

Queries

-QUERY1 :- Hotels in Jaipur

```
SELECT * FROM ACCOMMODATION WHERE (CITY='Jaipur');
```

The screenshot shows the pgAdmin 4 interface. On the left, the 'Browser' pane displays a tree view of database objects, including servers, databases, schemas, and tables. The 'DS_Questions.sql*' tab in the top right contains the SQL query: 'SELECT * FROM ACCOMMODATION WHERE (CITY='Jaipur');'. The main pane shows the results of this query in a grid format. The grid has columns: accommodation_id, accommodation_name, accommodation_type, street, landmark, city, state, pincode, and ch. The data consists of three rows:

	accommodation_id	accommodation_name	accommodation_type	street	landmark	city	state	pincode	ch
1	JP100	Hotel Silver Pride	Hotel	Tonk Road	Durgapura	Jaipur	Rajasthan	302018	09
2	JP209	Hotel Arco Palace	Hotel	Station Rd	Sindhi Camp	Jaipur	Rajasthan	302006	09
3	JP132	ITC Jaipur	Hotel	Near Ghanta Ghar	Sawai Street	Jaipur	Rajasthan	302001	09

At the bottom of the interface, there are status bars showing 'Total rows: 3 of 3' and 'Query complete 00:00:00.357' on the left, and 'Ln 81, Col 1' on the right. The taskbar at the bottom of the screen shows various application icons.

-QUERY2:- Room Types of hotels in Jaipur

```
SELECT Accommodation_Name,ROOM_TYPE.* FROM (ACCOMMODATION NATURAL JOIN ROOM_TYPE)
WHERE (CITY='Jaipur');
```

The screenshot shows the pgAdmin 4 interface. On the left, the 'Browser' pane displays the database structure under 'Servers (2)'. A connection to 'WFA' is selected, showing 'Databases (1)' containing '202101208'. The 'Query' tab in the center contains the following SQL code:

```
1 SET SEARCH_PATH TO WFA;
2
3 SELECT * FROM ACCOMMODATION WHERE (CITY='Jaipur');
4
5 SELECT Accommodation_Name,ROOM_TYPE.* FROM (ACCOMMODATION NATURAL JOIN ROOM_TYPE)
6 WHERE (CITY='Jaipur');
```

The 'Data Output' tab shows the results of the query, which are:

	accommodation_name	accommodation_id	room_type_id	room_type_name	capacity	bed_type	area	extra_capacity	total_rooms
1	Hotel Silver Pride	JP100	10001	Deluxe Room	2	Double Bed	224	1	5
2	Hotel Silver Pride	JP100	10002	Super Deluxe Room	2	Double Bed	288	0	5
3	Hotel Silver Pride	JP100	10003	Superior Room	2	Double Bed	360	1	10
4	Hotel Arco Palace	JP209	20901	Standard Room	2	Double Bed	300	1	10
5	Hotel Arco Palace	JP209	20902	Business Room	2	Double Bed	300	0	5
6	Hotel Arco Palace	JP209	20903	1BHK With GYM	2	Double Bed	300	1	5
7	ITC Jaipur	JP132	13201	Premium Room	2	King Bed	120	0	10
8	ITC Jaipur	JP132	13202	Deluxue Room	2	Single Bed	150	0	15

At the bottom, it says 'Total rows: 8 of 8 Query complete 00:00:00.072 Ln 5, Col 1'.

QUERY3 :- Prices of Particular type of Rooms in Jaipur

```
SELECT Accommodation_Name,Room_Type_Name,ACCOMMODATION_FARES.* FROM (ACCOMMODATION NATURAL  
JOIN ROOM_TYPE NATURAL JOIN ACCOMMODATION_FARES)  
WHERE (CITY='Jaipur');
```

The screenshot shows the pgAdmin 4 interface with the following details:

- Servers:** pgAdmin 4, File, Object, Tools, Help.
- Browser:** Servers (2), PgSQL Server, WFA, Databases (1), 202101208, Casts, Catalogs, Event Triggers, Extensions, Foreign Data Wrappers, Languages, Publications, Schemas (5), Sales Management, company, public, wfa, xit, Subscriptions, Login/Group Roles, Tablespaces.
- Query Editor:** Title: 202101208/202101208@WFA*. Query History:
2
3 SELECT * FROM ACCOMMODATION WHERE (CITY='Jaipur');
4
5 SELECT Accommodation_Name,ROOM_TYPE.* FROM (ACCOMMODATION NATURAL JOIN ROOM_TYPE)
WHERE (CITY='Jaipur');
6
7
8
9 SELECT Accommodation_Name,Room_Type_Name,ACCOMMODATION_FARES.* FROM (ACCOMMODATION NATURAL JOIN ROOM_TYPE NATURAL JOIN ACCOMMODATION_FARES)
10 WHERE (CITY='Jaipur');
- Data Output:** Shows a table with 8 rows of room information. The columns are: name, room_type_name, accommodation_id, room_type_id, base_price, mattresses, breakfast, lunch_or_dinner, lunch_and_dinner, zero_cancellation, and taxes.
- Messages:** Total rows: 8 of 8, Query complete 00:00:00.068, Ln 8, Col 2.
- System Bar:** Windows icon, search, task view, file explorer, settings, browser, calendar, taskbar, battery, signal, ENG, date: 24-04-2023, time: 11:15.

	name (50)	room_type_name	accommodation_id character varying (50)	room_type_id integer	base_price integer	mattresses smallint	breakfast integer	lunch_or_dinner integer	lunch_and_dinner integer	zero_cancellation integer	taxes integer
1		Deluxe Room	JP100	10001	1822	150	400	1500	1000	0	379
2		Super Deluxe Room	JP100	10002	2391	150	300	1500	1000	125	507
3		Superior Room	JP100	10003	2847	150	250	1500	1000	250	592
4		Standard Room	JP209	20901	1226	250	100	250	250	100	277
5		Business Room	JP209	20902	1549	250	200	500	500	200	350
6		1BHK With GYM	JP209	20903	2259	250	200	750	750	250	625
7		Premium Room	JP132	13201	500	100	300	500	700	150	120
8		Delux Room	JP132	13202	600	150	0	500	500	250	180

Query 4 :- Hotels of Shimla with starting price less than 1000

```
SELECT * FROM ACCOMMODATION  
WHERE CITY='Shimla' AND STARTING_PRICE<1000;
```

The screenshot shows the pgAdmin 4 interface with the following details:

- Toolbar:** Includes File, Object, Tools, Help, and various icons for browser, dashboard, properties, statistics, dependencies, dependents, processes, and queries.
- Header:** Shows "pgAdmin 4" and the current session "202101208/202101208@WFA".
- Left Sidebar (Browser):** Lists database objects under "Tables (21)":
 - 1.3 Sequences
 - accommmodation
 - accommmodation_fare
 - accommmodation_refu
 - available_rooms
 - available_seats
 - bank_details
 - classes
 - consumer
 - consumer_profile
 - offer_cities
 - offer_dates
 - offers
 - package
 - room_type
 - route
 - Columns (8)
 - Constraints
 - Indexes
 - RLS Policies
 - Rules
 - Triggers
 - station
 - transportation
 - transportation_class
 - transportation_fares
 - transportation_owner
 - transportation_refunc
 - Trigger Functions
- Central Area:** Shows the SQL query and its execution results.

```
8 SELECT ACCOMMODATION_NAME, ROOM_TYPE_NAME, ACCOMMODATION_FARES.*  
9     FROM (ACCOMMODATION NATURAL JOIN ROOM_TYPE NATURAL JOIN ACCOMMODATION_FARES)  
10    WHERE (CITY='Jaipur');  
11  
12   SELECT COUNT(Accommodation_ID), CITY FROM ACCOMMODATION GROUP BY CITY;  
13  
14 --4  
15   SELECT * FROM ACCOMMODATION  
16  WHERE CITY='Shimla' AND STARTING_PRICE<1000;  
17  
18 --5  
19  
20  SELECT T.TRANSPORTATION_NAME, T.USER_RATING, T.Transportation_Type, R.* FROM TRANSPORTATION AS T  
21  NATURAL JOIN  
22    (SELECT S.TRANSPORTATION_ID, S.STATION_CODE, S.ARRIVAL AS S_ARRIVAL, S.DEPARTURE AS S_DEPARTURE, S.HALT AS S_HALT, S.DAY_NO AS S_DAY_NO,  
23      D.STATION_CODE, D.ARRIVAL AS D_ARRIVAL, D.DEPARTURE AS D_DEPARTURE, D.HALT AS D_HALT, D.DAY_NO AS D_DAY_NO FROM  
24      (SELECT ROUTE.* FROM ROUTE NATURAL JOIN (SELECT * FROM STATION WHERE CITY='Navsari') AS N) AS S  
25    JOIN  
26      (SELECT ROUTE.* FROM ROUTE NATURAL JOIN (SELECT * FROM STATION WHERE CITY='Ahmedabad') AS A) AS D ON S.TRANSPORTATION_ID=D.TRANSP  
27  WHERE S.STOP_NUMBER < D.STOP_NUMBER) AS R;
```
- Data Output:** Displays the results of the query:

accommodation_id	accommodation_name	accommodation_type	street	landmark	city	state	pincode	che
THC123	The Oberoi Cecil	Hotel	Chaura Maidan Rd	Nabha	Shimla	Himachal Pradesh	171004	09
TTM399	Treebo Trend Maharaja	Hotel	Circular Rd	Victory Tunnel	Shimla	Himachal Pradesh	171003	09
- Bottom Status Bar:** Shows "Total rows: 2 of 2" and "Query complete 00:00:00.146".
- Taskbar:** Shows various system icons and the system tray with the message "40°C Sunny" and the date "23-04-2023".

Query 5 :- Transportation Facilities available between Ahmedabad and Navsari

```
SELECT T.TRANSPORTATION_NAME, T.USER_RATING, T.Transportation_Type, R.* FROM TRANSPORTATION AS T
NATURAL JOIN
  (SELECT S.TRANSPORTATION_ID, S.STATION_CODE, S.ARRIVAL AS S_ARRIVAL, S.DEPARTURE AS S_DEPARTURE, S.HALT AS S_HALT, S.DAY_NO AS S_DAY_NO,
    D.STATION_CODE, D.ARRIVAL AS D_ARRIVAL, D.DEPARTURE AS D_DEPARTURE, D.HALT AS D_HALT, D.DAY_NO AS D_DAY_NO FROM
    (SELECT ROUTE.* FROM ROUTE NATURAL JOIN (SELECT * FROM STATION WHERE CITY='Navsari') AS N) AS S
  JOIN
    (SELECT ROUTE.* FROM ROUTE NATURAL JOIN (SELECT * FROM STATION WHERE CITY='Ahmedabad ') AS A) AS D ON S.TRANSPORTATION_ID=D.TRANSPORTATION_ID
  WHERE S.STOP_NUMBER < D.STOP_NUMBER) AS R;
```

The screenshot shows the pgAdmin 4 interface with the following details:

- Toolbar:** Includes File, Object, Tools, Help, Dashboard, Properties, SQL, Statistics, Dependencies, Dependents, Processes, and a Queries tab.
- Browser:** Shows a tree view of database objects:
 - 1.3 Sequences
 - Tables (21)
 - accommodation
 - accommodation_fare
 - accommodation_refu
 - available_rooms
 - available_seats
 - bank_details
 - classes
 - consumer
 - consumer_profile
 - offer_cities
 - offer_dates
 - offers
 - package
 - room_type
 - route
 - Columns (8)
 - station
 - transportation
 - transportation_class
 - transportation_fares
 - transportation_owner
 - transportation_refunc
 - Trigger Functions
- Query Editor:** Displays the SQL query from the code block above. The results of the last executed part (lines 20-30) are shown below.
- Data Output:** A table showing transportation facilities between Navsari and Ahmedabad.

transportation_name	user_rating	transportation_type	transportation_id	station_code	s_arrival	s_departure	s_halt	s_sr
ARAVALI EXP	10	RAIL	14702	NVS	00:11:00	00:13:00	00:02:00	
Karnavati Express	8	RAIL	23104	NVS	16:50:00	16:52:00	00:02:00	
- Status Bar:** Shows "Total rows: 2 of 2" and "Query complete 00:00:00.088".
- System Tray:** Shows system icons including battery level, temperature (40°C), and date/time (23-04-2023).

Query 6 :- Classes available in particular type of transportation between Ahmedabad and Navsari

```

SELECT P.* ,TC.CLASS_NAME FROM
(SELECT T.TRANSPORTATION_NAME ,T.USER_RATING ,R.* FROM TRANSPORTATION AS T
NATURAL JOIN
    (SELECT S.TRANSPORTATION_ID ,S.STATION_CODE ,S.ARRIVAL AS S_ARRIVAL ,S.DEPARTURE AS
S_DEPARTURE ,S.HALT AS S_HALT ,S.DAY_NO AS S_DAY_NO ,S.KM_FROM_ORIGIN AS S_KM_FROM_ORIGIN ,
D.STATION_CODE ,D.ARRIVAL AS D_ARRIVAL ,D.DEPARTURE AS D_DEPARTURE ,D.HALT AS
D_HALT ,D.DAY_NO AS D_DAY_NO ,D.KM_FROM_ORIGIN AS D_KM_FROM_ORIGIN FROM
    (SELECT * FROM ROUTE NATURAL JOIN (SELECT * FROM STATION WHERE CITY='Navsari') AS N) AS S
JOIN
    (SELECT * FROM ROUTE NATURAL JOIN (SELECT * FROM STATION WHERE CITY='Ahmedabad ') AS A) AS D ON S.TRANSPORTATION_ID=D.TRANSPORTATION_ID
WHERE S.STOP_NUMBER < D.STOP_NUMBER) AS R ) AS P
NATURAL JOIN
(TRANSPORTATION_CLASS NATURAL JOIN CLASSES) AS TC;

```

The screenshot shows the pgAdmin 4 interface with the following details:

- Toolbar:** Includes File, Object, Tools, Help, and a search bar.
- Left Panel (Browser):** Shows the database schema with 1.3 Sequences and 21 Tables. Tables include accommodation, accommodation_fare, accommodation_refu, available_rooms, available_seats, bank_details, classes, consumer, consumer_profile, offer_cities, offer_dates, offers, package, room_type, and route.
- Central Panel (Query Editor):** Displays the SQL query from the code block above. The query retrieves transportation details, routes, and classes for routes between Navsari and Ahmedabad.
- Bottom Panel (Data Output):** Shows the results of the query as a table. The columns are:

	transportation_name	user_rating	transportation_id	station_code	s_arrival	s_departure	s_halt	s_day_no	s_km_from_origin
1	ARAVALI EXP	10	14702	NVS	00:11:00	00:13:00	00:02:00	2	
2	ARAVALI EXP	10	14702	NVS	00:11:00	00:13:00	00:02:00	2	
3	Karnavati Express	8	23104	NVS	16:50:00	16:52:00	00:02:00	1	
4	Karnavati Express	8	23104	NVS	16:50:00	16:52:00	00:02:00	1	
5	Karnavati Express	8	23104	NVS	16:50:00	16:52:00	00:02:00	1	
- Bottom Status Bar:** Shows system icons, a battery level of 40%, and the date/time (23-04-2023 22:08).

Query 7 :- Fares of Particular class in transportation facilities between Ahmedabad and Navsari

```

SELECT PP.* , (D.KM_FROM_ORIGIN-S.KM_FROM_ORIGIN)*BASE_PRICE AS BASE_PRICE, (D.KM_FROM_ORIGIN-S.KM_FROM_ORIGIN)*TAXES AS TAXES,
TF.BREAKFAST, TF.LUNCH, TF.DINNER, TF.ZERO_CANCELLATION FROM
(SELECT P.* , TC.CLASS_NAME FROM
(SELECT T.TRANSPORTATION_NAME, T.USER_RATING, R.* FROM TRANSPORTATION AS T
NATURAL JOIN
(SELECT S.TRANSPORTATION_ID, S.STATION_CODE, S.ARRIVAL AS S_ARRIVAL, S.DEPARTURE AS S_DEPARTURE, S.HALT AS S_HALT, S.DAY_NO AS S_DAY_NO, S.KM_FROM_ORIGIN AS S_KM_FROM_ORIGIN,
D.STATION_CODE, D.ARRIVAL AS D_ARRIVAL, D.DEPARTURE AS D_DEPARTURE, D.HALT AS D_HALT, D.DAY_NO AS D_DAY_NO, D.KM_FROM_ORIGIN AS D_KM_FROM_ORIGIN FROM
(SELECT * FROM ROUTE NATURAL JOIN (SELECT * FROM STATION WHERE CITY='Navsari') AS A) AS S
JOIN
(SELECT * FROM ROUTE NATURAL JOIN (SELECT * FROM STATION WHERE CITY='Ahmedabad ') AS A) AS D ON S.TRANSPORTATION_ID=D.TRANSPORTATION_ID
WHERE S.STOP_NUMBER < D.STOP_NUMBER) AS R ) AS P
NATURAL JOIN
(TRANSPORTATION_CLASS NATURAL JOIN CLASSES) AS TC) AS PP
NATURAL JOIN
TRANSPORTATION_FARES AS TF ;

```

The screenshot shows the pgAdmin 4 interface with the following details:

- Toolbar:** Includes File, Object, Tools, Help, and a search bar.
- Left Panel (Browser):** Shows the database schema with tables like sequences, tables (21), accommodation, accommodation_fare, etc., and columns (8).
- Central Panel (Query Editor):** Displays the SQL query from the previous code block. The results of the query are shown below the code.
- Result Table:** A data grid showing the results of the query. The columns are:

	transportation_name	user_rating	transportation_id	station_code	s_arrival	s_departure	s_halt	s_day_no	s_km_from_
1	ARAVALI EXP	10	14702	NVS	00:11:00	00:13:00	00:02:00	2	
2	ARAVALI EXP	10	14702	NVS	00:11:00	00:13:00	00:02:00	2	
3	ARAVALI EXP	10	14702	NVS	00:11:00	00:13:00	00:02:00	2	
4	ARAVALI EXP	10	14702	NVS	00:11:00	00:13:00	00:02:00	2	
5	Karnavati Express	8	23104	NVS	16:50:00	16:52:00	00:02:00	1	
6	Karnavati Express	8	23104	NVS	16:50:00	16:52:00	00:02:00	1	
7	Karnavati Express	8	23104	NVS	16:50:00	16:52:00	00:02:00	1	
- Bottom Status Bar:** Shows "Total rows: 13 of 13" and "Query complete 00:00:00.239".
- System Icons:** Standard Windows taskbar icons for search, file, and system status.

Query 8 :-

Accommodation Prices of All Offers

```
CREATE VIEW PAP AS
SELECT OFFER_ID, PACKAGE_ID, SUM(BASE_PRICES) AS BASE_PRICES, SUM(TAX_AMT) AS TAX_AMT FROM
(SELECT OFFER_ID, PACKAGE_ID, ACCOMMODATION_ID, SUM(BASE_PRICE*DAYS)/2 AS BASE_PRICES,
SUM(TAXES*DAYS)/2 AS TAX_AMT
FROM OFFER_ACcomMODATION
NATURAL JOIN ACCOMMODATION_FARES
GROUP BY OFFER_ID, PACKAGE_ID, ACCOMMODATION_ID) AS AF GROUP BY OFFER_ID, PACKAGE_ID;

SELECT * FROM PAP;
```

The screenshot shows the pgAdmin 4 interface. The left sidebar displays the database browser with the 'wfa' schema selected, showing various objects like tables, functions, and materialized views. The main area shows a query editor with the following content:

```
202101208/202101208@wfa*
1 SET SEARCH_PATH TO WFA;
2
```

The results pane displays a table with the following data:

	offer_id	package_id	base_prices	tax_amt
1	30101	30100	8846	1936
2	30102	30100	19485	3909
3	30103	30100	26881	6087
4	30104	30100	6196	1400
5	30105	30100	6580	1449
6	30201	30200	38612	9066
7	30202	30200	3800	1224
8	30401	30400	8441	2702
9	30502	30500	2800	674

Total rows: 9 of 9 Query complete 00:00:01.180 Ln 10, Col 1

Query 9 :-

Source Travel Information of All Offers

```
CREATE VIEW RT AS
SELECT OFFER_ID, PACKAGE_ID, R.TRANSPORTATION_ID, CLASS_ID, FROM_STATION, stop_number AS
FROM_stop_no, km_from_origin AS FROM_km
FROM (SELECT OFFER_ID, PACKAGE_ID, TRANSPORTATION_ID, CLASS_ID, FROM_STATION
FROM OFFER_TRANSPORTATION) AS OT JOIN
(SELECT * FROM ROUTE ) AS R
ON (OT.TRANSPORTATION_ID = R.TRANSPORTATION_ID
AND OT.FROM_STATION = R.STATION_CODE);

SELECT * FROM RT;
```

The screenshot shows the pgAdmin 4 interface with the following details:

- File Object Tools Help**: The top menu bar.
- Browser**: A tree view of database objects including Catalogs, Event Triggers, Extensions, Foreign Data Wrappers, Languages, Publications, Schemas (5), Sales Management, company, public, and wfa.
- Dashboard Properties SQL Statistics Dependencies Dependents Processes**: The tabs for the current session.
- QUE.sql***: The active query editor tab.
- Query History**: A list of previous queries.
- Query**: The SQL code for creating the view RT.
- Data Output**: The results of the query, displayed as a table.
- Messages Notifications**: Status and notification panels.
- Table Results**: A detailed view of the query results.
- Total rows: 72 of 72 Query complete 00:00:00.124**: The status message at the bottom.
- Toolbar**: Standard Windows-style toolbar icons.
- System Tray**: Icons for network, battery, volume, and date/time (24-04-2023, 02:18).

offer_id	package_id	transportation_id	class_id	from_station	from_stop_no	from_km
1	30101	30100	22977	JP	1	0
2	30101	30100	1478	JPR	1	0
3	30101	30100	12467	BKN	2	319
4	30101	30100	22977	JP	1	0
5	30101	30100	1478	JPR	1	0
6	30101	30100	12467	BKN	2	319
7	30102	30100	164403	GTM	1	0
8	30102	30100	UK208	UDZ	1	0
9	30102	30100	KC568	KMBR	1	0
10	30102	30100	176454	CHTR	1	0
11	30102	30100	5699	ARD	1	0
12	30102	30100	164403	GTM	1	0
13	30102	30100	UK208	UDZ	1	0
14	30102	30100	KC568	KMBR	1	0
15	30102	30100	176454	CHTR	1	0
16	30102	30100	5699	ARD	1	0

Query 10 :-

Destination Travel Information of All Offers

```
CREATE VIEW RF AS
SELECT OFFER_ID, PACKAGE_ID, R.TRANSPORTATION_ID, CLASS_ID, TO_STATION, stop_number AS
TO_stop_no, km_from_origin AS TO_km FROM
(SELECT OFFER_ID, PACKAGE_ID, TRANSPORTATION_ID, CLASS_ID, TO_STATION
FROM OFFER_TRANSPORTATION) AS OTT JOIN
(SELECT * FROM ROUTE ) AS R
ON OTT.TRANSPORTATION_ID = R.TRANSPORTATION_ID
AND OTT.TO_STATION = R.STATION_CODE;

SELECT * FROM RF;
```

pgAdmin 4

File Object Tools Help

Browser

- Catalogs
- Event Triggers
- Extensions
- Foreign Data Wrappers
- Languages
- Publications
- Schemas (5)
 - Sales Management
 - company
 - public
 - wfa
 - Aggregates
 - Collations
 - Domains
 - FTS Configurations
 - FTS Dictionaries
 - FTS Parsers
 - FTS Templates
 - Foreign Tables
 - Functions
 - Materialized Views
 - Operators
 - Sequences
 - Tables
 - Trigger Functions
 - Types
 - Views (4)
 - o_accm_b
 - o_trans_b
 - rf
 - rt

Dashboard Properties SQL Statistics Dependencies Dependents Processes QUE.sql*

202101208/202101208@WFA

No limit

Query History

```
32 ON (OT.TRANSPORTATION_ID = R.TRANSPORTATION_ID
33 AND OT.FROM_STATION = R.STATION_CODE);
34
35 SELECT * FROM RT;
36
37 CREATE VIEW RF AS
```

Data Output Messages Notifications

	offer_id	package_id	transportation_id	class_id	to_station	to_stop_no	to_km
1	30101	30100	22977	10	JU	3	313
2	30101	30100	1478	31	BKR	2	334
3	30101	30100	12467	13	JP	4	701
4	30101	30100	22977	10	JU	3	313
5	30101	30100	1478	31	BKR	2	334
6	30101	30100	12467	13	JP	4	701
7	30102	30100	164403	32	UDP	2	260
8	30102	30100	UK208	41	KMBR	2	85
9	30102	30100	KC568	42	CHTR	2	152
10	30102	30100	176454	32	ABR	2	255
11	30102	30100	5699	32	GTM	2	230
12	30102	30100	164403	32	UDP	2	260
13	30102	30100	UK208	41	KMBR	2	85
14	30102	30100	KC568	42	CHTR	2	152
15	30102	30100	5699	32	GTM	2	230
16	30102	30100	14780	12	AH	10	672

Total rows: 71 of 71 Query complete 00:00:00.088

Ln 37, Col 1

02:19 24-04-2023

Query 11 :-

Transportation Prices of All Offers

```
CREATE VIEW PTP AS
(SELECT OFFER_ID, PACKAGE_ID, SUM(BASE_PRICES) AS TRAN_BASE_PRICE, SUM(TAXES_AMT) AS
TRAN_TAXES FROM
(SELECT OFFER_ID, PACKAGE_ID, TRANSPORTATION_ID, CLASS_ID, BASE_PRICE*(RTF.TO_km -
RTF.FROM_km) AS BASE_PRICES,
(TAXES)*(RTF.TO_km - RTF.FROM_km) AS TAXES_AMT FROM
((SELECT * FROM RT
NATURAL JOIN RF) AS RTF
NATURAL JOIN TRANSPORTATION_FARES) AS RTF WHERE TO_STOP_NO>FROM_STOP_NO) AS OTF
GROUP BY OFFER_ID, PACKAGE_ID);

SELECT * FROM PTP;
```

The screenshot shows the pgAdmin 4 interface with the following details:

- File Bar:** File, Object, Tools, Help.
- Toolbar:** Includes icons for Catalogs, Event Triggers, Extensions, Foreign Data Wrappers, Languages, Publications, Schemas (5), Sales Management, company, public, wfa, Aggregates, Collations, Domains, FTS Configurations, FTS Dictionaries, FTS Parsers, FTS Templates, Foreign Tables, Functions, Materialized Views, Operators, Sequences, and Tables (36).
- Left Sidebar (Browser):** Shows the schema tree under the 'wfa' schema, including tables like accommodation, accommodation_add_on, accommodation_booking, accommodation_fares, accommodation_refund_rules, available_rooms, and available_seats.
- Central Area:**
 - Query Editor:** Displays the SQL code for creating the PTP view.
 - Data Output:** Shows the results of the query, which consists of 9 rows of data.
 - Messages:** Shows the message "Query complete 00:00:00.920".
 - Notifications:** Shows the message "Ln 3, Col 1".
- Bottom Status Bar:** Shows system status including temperature (33°C), battery level (Smoke), and system time (15:26 24-04-2023).

Query 12 :-

Total Prices of Each Offer

```
SELECT PAP.OFFER_ID, PAP.PACKAGE_ID, PAP.BASE_PRICES AS TOTAL_ACCM, PTP.TRAN_BASE_PRICE AS  
TOTAL_TRAN,  
(PAP.TAX_AMT + PTP.TRAN_TAXES) AS TOTAL_TAXES,  
PAP.BASE_PRICES + PTP.TRAN_BASE_PRICE + (PAP.TAX_AMT + PTP.TRAN_TAXES) AS TOTAL_PRICE FROM  
PAP JOIN PTP ON PAP.OFFER_ID = PTP.OFFER_ID;
```

The screenshot shows the pgAdmin 4 interface. The left sidebar is the 'Browser' pane, displaying the database schema structure. The main area is the 'Query' pane where the SQL query has been entered and executed. The results are displayed in a table.

Query:

```
1 SET SEARCH_PATH TO WFA;  
2  
3 SELECT PAP.OFFER_ID, PAP.PACKAGE_ID, PAP.BASE_PRICES AS TOTAL_ACCM, PTP.TRAN_BASE_PRICE AS TOTAL_TRAN,  
4 (PAP.TAX_AMT + PTP.TRAN_TAXES) AS TOTAL_TAXES,  
5 PAP.BASE_PRICES + PTP.TRAN_BASE_PRICE + (PAP.TAX_AMT + PTP.TRAN_TAXES) AS TOTAL_PRICE FROM  
6 PAP JOIN PTP ON PAP.OFFER_ID = PTP.OFFER_ID;
```

Data Output:

	offer_id	package_id	total_accm	total_tran	total_taxes	total_price
1	30101	30100	8846	5686	2959.48004066944	17491.4800406694
2	30102	30100	19485	13239.2000203133	6292.74001824856	39016.9400385618
3	30103	30100	26881	36896.7992296219	12725.3201626539	76503.1193922758
4	30104	30100	6196	21208	5217.44003677368	32621.4400367737
5	30105	30100	6580	35789.600025177	7893.23987412453	50262.8398993015
6	30201	30200	38612	56438.1014728546	19232.5800408125	114282.681513667
7	30202	30200	3800	40691.6001281738	8552.64004230499	53044.2401704788
8	30401	30400	8441	57729.1600246429	13091.4398880005	79261.5999126434
9	30502	30500	2800	55444	10653.9198534489	68897.9198534489

Total rows: 9 of 9 Query complete 00:00:00.423 Ln 3, Col 1

Query 13 :- Details of Sole Accommodation Booking

```
SELECT CD.* , INVOICE_ID FROM
(SELECT * FROM
(SELECT BOOKING_ID FROM BOOKING
EXCEPT
SELECT BACKEND_BOOKING_ID FROM PACKAGE_BOOKING_DETAILS) AS AB
NATURAL JOIN ACCOMMODATION_BOOKING) AS CD
NATURAL JOIN BOOKING
;
```

The screenshot shows the pgAdmin 4 interface with the following details:

- File Bar:** File, Object, Tools, Help.
- Toolbar:** Includes icons for FTS Configurations, FTS Dictionaries, FTS Parsers, FTS Templates, Foreign Tables, Functions, Materialized Views, Operators, Sequences, and Tables (33).
- Query Editor:** Shows the SQL query from the code block above. The results of the query are displayed below the editor.
- Result Grid:** Displays the data returned by the query. The columns are:

booking_id	room_type_id	accommodation_id	check_in_date	check_out_date	no_of_rooms	no_of_nights	booking_status	invoice_id	
1	9250143	46708	MAL467	2023-05-15	2023-05-17	1	2	Booked	8G2B7C1F6E
2	1523847	12301	JSM123	2023-05-15	2023-05-16	1	1	Booked	1B7E3F0G9D
3	7130825	39902	TTM399	2023-05-19	2023-05-20	1	1	Booked	9J0I5D4H3A
4	4065129	38601	HMT386	2023-05-15	2023-05-16	1	1	Booked	2H5C4A8J6I
5	5398201	82302	MBR823	2023-05-18	2023-05-20	1	2	Booked	3F6D1G9B0J
6	6710235	12301	THC123	2023-05-29	2023-05-30	1	1	Booked	0E4I7A8H3D
- Status Bar:** Total rows: 6 of 6 | Query complete 00:00:00.075 | Ln 76, Col 1.
- System Icons:** Standard Windows taskbar icons for search, file, and system status.

Query 14 :- INVOICE DETAILS OF ONLY ACCOMMODATION BOOKING

```
CREATE VIEW O_ACCM_B AS
SELECT CD.*, INVOICE_ID FROM
(SELECT * FROM
(SELECT BOOKING_ID FROM BOOKING
EXCEPT
SELECT BACKEND_BOOKING_ID FROM PACKAGE_BOOKING_DETAILS) AS AB
NATURAL JOIN ACCOMMODATION_BOOKING) AS CD
NATURAL JOIN BOOKING
;
-- 
SELECT * FROM O_ACCM_B ;
```

The screenshot shows the pgAdmin 4 interface with the following details:

- Toolbar:** File, Object, Tools, Help.
- Left Panel (Browser):** Shows various database objects: FTS Configurations, FTS Dictionaries, FTS Parsers, FTS Templates, Foreign Tables, Functions, Materialized Views, Operators, Sequences, and Tables (33). The 'Tables' node is expanded, showing accommodation, accommodation_add, accommodation_bool, accommodation_fare, accommodation_refu, available_rooms, available_seats, bank_details, booking, classes, consumer, consumer_profile, invoice_details, offer_accommodation, offer_cities, offer_dates, offer_transportation, offers, package, and package_booking.
- Central Panel (Query Editor):** Displays the SQL code for creating the view O_ACCM_B. The code includes joins between BOOKING, PACKAGE_BOOKING_DETAILS, and ACCOMMODATION_BOOKING tables.
- Results Panel:** Shows a table with the following data:

	booking_id	room_type_id	accommodation_id	check_in_date	check_out_date	no_of_rooms	no_of_nights	booking_status	invoice_id
1	9250143	46708	MAL467	2023-05-15	2023-05-17	1	2	Booked	8G2B7C1F6E
2	1523847	12301	JSM123	2023-05-15	2023-05-16	1	1	Booked	1B7E3FG9D
3	7130825	39902	TTM399	2023-05-19	2023-05-20	1	1	Booked	9J015D4H3A
4	4065129	38601	HMT386	2023-05-15	2023-05-16	1	1	Booked	2H5C4A8J6I
5	5398201	82302	MBR823	2023-05-18	2023-05-20	1	2	Booked	3F6D1G9B0J
6	6710235	12301	THC123	2023-05-29	2023-05-30	1	1	Booked	0E417A8H3D

Total rows: 6 of 6 Query complete 00:00:00.116 Ln 95, Col 1

Query 15:- ONLY ACCOMMODATION BOOKING PRICE DETAILS

```
SELECT * FROM
(SELECT * FROM O_ACCM_B
NATURAL JOIN ACCOMMODATION_FARES) AS AF
JOIN ACCOMMODATION_ADD_ON AS AD ON (AF.BOOKING_ID=AD.BOOKING_ID)
;
```

The screenshot shows the pgAdmin 4 interface with the following details:

- File Object Tools Help** menu bar.
- Browser** pane on the left listing database objects like Catalogs, Event Triggers, Extensions, Foreign Data Wrappers, Languages, Publications, Schemas (5), Sales Management, company, public, and wfa.
- Query History** pane showing the executed query.
- Query Editor** pane containing the SQL code for Query 15.
- Data Output** pane displaying the results of the query, which are as follows:

room_type_id	accommodation_id	booking_id	check_in_date	check_out_date	no_of_rooms	no_of_nights	booking_status	invoice_id	base_price	mattresses	
1	46708	MAL467	9250143	2023-05-15	2023-05-17	1	2	Booked	8G2B7C1F6E	4255	25
2	12301	JSM123	1523847	2023-05-15	2023-05-16	1	1	Booked	1B7E3FOG9D	1200	15
3	39902	TTM399	7130825	2023-05-19	2023-05-20	1	1	Booked	9J015D4H3A	300	15
4	38601	HMT386	4065129	2023-05-15	2023-05-16	1	1	Booked	2H5C4A8J6I	400	15
5	82302	MBR823	5398201	2023-05-18	2023-05-20	1	2	Booked	3F6D1G9B0J	600	10
6	12301	THC123	6710235	2023-05-29	2023-05-30	1	1	Booked	0E4I7A8H3D	500	10

Total rows: 6 of 6 | Query complete 00:00:00.094 | Ln 3, Col 1

Query 16:- Price distribution

```
SELECT *, (FF.TOTAL_AMT+ FF.TAXES) AS GRAND_TOTAL FROM
(SELECT F.* , (F.Mattresses+F.BASE_PRICE+F.BREAKFAST + F.LUNCH_OR_DINNER + F.DINNER +
F.ZERO_CANCELLATION) AS TOTAL_AMT FROM
(SELECT INVOICE_ID,AF.BASE_PRICE,
(AF.BREAKFAST*AD.BREAKFAST) AS BREAKFAST,
(AF.LUNCH_OR_DINNER*AD.LUNCH_OR_DINNER) AS LUNCH_OR_DINNER,
(AF.LUNCH_AND_DINNER*AD.DINNER) AS DINNER,
(AF.ZERO_CANCELLATION*AD.ZERO_CANCELLATION) AS ZERO_CANCELLATION,
(AF.Mattresses*AD.No_Of_Mattresses) AS Mattresses,
AF.TAXES FROM
(SELECT * FROM O_ACCM_B
NATURAL JOIN ACCOMMODATION_FARES) AS AF
JOIN ACCOMMODATION_ADD_ON AS AD ON (AF.BOOKING_ID=AD.BOOKING_ID) ) AS F) AS FF
;
```

The screenshot shows the pgAdmin 4 interface with the following details:

- Toolbar:** Includes File, Object, Tools, Help, and various icons for browser, dashboard, properties, statistics, dependencies, dependents, processes, and queries.
- Left Sidebar (Browser):** Shows a tree view of database objects including Collations, Domains, FTS Configuration, FTS Dictionaries, FTS Parsers, FTS Templates, Foreign Tables, Functions, Materialized Views, Operators, Sequences, and Tables (36).
- Central Area:**
 - Query Editor:** Displays the SQL query (labeled 143--157) and its execution results.
 - Data Output:** A table showing the results of the query execution.
- Bottom Status Bar:** Shows "Total rows: 6 of 6" and "Query complete 00:00:00.082".
- System Tray:** Shows system information including temperature (40°C), battery level (Sunny), and date/time (24-04-2023 16:13).

Table Data:

	invoice_id	base_price	breakfast	lunch_or_dinner	dinner	zero_cancellation	mattresses	taxes	total_amt	grand_total
1	8G2B7C1F6E	4255	0	0	0	250	0	1107	4505	5612
2	1B7E3F0G9D	1200	0	0	500	0	0	676	1700	2376
3	9J0I5D4H3A	300	0	500	0	150	150	100	1100	1200
4	2H5C4A8J6I	400	0	300	0	0	0	120	700	820
5	3F6D1G9B0J	600	0	350	400	0	0	180	1350	1530
6	0E4I7A8H3D	500	300	0	700	150	0	120	1650	1770

Query 17:- ONLY TRANSPORTATION BOOKING PRICE DETAILS

```
SELECT * FROM
(SELECT BOOKING_ID FROM BOOKING
EXCEPT
SELECT BACKEND_BOOKING_ID FROM PACKAGE_BOOKING_DETAILS) AS AB
NATURAL JOIN TRANSPORTATION_BOOKING;
```

The screenshot shows the pgAdmin 4 interface with the following details:

- Toolbar:** Includes File, Object, Tools, Help, and various icons for connection management.
- Left Panel (Browser):** Shows a tree view of database objects under "Tables (36)". The node for "available_rooms" is currently selected.
- Central Panel (Query Editor):** Displays the SQL query for "ONLY TRANSPORTATION BOOKING PRICE DETAILS".

```
154 NATURAL JOIN ACCOMMODATION_FARES) AS AF
155 JOIN ACCOMMODATION_ADD_ON AS AD ON (AF.BOOKING_ID=AD.BOOKING_ID) ) AS F) AS FF
156 ;
157 --11
158 SELECT * FROM
159 (SELECT BOOKING_ID FROM BOOKING
160 EXCEPT
161 SELECT BACKEND_BOOKING_ID FROM PACKAGE_BOOKING_DETAILS) AS AB
162 NATURAL JOIN TRANSPORTATION_BOOKING*
```
- Bottom Panel (Results):** Shows the results of the query as a table with 6 rows. The columns are: booking_id, transportation_id, class_id, from_station, to_station, departure_timestamp, arrival_timestamp, pnr, and booking_status.

	booking_id	transportation_id	class_id	from_station	to_station	departure_timestamp	arrival_timestamp	pnr	booking_status
1	7038195	6E2518	21	BOM	DEL	2023-05-21 20:10:00	2023-05-21 22:25:00	8657139425	BOOKED
2	6519238	22185	11	ANND	NVS	2023-05-27 21:15:00	2023-05-28 00:21:00	9138652475	WAITING
3	4876392	23104	11	CSMT	ADI	2023-05-17 14:05:00	2023-05-17 21:05:00	3724968513	WAITING
4	2986415	14702	13	ST	JP	2023-05-15 00:55:00	2023-05-15 17:50:00	2475913865	BOOKED
5	3601975	6E2016	23	DEL	BOM	2023-05-20 15:45:00	2023-05-20 18:00:00	4851693724	WAITING
6	8502167	1479	31	JPR	BKR	2023-05-31 13:30:00	2023-05-31 18:45:00	9425368715	BOOKED

Total rows: 6 of 6 Query complete 00:00:00.325 Ln 159, Col 2

The system tray at the bottom right shows the date as 24-04-2023, time as 15:57, and weather as 40°C Sunny.

Query 18:- Transportation Details with Amenities

```
CREATE VIEW O_TRANS_B AS
SELECT CD.* , INVOICE_ID FROM
(SELECT * FROM
(SELECT BOOKING_ID FROM BOOKING
EXCEPT
SELECT BACKEND_BOOKING_ID FROM PACKAGE_BOOKING_DETAILS) AS AB
NATURAL JOIN TRANSPORTATION_BOOKING) AS CD
NATURAL JOIN BOOKING
;

SELECT * FROM O_TRANS_B ;
```

The screenshot shows the pgAdmin 4 interface with the following details:

- Toolbar:** File, Object, Tools, Help.
- Connections:** A connection named "202101208/202101208@WFA" is selected.
- Queries Tab:** The "Queries" tab is active, showing the executed SQL code for creating the view O_TRANS_B.
- Data Output:** The results of the query are displayed in a table format.
- Table Headers:** booking_id, transportation_id, class_id, from_station, to_station, departure_timestamp, arrival_timestamp, pnr, booking_status, invoice_id.
- Table Data:** Six rows of data are shown, corresponding to the six entries in the screenshot.
- Bottom Status Bar:** Total rows: 6 of 6, Query complete 00:00:00.157, Ln 175, Col 1.
- System Tray:** Shows system icons including battery level, signal strength, and date/time (24-04-2023, 16:01).

booking_id	transportation_id	class_id	from_station	to_station	departure_timestamp	arrival_timestamp	pnr	booking_status	invoice_id	
1	7038195	6E2518	21	BOM	DEL	2023-05-21 20:10:00	2023-05-21 22:25:00	8657139425	BOOKED	2J4E6F8D9G
2	6519238	22185	11	NNND	NVS	2023-05-27 21:15:00	2023-05-28 00:21:00	9138652475	WAITING	6F3D9G2J1E
3	4876392	23104	11	CSMT	ADI	2023-05-17 14:05:00	2023-05-17 21:05:00	3724968513	WAITING	7H0C8ISB3G
4	2986415	14702	13	ST	JP	2023-05-15 00:55:00	2023-05-15 17:50:00	2475913865	BOOKED	5B7A0H1C8I
5	3601975	6E2016	23	DEL	BOM	2023-05-20 15:45:00	2023-05-20 18:00:00	4851693724	WAITING	4A1J6E9F7D
6	8502167	1479	31	JPR	BKR	2023-05-31 13:30:00	2023-05-31 18:45:00	9425368715	BOOKED	3G9I1B5C7A

Query 19:- Price distribution

```
CREATE VIEW RFB AS
SELECT BOOKING_ID,R.TRANSPORTATION_ID,CLASS_ID,FROM_STATION,INVOICE_ID,stop_number AS
FROM_stop_no,km_from_origin AS FROM_km
FROM (SELECT BOOKING_ID,TRANSPORTATION_ID,CLASS_ID,FROM_STATION,INVOICE_ID FROM O_TRANS_B)
AS OTBF
JOIN
(SELECT * FROM ROUTE) AS R
ON (OTBF.TRANSPORTATION_ID = R.TRANSPORTATION_ID
AND OTBF.FROM_STATION = R.STATION_CODE);

--SELECT * FROM RFB;

CREATE VIEW RTB AS
SELECT BOOKING_ID,R.TRANSPORTATION_ID,CLASS_ID,TO_STATION,INVOICE_ID,stop_number AS
TO_stop_no,km_from_origin AS TO_km
FROM (SELECT BOOKING_ID,TRANSPORTATION_ID,CLASS_ID,TO_STATION,INVOICE_ID FROM O_TRANS_B) AS
OTBT
JOIN
(SELECT * FROM ROUTE) AS R
ON (OTBT.TRANSPORTATION_ID = R.TRANSPORTATION_ID
AND OTBT.TO_STATION = R.STATION_CODE);

--SELECT * FROM RTB;

SELECT *,(FF.TOTAL_AMT+ FF.TAXES) AS GRAND_TOTAL FROM
(SELECT F.*,(F.BASE_PRICE+F.BREAKFAST + F.LUNCH + F.DINNER + F.ZERO_CANCELLATION) AS
TOTAL_AMT FROM
(SELECT INVOICE_ID,((to_km-from_km)*PRTFB.BASE_PRICE) as
BASE_PRICE,(PRTFB.BREAKFAST*TADD.BREAKFAST)AS BREAKFAST,
(PRTFB.LUNCH*TADD.LUNCH) AS LUNCH,
(PRTFB.DINNER*TADD.DINNER) AS DINNER,
(PRTFB.ZERO_CANCELLATION*TADD.ZERO_CANCELLATION) AS ZERO_CANCELLATION,
((to_km-from_km) * PRTFB.TAXES) AS TAXES FROM
(SELECT * FROM
((SELECT * FROM RTB
NATURAL JOIN RFB)
AS RTFB NATURAL JOIN TRANSPORTATION_FARES)
AS RTFB WHERE TO_stop_no>FROM_stop_no)
AS PRTFB JOIN TRANSPORTATION_ADD_ON AS TADD ON TADD.BOOKING_ID = PRTFB.BOOKING_ID) AS F)AS
FF;
```


Query 20 :- Seat Availability in Particular Train between Jaipur and Jodhpur

```
SELECT TRANSPORTATION_NAME, CLASS_NAME, AVL.* FROM
(SELECT P.*, TC.CLASS_NAME, TC.CLASS_ID FROM
(SELECT T.TRANSPORTATION_NAME, T.USER_RATING, R.* FROM TRANSPORTATION AS T
NATURAL JOIN
    (SELECT S.TRANSPORTATION_ID, S.STATION_CODE, S.ARRIVAL AS S_ARRIVAL, S.DEPARTURE AS
S_DEPARTURE, S.HALT AS S_HALT, S.DAY_NO AS S_DAY_NO, S.KM_FROM_ORIGIN AS S_KM_FROM_ORIGIN,
D.STATION_CODE, D.ARRIVAL AS D_ARRIVAL, D.DEPARTURE AS D_DEPARTURE, D.HALT AS
D_HALT, D.DAY_NO AS D_DAY_NO, D.KM_FROM_ORIGIN AS D_KM_FROM_ORIGIN FROM
    (SELECT * FROM ROUTE NATURAL JOIN (SELECT * FROM STATION WHERE CITY='Jaipur') AS N)
AS S
    JOIN
        (SELECT * FROM ROUTE NATURAL JOIN (SELECT * FROM STATION WHERE CITY='Jodhpur') AS
A) AS D ON S.TRANSPORTATION_ID=D.TRANSPORTATION_ID
    WHERE S.STOP_NUMBER < D.STOP_NUMBER) AS R ) AS P
NATURAL JOIN
    (TRANSPORTATION_CLASS NATURAL JOIN CLASSES) AS TC) AS F
JOIN AVAILABLE_SEATS AS AVL ON (AVL.TRANSPORTATION_ID=F.TRANSPORTATION_ID AND
AVL.CLASS_ID=F.CLASS_ID) WHERE DEPARTURE_TIMESTAMP BETWEEN '2023-05-16 06:00:00' AND
'2023-05-23 06:00:00';
```

The screenshot shows the pgAdmin 4 interface with the following details:

- File Object Tools Help**: The top navigation bar.
- Browser**: The left sidebar showing database objects like Catalogs, Event Triggers, Extensions, Languages, Publications, Schemas (5), Sales Management, company, public, and wfa.
- Query Editor**: The main area where the SQL query is typed. It includes a search bar, file operations, and a toolbar. The query is pasted into the editor.
- Data Output**: A table showing the results of the query. The columns are:

transportation_name	class_name	transportation_id	class_id	from_station	to_station	departure_timestamp
JP JU EXP	Second sitting	22977	10	JP	JU	2023-05-1
JP JU EXP	Second sitting	22977	10	JP	JU	2023-05-1
JP JU EXP	Second sitting	22977	10	JP	JU	2023-05-1
JP JU EXP	Second sitting	22977	10	JP	JU	2023-05-1
JP JU EXP	Second sitting	22977	10	JP	JU	2023-05-2
JP JU EXP	Second sitting	22977	10	JP	JU	2023-05-2
- Messages**: Shows the status "Query complete 00:00:00.191".
- Notifications**: Shows "Ln 1, Col 1".
- System Bar**: The bottom bar showing system icons, including a weather icon (33°C Smoke), a search bar, and various application icons.
- System Status**: The bottom right corner showing the date (24-04-2023), time (15:05), and battery level (16%).

Query 21 :- Total fares after complete booking

```
CREATE VIEW O_ACCM_B AS
SELECT CD.* , INVOICE_ID FROM
(SELECT * FROM
(SELECT BOOKING_ID FROM BOOKING
EXCEPT
SELECT BACKEND_BOOKING_ID FROM PACKAGE_BOOKING_DETAILS) AS AB
NATURAL JOIN ACCOMMODATION_BOOKING) AS CD
NATURAL JOIN BOOKING
;

SELECT *,(FF.TOTAL+FF.TAXES) AS GRAND_TOTAL FROM
(SELECT INVOICE_ID,(F.BASE_PRICE+F.BREAKFAST + F.LUNCH_OR_DINNER + F.DINNER +
F.ZERO_CANCELLATION) AS TOTAL ,TAXES FROM
(SELECT INVOICE_ID,AF.BASE_PRICE,(AF.BREAKFAST*AD.BREAKFAST) AS
BREAKFAST,(AF.LUNCH_OR_DINNER*AD.LUNCH_OR_DINNER) AS
LUNCH_OR_DINNER,(AF.LUNCH_AND_DINNER*AD.DINNER) AS
DINNER,(AF.ZERO_CANCELLATION*AD.ZERO_CANCELLATION) AS ZERO_CANCELLATION,AF.TAXES FROM
(SELECT * FROM O_ACCM_B
NATURAL JOIN ACCOMMODATION_FARES) AS AF
JOIN ACCOMMODATION_ADD_ON AS AD ON (AF.BOOKING_ID=AD.BOOKING_ID) ) AS F) AS FF;
```

The screenshot shows the pgAdmin 4 application window. The left sidebar is titled "Browser" and contains a tree view of database objects under "Schemas (5)". The "wfa" schema is expanded, showing tables like "accommodation", "available_rooms", and "available_seats". The top right features a "Properties" tab and a "SQL" tab where a query has been run. The results of the query are displayed in a table:

	invoice_id	total	taxes	grand_total
1	8G2B7C1F6E	4505	1107	5612
2	1B7E3F0G9D	1700	676	2376
3	9J015D4H3A	950	100	1050
4	2H5C4A8J6I	700	120	820
5	3F6D1G9B0J	1350	180	1530
6	0E4I7A8H3D	1650	120	1770

At the bottom, status bars show "Total rows: 6 of 6" and "Query complete 00:00:00.418". The system tray at the bottom right includes icons for battery level (33°C Smoke), network, volume, and date/time (24-04-2023 15:16).

Query 22 :- Transaction Count of payments made using UPI

```
SELECT COUNT(TRANSACTION_ID) FROM TRANSACTION WHERE TRANSACTION_MODE = 'UPI';
```

The screenshot shows the pgAdmin 4 interface. The left sidebar is the 'Browser' pane, displaying a tree view of database objects under the 'wfa' schema, including tables like 'accommodation', 'available_rooms', and 'available_seats'. The main area is the 'Query' editor, which contains the following SQL code:

```
1 SET SEARCH_PATH TO WFA;
2
3 SELECT COUNT(TRANSACTION_ID) FROM TRANSACTION WHERE TRANSACTION_MODE = 'UPI';
```

The 'Data Output' tab shows the result of the query:

count	bigint
1	13

At the bottom of the pgAdmin window, a status bar indicates: 'Total rows: 1 of 1 Query complete 00:00:00.235'. To the right of the status bar is a green success message: 'Successfully run. Total query runtime: 235 msec. 1 rows affected.' The system tray at the bottom of the screen shows various icons and the date/time: '24-04-2023 15:28'.

Query 23 :- Number of Cancellations and the amount to be refunded

```
SELECT COUNT(CANCELLATION_ID), SUM(TOTAL_REFUNDABLE_AMOUNT) FROM CANCELLATION WHERE  
REFUND_STATUS = 'IN PROCESS';
```

The screenshot shows the pgAdmin 4 interface. The left sidebar is titled 'Browser' and lists various database objects under the schema 'wfa'. The main area is a query editor with the following content:

```
202101208/202101208@wfa*  
Query History  
1 SET SEARCH_PATH TO WFA;  
2  
3 SELECT COUNT(CANCELLATION_ID), SUM(TOTAL_REFUNDABLE_AMOUNT) FROM CANCELLATION WHERE REFUND_STATUS = 'IN PRO'
```

The 'Data Output' tab is selected, displaying the results of the query:

	count	sum
	bigrint	real
1	2	10455.5

At the bottom of the pgAdmin window, the taskbar shows the system tray and system status.

Query 24 :- Rooms available for particular period

```
SELECT ACCOMMODATION_NAME, ROOM_TYPE_NAME, MIN(AVAILABLE_ROOMS) AS AVAILABLE FROM  
AVAILABLE_ROOMS AS AR NATURAL JOIN ROOM_TYPE NATURAL JOIN ACCOMMODATION  
WHERE CITY = 'Manali' AND (CHECK_IN_DATE = '2023-05-20' OR CHECK_OUT_DATE = '2023-05-22')  
GROUP BY ACCOMMODATION_NAME, ROOM_TYPE_NAME;
```

The screenshot shows the pgAdmin 4 application window. The left sidebar displays a tree view of database objects under the 'Browser' tab, including Catalogs, Event Triggers, Extensions, Foreign Data Wrappers, Languages, Publications, Schemas (5), Sales Management, company, public, wfa, Aggregates, Collations, Domains, FTS Configuration, FTS Dictionary, FTS Parsers, FTS Templates, Foreign Tables, Functions, Materialized Views, Operators, Sequences, and Tables (36). The 'Tables (36)' node is currently selected. The main pane contains a query editor with the following SQL code:

```
1 SET SEARCH_PATH TO WFA;
2
3 SELECT ACCOMMODATION_NAME, ROOM_TYPE_NAME, MIN(AVAILABLE_ROOMS) AS AVAILABLE FROM
4 AVAILABLE_ROOMS AS AR NATURAL JOIN ROOM_TYPE NATURAL JOIN ACCOMMODATION
5 WHERE CITY = 'Manali' AND (CHECK_IN_DATE = '2023-05-20' OR CHECK_OUT_DATE = '2023-05-22')
6 GROUP BY ACCOMMODATION_NAME, ROOM_TYPE_NAME;
```

Below the query editor is a 'Data Output' section with tabs for Messages and Notifications. A results grid displays the following data:

	accommmodation_name	room_type_name	available
1	Hotel Mountain Top	Honeymoon Room	7
2	Sarthak Resorts	Honeymoon Room	8

At the bottom of the screen, the taskbar shows system icons for weather (33°C Smoke), search, file explorer, and various applications like Microsoft Word, Outlook, and browser. The status bar indicates 'Total rows: 2 of 2' and 'Query complete 00:00:00.261'. The bottom right corner shows the date and time as '24-04-2023 16:24'.

Query 25 :- Displaying Invoice details

```
SELECT * FROM INVOICE_DETAILS;
```

The screenshot shows the pgAdmin 4 interface with the following details:

- File Object Tools Help** (top menu)
- Browser** (left sidebar, expanded to show Catalogs, Event Triggers, Extensions, Foreign Data Wrappers, Languages, Publications, Schemas (5), Sales Management, company, public, wfa, Aggregates, Collations, Domains, FTS Configuration, FTS Dictionaries, FTS Parsers, FTS Templates, Foreign Tables, Functions, Materialized Views, Operators, Sequences, and Tables (36)).
- Query History** (dropdown menu) set to "202101208/202101208@wfa".
- Query** (main area):

```
SET SEARCH_PATH TO WFA;
```
- Data Output** (selected tab) showing the results of the query:

invoice_id	total_amount	taxes	grand_total	total_amount_paid	amount_left	next_installment_date
1	29064	5918.96	34983	20000	14983	2023-05-20
2	169478	31573	201051	75000	126051	2023-05-18
3	127109	23679.7	150789	150789	0	[null]
4	190100	38465.2	228565	150000	78565.4	2023-05-17
5	133475	25657.9	159133	10000	149133	2023-05-21
6	1650	120	1770	1890	-120	2023-04-25
7	1350	180	1530	450	1080	2023-04-15
8	1400	240	1640	420	1220	2023-04-24
9	1700	676	2376	1100	1276	2023-04-19
10	2850	300	3150	750	2400	2023-04-20
11	4505	1107	5612	1250	4362	2023-04-25
12	10987.6	1802.36	12790	10000	2789.96	2023-04-23
13	790	88.2	878.2	878.2	0	[null]
14	686	69.48	755.48	755.48	0	[null]
15	5B7A0H1C8I	1885	307.8	2192.8	0	[null]
16	2J4E6FB9G	13239.6	2100.84	15340.4	10000	5340.44
17	3G9I1B5C7A	400.8	73.4799	474.28	474.28	0

- Total rows: 17 of 17 Query complete 00:00:00.465 Ln 3, Col 1** (status bar)
- System tray icons** (bottom): 33°C, Smoke, Search, File Explorer, Task View, Mail, Edge, Google Chrome, WhatsApp, Microsoft Teams, Visual Studio Code, Python, 16:30, ENG IN, Wi-Fi, Battery, 24-04-2023.