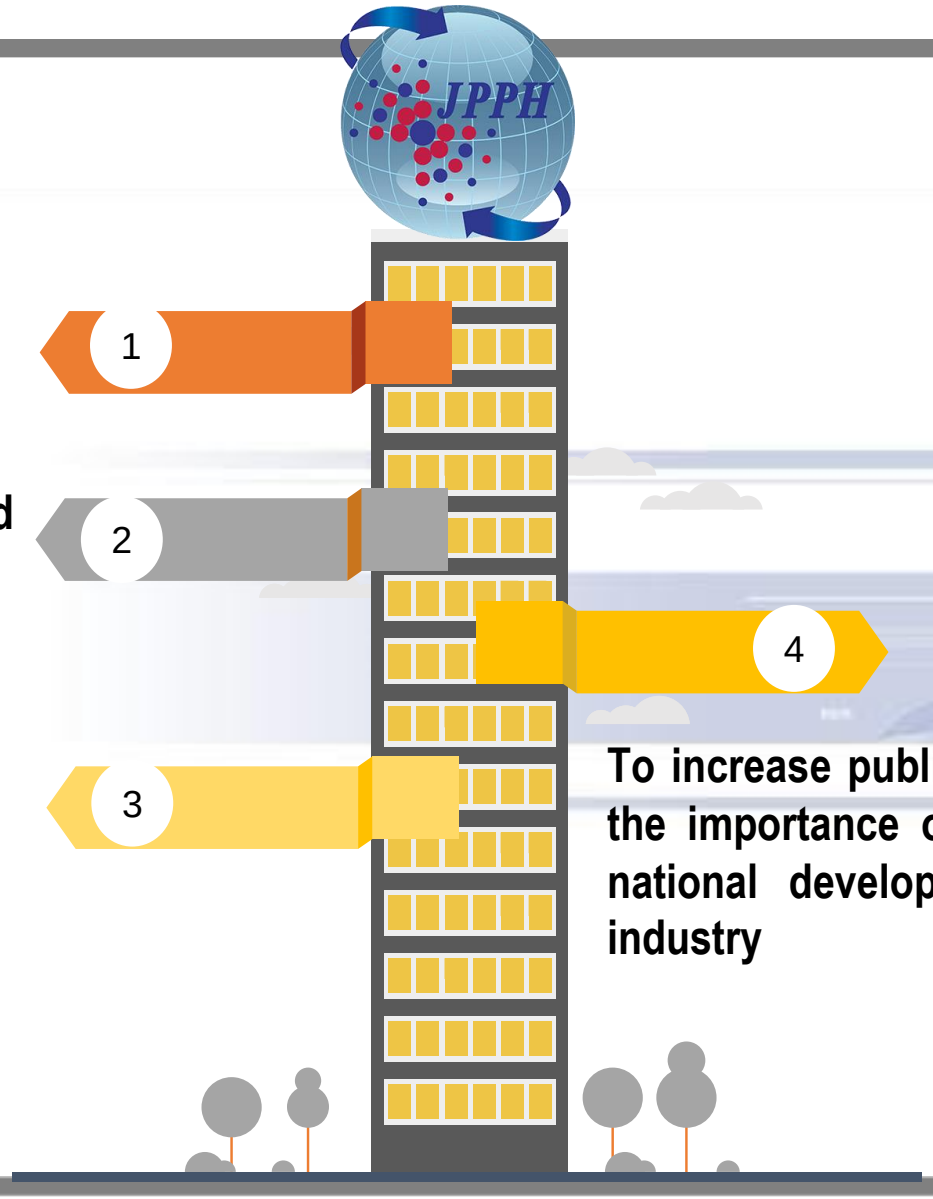
CAPSTONE PROJECT PRESENTATION
PROPERTY INVENTORY DIVISION
NATIONAL PROPERTY INFORMATION CENTRE (NAPIC)

"Identifying the ideal demand and supply in the residential sector in Johor Darul Takzim"

Shahrul bin Abd Zoher
Director Assistant
Property Inventory Division (PRISM)

CONTENT

BUSINESS UNDERSTANDING - NAPIC



To collect property demand and supply data from various parties

To provide accurate, comprehensive and timely information to government agencies and all other parties involved in the property industry

To advice the government on property development in the country

To increase public awareness regarding the importance of property data to the national development of the property industry

PROBLEM STATEMENT

There's a mismatch in the housing market'

> Residential property sector experiencing a supply-demand imbalance, says expert

BY **EVA YEONG**
sunbiz@thesundaily.com

PETALING JAYA: The residential property market is experiencing a mismatch in terms of demand and supply, and prices need to be managed so that the market remains sustainable, said Jones Lang Wootton executive director Malathi Thevendran.

"There's totally a demand-supply mismatch. There is demand but to be sustainable we have to make sure that it is attractive pricing. At the same time, occupancy and product differentiation are important," she said during her presentation on the residential and condominium segment at the 17th National Housing & Property Summit 2014 here yesterday.

Malathi said the supply and demand imbalance came about when developers launched too many residential units priced RM1 million and above.

In 2005, only 10% of the launches were RM1 million and... However, in 2012, 40%... priced RM1 million a...

"How many peopl... You can upgrade one... you upgrade to so ma... detached houses? Yo... many bungalows," sh...

As a result, launch... prices rose. Accordin... Jones Lang Wootton,... 400 residential units... seven projects so far...

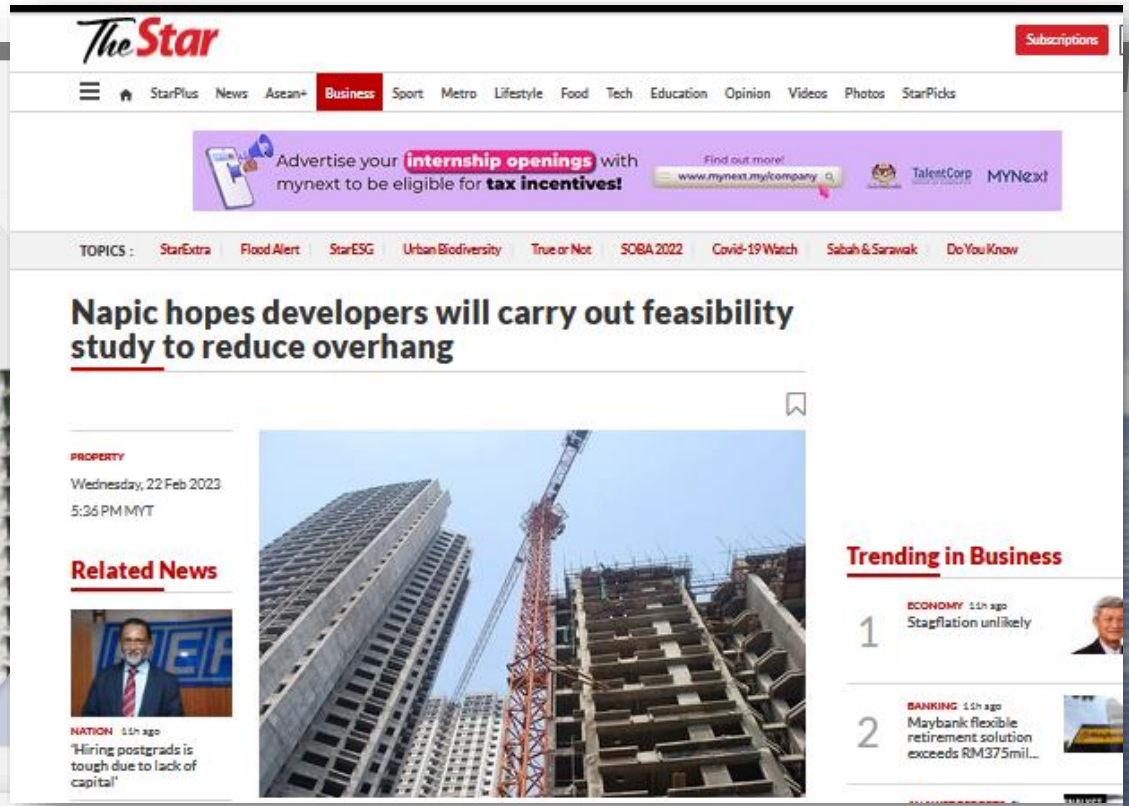
"It shows that in t... there are not many h... they were to launch,... no less than RM750,0... people can actually bu...

"At the same time, it... or not they can get a hous...

Mismatch between purchasing power and market prices making houses unaffordable

Shivani Supramani

15-08- 2022 08:45 AM



PROBLEM STATEMENT QUESTIONS

Study 1

What is the current supply of residential properties and the current population of households in Johor?

Study 2

What is the trend in the supply of residential properties compared to the total household population?

Based on the results of studies 1 and 2 conducted previously, further questions need to answer as follow:

1.
- If the current property stock is insufficient, what type of property can be built and what is the ideal price range should be offer for a certain area?

2.
- If the property stock quantity is sufficient and excessive, the relevant parties need to consider who the target group is that is deemed potential buyers.

DATA RESOURCE DESCRIPTIONS

1. Residential Property Stock Data (NAPIC) : Year 2012 until 2023, 120 row and 18 column

Review Period	Quarter	Year	Development Stage	State	District	Single Storey Terrace	2 - 3 Storey Terrace	Single Storey Semi-Detach	2 - 3 Storey Semi-Detach	Detach	Town House	Cluster	Low Cost House	Low Cost Flat	Flat	Condominium / Apartment	Total
Q4 2018	Q4	2018	Existing Stock	Johor	Batu Pahat	22806	14890	4712	2629	16890	84	124	21609	1060	714	1160	86678
Q4 2018	Q4	2018	Existing Stock	Johor	Johor Bahru	80530	141935	5139	12232	11208	1028	8619	49392	46163	21631	51921	429798
Q4 2018	Q4	2018	Existing Stock	Johor	Kluang	21396	12189	3262	1644	8980		156	18628	200	638		67093

2. Residential Property Status Data (NAPIC) : Year 2018 until 2023, 1219 row and 14 column

Review_Period	Quarter	Year	Unsold Status	State	District	Mukim	Local Auhtority Name	Property Type	Total launched	Total Unsold	Price Range	Total Unsold Value	Launched Date
Q4 2018	Q4	2018	Completed	Johor	Johor Bahru	Tebrau	Majlis Bandaraya Johor Bahru	Cluster	150	14	Above RM1,000,000	14350000	19/1/2015
Q4 2018	Q4	2018	Completed	Johor	Johor Bahru	Tebrau	Majlis Bandaraya Johor Bahru	Cluster	154	13	Above RM1,000,000	16770000	19/1/2015
Q4 2018	Q4	2018	Completed	Johor	Johor Bahru	Tebrau	Majlis Bandaraya Johor Bahru	2-3 Storey Semi-Detach	90	37	Above RM1,000,000	66970000	19/1/2015

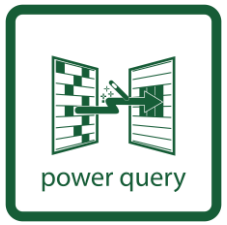
3. Household Population Data (DOSM) : Year 2014 until 2023, 100 row and 4 column

Year	State	District	Household Population
2014	Johor	Batu Pahat	94314
2014	Johor	Johor Bahru	358715
2014	Johor	Kluang	67712

4. Household Income Data (DOSM) : Year 2018 until 2023, 58 row and 4 column

Year	State	District	Household Income
2018	Johor	Kluang	RM4,547.38
2018	Johor	Kota Tinggi	RM5,247.92
2018	Johor	Mersing	RM3,772.32

DATA RESOURCE PREPROCESSING



Tableau

Pyhton

Ms Excel /
Power Querry

DATA RESOURCE PREPROCESSING

Data Cleaning

Using Ms Excel

Table 6

PENAWARAN UNIT KEDIAMAN MENGIKUT JENIS DI JOHOR

SUPPLY OF RESIDENTIAL UNITS BY TYPE IN JOHOR

Review Period	States	Single Storey Terrace	2 - 3 Storey Terrace	Single Storey Semi-Detach	2 - 3 Storey Semi-Detach	Detach	Town House	Cluster	Low Cost House	Low Cost Flat	Flat	Condominium / Apartment	Total
EXISTING STOCK													
Q4 2023	Batu Pahat	26,665	19,390	5,246	3,514	17,010	172	412	21,919	1,132	698	1,189	97,347
	Johor Bahru	81,868	170,859	5,285	13,949	11,616	3,993	14,585	49,392	47,002	24,338	65,571	488,458
	Kluang	23,208	14,872	3,383	1,964	9,316	0	560	18,768	200	638	0	72,909
	Kota Tinggi	11,236	8,200	575	538	4,965	0	424	4,422	180	64	375	30,979
	Kulai	27,006	24,867	865	2,136	4,427	1,683	1,204	9,247	2,114	1,180	1,009	75,738
	Mersing	2,009	645	218	40	1,779	0	0	2,565	0	0	0	7,256
	Muar	9,390	6,433	4,730	1,792	2,496	22	152	20,796	274	344	1,452	47,881
	Pontian	5,265	3,274	2,155	677	2,777	88	192	5,773	45	233	0	20,479
	Segamat	13,582	8,507	2,184	952	10,059	0	176	11,312	268	30	0	47,070

COVER | CONTENTS | TABLE 1 | TABLE 2 | TABLE 3 | TABLE 4 | TABLE 5 | TABLE 6 | TABLE 7 | TABLE 8 | TABLE 9 | TABLE 10

Review Period	Quarter	Year	Development Stage	State	District	Single Storey Terrace	2 - 3 Storey Terrace	Single Storey Semi-Detach	2 - 3 Storey Semi-Detach	Detach	Town House	Cluster	Low Cost House	Low Cost Flat	Flat	Condominium / Apartment	Total
Q4 2018	Q4	2018	Existing Stock	Johor	Batu Pahat	22806	14890	4712	2629	16890	84	124	21609	1060	714	1160	86678
Q4 2018	Q4	2018	Existing Stock	Johor	Johor Bahru	80530	141935	5139	12232	11208	1028	8619	49392	46163	21631	51921	429798
Q4 2018	Q4	2018	Existing Stock	Johor	Kluang	21396	12189	3262	1644	8980		156	18628	200	638		67093

DATA RESOURCE PREPROCESSING

Table 2
BILANGAN DAN NILAI UNIT KEDIAMAN SIAP DIBINA TIDAK TERJUAL MENGIKUT NEGERI, JENIS DAN LINGKUNGAN HARGA Q4 2023
NUMBER AND VALUE OF OVERHANG RESIDENTIAL UNITS BY STATE, TYPE AND PRICE RANGE Q4 2023

State	Price Range	Single Storey Terraced	2 -3 Storey Terraced	Single Storey Semi-Detached	2-3 Storey Semi-Detached	Detached	Town House	Cluster	Low Cost House	Low Cost Flat	Flat	Condominium/ Apartment	Total
JOHOR													
	RM0 - RM100,000	0	90	0	0	0	0	0	70	0	0	0	160
		0	59	0	0	0	0	0	17	0	0	0	76
		0.00	4.72	0.00	0.00	0.00	0.00	0.00	0.84	0.00	0.00	0.00	5.56
	RM100,001 - RM200,000	324	171	0	0	0	496	0	0	0	0	222	1,213
		7	73	0	0	0	125	0	0	0	0	7	212
		1.05	10.95	0.00	0.00	0.00	18.75	0.00	0.00	0.00	0.00	1.05	31.80
	RM200,001 - RM300,000	478	0	0	0	0	0	0	0	0	0	1,444	1,922

Data Cleaning

Using Ms Excel

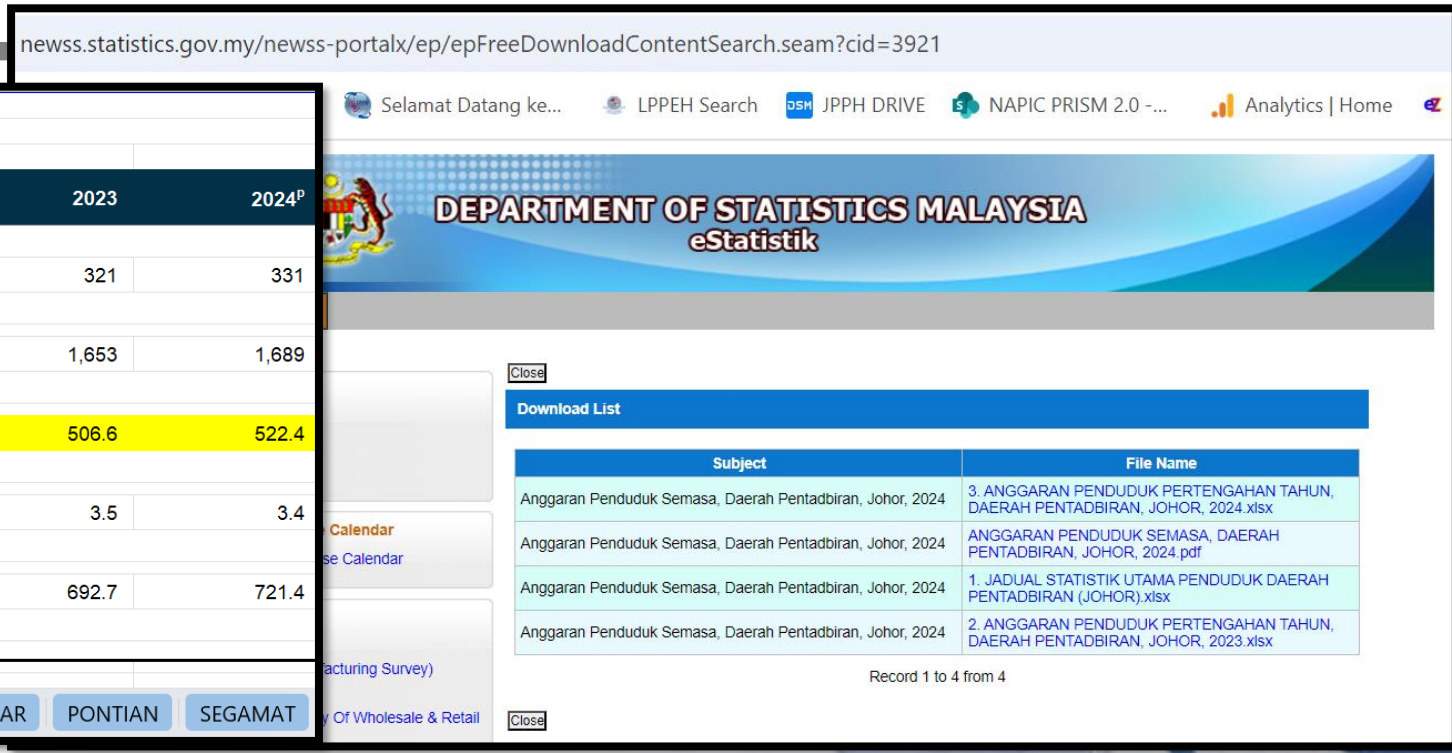
Review_Period	Quarter	Year	Unsold Status	State	District	Mukim	Local Auhtority Name	Property Type	Total launched	Total Unsold	Price Range	Total Unsold Value	Launched Date
Q4 2018	Q4	2018	Completed	Johor	Johor Bahru	Tebrau	Majlis Bandaraya Johor Bahru	Cluster	150	14	Above RM1,000,000	14350000	19/1/2015
Q4 2018	Q4	2018	Completed	Johor	Johor Bahru	Tebrau	Majlis Bandaraya Johor Bahru	Cluster	154	13	Above RM1,000,000	16770000	19/1/2015
Q4 2018	Q4	2018	Completed	Johor	Johor Bahru	Tebrau	Majlis Bandaraya Johor Bahru	2-3 Storey Semi-Detach	90	37	Above RM1,000,000	66970000	19/1/2015

DATA RESOURCE PREPROCESSING

Jadual 1.1.2: Statistik utama penduduk, Johor Bahru, Johor, 2020-2024
Table 1.1.2: Principal statistics of population, Johor Bahru, Johor, 2020-2024

	2020	2021	2022	2023	2024[p]
Citizens					
Bukan warganegara	310	299	310	321	331
Non-citizens					
Kepadatan penduduk (per km^2)	1,605	1,609	1,617	1,653	1,689
Population density (per km^2)					
Bilangan isi rumah ('000)	460.7	475.7	491.1	506.6	522.4
Number of households ('000)					
Purata saiz isi rumah	3.7	3.6	3.5	3.5	3.4
Average household size					
Bilangan tempat kediaman ('000)	610.3	637.2	664.7	692.7	721.4
Number of living quarters ('000)					

JOHOR | BATU PAHAT | JOHOR BAHRU | KLUANG | KOTA TINGGI | MERSING | MUAR | PONTIAN | SEGAMAT



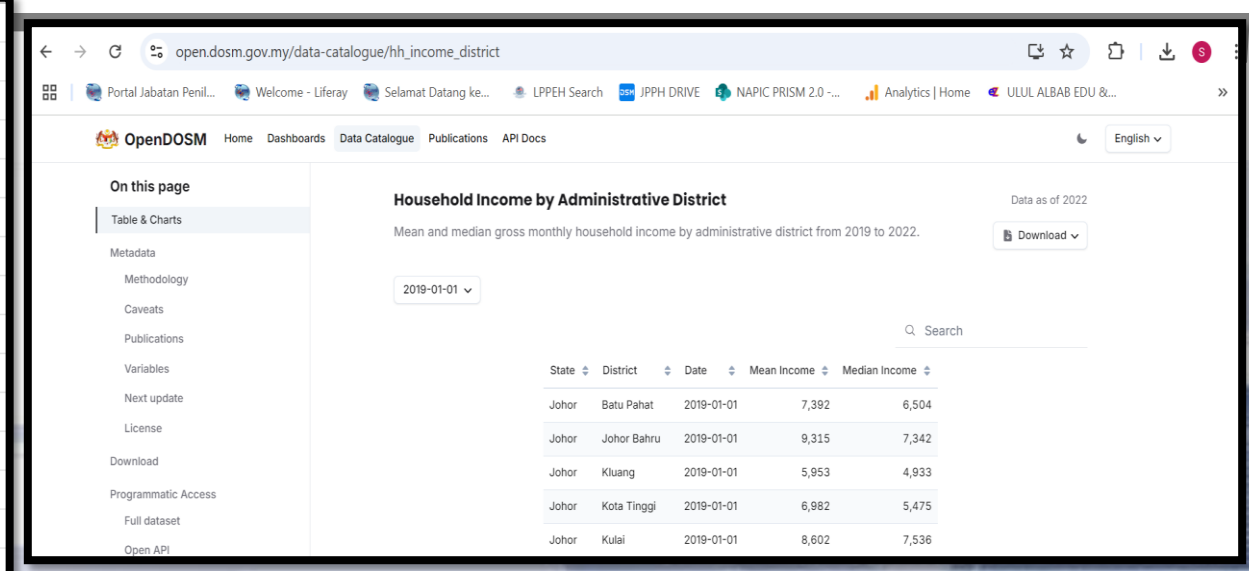
Data Cleaning

Using Ms Excel

Year	Sta	District	Household Populatio
2023	Johor	Batu Pahat	124700
2023	Johor	Johor Bahru	506600
2023	Johor	Kluang	85700
2023	Johor	Kota Tinggi	62300
2023	Johor	Kulai	85700
2023	Johor	Mersing	22100

DATA RESOURCE PREPROCESSING

state	district	date	income_mean	income_median
Johor	Batu Pahat	1/1/2022	7419	6347
Johor	Johor Bahru	1/1/2022	9869	8232
Johor	Kluang	1/1/2022	6461	5204
Johor	Kota Tinggi	1/1/2022	7529	6227
Johor	Kulai	1/1/2022	9177	7460
Johor	Mersing	1/1/2022	5426	4445
Johor	Muar	1/1/2022	7572	5724
Johor	Pontian	1/1/2022	6913	5616
Johor	Segamat	1/1/2022	6578	5318
Johor	Tangkak	1/1/2022	7093	5622
Kedah	Baling	1/1/2022	4263	3400
Kedah	Bandar Baharu	1/1/2022	4565	3922
Kedah	Kota Setar	1/1/2022	5959	4751
Kedah	Kuala Muda	1/1/2022	5624	4200
Kedah	Kubang Pasu	1/1/2022	6221	5201
Kedah	Kulim	1/1/2022	5636	4498
Kedah	Langkawi	1/1/2022	6087	5250



Data Cleaning

Using Ms Excel

Year	State	District	Household Income
2022	Johor	Batu Pahat	RM6,347.00
2022	Johor	Johor Bahru	RM8,232.00
2022	Johor	Kluang	RM5,204.00
2022	Johor	Kota Tinggi	RM6,227.00
2022	Johor	Mersing	RM4,445.00
2022	Johor	Muar	RM5,724.00
2022	Johor	Pontian	RM5,616.00
2022	Johor	Segamat	RM5,318.00
2022	Johor	Kulai	RM7,460.00
2022	Johor	Tangkak	RM5,622.00

DATA RESOURCE PREPROCESSING

Import Data

Using Pyhton

1. Muat naik data set serta variables yang telah dikenalpasti selepas EDA digunakan bagi kerja - kerja forecasting berkaitan jumlah total existing stock bagi tahun 2024, 2025 dan 2026

```
from google.colab import drive
drive.mount('/content/drive')

import pandas as pd
import numpy as np
import gdown

Supply = pd.read_csv("/content/drive/My Drive/Existingstok_Johornew.csv")
print(f'Data size: {Supply.shape[0]} samples, {Supply.shape[1]} features')
Supply.head()
```

2. Setelah data set dimuatnaik, perlu menyemak jenis data bagi setiap variables yang digunakan bagi kelestarian proses penyediaan ML.

```
Supply.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 120 entries, 0 to 119
Data columns (total 5 columns):
 #   Column               Non-Null Count  Dtype
---  ------               --------------  -----
 0   Year                 120 non-null    object
 1   Development_Stage    120 non-null    object
 2   State                120 non-null    object
 3   District             120 non-null    object
 4   Total_ExistingStock  120 non-null    int64
dtypes: int64(1), object(4)
memory usage: 4.8+ KB
```

Data Cleaning

Using Pyhton

DATA RESOURCE PREPROCESSING

3. Daripada semakan jenis data bagi setiap variables diatas, didapati variables Year perlu ditukar kepada format datetime64 bagi memudahkan proses ML.

```
1  Supply['Year'] = pd.to_datetime(Supply['Year'])
2
3  Supply.dtypes
```

	0
Year	datetime64[ns]
Development_Stage	object
State	object
District	object
Total_ExistingStock	int64

dtype: object

Data Transformation

Using Pyhton

Missing Value Check

Using Pyhton

4. Setelah semua variables yang telah dikenalpasti ditukar kepada format data yang tepat, tindakan semakan samada terdapat missing value, jika ada, perlu tindakan pembersihan dengan pengisisan nilai null/nan.

```
1  Supply. isna().sum()
```

	0
Year	0
Development_Stage	0
State	0
District	0
Total_ExistingStock	0

dtype: int64

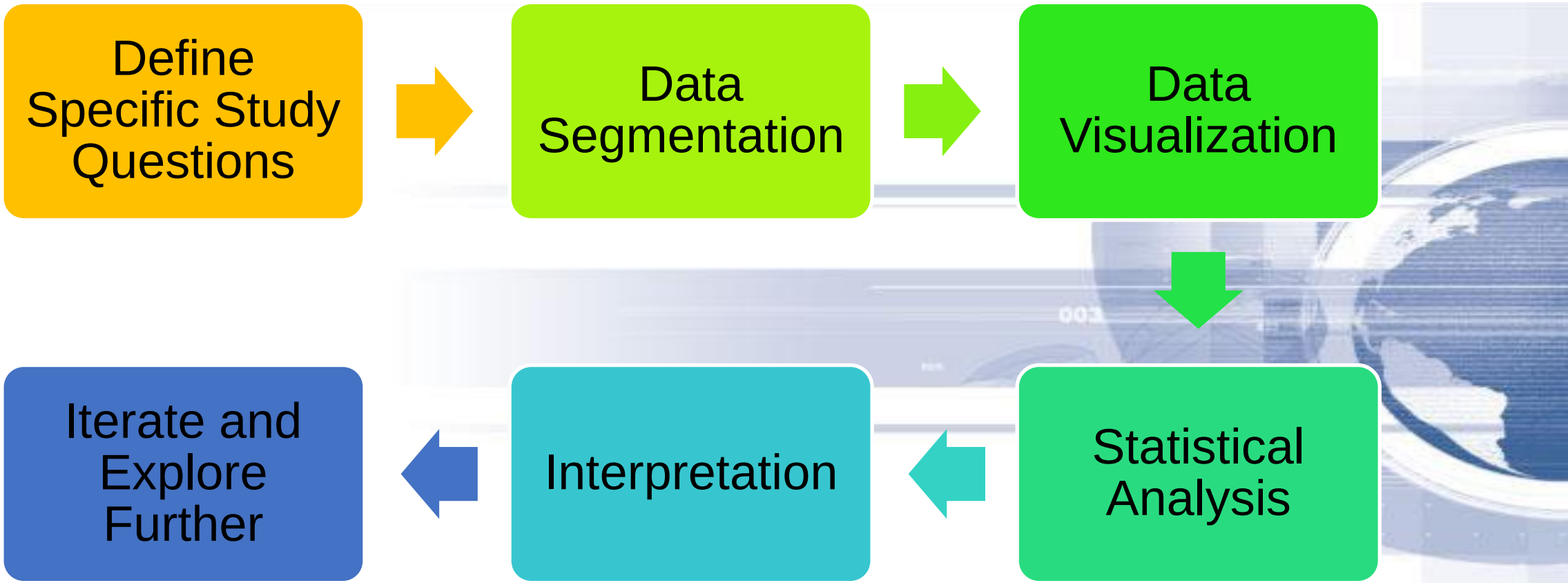
DATA RESOURCE PREPROCESSING

5. Apabila proses semakan di atas telah selesai, maka data set tersebut telah berada dalam keadaan yang baik untuk diproses melalui kaedah ML untuk mendapatkan anggaran forecasting.

6. Untuk memudahkan aturan coding, maklumat yang hendak diuji iaitu Y Target = Total_ExistingStock bagi setiap daerah. Pengujian time series ini dilakukan mengikut peringkat daerah dan dibandingkan dengan daerah yang lain dalam negeri yang sama secara by default disetkan sebagai filtered_data

```
[ ]  1  district = 'Johor Bahru'
     2
     3  filtered_data = Supply[Supply['District']==district]['Total_ExistingStock']
```

EXPLORATORY DATA ANALYSIS (EDA)

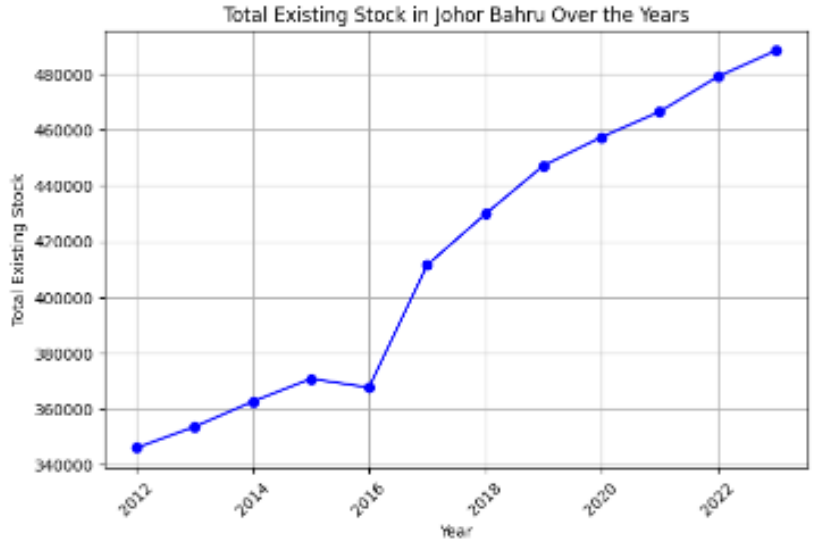


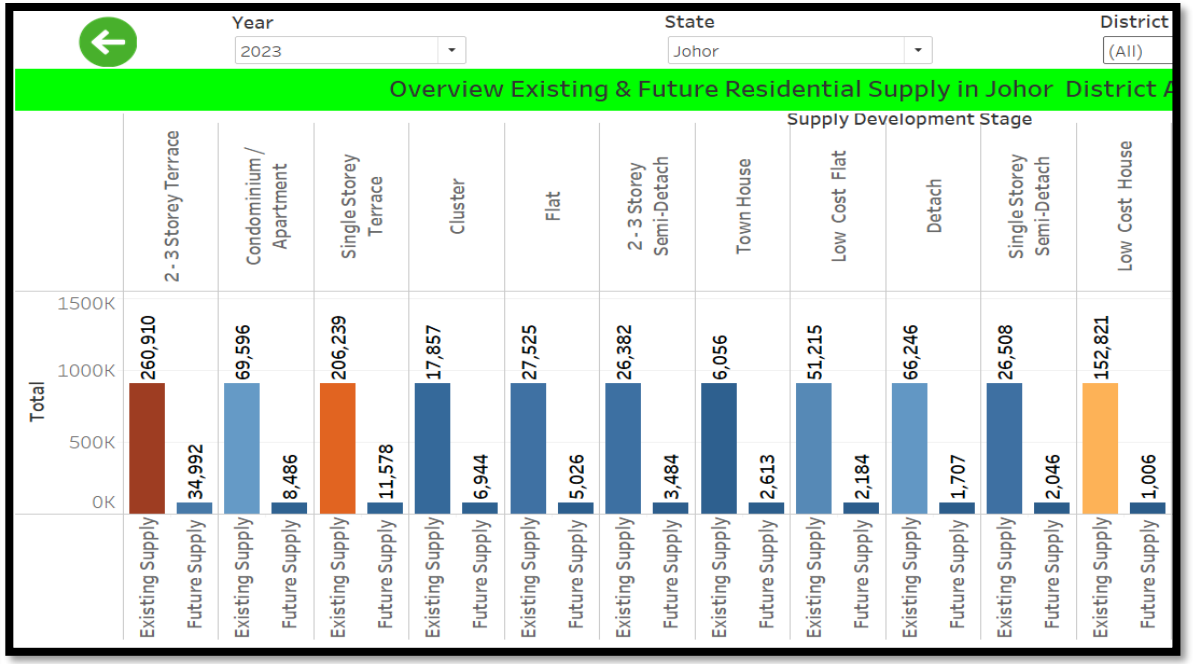
EXPLORATORY DATA ANALYSIS (EDA)

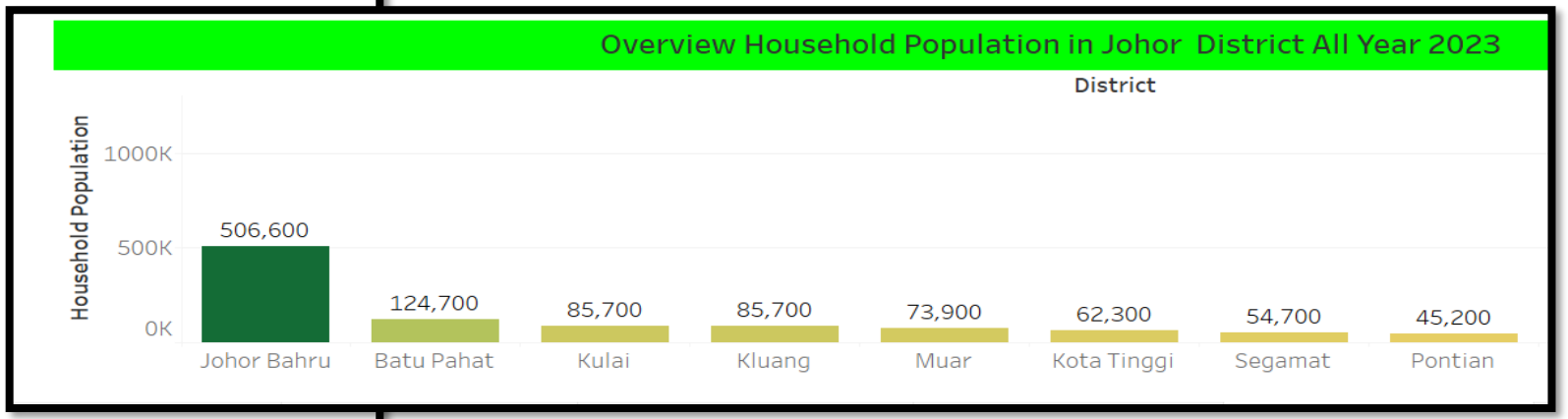
Specific Question 1	Data Segmentation	Data Segment Justification	Study Benefits
What is the current supply of residential properties and the current population of households in Johor?	• Property Stock Report (NAPIC) • Household Population Census Report (DOSM)	• Knowing the number of residential supply according to State, District, Year, and Type of House based on the development stage, namely:- a. Completed stock (Existing Stock) b. Under Construction Supply (Incoming Supply) c. Not Constructed Supply (Planned Supply) For items (b) and (c), it refers to Future Supply Data. • Knowing the household population by State, District, and Year	• Providing an initial overview to stakeholders regarding the trend in the number of existing stock and upcoming supply, broken down by each type of house and compared with • Distribution of household population for each development area • Becoming an initial indicator to stakeholders regarding the supply status based on the number of potential homebuyers (household population) whether:- a. insufficient b. sufficient c. excessive

EXPLORATORY DATA ANALYSIS (EDA)

```python
import matplotlib.pyplot as plt

district = 'Johor Bahru'

# Filter data for the specified district
filtered_data1 = Supply[Supply['District'] == district]

# Plot the data with 'Year' on the x-axis and 'Total_ExistingStock' on the y-axis
plt.figure(figsize=(8, 5))
plt.plot(filtered_data1['Year'], filtered_data1['Total_ExistingStock'], marker='o', linestyle='-', color='b')

# Set plot labels and title
plt.xlabel('Year')
plt.ylabel('Total Existing Stock')
plt.title(f'Total Existing Stock in {district} Over the Years')

# Show the plot
plt.xticks(rotation=45)  # Rotate x-axis labels if needed
plt.grid(True)
plt.show()
```



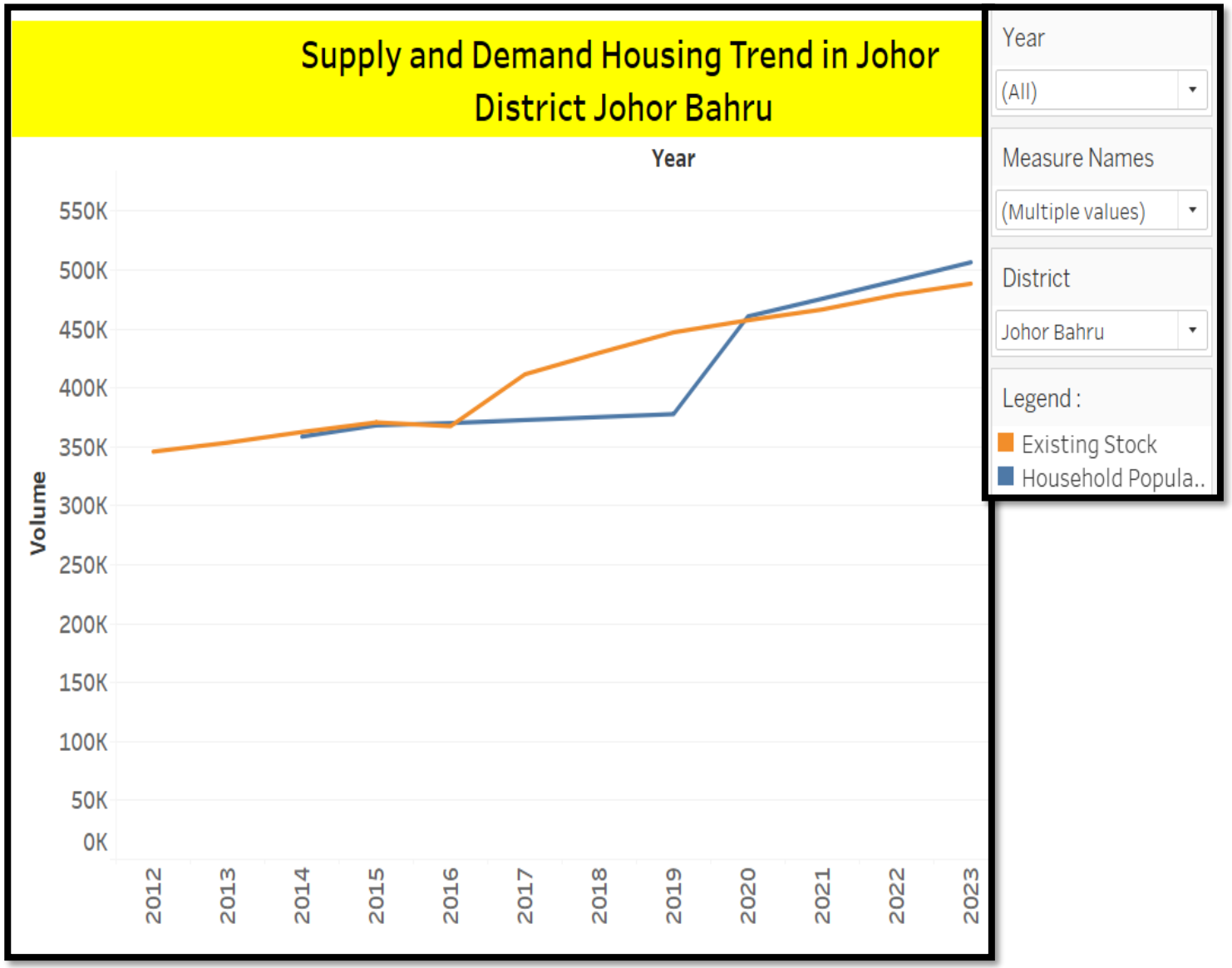




EXPLORATORY DATA ANALYSIS (EDA)

Specific Question 2	Data Segmentation	Data Segment Justification	Study Benefits
What is the trend in the supply of residential properties compared to the total household population? Study view from the perspective: 1. The condition of the current property stock 2. The condition of the current property stock + the upcoming supply 3. The forecasted amounts for the years 2024, 2025, and 2026	• Property Stock Report (NAPIC) • Household Population Census Report (DOSM)	• Knowing the number of residential supply according to State, District, Year, and Type of House based on the development stage, namely:- a. Completed stock (Existing Stock) b. Under Construction Supply (Incoming Supply) c. Not Constructed Supply (Planned Supply) For items (b) and (c), it refers to Future Supply Data. • Knowing the household population by State, District, and Year	• Providing an initial overview to stakeholders regarding the trend in the number of existing stock and upcoming supply, broken down by each type of house and compared with • Distribution of household population for each development area • Becoming an initial indicator to stakeholders regarding the supply status based on the number of potential homebuyers (household population) whether:- a. insufficient b. sufficient c. excessive

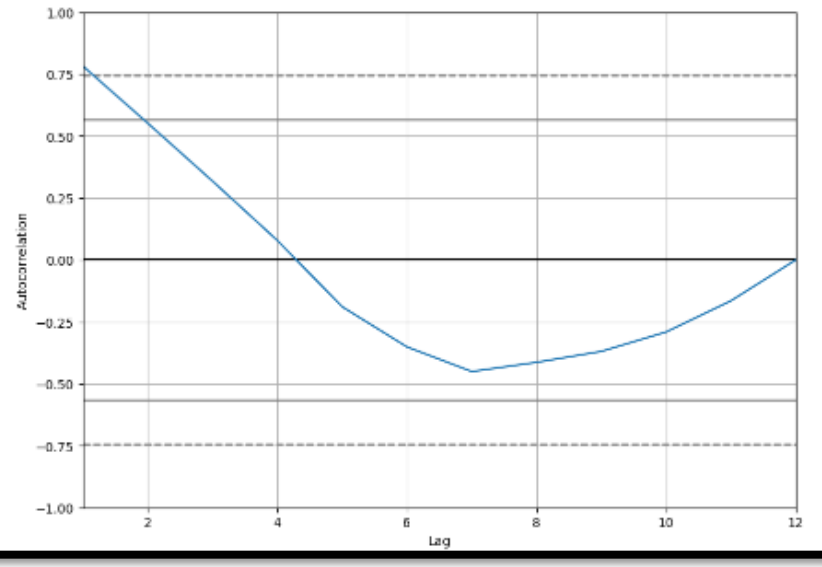
EXPLORATORY DATA ANALYSIS (EDA)




```python
from statsmodels.tsa.stattools import adfuller
test_result=adfuller(filtered_data)
from statsmodels.tsa.stattools import adfuller

def adfuller_test(filtered_data):
    result = adfuller(filtered_data)
    labels = ['ADF Test Statistic', 'p-value', '#Lags Used', 'Number of Observations']
    for value, label in zip(result, labels):
        print(label + ' : ' + str(value))

    if result[1] <= 0.05:
        print("Strong evidence against the null hypothesis (Ho), reject the null hypothesis. Data is stationary")
    else:
        print("Weak evidence against the null hypothesis, indicating it is non-stationary")

#adfuller_test(Supply['Total_ExistingStock'])

adfuller_test(filtered_data)
```

```
ADF Test Statistic : -40.366314410513446
p-value : 0.0
#Lags Used : 4
Number of Observations : 7
Strong evidence against the null hypothesis (Ho), reject the null hypothesis. Data is stationary
```

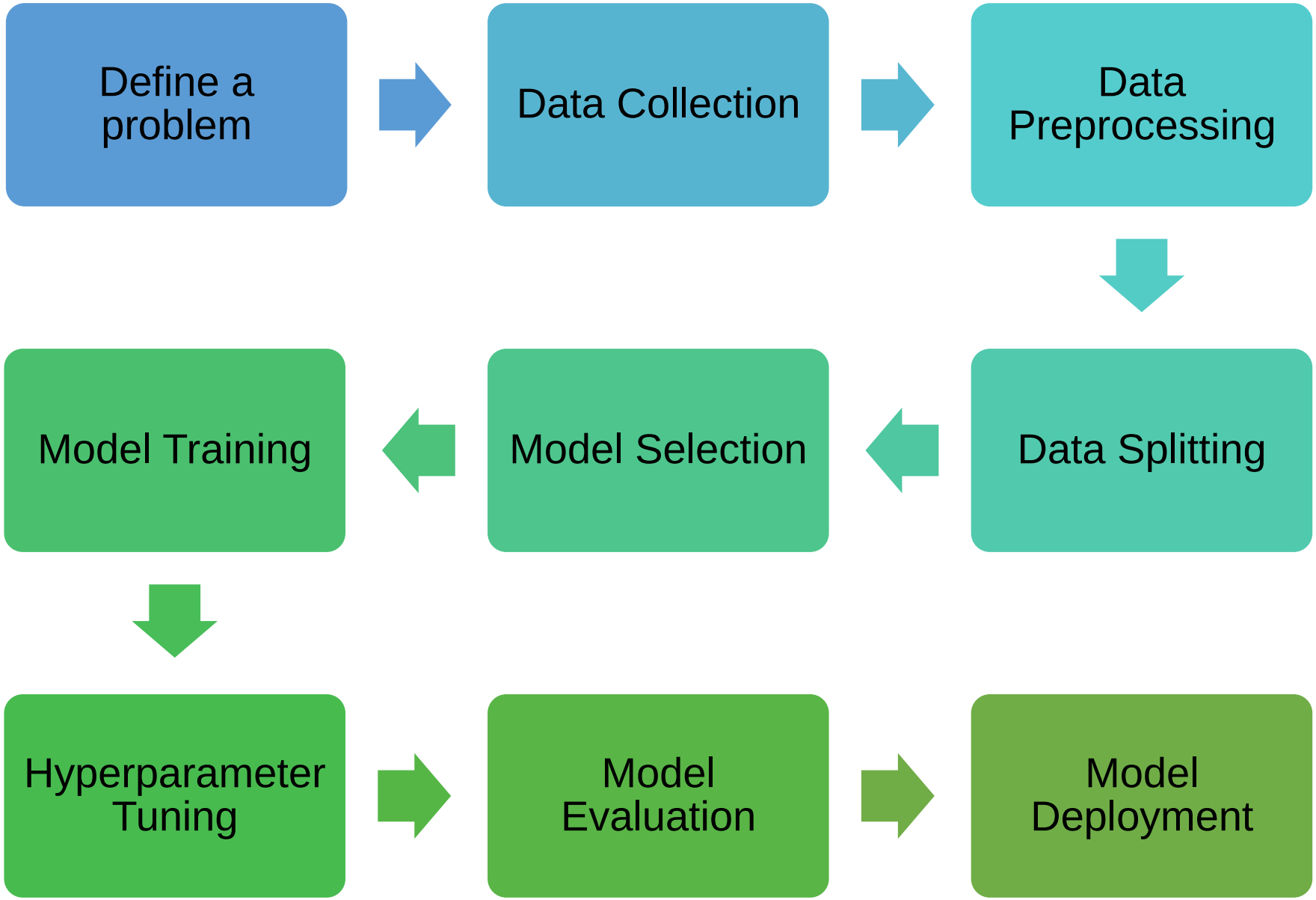
```python
from pandas.plotting import autocorrelation_plot
autocorrelation_plot(filtered_data)
plt.show()
```



EXPLORATORY DATA ANALYSIS (EDA)

Specific Question 3	Data Segmentation	Data Segment Justification	Study Benefits
Based on the results of studies 1 and 2 conducted previously, further questions need to answer as follow: 1. If the current property stock is insufficient, what type of property can be built and what is the ideal price range should be offer for a certain area? 2. If the property stock quantity is sufficient and excessive, the relevant parties need to consider who the target group is that is deemed potential buyers.	• Property Stock Report (NAPIC) • Property Status Report (NAPIC) • Household Income Census Report (DOSM)	• Knowing the number of residential supply by State, District, Year, Price Range, and Maximum Price Knowing the status of unsold residential units at the existing stock performance (the building has been 100% completed with CCC). Knowing the median household income by State, District, Year, and maximum home affordability level	• Providing an initial overview to stakeholders regarding the range of constructed housing prices versus household income. Becoming a guide for stakeholders regarding more competitive prices and types of housing based on the number of potential homebuyers and household income. Measuring the ability to own/buy a house, namely: a. Income multiple median b. Housing cost burden By Bank Negara Malaysia (BNM)

MODELLING AND ANALYSIS (MACHINE LEARNING)



MODELLING AND ANALYSIS (MACHINE LEARNING)

6. Untuk memudahkan aturan coding, maklumat yang hendak diuji iaitu Y Target = Total_ExistingStock bagi setiap daerah. Pengujian time series ini dilakukan mengikut peringkat daerah dan dibandingkan dengan daerah yang lain dalam negeri yang sama secara by default disetkan sebagai filtered_data

```
[ ]  1  district = 'Johor Bahru'
     2
     3  filtered_data = Supply[Supply['District']==district]['Total_ExistingStock']
```

NewCombineForecast_ExistingJohor.ipynb

File Edit View Insert Runtime Tools Help Last saved at 1:23 PM

+ Code + Text

```
1  values = filtered_data.values.reshape(-1,1)
2  values
```

```
array([[345970],
       [353563],
       [362598],
       [370753],
       [367594],
       [411564],
       [429798],
       [447180],
       [457425],
       [466551],
       [479106],
       [488458]])
```

```
[ ]  1  from sklearn.preprocessing import MinMaxScaler
     2
     3  scaler = MinMaxScaler()
     4
     5  scaled_values = scaler.fit_transform(values)
     6
     7  scaled_values[0:20]
```

```
array([[0.       ],
       [0.0532887 ],
       [0.11669755],
```

```
[ ]  1  test_ratio = 0.2 #need update here if needed
     2
     3  train_size = int(len(filtered_data) * (1-test_ratio))
     4
     5  train_data = scaled_values[:train_size]
     6  test_data = scaled_values[train_size:]
     7
     8  print(f'Train size: {train_size}, Test size: {len(test_data)}')
```

```
Train size: 9, Test size: 3
```

```
[ ]  1  def createSeq(dat, time_steps=1):
     2    X, y = [],[]
     3    for i in range(len(dat)-time_steps):
     4      X.append(dat[i:(i+time_steps),0])
     5      y.append(dat[i+time_steps, 0])
     6    return np.array(X), np.array(y)
```

```
1  time_steps = 2 # update here if needed
2
3  X_train, y_train = createSeq(train_data, time_steps)
4  X_test, y_test = createSeq(test_data, time_steps)
```

```
[ ]  1  X_train
```

```
array([[0.       , 0.0532887 ],
       [0.0532887 , 0.11669755],
       [0.11669755, 0.17393044],
       [0.17393044, 0.15176015],
       [0.15176015, 0.46034754],
       [0.46034754, 0.58831621],
       [0.58831621, 0.71030543]])
```

```
[ ]  1  y_train
```

```
array([0.11669755, 0.17393044, 0.15176015, 0.46034754, 0.58831621,
       0.71030543, 0.78220622])
```

MODELLING AND ANALYSIS (MACHINE LEARNING)

NewCombineForecast_ExistingJohor.ipynb

File Edit View Insert Runtime Tools Help Last saved at 1:23 PM

+ Code + Text

Pengujian 1 : Menggunakan Tensorflow Sequential Model

```
1  import tensorflow as tf
2
3  model = tf.keras.models.Sequential()
4
5  model.add(tf.keras.layers.Input(shape=(X_train.shape[1],1)))
6  model.add(tf.keras.layers.LSTM(20, return_sequences=False))
7  model.add(tf.keras.layers.Dense(1))
8
9  model.compile(optimizer='adam', loss='mse')
10
11 model.summary()
```

```
Model: "sequential_10"
```

Layer (type)	Output Shape	Param #
lstm_10 (LSTM)	(None, 20)	1,760
dense_10 (Dense)	(None, 1)	21

```
Total params: 1,781 (6.96 KB)
Trainable params: 1,781 (6.96 KB)
Non-trainable params: 0 (0.00 B)
```

```
[ ] 1  model.fit(X_train, y_train, epochs=200, batch_size=X_train.shape[0],
       validation_data=(X_test, y_test), verbose=1)
```

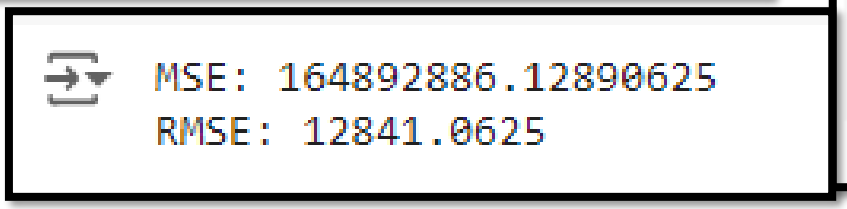
```
Epoch 1/200
1/1 ━━━━━━━━━━━━━━━━━━━━ 2s 2s/step - loss: 0.3188 - val_loss: 1.3455
Epoch 2/200
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 48ms/step - loss: 0.3132 - val_loss: 1.3253
```

```
[ ] 1  predicted = model.predict(X_test)
2
3
4  # following two lines is to un-normalized the value to readable values
5
6  predicted = scaler.inverse_transform(predicted)
7
8  y_test_unscaled = scaler.inverse_transform(y_test.reshape(-1,1))
9
10 print(f"Predicted 2024: {predicted}")
11
12 next = model.predict(np.array(predicted).reshape(-1,1))
13
14 print(f"Predicted 2025: {scaler.inverse_transform(next)}")
15
16 next = model.predict(np.array(next).reshape(-1,1))
17
18 print(f"Predicted 2026: {scaler.inverse_transform(next)}")
19
```

```
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 154ms/step
Predicted 2024: [[475616.94]]
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 159ms/step
Predicted 2025: [[670714.56]]
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 19ms/step
Predicted 2026: [[448519.56]]
```

```
1  from sklearn.metrics import mean_squared_error
2  import math
3
4  mse = mean_squared_error(y_test_unscaled, predicted)
5  rmse_tsm = math.sqrt(mse)
6
7  print(f'MSE: {mse}')
8  print(f'RMSE: {rmse_tsm}')
```

```
MSE: 164892886.12890625
RMSE: 12841.0625
```



MODELLING AND ANALYSIS (MACHINE LEARNING)

Pengujian 2 : Menggunakan AutoRegressive Integrated Moving Average (ARIMA) Model

```
[ ]  1  import pandas as pd
     2  import matplotlib.pyplot as plt
     3  import numpy as np
     4  import statsmodels.api as sm
```

```
     1  from pylab import rcParams
     2  rcParams['figure.figsize'] = 10, 7
     3  filtered_data.plot()
```

```
     1  # fit an ARIMA model and plot residual errors
     2  from statsmodels.tsa.arima.model import ARIMA
     3  from matplotlib import pyplot
     4  # fit model
     5  model = ARIMA (filtered_data, order = (1,1,1))
     6  model_fit = model.fit()
     7  # summary of fit model
     8  print(model_fit.summary())
```

```
                               SARIMAX Results
==============================================================================
Dep. Variable:     Total_ExistingStock   No. Observations:                   12
Model:                  ARIMA(1, 1, 1)   Log Likelihood                -119.439
Date:                 Wed, 23 Oct 2024   AIC                            244.878
Time:                         05:23:11   BIC                            246.072
Sample:                              0   HQIC                           244.126
                                   - 12
Covariance Type:                   opg
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
ar.L1          0.9934      0.426      2.334      0.020       0.159       1.828
ma.L1         -0.9764      0.839     -1.164      0.244      -2.620       0.667
sigma2     2.587e+08   2.74e-09   9.43e+16      0.000    2.59e+08    2.59e+08
===================================================================================
Ljung-Box (L1) (Q):                   0.28   Jarque-Bera (JB):                 9.58
Prob(Q):                              0.60   Prob(JB):                         0.01
Heteroskedasticity (H):               0.09   Skew:                             1.74
Prob(H) (two-sided):                  0.04   Kurtosis:                         5.97
===================================================================================
```

```
     1  from sklearn.metrics import mean_squared_error
     2  from math import sqrt
     3  # To Forecast
     4  # split into train and test sets
     5  X = filtered_data
     6  size = int(len(X) * 0.70)
     7  train, test = X[0:size], X[size:len(X)]
     8  history = [x for x in train]
     9  predictions = list()
    10  # walk-forward validation
    11  for t in range(len(test)):
    12    model = ARIMA (history, order = (1, 1, 1))
    13    model_fit = model.fit()
    14    output = model_fit.forecast()
    15    yhat = output[0]
    16    predictions.append(yhat)
    17    obs = test.iloc[t]
    18    history.append(obs)
    19    print('predicted=%f, expected=%f' % (yhat, obs))
    20  # evaluate forecasts
    21  rmse_arima = sqrt(mean_squared_error(test, predictions))
    22  print('Test RMSE: %.3f' % rmse_arima)
```

```
predicted=461109.420914, expected=457425.000000
/usr/local/lib/python3.10/dist-packages/statsmodels/tsa/statespace/sarimax.py:966: User
  warn('Non-stationary starting autoregressive parameters'
/usr/local/lib/python3.10/dist-packages/statsmodels/tsa/statespace/sarimax.py:978: User
  warn('Non-invertible starting MA parameters found.'
predicted=479785.042934, expected=479106.000000
predicted=492274.815818, expected=488458.000000
Test RMSE: 3513.285
```

predicted=479785.042934, expected=479106.000000
predicted=492274.815818, expected=488458.000000
Test RMSE: 3513.285

MODELLING AND ANALYSIS (MACHINE LEARNING)

Pengujian 3 : Menggunakan Error, Trend, Seasonality (ETS) Model

```
[ ]  1  import pandas as pd
     2  import matplotlib.pyplot as plt
     3  import numpy as np
     4  import statsmodels.api as sm
```

```
[ ]  1  Supply.head()
     2  Supply.describe()
     3  Supply.set_index('Year',inplace=True)
```

```
     1  from pylab import rcParams
     2  rcParams['figure.figsize'] = 10, 7
```

```
     1  # fit an ETS model and plot residual errors
     2  from statsmodels.tsa.holtwinters import SimpleExpSmoothing
     3  from matplotlib import pyplot
     4  # fit model
     5  model = SimpleExpSmoothing(filtered_data)
     6  model_fit = model.fit()
     7  # summary of fit model
     8  print(model_fit.summary())
```

```
                       SimpleExpSmoothing Model Results
==============================================================================
Dep. Variable:     Total_ExistingStock   No. Observations:                   12
Model:              SimpleExpSmoothing   SSE                 3233234242.675
Optimized:                        True   AIC                        236.942
Trend:                            None   BIC                        237.912
Seasonal:                         None   AICC                       242.656
Seasonal Periods:                 None   Date:               Wed, 23 Oct 2024
Box-Cox:                         False   Time:                       05:23:12
Box-Cox Coeff.:                   None
==============================================================================
                       coeff                 code              optimized
------------------------------------------------------------------------------
smoothing_level            0.9950000             alpha                 True
initial_level              3.4597e+05              1.0                False
------------------------------------------------------------------------------
```

```
     1  from sklearn.metrics import mean_squared_error
     2  from math import sqrt
     3  # To Forecast
     4  # split into train and test sets
     5  X = filtered_data
     6  size = int(len(X) * 0.70)
     7  train, test = X[0:size], X[size:len(X)]
     8  history = [x for x in train]
     9  predictions = list()
    10  # walk-forward validation
    11  for t in range(len(test)):
    12    model = SimpleExpSmoothing (history)
    13    model_fit = model.fit()
    14    output = model_fit.forecast()
    15    yhat = output[0]
    16    predictions.append(yhat)
    17    obs = test.iloc[t]
    18    history.append(obs)
    19    print('predicted=%f, expected=%f' % (yhat, obs))
    20  # evaluate forecasts
    21  rmse_ets = sqrt(mean_squared_error(test, predictions))
    22  print('Test RMSE: %.3f' % rmse_ets)
```

```
predicted=447092.628656, expected=457425.000000
predicted=457373.338143, expected=466551.000000
predicted=466505.111691, expected=479106.000000
predicted=479042.995558, expected=488458.000000
Test RMSE: 10469.146
```

MODELLING AND ANALYSIS (MACHINE LEARNING)

NewCombineForecast_ExistingJohor.ipynb

File Edit View Insert Runtime Tools Help Last saved at 1:23 PM

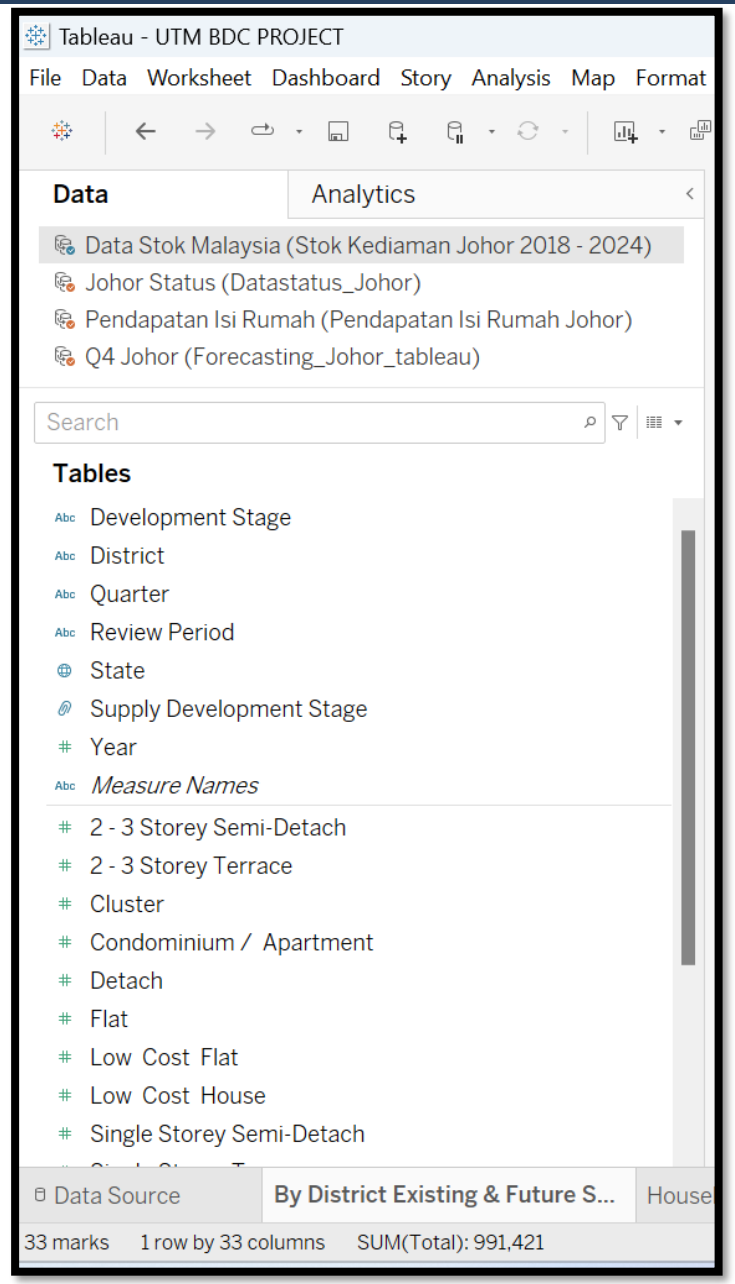
+ Code + Text

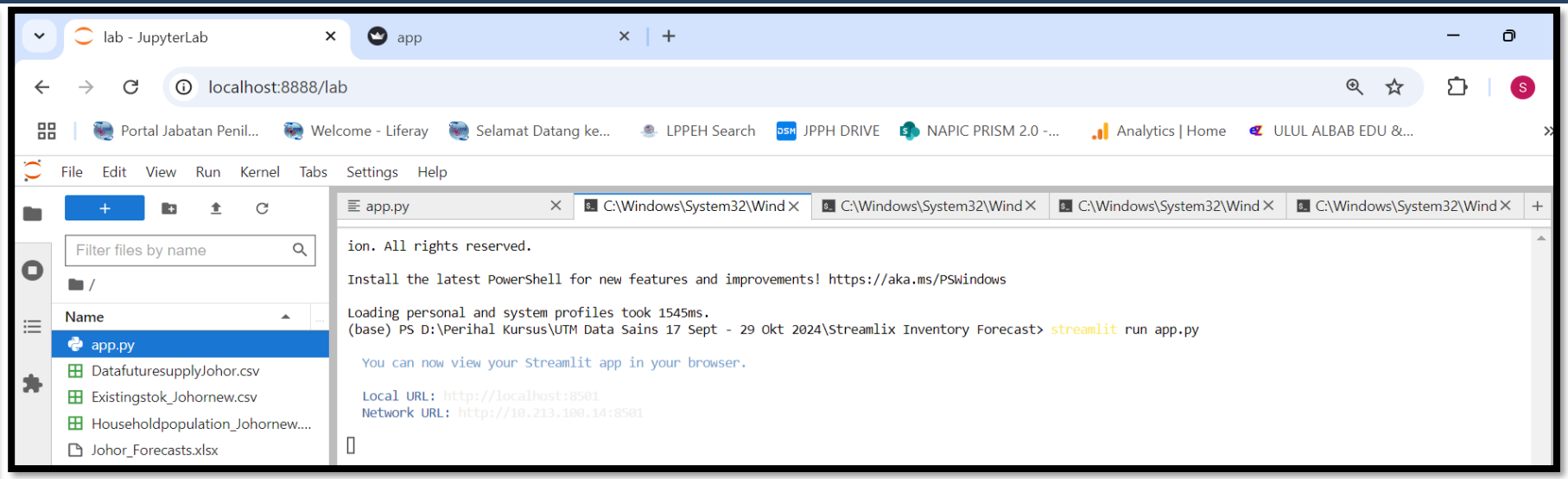
10. Setelah 3 model bagi forecasting time series dilakukan, maka perbandingan antara 3 model tersebut perlu dilakukan bagi memilih model yang memberikan nilai Root Mean Square Error (RMSE) yang paling munasabah diterima dan digunapakai

```
[ ]  1  # Print performance metrics
     2  print(f"RMSE TSM: {rmse_tsm:.2f}")
     3  print(f"RMSE ARIMA: {rmse_arima:.2f}")
     4  print(f"RMSE ETS: {rmse_ets:.2f}")
```

```
RMSE TSM: 12841.06
RMSE ARIMA: 3513.29
RMSE ETS: 10469.15
```

Dashboard and Streamlit App

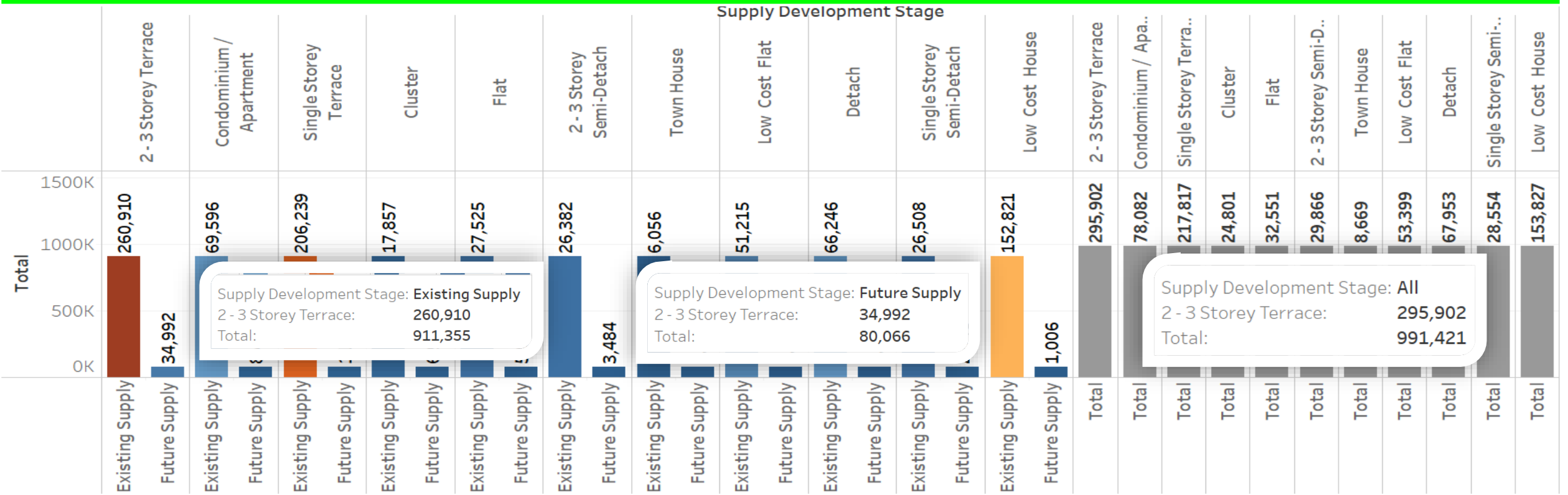




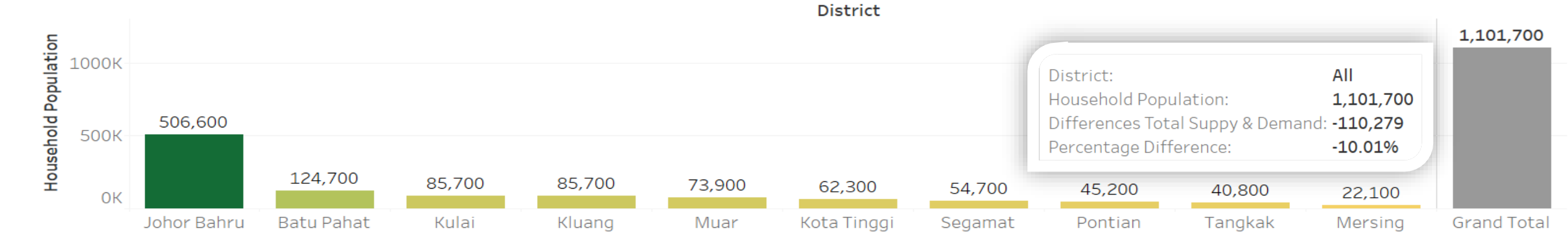
Question 1

Year: 2023 | State: Johor | District: (All)

Overview Existing & Future Residential Supply in Johor District All Year 2023



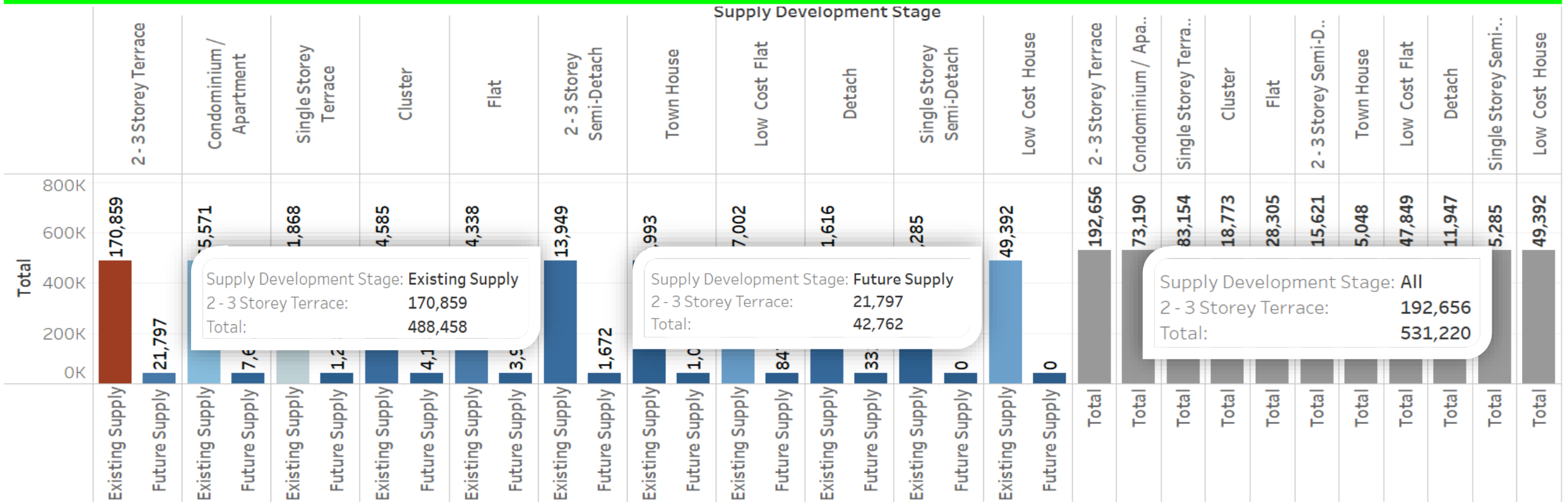
Overview Household Population in Johor District All Year 2023



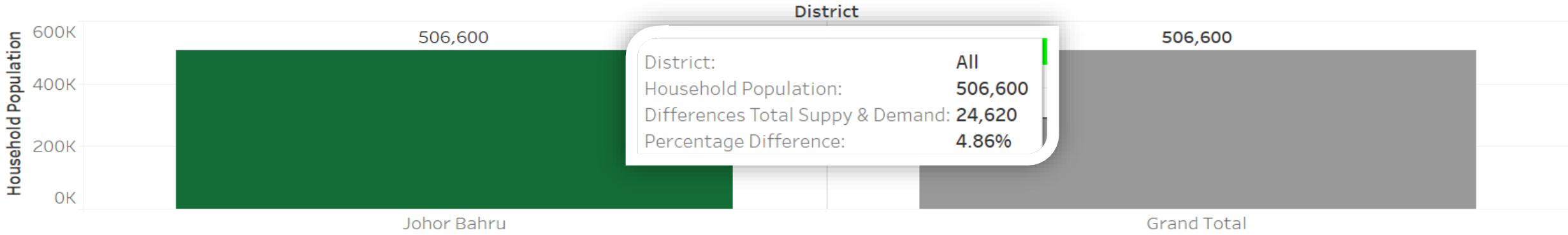
Year: 2023 | State: Johor | District: Johor Bahru

Question 1

Overview Existing & Future Residential Supply in Johor District Johor Bahru Year 2023



Overview Household Population in Johor District Johor Bahru Year 2023



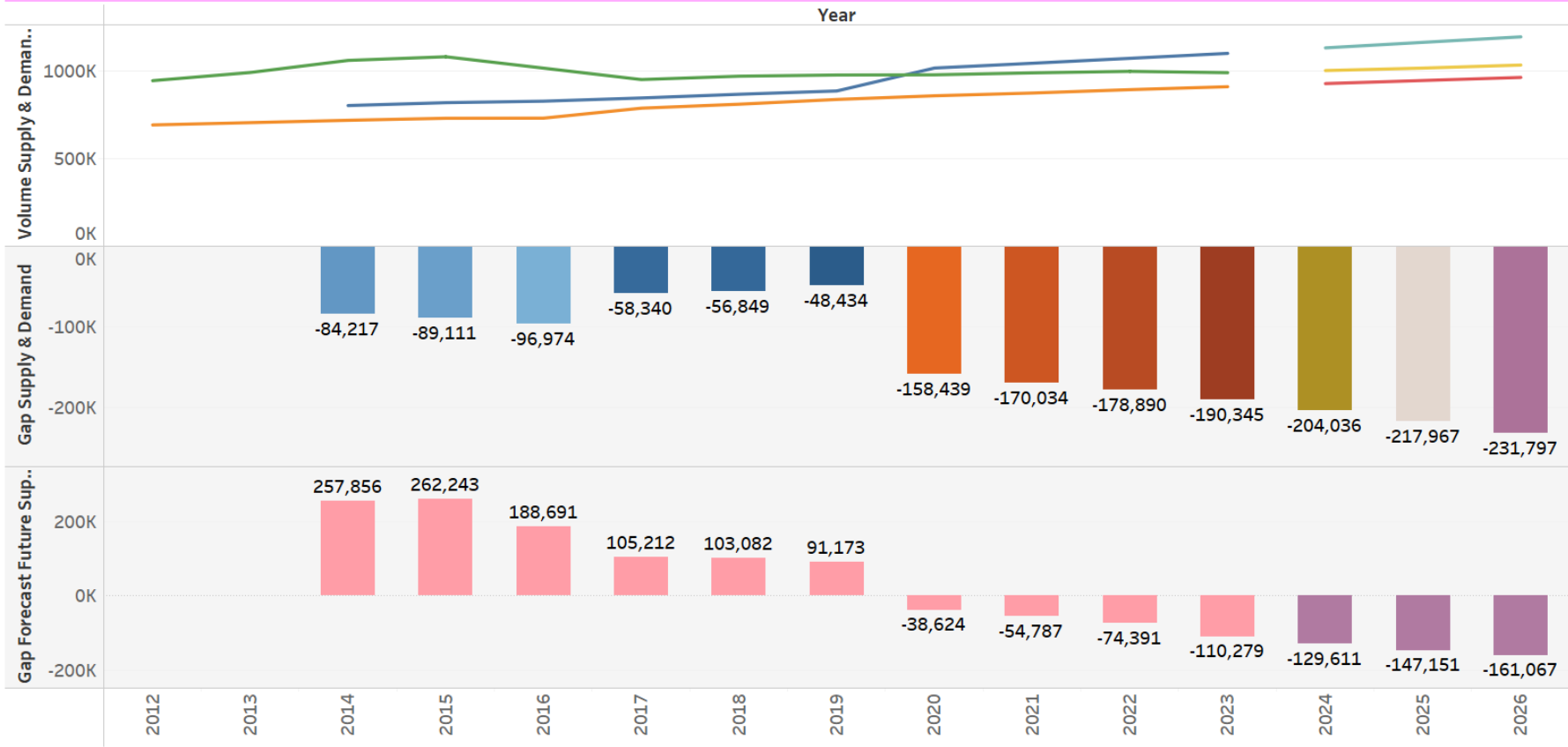
State: Johor

District: (All)

Question 2

Legend:
- Existing Stock
- Existing Stock + Future Supply
- Household Population
- Forecasting Existing Stock
- Forecasting Existing Stock + Future S..
- Forecasting Household Population
- Forecasting Existing Stock + Future S..
- Existing Stock + Future Supply vs Hou..

Supply and Demand Housing Gap Trend in Johor District All



Year: 2023
Existing Stock: 911,355

Year: 2026
Forecasting Existing Stock: 963,126

Year: 2023
Household Population: 1,101,700

Year: 2026
Forecasting Household Population: 1,191,614

Year: 2023
Existing Stock + Future Supply: 991,421

Year: 2026
Forecasting Existing Stock + Future Supply: 1,033,856

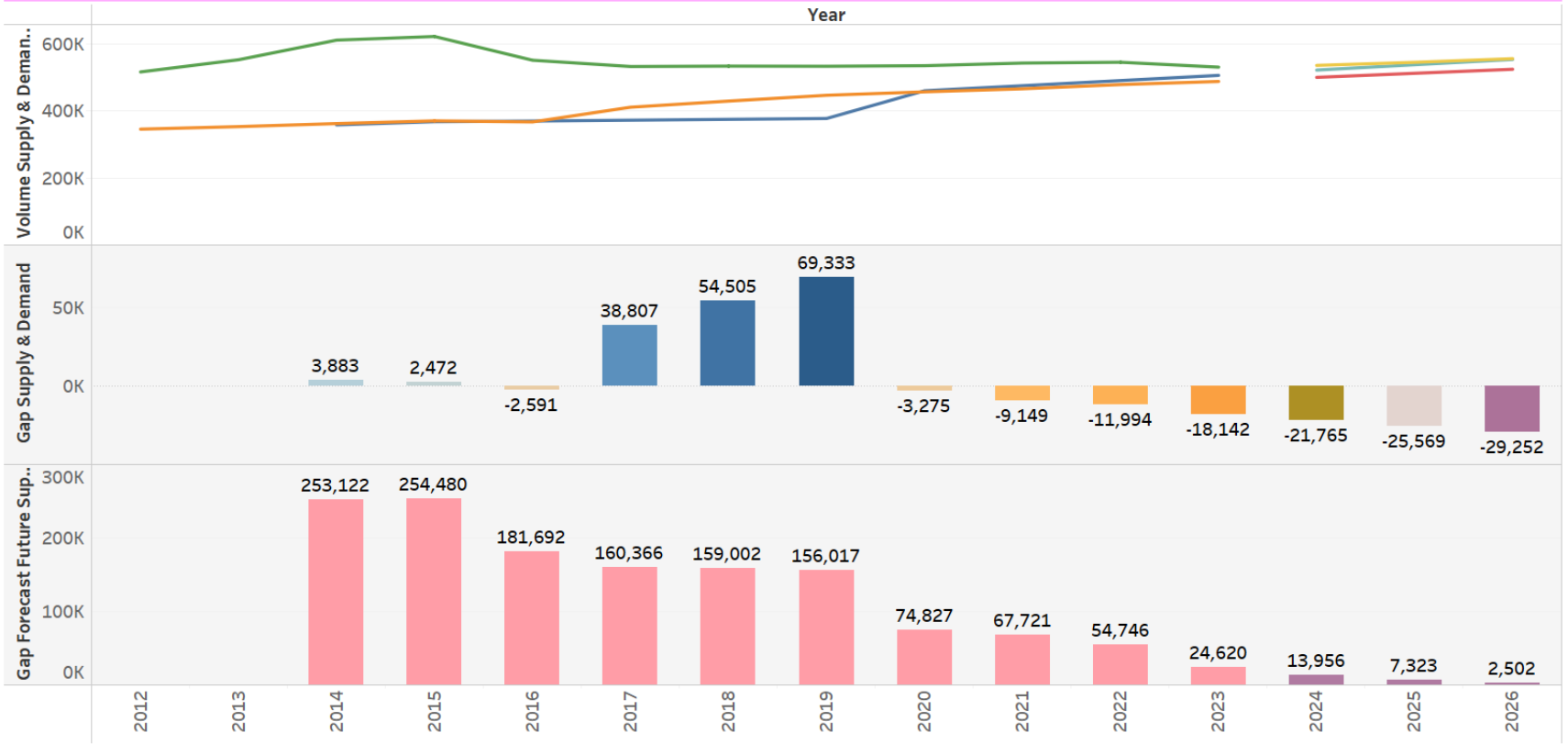
Question 2

State: Johor

District: Johor Bahru

Legend:
- Existing Stock
- Existing Stock + Future Supply
- Household Population
- Forecasting Existing Stock
- Forecasting Existing Stock + Future S..
- Forecasting Household Population
- Forecasting Existing Stock + Future S..
- Existing Stock + Future Supply vs Hou..

Supply and Demand Housing Gap Trend in Johor District Johor Bahru



Year: 2023
Existing Stock: 488,458

Year: 2023
Household Population: 506,600

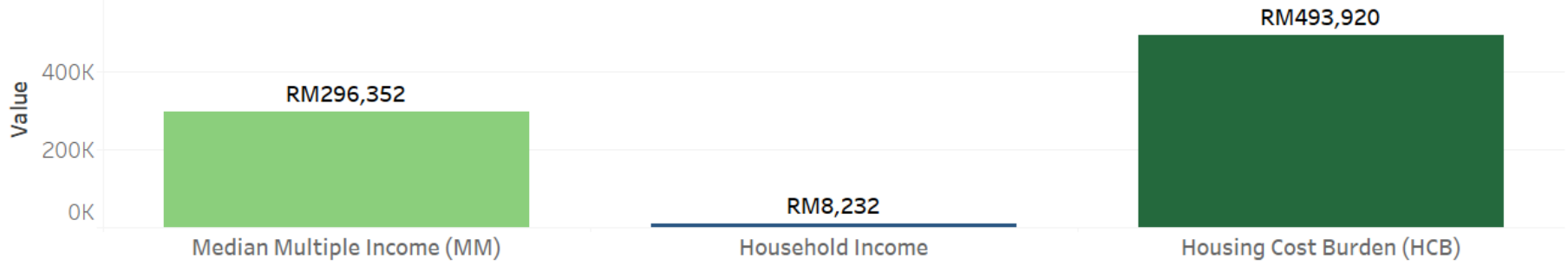
Year: 2023
Existing Stock + Future Supply: 531,220

Year: 2026
Forecasting Existing Stock: 522,916

Year: 2026
Forecasting Household Population: 553,644

Year: 2026
Forecasting Existing Stock + Future Supply: 554,670

Overview Household Income in Johor District Johor Bahru Year 2023



Question 3

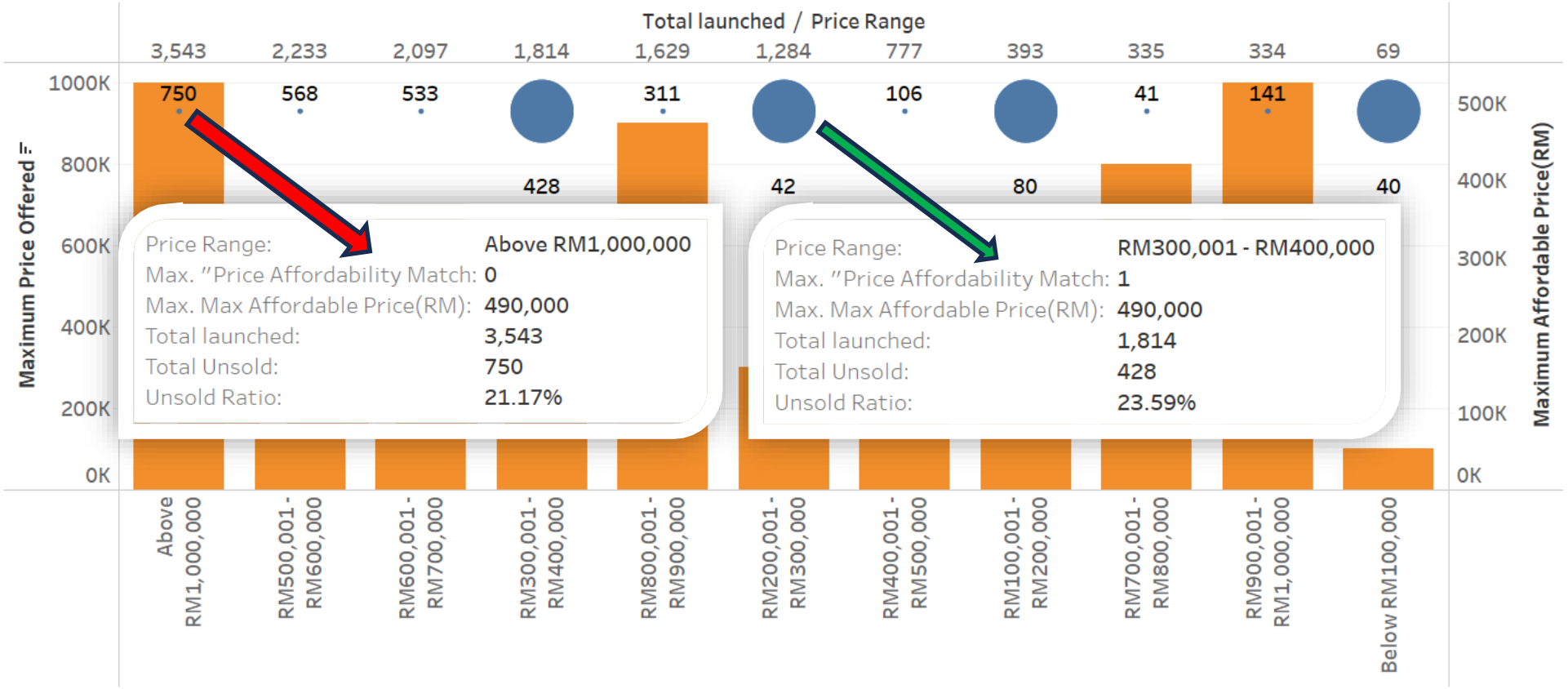
Year: 2023

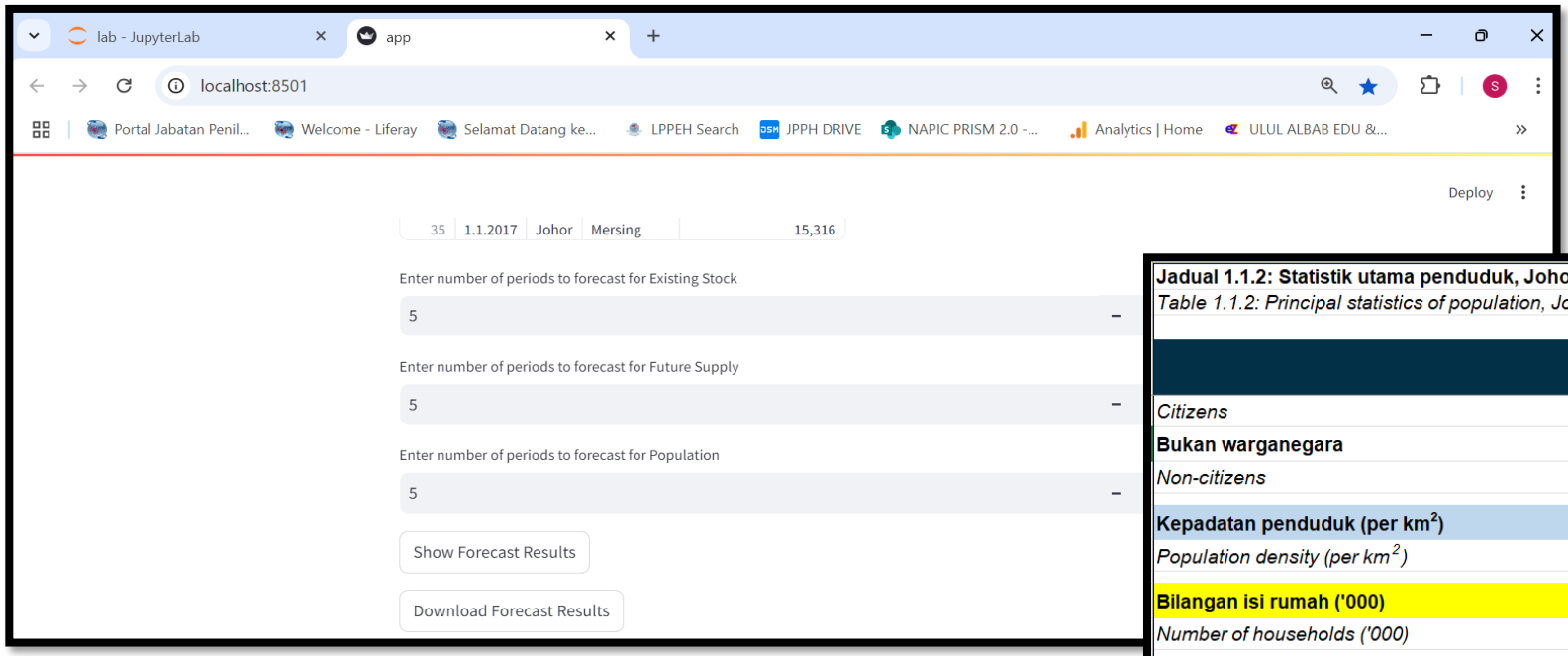
State: Johor

District: Johor Bahru

Property Type: (All)

Overhang Performance vs Maximum Affordability Household Income in Johor District Johor Bahru for year 2023





Jadual 1.1.2: Statistik utama penduduk, Johor Bahru, Johor, 2020-2024

Table 1.1.2: Principal statistics of population, Johor Bahru, Johor, 2020-2024

	2020	2021	2022	2023	2024[p]
Citizens					
Bukan warganegara	310	299	310	321	331
Non-citizens					
Kepadatan penduduk (per km^2)	1,605	1,609	1,617	1,653	1,689
Population density (per km^2)					
Bilangan isi rumah ('000)	460.7	475.7	491.1	506.6	522.4
Number of households ('000)					
saiz isi rumah	3				3.4
e household size					
an tempat kediaman ('000)	610				721.4
er of living quarters ('000)					

JOHOR | BATU PAHAT | JOHOR BAHRU | EGAMAT

Existing Stock Forecast Results

Batu Pahat

	Date	Forecast
2024-01-01 00:00:00	2024-12-31 00:00:00	99,651.8427
2025-01-01 00:00:00	2025-12-31 00:00:00	101,956.4754
2026-01-01 00:00:00	2026-12-31 00:00:00	104,260.8981
2027-01-01 00:00:00	2027-12-31 00:00:00	106,565.1109
2028-01-01 00:00:00	2028-12-31 00:00:00	108,869.1137

Johor Bahru

	Date	Forecast
2024-01-01 00:00:00	2024-12-31 00:00:00	500,677.1159
2025-01-01 00:00:00	2025-12-31 00:00:00	512,815.1954

Future Supply Forecast Results

Batu Pahat

	Date	Forecast
2024-01-01 00:00:00	2024-12-31 00:00:00	5,209.7802
2025-01-01 00:00:00	2025-12-31 00:00:00	5,222.6648
2026-01-01 00:00:00	2026-12-31 00:00:00	5,228.2393
2027-01-01 00:00:00	2027-12-31 00:00:00	5,230.6512
2028-01-01 00:00:00	2028-12-31 00:00:00	5,231.6947

Johor Bahru

	Date	Forecast
2024-01-01 00:00:00	2024-12-31 00:00:00	35,721.44
2025-01-01 00:00:00	2025-12-31 00:00:00	32,892.0291
2026-01-01 00:00:00	2026-12-31 00:00:00	31,754.9654

Johor Bahru

	Date	Forecast
2024-01-01 00:00:00	2024-12-31 00:00:00	522,442.1122

Population Forecast Results

Batu Pahat

	Date	Forecast
2024-01-01 00:00:00	2024-12-31 00:00:00	128,034.2525
2025-01-01 00:00:00	2025-12-31 00:00:00	131,368.2289
2026-01-01 00:00:00	2026-12-31 00:00:00	134,701.9292
2027-01-01 00:00:00	2027-12-31 00:00:00	138,035.3535
2028-01-01 00:00:00	2028-12-31 00:00:00	141,368.5019

Johor Bahru

	Date	Forecast
2024-01-01 00:00:00	2024-12-31 00:00:00	522,442.1122
2025-01-01 00:00:00	2025-12-31 00:00:00	538,282.7982
2026-01-01 00:00:00	2026-12-31 00:00:00	554,122.0581

CONCLUSION

Based on all the objectives, the study shows that DOSM data can provide additional input to NAPIC's publication
on the current residential property market.

Providing more comprehensive input to the Government in formulating housing-related policies and regulations.

The study can be extended for periodic publication based on the availability of a combination
of internal and external data.

This study is suitable for publication for public exposure.

THANK YOU