```
!pip install bs4
```

```
Collecting bs4
  Downloading bs4-0.0.2-py2.py3-none-any.whl.metadata (411 bytes)
Requirement already satisfied: beautifulsoup4 in /usr/local/lib/python3.10/dist-packages (from bs4) (4.12.3)
Requirement already satisfied: soupsieve>1.2 in /usr/local/lib/python3.10/dist-packages (from beautifulsoup4->bs4) (2.6)
Downloading bs4-0.0.2-py2.py3-none-any.whl (1.2 kB)
Installing collected packages: bs4
Successfully installed bs4-0.0.2
```

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib
```

Scapping all route available

```
from bs4 import BeautifulSoup
import requests
url = "https://mtcbus.tn.gov.in/Home/routewiseinfo"
req = requests.get(url)
soup = BeautifulSoup(req.content, "html.parser")
route_html=soup.find("select", {"name": "selroute"})
route_list = [ route.text for route in route_html.find_all("option")]
print(route_list)
```

```
['--Route--', '1', '101', '102', '102A', '102C', '102CT', '102K', '102K#', '102M', '102P', '102S', '102X', '104', '104A'
```

```
# removing heading
route_list = route_list[1:]
pd.DataFrame(route_list)
```

	0
0	1
1	101
2	102
3	102A
4	102C
...	...
667	T23C
668	T29
669	T29C
670	V51
671	V51X

672 rows × 1 columns

Creating empty dataframe

```
import pandas as pd

route_df = pd.DataFrame(columns = ['route_id', 'stop_id', 'stop_name'])
```

For having progress bar

```
!pip install tqdm
```

```
Requirement already satisfied: tqdm in /usr/local/lib/python3.10/dist-packages (4.66.6)
```

```
from tqdm import tqdm
```

Scrap Route Detail

```
import requests
from bs4 import BeautifulSoup
import pandas as pd
from tqdm import tqdm

# Initialize an empty list to store route data
route_data = []

# Iterate over each route in route_list
for route in tqdm(route_list):
    url = f"https://mtcbus.tn.gov.in/Home/routewiseinfo?csrf_test_name=3a87933c0527dbc37410c133a7f30868&selroute={route}&subm
    req = requests.get(url)
    soup = BeautifulSoup(req.content, "html.parser")

    # Find route details
    route_detail = soup.find("ul", {"class": "route"})

    if route_detail:  # Ensure route_detail is not None
        for stop in route_detail.find_all("li"):
            try:
                stop_id, stop_name = stop.text.split("\t")
                # Append data to the list, not to the DataFrame
                route_data.append({"route_id": route, "stop_id": stop_id, "stop_name": stop_name})
            except ValueError:
                # Handle cases where the text split doesn't work as expected
                print(f"Skipping stop on route {route} due to unexpected format: {stop.text}")

# Convert the list to a DataFrame once all data is collected
route_df = pd.DataFrame(route_data)

# Show the resulting DataFrame
print(route_df)
```

```
100%|██████████| 672/672 [21:11<00:00,  1.89s/it]     route_id stop_id             stop_name
0            1       1         THIRUVOTRIYUR
1            1       2  THIRUVOTRIYUR TEMPLE
2            1       3               THANGAL
3            1       4            ANNA NAGAR
4            1       5          ROYAPURAM P.S
...        ...     ...                   ...
9227      V51X      15           IRUMBULIYUR
9228      V51X      16          PERUNGALATHUR
9229      V51X      17          VANDALUR GATE
9230      V51X      18           VANDALUR ZOO
9231      V51X      19      KILAMBAKKAM B.S.

[9232 rows x 3 columns]
```

```
# saving into csv file
route_df.head()
route_df.to_csv('route_detail.csv')
```