in part 1

Example Output

Improved:
No repeats

```
The most popular storey in 2020 is 04 TO 06 with total of 1303 flats purchased.
The most popular storey in 2021 is 04 TO 06 with total of 1582 flats purchased.
The most popular storey in 2022 is 07 TO 09 with total of 1464 flats purchased.
The most popular storey in 2023 is 07 TO 09 with total of 1352 flats purchased.
The most popular storey in 2024 is 04 TO 06 with total of 1500 flats purchased.
The most popular storey throughout the years is 04 TO 06.
```

Adjoined flat: >1 storey is highest

```
The most popular storey in 2020 is 07 TO 09 with total of 19 flats purchased.
The most popular storey in 2021 is 01 TO 03 with total of 13 flats purchased.
The most popular storey in 2022 is 04 TO 06, 07 TO 09, 10 TO 12 with total of 10 flats purchased.
The most popular storey in 2023 is 07 TO 09 with total of 11 flats purchased.
The most popular storey in 2024 is 04 TO 06 with total of 13 flats purchased.
The most popular storey throughout the years is 07 TO 09.
```

Function Calling

```
#MAIN PROGRAM
print("""Welcome to my program!
In this program, you are able to check which storey remains most popular among Singapore residences by flat type across the years.
The flat models available to check are:""")
models = ["2-room","Adjoined flat","Apartment","DBSS","Improved","Maisonette","Model A","Model A-Maisonette","Model A2",
          "Multi Generation","New Generation","Premium Apartment","Simplified","Standard","Terrace","Type S1","Type S2"]
for model in models:
    print(model)
while True:
    filter_criterion = input(str("\nEnter flat model you would like to check (or press 'Enter' to exit the program):"))
    if filter_criterion == "":
        print("Thank you for using my program!")
        break
    elif filter_criterion not in models:
        print("There is no such model. Try again.")
    else:
        result = sort_data(filter_criterion)
        print_table(filter_criterion, result)
        insight = insights(result)
```

Repeat until blank input

04

Additional Insights

Trend analysis for most bought storey

Trend analysis

Calculate percentage change in the highest bought storey for the flat model from year to year.

```
def add_insights(data, first_insight):
    year = [2020, 2021, 2022, 2023, 2024]
    trend = []
    for y in year:
        count = data[y][first_insight]
        trend.append(count)

    for n in range(len(trend)-1):
        change = ((trend[n+1]-trend[n]) / trend[n]) * 100
        print(f"{year[n]} -> {year[n+1]}: {round(change,2)}% change")
```

```
else:
    result = sort_data(filter_criterion)
    print_table(filter_criterion, result)
    insight = insights(result)
    answer = input(str(f"\nWould you like to know trends in {insight} storey {filter_criterion} flats purchased over the years? (Yes/No):"))
    if answer == "No":
        continue
    elif answer == "Yes":
        add_insights(result, insight)
```

```
Would you like to know trends in 07 TO 09 storey Adjoined flat flats purchased over the years? (Yes/No):Yes
2020 -> 2021: -42.11% change
2021 -> 2022: -9.09% change
2022 -> 2023: 10.0% change
2023 -> 2024: -18.18% change
```

Value added:

Identify growing and declining trends to better understand characteristic of most popular storey and possibly investigate more on declining years.

05

Reflection

Useful codes

Loops

Helped in simplifying code since repetition is needed a lot

Function defining and calling

Helped to organise code and easy to reuse

Formatting (f"...")

Helped in arranging output to print out neatly

Learning points

Trial and error

Keep “debugging” by focusing on the line that is printed wrongly/ produces an error

Use sample codes

Refer to codes already done (I did from tutorial examples) and reformat to suit the problem you are working on

Find the easiest route

To avoid code being unnecessarily long and messy, always refine code to achieve the same goal in the shortest route

Thank you!

CREDITS: This presentation template was created by **Slidesgo,** and includes icons, infographics & images by **Freepik**

Please keep this slide for attribution