Redbus Data Scraping with Selenium & Dynamic Filtering using Streamlit

Approach:

1. **Data Scraping:**

Web scraping with Selenium to extract bus details from RedBus and storing them in a MySQL database and **Dynamic Filtering using Streamlit Application**

Set up the environment:

Installed Selenium and the appropriate web driver for chrome browser.

➤ 'selenium' library is used for web scraping.

➤ 'time' library is used to add delays during web scraping to ensure elements have loaded.

Load the web page:

Used Selenium to navigate to the bus routes web page.

Extract the bus route details:

Located the elements containing the bus route details and extract the information.

Navigate through the pages:

Used Selenium to click the "Page Icon" button to navigate through all the pages and repeat the extraction process.

CODE:

```python
from selenium import webdriver
from selenium.webdriver.common.by import By
from selenium.webdriver.common.keys import Keys
from selenium.webdriver.support.ui import WebDriverWait
from selenium.webdriver.support import expected_conditions as EC
import time
import mysql.connector
# KSRTC BUS ROUTE EXTRACTION
driver = webdriver.Chrome()  # Optional argument, if not specified will search
path.
driver.get('https://www.redbus.in/online-booking/ksrtc-kerala'); #url to
scrape
time.sleep(10) # Let the user actually see something!
driver.maximize_window()
def extract_bus_routes():
    routes=[]
    body=driver.find_element(By.TAG_NAME,'body')
    for i in range(5):
        for j in range(2):
```

```python
            body.send_keys(Keys.PAGE_DOWN)
    time.sleep(10)
    route_details=driver.find_elements(By.CLASS_NAME,'route_details')
    for route in route_details:
        route_info=route.find_element(By.CLASS_NAME,'route')
        route_text=route_info.text
        source,destination=route_text.split(' to ')
        routes.append({"source":source,"destination":destination})
    return routes
all_routes=[]

pagination=driver.find_element(By.CLASS_NAME,'DC_117_paginationTable')
page_buttons=pagination.find_elements(By.TAG_NAME,'div')
for page_button in page_buttons:
    try:
        page_button.click()
        route_on_page=extract_bus_routes()
        all_routes.extend(route_on_page)
    except Exception as e:
        print(f"An error occured:{e}")
        break
driver.quit()
for route in all_routes:
    print(route)
# connecting my sql data base
```

Explanation:

1. **Open the Website**: The script navigates to the specified URL.

2. **Wait for the Page to Load**: Ensures that the page is fully loaded before proceeding.

3. **Extract Bus Routes**: Uses find_elements to locate all bus route elements on the current page and extracts their details in the format "Source to Destination".

4. **Page Navigation**:

   ➤ Locate the pagination container using a CLASS_NAME.

   ➤ Find all page buttons within the pagination container.

   ➤ Iterate through each page button, clicking it and then extracting the bus route details from the resulting page.

5. **Store in Dictionary**: Adds each route as a dictionary with keys "source" and "destination" to a list.

6. **Output the Dictionary**: After scraping all the pages, it prints the list containing dictionaries of source and destination for all routes.

Output:

```
{'source': 'Bangalore', 'destination': 'Kozhikode'}
{'source': 'Kozhikode', 'destination': 'Ernakulam'}
{'source': 'Kozhikode', 'destination': 'Bangalore'}
{'source': 'Ernakulam', 'destination': 'Kozhikode'}
{'source': 'Kozhikode', 'destination': 'Mysore'}
{'source': 'Kozhikode', 'destination': 'Thiruvananthapuram'}
{'source': 'Bangalore', 'destination': 'Kalpetta (kerala)'}
{'source': 'Mysore', 'destination': 'Kozhikode'}
{'source': 'Kalpetta (kerala)', 'destination': 'Bangalore'}
{'source': 'Kozhikode', 'destination': 'Thrissur'}
{'source': 'Thiruvananthapuram', 'destination': 'Kozhikode'}
{'source': 'Bangalore', 'destination': 'Kannur'}
{'source': 'Kozhikode', 'destination': 'Kottayam'}
{'source': 'Kannur', 'destination': 'Bangalore'}
{'source': 'Kottayam', 'destination': 'Kozhikode'}
{'source': 'Thrissur', 'destination': 'Kozhikode'}
{'source': 'Kozhikode', 'destination': 'Kalpetta (kerala)'}
{'source': 'Coimbatore', 'destination': 'Ooty'}
{'source': 'Kalpetta (kerala)', 'destination': 'Kozhikode'}
```

2. **Data Storage:**

   ➤ 'mysql.connector' library is used for connecting and interacting with MySQL databases

   ➤ Establishes a connection to the MySQL database.

   ➤ db.cursor() creates a cursor object which is used to execute SQL queries My SQL Database Connection:

```python
# connecting my sql data base
db = mysql.connector.connect(
        host='127.0.0.1',
        port='3306',
        user='root',
        password='123456789',
        database='redbus_db'
    )
if db:
        print('Connected to MySQL database')
else:
        print('Error connecting to MySQL database')
cursor= db.cursor()
```

```
cursor.execute("""CREATE TABLE IF NOT EXISTS bus_routes(
                ID integer auto_increment primary key,
                STATE TEXT,
                SOURCE text,
                DESTINATION text,
                BUS_NAME text,
                BUS_TYPE text,
                DEPARTING_TIME text,
                ARRIVING_TIME text,
                DURATION text,
                STAR_RATING float,
                PRICE decimal,
                SEATS_AVAILABLE text,
                BUS_CATEGORY text,
                AC_TYPE text,
                STAR_RANGE text)"""
                )
```

Output:

```
Connected to MySQL database
```

Explanation:

> ➢ Creates a table bus_routes if it doesn't already exist, with columns for source, destination, bus name, departure time, arrival time, duration, star rating, seats and price.

Bus details Scraping

```
# Scraping of Bus Details
driver = webdriver.Chrome()
def scrape_bus_details(source,destination):
    driver.get('https://www.redbus.in/online-booking/ksrtc-kerala'); #url to
scrape
    WebDriverWait(driver,10).until(EC.presence_of_element_located((By.CSS_SELE
CTOR, 'body')))
    driver.maximize_window()
    time.sleep(5) # Let the user actually see something!
    source_input= driver.find_element (By.ID,'txtSource')
    time.sleep(2)
```

```python
    source_input.click()
    driver.execute_script("arguments[0].value= '';",source_input)
    source_input.send_keys(source)
    time.sleep(5)
    source_input.send_keys(Keys.ENTER)
    time.sleep(2)
    dest_input= driver.find_element (By.ID,'txtDestination')
    time.sleep(2)
    dest_input.click()
    driver.execute_script("arguments[0].value= '';",dest_input)
    dest_input.send_keys(destination)
    time.sleep(5)
    dest_input.send_keys(Keys.ENTER)
    time.sleep(2)
    select_date="23-Jul-2024"
    dates=select_date.split("-")
    Calendar=driver.find_element(By.ID, 'txtOnwardCalendar').click()
    date=driver.find_elements(By.XPATH,'//*[@id="rb-calendar"]//li')

    for ele in date:
        if ele.text==dates[0]:
            print(ele.text)
            ele.click()
            break

    Search_button=driver.find_element (By.CSS_SELECTOR,
"button[class='D120_search_btn_v2 searchBuses']").click()
    time.sleep(20)
    WebDriverWait(driver,10).until(EC.presence_of_element_located((By.CSS_SELE
CTOR, 'body')))
    filters=driver.find_elements(By.XPATH,'//div[@class="filter-details f-12
d-color"]')
    for filter in filters:
        reset=driver.find_element(By.XPATH,'//*[@id="filter-
block"]/div/div[1]/span')
        reset.click()
    body=driver.find_element(By.TAG_NAME,'body')
    for i in range(50):
        for j in range(2):
            body.send_keys(Keys.PAGE_DOWN)

    govt_bus_detail=[]
    bus_items=driver.find_elements(By.XPATH,'//div[@class="clearfix bus-
item"]')
    WebDriverWait(driver,30).until(EC.presence_of_element_located((By.XPATH,'/
/div[@class="clearfix bus-item"]')))
    time.sleep(10)
    for bus in bus_items:
```

```python
        try:
            bus_name = bus.find_element(By.XPATH, './/div[@class="travels lh-
24 f-bold d-color"]').text
            if "KSRTC" in bus_name:
                bus_category="Government"
            else:
                bus_category="private"
            bus_type = bus.find_element(By.XPATH, './/div[@class="bus-type f-
12 m-top-16 l-color evBus"]').text
            if "Non" in bus_type:
                AC_type="NON AC"
            else:
                AC_type="AC"

            departure_time = bus.find_element(By.XPATH, './/div[@class="dp-
time f-19 d-color f-bold"]').text
            arrival_time = bus.find_element(By.XPATH, './/div[@class="bp-time
f-19 d-color disp-Inline"]').text
            duration=bus.find_element(By.XPATH,'//div[@class="dur l-color lh-
24"]').text
            star_rating=bus.find_element(By.XPATH,'.//div[@class="rating-sec
lh-24"]').text
            star=float(star_rating)
            if star<2:
                star_range='1-2'
            elif 2<=star<3:
                star_range='2-3'
            elif 3<=star<4:
                star_range='3-4'
            else:
                star_range='4 above'
            fare = bus.find_element(By.XPATH, ".//span[contains(@class,'f-
19')]").text
            seats_avail=
bus.find_element(By.XPATH,".//div[contains(@class,'seat-left')]").text
            seats=seats_avail.split(' ')[0]
            state="Kerala"
            govt_bus_detail.append((bus_name, bus_type, departure_time,
arrival_time,duration,star_rating,fare,seats,bus_category,AC_type,star_range))
            cursor.execute("""INSERT INTO
bus_routes(STATE,SOURCE,DESTINATION,BUS_NAME,BUS_TYPE,DEPARTING_TIME,ARRIVING_
TIME,DURATION,STAR_RATING,PRICE,SEATS_AVAILABLE,BUS_CATEGORY,AC_TYPE,STAR_RANG
E
        )VALUES(%s,%s,%s,%s,%s,%s,%s,%s,%s,%s,%s,%s,%s,%s)""",(state,source,de
stination,bus_name,bus_type,departure_time,arrival_time,duration,star_rating,f
are,seats,bus_category,AC_type,star_range))
            db.commit()
        except Exception as e:
```

```
            print(f"An error occured:{e}")
            continue
    print(govt_bus_detail)
for route in all_routes:
    if "source" in route and "destination" in route:
            scrape_bus_details(route["source"],route["destination"])


driver.quit()
cursor.close()
db.close()
```

Explanation:

- 'all_routes' defines a list of dictionaries, each containing a source and a destination city

- Initialize the Chrome WebDriver to control the Chrome browser

- Opens the Redbus website.

- Waits for the source and destination input field to be present and then clears it by click()

- Enters the source and destination city and send "ENTER KEY" to select city from drop down

- Waits for the calendar to be clickable and selects date.

- Adds a delay to ensure the search results page has loaded

- Finds all bus items on the page.

- Iterates over each bus item to extract details like bus name, departure time, arrival time, price, etc.

- Inserts the extracted bus details into the bus_routes table.

- Commits the transaction to save the data.

- Iterates over the list of source-destination pairs and calls the scrape_bus_details function for each pair.

My SQL Data Base Table Output:
1. KSRTC (KERALA STATE TRANSPORT CORPORATION)

URL: https://www.redbus.in/online-booking/ksrtc-kerala

ID	STATE	SOURCE	DESTINATION	BUS_NAME	BUS_TYPE	DEPARTI	ARRIVIN	DURATIO	STAR	PRIC	SEAT	BUS_CATEGORY	AC_TYPE	STAR_RANGE
1	Kerala	Bangalore	Kozhikode	KSRTC (Kerala) - 581	Super Deluxe Non A...	10:03	18:48	08h 45m	4.1	469	27	Government	NON AC	4 above
2	Kerala	Bangalore	Kozhikode	KSRTC (Kerala) - 52	SWIFT-GARUDA A/C ...	12:01	19:55	08h 45m	3.3	627	37	Government	AC	3-4
3	Kerala	Bangalore	Kozhikode	KSRTC (Kerala) - 1378	AC MULTI AXLE	13:46	20:50	08h 45m	2.6	729	46	Government	AC	2-3
4	Kerala	Bangalore	Kozhikode	KSRTC (Kerala) - 129	SWIFT-GARUDA A/C ...	14:15	22:50	08h 45m	4.1	627	41	Government	AC	4 above
5	Kerala	Bangalore	Kozhikode	KSRTC (Kerala) - 235	Swift Deluxe Non AC ...	15:00	00:01	08h 45m	4.6	620	38	Government	NON AC	4 above

bus routes 5 ✕

2. APSRTC (ANDRA PRADESH STATE TRANSPORT CORPORATION)

URL: https://www.redbus.in/online-booking/apsrtc/?utm_source=rtchometile

ID	STATE	SOURCE	DESTINATION	BUS_NAME	BUS_TYPE	DEPARTI	ARRIVIN	DURATIO	STAR.	PRIC	SEA1	BUS_CATEGORY	AC_TYPE	STAR_RANGE
4651	Andra Pradesh	Vijayawada	Hyderabad	TSRTC - 9887	RAJDHANI (A.C. Se...	00:10	06:20	06h 10m	3.1	374	37	Government	AC	3-4
4652	Andra Pradesh	Vijayawada	Hyderabad	TSRTC - 1424	GARUDA PLUS (VOLV...	00:20	06:10	06h 10m	3.5	374	41	Government	AC	3-4
4653	Andra Pradesh	Vijayawada	Hyderabad	TSRTC - 31362	RAJDHANI (A.C. Se...	00:30	06:00	06h 10m	3.3	374	37	Government	AC	3-4
4654	Andra Pradesh	Vijayawada	Hyderabad	TSRTC - 42162	SUPER LUXURY (NON...	01:10	07:20	06h 10m	4.3	396	33	Government	AC	4 above
4655	Andra Pradesh	Vijayawada	Hyderabad	TSRTC - 30710	RAJDHANI (A.C. Se...	01:20	07:00	06h 10m	4.1	374	37	Government	AC	4 above

bus_routes 6 ×   Apply   R

3. TSRTC(TELANGANA STATE TRANSPORT CORPORATION)

URL: https://www.redbus.in/online-booking/tsrtc/?utm_source=rtchometile

Result Grid | 📊 | 🔄 Filter Rows: [   ] | Edit: 📝 🖺 🖺 | Export/Import: 🖺 🖺 | Wrap Cell Content: 🔠

ID	STATE	SOURCE	DESTINATION	BUS_NAME	BUS_TYPE	DEPARTI	ARRIVIN	DURATIO	STAR.	PRIC	SEA1	BUS_CATEGO	AC_T	STAR_RA
11707	Telangana	Khammam	Hyderabad	TSRTC - 64959	SUPER LUXURY (NON-AC, 2 + 2 PUSH BACK)	05:40	10:10	03h 55m	4.5	320	28	Government	AC	4 above
11708	Telangana	Khammam	Hyderabad	TSRTC - 64187	SUPER LUXURY (NON-AC, 2 + 2 PUSH BACK)	05:45	10:30	03h 55m	3.7	320	31	Government	AC	3-4
11709	Telangana	Khammam	Hyderabad	TSRTC - 8633	SUPER LUXURY (NON-AC, 2 + 2 PUSH BACK)	06:45	10:55	03h 55m	3.8	320	31	Government	AC	3-4
11710	Telangana	Khammam	Hyderabad	TSRTC - 64826	SUPER LUXURY (NON-AC, 2 + 2 PUSH BACK)	07:10	11:30	03h 55m	4.8	320	31	Government	AC	4 above

4. SBSTC (SOUTH BENGAL STATE TRANSPORT CORPORATION)

URL: https://www.redbus.in/online-booking/south-bengal-state-transport-corporation-sbstc/?utm_source=rtchometile

ID	STATE	SOURCE	DESTINATION	BUS_NAME	BUS_TYPE	DEPARTI	ARRIVIN	DURATIO	STAR.	PRIC	SEA1	BUS_CATEGO	AC_TYPE	STAR_RANGE
14502	South Bengal	Burdwan	Kolkata	SBSTC-BURDWAN - KOLKA...	Non AC Seater (2+3)	05:45	07:45	02h 00m	4.2	95	48	Government	NON AC	4 above
14503	South Bengal	Burdwan	Kolkata	SBSTC-BURDWAN - KARUN...	Non AC Seater (2+3)	05:50	08:20	02h 00m	4.2	100	31	Government	NON AC	4 above
14504	South Bengal	Burdwan	Kolkata	SBSTC-BURDWAN - KOLKA...	Non AC Seater (2+3)	06:15	08:15	02h 00m	3.8	95	50	Government	NON AC	3-4
14505	South Bengal	Burdwan	Kolkata	SBSTC-BURDWAN - KARUN...	Non AC Seater (2+3)	06:25	08:55	02h 00m	4.2	100	16	Government	NON AC	4 above
14506	South Bengal	Burdwan	Kolkata	SBSTC-BURDWAN - KARUN...	Non AC Seater (2+3)	07:10	09:40	02h 00m	4.2	100	31	Government	NON AC	4 above

bus_routes 6 ×

5. WBSTC(WEST BENGAL STATE TRANSPORT CORPORATION)

URL: https://www.redbus.in/online-booking/west-bengal-transport-corporation?utm_source=rtchometile

ID	STATE	SOURCE	DESTINATION	BUS_NAME	BUS_TYPE	DEPARTI	ARRIVIN	DURATIO	STAR.	PRIC	SEA1	BUS_CATEGO	AC_TYPE	STAR_RANGE
16613	West Bengal	Kolkata	Digha	Santosh Bus Service	A/C Seater Push Bac...	06:10	11:00	20h 00m	2.9	323	24	private	AC	2-3
16614	West Bengal	Kolkata	Digha	Snemita Paribahan (Angel)	A/C Seater (2+3)	08:30	13:00	20h 00m	3.7	370	15	private	AC	3-4
16615	West Bengal	Kolkata	Digha	Snemita Parisheba (Manika...	AC Seater (2+3)	06:35	11:20	20h 00m	3.1	352	9	private	AC	3-4
16616	West Bengal	Kolkata	Digha	Sagufta Travels (Sneha)	A/C Seater (2+3)	09:10	13:40	20h 00m	2.8	315	25	private	AC	2-3
16617	West Bengal	Kolkata	Digha	Sagufta Travels (Sagar Kan...	A/C Seater Push Bac...	23:30	04:05	20h 00m	2.9	315	9	private	AC	2-3

bus_routes 6 ×   Apply

6. HRTC(HARYANA STATE TRANSPORT CORPORATION)

URL: https://www.redbus.in/online-booking/hrtc/?utm_source=rtchometile

ID	STATE	SOURCE	DESTINATION	BUS_NAME	BUS_TYPE	DEPARTI	ARRIVIN	DURATIO	STAR.	PRIC	SEA1	BUS_CATEGO	AC_TYPE	STAR_RANGE
17333	HARYANA	Delhi	Shimla	HRTC - 69	Himsuta AC Seater V...	00:40	09:30	08h 50m	4.6	892	35	Government	AC	4 above
17334	HARYANA	Delhi	Shimla	HRTC - 6	Himsuta AC Seater V...	06:45	16:10	08h 50m	4.6	892	33	Government	AC	4 above
17335	HARYANA	Delhi	Shimla	HRTC - 592	A/C Executive (2+3)	08:05	18:10	08h 50m	3.1	602	42	Government	AC	3-4
17336	HARYANA	Delhi	Shimla	HRTC - 129	Ordinary 3+2 Non A...	08:50	18:50	08h 50m	2.4	507	37	Government	NON AC	2-3
17337	HARYANA	Delhi	Shimla	HRTC - 7	Himsuta AC Seater V...	09:25	18:30	08h 50m	4.5	892	34	Government	AC	4 above

bus_routes 6 ×

7. TNSTC(TAMILNADU STATE TRANSPORT CORPORATION)

URL: https://www.redbus.in/online-booking/tnstc

ID	STATE	SOURCE	DESTINATION	BUS_NAME	BUS_TYPE	DEPARTI	ARRIVIN	DURATIO	STAR.	PRIC	SEA1	BUS_CATEGO	AC_TYPE	STAR_RA
19489	Tamilnadu	Chennai	Tiruchirapalli	Sri Renu Travels	A/C Seater / Sleeper...	14:10	20:30	06h 20m	4.6	499	38	private	AC	4 above
19490	Tamilnadu	Chennai	Tiruchirapalli	SK Balu Bus	A/C Sleeper (2+1)	06:00	12:30	06h 20m	4.7	602	22	private	AC	4 above
19491	Tamilnadu	Chennai	Tiruchirapalli	MBT TRAVELS	A/C Sleeper (2+1)	23:35	05:35	06h 20m	4.6	650	18	private	AC	4 above
19492	Tamilnadu	Chennai	Tiruchirapalli	AdACLS Navigator	Bharat Benz A/C Sle...	20:20	03:15	06h 20m	4.6	899	24	private	AC	4 above
19493	Tamilnadu	Chennai	Tiruchirapalli	Excel Travels	A/C Sleeper (2+1)	23:00	05:30	06h 20m	4.6	550	9	private	AC	4 above

bus_routes 2 ×

8. PRTC(PUDUCHERRY STATE TRANSPORT CORPORATION)

URL: https://www.redbus.in/online-booking/puducherry-road-transport-corporation-prtc

ID	STATE	SOURCE	DESTINATION	BUS_NAME	BUS_TYPE	DEPARTI	ARRIVIN	DURATIO	STAR	PRIC	SEAT	BUS_CATEGO	AC_TYPE	STAR_RANGE
19253	Puducherry	Tindivanam	Puducherry	Vetri Travels	A/C Sleeper (2+1)	04:00	05:00	01h 30m	3.2	777	2	private	AC	3-4
19254	Puducherry	Tindivanam	Puducherry	Vetri Travels	A/C Sleeper (2+1)	04:00	05:00	01h 30m	3.2	777	4	private	AC	3-4
19255	Puducherry	Tindivanam	Puducherry	Vetri Travels	A/C Sleeper (2+1)	04:00	05:00	01h 30m	3.2	777	1	private	AC	3-4
19256	Puducherry	Tindivanam	Puducherry	Vetri Travels	A/C Sleeper (2+1)	04:55	05:55	01h 30m	3.2	777	1	private	AC	3-4
19257	Puducherry	Tindivanam	Puducherry	Sri Vaari Travels	A/C Sleeper (2+1)	05:00	05:30	01h 30m	4.1	700	18	private	AC	4 above

bus_routes 3 ×

9. PEPSU(PANJAB)

URL: https://www.redbus.in/online-booking/pepsu

ID	STATE	SOURCE	DESTINATION	BUS_NAME	BUS_TYPE	ARRIVIN	DURATIO	STAR	PRIC	SEAT	BUS_CATEGO	AC_TYPE	STAR_RANGE
22073	Punjab	Patiala	Delhi	PEPSU (Punjab) - ...	Volvo AC Seater (2+2) 04:20	09:21	05h 01m	3.6	692	39	Government	AC	3-4
22074	Punjab	Patiala	Delhi	PEPSU (Punjab) - ...	Volvo AC Seater (2+2) 05:00	10:01	05h 01m	4.1	692	39	Government	AC	4 above
22075	Punjab	Patiala	Delhi	PEPSU (Punjab) - ...	Volvo AC Seater (2+2) 06:15	11:16	05h 01m	3.9	692	39	Government	AC	3-4
22076	Punjab	Patiala	Delhi	PEPSU (Punjab) - ...	AC Seater Hvac 2+2 09:05	14:06	05h 01m	3.1	440	39	Government	AC	3-4
22077	Punjab	Patiala	Delhi	PEPSU (Punjab) - ...	Volvo AC Seater (2+2) 09:50	14:51	05h 01m	3.6	692	38	Government	AC	3-4

bus routes 4 ×

10. ASTC(ASSAM STATE TRANSPORT CORPORATION)

URL: https://www.redbus.in/online-booking/astc

ID	STATE	SOURCE	DESTINATION	BUS_NAME	BUS_TYPE	DEPARTI	ARRIVIN	DURATIO	STAR	PRIC	SEAT	BUS_CATEGO	AC_TYPE	STAR_RANGE
22916	Assam	Tezpur	Guwahati	Assam State Tran...	Bharat Benz A/C Sea...	05:00	09:00	04h 00m	3.5	270	34	Government	AC	3-4
22917	Assam	Tezpur	Guwahati	Assam State Tran...	A/C Seater (2+1)	06:15	10:30	04h 00m	2.5	298	30	Government	AC	2-3
22918	Assam	Tezpur	Guwahati	Assam State Tran...	Bharat Benz A/C Sea...	07:10	11:00	04h 00m	4.7	298	35	Government	AC	4 above
22919	Assam	Tezpur	Guwahati	Assam State Tran...	Bharat Benz A/C Sea...	09:30	14:30	04h 00m	3.9	298	35	Government	AC	3-4
22920	Assam	Tezpur	Guwahati	Assam State Tran...	Bharat Benz A/C Sea...	10:35	15:15	04h 00m	4.1	298	34	Government	AC	4 above

3. Streamlit Application:

Code:

```
4.  import streamlit as st
5.  import mysql.connector
6.  import pandas as pd
7.  def get_db_connection():
8.      db = mysql.connector.connect(
9.          host='127.0.0.1',
10.         port=3306,
11.         user='root',
12.         password='123456789',
13.         database='redbus_db'
14.     )
15.     return db
16. def fetch_distinct_values(column,state=None,source=None):
17.     db=get_db_connection()
18.     cursor=db.cursor()
19.     query=f"SELECT DISTINCT {column} FROM bus_routes"
20.     conditions=[]
21.     params=[]
22.     if state:
23.         conditions.append("state=%s")
24.         params.append(state)
25.     if source:
26.         conditions.append("state=%s")
27.         params.append(state)
28.     if conditions:
29.         query += " WHERE " + " AND ".join(conditions)
30.     cursor.execute(query,tuple(params))
```

```python
31.    values=cursor.fetchall()
32.    db.close()
33.    return[value[0] for value in values]
34.
35.def
   fetch_bus_details(SOURCE,DESTINATION,BUS_TYPE=None,STAR_RANGE=None,AC_T
   YPE=None,PRICE=None):
36.    db=get_db_connection()
37.    cursor=db.cursor()
38.    query="""
39.    SELECT
   BUS_CATEGORY,SOURCE,DESTINATION,BUS_NAME,BUS_TYPE,DEPARTING_TIME,ARRIVI
   NG_TIME,DURATION,STAR_RATING,PRICE,SEATS_AVAILABLE,AC_TYPE,STAR_RANGE
40.    FROM bus_routes
41.    WHERE SOURCE = %s AND DESTINATION = %s
42.    """
43.    params=[SOURCE,DESTINATION]
44.
45.    if STAR_RANGE:
46.        query+=" AND STAR_RANGE=%s"
47.        params.append(STAR_RANGE)
48.    if BUS_TYPE:
49.        query+=" AND BUS_TYPE=%s"
50.        params.append(BUS_TYPE)
51.    if AC_TYPE:
52.        query+=" AND AC_TYPE=%s"
53.        params.append(AC_TYPE)
54.    if PRICE:
55.        query += " AND price BETWEEN %s AND %s"
56.        params.extend(PRICE)
57.
58.    cursor.execute(query,tuple(params))
59.    bus_details=cursor.fetchall()
60.    db.close()
61.    return bus_details
62.
63.def home_page():
64.    st.title('REDBUS APP')
65.    st.subheader('Welcome to the Redbus App')
66.    st.write('please select the State to get started.....')
67.    states= fetch_distinct_values('STATE')
68.    sel_state=st.selectbox('Selected State:',[''] + states)
69.    if sel_state:
70.        st.session_state['sel_state']=sel_state
71.        st.write(f"selected State: {sel_state}")
72.        st.write("Navigate to the bus selection page to Continue.")
73.
74.#Bus Selection Page
```

```python
75.def bus_selection_page():
76.    if "visibility" not in st.session_state:
77.        st.session_state.visibility = "visible"
78.        st.session_state.disabled = False
79.    st.title('Bus Selection')
80.    st.write('Please select the source, destination and optional
   filters')
81.    col1, col2, col3 = st.columns(3)
82.
83.    with col1:
84.
85.        if 'sel_state' not in st.session_state:
86.            st.write("Please select the state from the home page")
87.            return
88.        sel_state=st.session_state['sel_state']
89.        sources = fetch_distinct_values('SOURCE',sel_state)
90.        sel_source=st.selectbox('FROM :oncoming_bus:',[''] + sources)
91.        destinations=fetch_distinct_values('DESTINATION',sel_state,sel_
   source)
92.        sel_destination=st.selectbox('TO :busstop:',[''] +
   destinations)
93.    with col2:
94.        bus_types=fetch_distinct_values('BUS_TYPE',sel_state)
95.        ac_types=fetch_distinct_values('AC_TYPE',sel_state)
96.
97.        sel_bustype=st.selectbox('BUS TYPE (Optional):bus:',[''] +
   bus_types)
98.        sel_ac_type=st.selectbox('AC_TYPE (Optional)',[''] + ac_types)
99.    with col3:
100.            star_ratings=fetch_distinct_values('STAR_RANGE',sel_state
   )
101.            sel_starrating=st.selectbox('STAR RATING
   (Optional):star:',[''] + star_ratings)
102.
103.            min_price, max_price = 0, 5000
104.            price = st.slider('Select Price Range',
   min_value=min_price, max_value=max_price, value=(min_price, max_price))
105.
106.        if sel_source and sel_destination:
107.            star_rating=sel_starrating if sel_starrating else None
108.            ac_type=sel_ac_type if sel_ac_type else None
109.            bus_type=sel_bustype if sel_bustype else None
110.
111.        bus_details=fetch_bus_details(sel_source,
   sel_destination,sel_bustype,sel_starrating,sel_ac_type)
112.            if bus_details:
113.                st.write("Bus Details:")
114.
```

```
115.              bus_details_df = pd.DataFrame(bus_details,
     columns=['Bus_Category','Source','Destination','Bus_Name','Bus_Type','D
     eparting_Time','Arriving_Time','Duration','Star_Rating','Price','Seats_
     Available','Ac_Type','Star_Range'])
116.              st.table( bus_details_df)
117.          else:
118.              st.write("No Buses found for the selection.")
119.
120.      #----------Main-------------
121.      st.sidebar.title("Navigation")
122.      menu=["Home","Bus Selection"]
123.      choice=st.sidebar.radio("Go to",menu)
124.      if choice=="Home":
125.          home_page()
126.      elif choice=="Bus Selection":
127.          if 'sel_state' in st.session_state:
128.              bus_selection_page()
129.          else:
130.              st.write("please select the state from the home page")
131.
```

Explanation of the Code

1. **Database Connection**:

   ➢ get_db_connection(): Establishes a connection to the MySQL database.

2. **Fetching Distinct Values**:

   ➢ fetch_distinct_values(column): Fetches distinct values for a specified column from the buses table.

3. **Fetching Bus Details**:

   ➢ fetch_bus_details(source, destination, star_rating=None, seats=None, ac_type=None): Fetches bus details based on the provided source, destination, and optional filters for star rating and seats.

4. **Home Page**:

   ➢ home_page(): Displays a welcome message and a dropdown for State selection. The selected state is stored in st.session_state.

5. **Bus Selection Page**:

   ➢ bus_selection_page(): Displays dropdowns for selecting source, destination, star rating (optional), Bus type, AC_type (Optional), price slider and seats (optional). Displays bus details in a table format based on the selected values.

6. **Navigation**:

   ➢ Sets up navigation between the home page and bus selection page using a sidebar menu.

7. **Importing pandas:**

   ➢ import pandas as pd: This imports the pandas library, which is used for data manipulation and analysis.
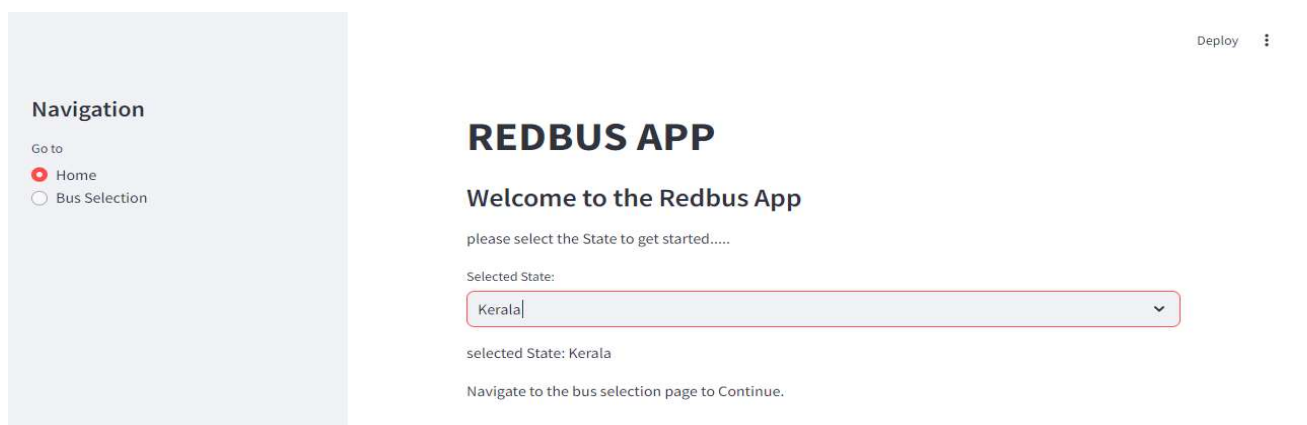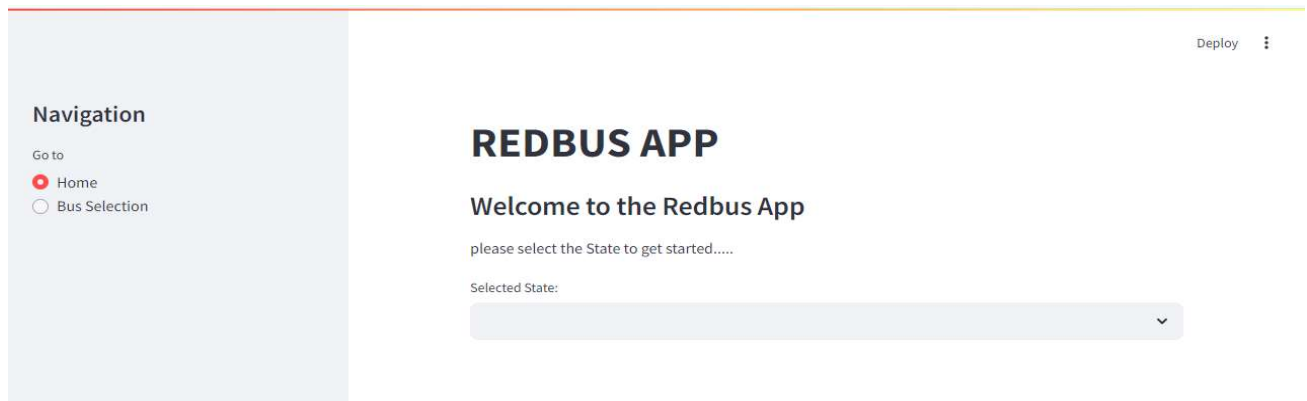
8. **Converting the Result to a DataFrame:**

   ➢ bus_details_df = pd.DataFrame(bus_details, columns=['Source', 'Destination', 'Bus Type', 'Duration', 'Star Rating', 'Departure Time', 'Arrival Time', 'Seats', 'Bus Category']): This line converts the list of tuples bus_details into a pandas DataFrame and assigns column names to it.

9. **Displaying the DataFrame:**

   ➢ st.table(bus_details_df): This displays the pandas DataFrame as a table in Streamlit. The DataFrame will include the specified column headings.

RedBus APP Screen Shot

Home Page

Bus Selection Page:

Project link: https://github.com/dineshanbu1988/Datascience_Projects.git