Tourism Management System

Group Number: 7

TA: Kashyap Nirmal (202011031)

- Riya Dineshkumar Soni (202012025)
- Kakkan Anurag Kishor (202012026)
- Gandhi Viral Ashok (202012027)
- Sukhadia Rutvi Kumarpal (202012028)
- Shah Siddhant Alkeshbhai (202018013)
- Shah Nihar Shaileshbhai (202018014)

SQL Queries and Relational Algebra

SET search_path TO Tourism_Management_System;

1) Retrieve the packages within a particular price range

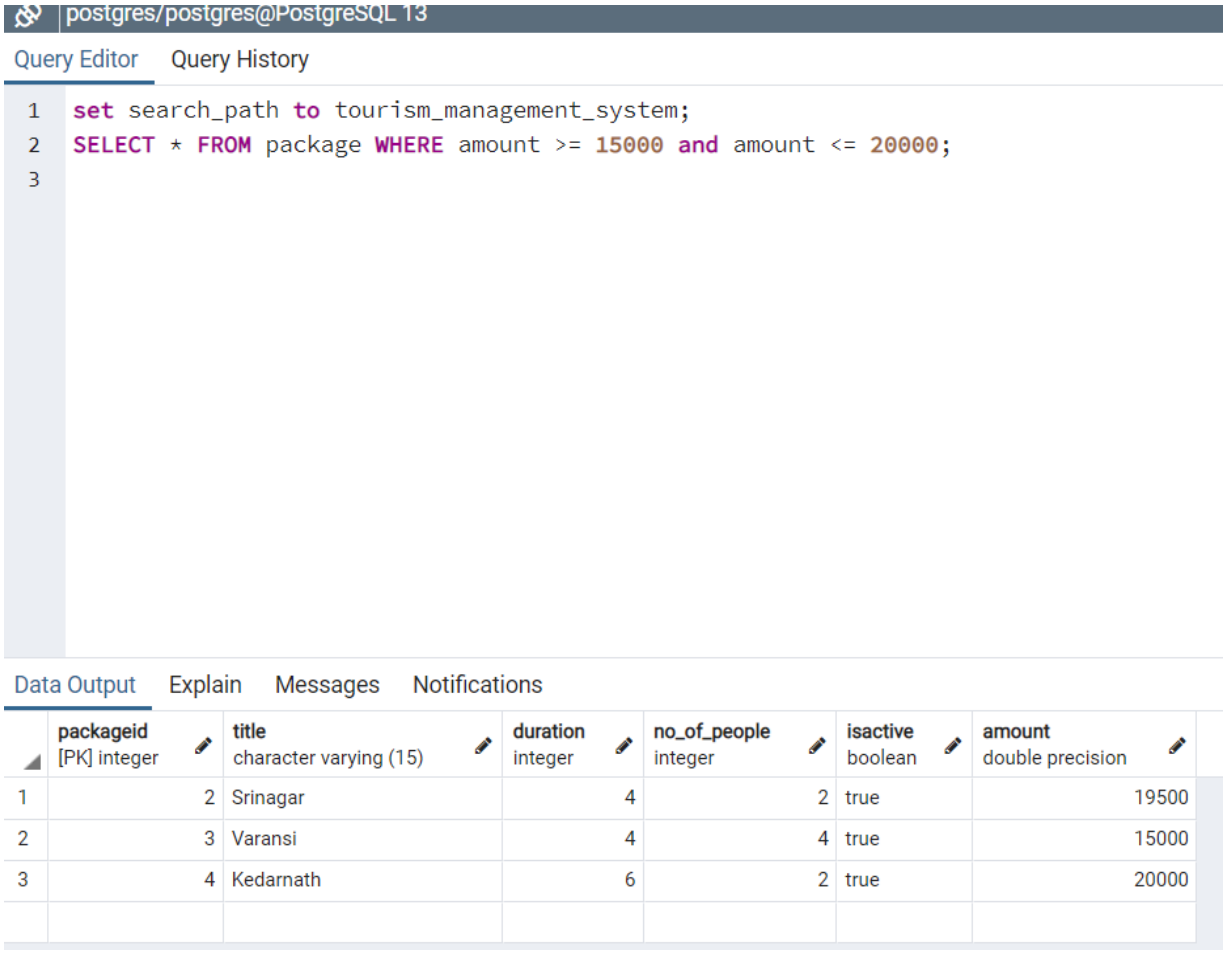
--Relational Algebra:

$\sigma_{\text{(amount >=15000 AND amount <= 20000)}}$ (package)

--SQL Query:

```
SELECT * FROM package WHERE amount >= 15000 and amount <= 20000;
```

--Output:



	packageid [PK] integer	title character varying (15)	duration integer	no_of_people integer	isactive boolean	amount double precision
1	2	Srinagar	4	2	true	19500
2	3	Varansi	4	4	true	15000
3	4	Kedarnath	6	2	true	20000

2) Show the list of top 3 packages based on the number of users who booked it.

--Relational Algebra:

r1 -> ρ (bfp, (σ Order By packageid,Count(bid) desc LIMIT 3 ($_{packageid}\mathcal{F}_{COUNT(bid)}$(Booking_for_package)))

r2 -> ρ (p, package) $\bowtie_{<p.packageid = bfp.packageid>}$(r1)

result -> $\prod_{p.packageid, title, duration, no\_of\_people, amount}$(r2)

--SQL Query:

```
SELECT bfp.number_of_users, p.packageid, title, duration, no_of_people, amount
FROM package as p JOIN
(SELECT COUNT (bid) AS number_of_users, packageid FROM Booking_for_package
 GROUP BY packageid order by packageid,COUNT(bid) desc LIMIT 3) as bfp
ON (bfp.packageID = p.packageID);
```

--Output:

```
1  SET search_path TO Tourism_Management_System;
2
3  --Show the list of top 3 packages based on the number of users who booked it.
4  SELECT bfp.number_of_users,p.packageid, title, duration, no_of_people, amount FROM package as p
5  JOIN
6  (SELECT COUNT(bid) AS number_of_users, packageid FROM Booking_for_package
7   GROUP BY packageid order by packageid,COUNT(bid) desc LIMIT 3) as bfp
8  ON (bfp.packageID = p.packageID);
```

Data Output | Explain | Messages | Query History | Notifications

	number_of_users bigint	packageid [PK] integer	title character varying (15)	duration integer	no_of_people integer	amount double precision
1	3	1	Manali Tour	9	8	64000
2	1	5	Taj Mahal Tour	2	4	12000
3	1	9	Lonavla Tour	2	6	12000

3) Retrieve list of all the active packages with full details.

--Relational Algebra:

r1 -> ρ (p, package) LEFT $\bowtie_{<p.packageid = ps.packageid>}$ ρ (ps, package_includes_spots)

LEFT $\bowtie_{<ps.spotid = ts.spotid>}$ ρ (ts,tourist_spots)

LEFT $\bowtie_{<p.packageid = ph.packageid>}$ ρ (ph, package_includes_hotels)

LEFT $\bowtie_{<ph.hotelid = r.hotelid AND ph.room_no=r.room_no>}$ ρ (r,room)

LEFT $\bowtie_{<r.hotelid = h.hotelid>}$ ρ (h, hotel)

LEFT $\bowtie_{<p.packageid = pg.packageid>}$ ρ (pg, package_includes_guides)

LEFT $\bowtie_{<pg.guideAadharNo = g.guideAadharNo>}$ ρ (g, guide)

result -> $\prod_{(p.packageid, p.title, p.duration, p.no\_of\_people, p.amount, ts."Name" -> Tourist\_spot, CONCAT (g.fname,' ',g.lname) -> Guide, h."Name" -> Hotel, r.room\_no -> Room\_No, r.rate ->Room\_rate)}$ (r1)

--SQL Query:

```
SELECT p.packageId,p.title,p.duration,p.no_of_people,p.amount,
ts."Name" AS Tourist_Spot,CONCAT(g.fname, ' ', g.lname) AS Guide,
h."Name" As Hotel,r.room_no,r.rate AS Room_Rate FROM package as p
Left JOIN package_includes_spots as ps ON (p.packageid=ps.packageid)
Left JOIN tourist_spots AS ts ON (ps.spotid = ts.spotid)
Left JOIN package_includes_hotels as ph ON (p.packageid=ph.packageid)
Left JOIN ROOM AS r ON (ph.hotelid = r.hotelid AND ph.room_no = r.room_no)
Left JOIN Hotel AS h ON (r.hotelid = h.hotelid)
Left JOIN package_includes_guides as pg ON (p.packageid=pg.packageid)
Left JOIN guide AS g ON (pg.guideAadharNo=g.guideAadharNo)
where p.isActive=true;
```

--Output:

Data Output | Explain | Messages | Query History | Notifications

	packageid integer	title character varying (15)	duration integer	no_of_people integer	amount double prec	tourist_spot character varying (20)	guide text	hotel character varying (50)	room_no numeric (3)	room_rate double precisio
1	1	Manali Tour	9	8	64000	Manali	Sachin Bhide	Hotel Thomas Villa	4	5000
2	1	Manali Tour	9	8	64000	Manali	Sachin Bhide	Hotel Thomas Villa	3	5000
3	2	Srinagar Tour	4	2	19500	Srinagar	Jay Prajapati	Billberry Hotel	1	3000
4	3	Varansi Tour	4	4	15000	Dashashwamedh Ghat	Jiimy Nisham	Hotel Sai International	3	5000
5	4	Kedarnath Tour	6	2	20000	Kedarnath Temple	Suresh Raina	Punjab Sindh Awas	1	4000
6	5	Taj Mahal Tour	2	4	12000	Taj Mahal	Virat Kohli	Anmol Hotel Pvt.Ltd	2	6500
7	6	Jaisalmer Tour	2	6	12500	Jaisalmer Fort	Ajay Bishnoi	Jaisalkot - A Luxury Boutiqu...	2	5500
8	7	Somnath Tour	2	4	6000	Somnath Temple	Abhi Baluni	The Fern Residency	1	3000
9	8	Mumbai Tour	2	4	10000	Gate Way Of India	Shikhar Dhoni	Meritas Picaddle Resort	1	3000
10	8	Mumbai Tour	2	4	10000	Gate Way Of India	Shikhar Dhoni	Central Hotel	3	3000
11	9	Lonavla Tour	2	6	12000	Lonavla Hills	Shakil Narayan	[null]	[null]	[null]

4) View list of all the tourist spots at a particular location.

--Relational Algebra:

r1 -> ρ (ts, tourist_spots) $\bowtie_{<ts.pincode = l.pincode>}$ ρ (l, location)

result -> Π (“Name”, season, ratings, CONCAT (ts.address,', ',l.city,', ',l.state, ' - ',ts.pincode) -> Address) (σ l.city=”kullu”(r1))

--SQL Query:

```
SELECT ts."Name", ts.season, ts.ratings,
 CONCAT (ts.address,', ',l.city,', ',l.state, ' - ',ts.pincode) AS "Address"
FROM tourist_spots AS ts JOIN location AS l ON ts.pincode = l.pincode
where l.city = 'Kullu';
```

--Output:

Query Editor Query History

```
1  set search_path to tourism_maSELECT ts."Name", ts.season, ts.ratings,
2  SELECT ts."Name", ts.season, ts.ratings,
3   CONCAT (ts.address,', ',l.city,', ',l.state, ' - ',ts.pincode) AS "Address"
4  FROM tourist_spots AS ts JOIN location AS l ON ts.pincode = l.pincode
5  where l.city = 'Kullu';
6
7
8
```

Data Output Explain Messages Notifications

	Name character varying (20)	season character varying (10)	ratings double precision	Address text
1	Manali	Winter	4.7	Rotang Pass...

5) View the tourist spots included in "abc" package.

--Relational Algebra:

r1 -> ρ (l, location) $\bowtie_{<l.pincode = ts.pincode>}$ ρ (ts, tourist_spots) $\bowtie_{<ts.spotid = pt.spotid>}$ ρ (pt, package_includes_spots) $\bowtie_{<pt.packageid = p.packageid>}$ ρ (p, package)

result -> $\prod_{(title, duration, no\_of\_people, amount, \text{“Name”}, rating, CONCAT (ts.address,', ',l.city,', ',l.state, ' - ',ts.pincode) -> Address)}$ ($\sigma_{title=\text{”Manali Tour”}}$(r1))

--SQL Query:

SELECT p.title, p.duration, p.no_of_people, p.amount, ts."Name", ts.season, ts.ratings,
CONCAT (ts.address,', ',l.city,', ',l.state, ' - ',ts.pincode) AS "Address"
FROM tourist_spots AS ts
JOIN package_includes_spots pt ON(pt.spotid = ts.spotid)

JOIN package AS p ON (p.packageid = pt.packageid)

JOIN location AS l ON ts.pincode = l.pincode

where p.title = 'Manali Tour';

--Output:

Query Editor | Query History | Scratch Pad

```
1  set search_path to tourism_maSELECT ts."Name", ts.season, ts.ratings,
2  SELECT p.title, p.duration, p.no_of_people, p.amount, ts."Name", ts.season, ts.ratings,
3  CONCAT (ts.address,', ',l.city,', ',l.state, ' - ',ts.pincode) AS "Address"
4  FROM tourist_spots AS ts
5  JOIN package_includes_spots pt ON(pt.spotid = ts.spotid)
6  JOIN package AS p ON (p.packageid = pt.packageid)
7  JOIN location AS l ON ts.pincode = l.pincode
8  where p.title = 'Manali Tour';
9
10
11
12
```

Data Output | Explain | Messages | Notifications

	title character varying (15)	duration integer	no_of_people integer	amount double precision	Name character varying (20)	season character varying (10)	ratings double precision	Address text
1	Manali Tour	9	8	64000	Manali	Winter	4.7	Rotang Pass...

6) Retrieve the tourist spot with highest user ratings

--Relational Algebra:

r1 -> $\mathcal{F}_{MAX(ratings)->ratings}$ (ρ (ts2, tourist_spots))

r2 -> r1 $\bowtie_{<ts2.ratings = ts1.ratings>}$ ρ (ts1, tourist_spots) $\bowtie_{<ts1.pincode = l.pincode>}$ ρ (l, location))

result -> $\prod$ "Name", season, ts2.ratings, address, t1.pincode, city, state(r2)

--SQL Query:

SELECT "Name", season, ts2.ratings,

CONCAT (ts1.address,', ', l.city,', ',l.state, ' - ',ts1.pincode) AS "Address"

FROM tourist_spots AS ts1

JOIN

(SELECT MAX (ratings) AS ratings FROM tourist_spots) AS ts2

ON (ts1.ratings = ts2.ratings)

JOIN Location as l

ON (ts1.pincode = l.pincode);

--Output:

Query Editor Query History

```
1  set search_path to tourism_maSELECT ts."Name", ts.season, ts.ratings,
2  SELECT "Name", season, ts2.ratings,
3  CONCAT (ts1.address,', ', l.city,', ',l.state, ' - ',ts1.pincode) AS "Address"
4  FROM tourist_spots AS ts1
5  JOIN
6  (SELECT MAX (ratings) AS ratings FROM tourist_spots) AS ts2
7  ON (ts1.ratings = ts2.ratings)
8  JOIN Location as l
9  ON (ts1.pincode = l.pincode);
10
11
12
13
14
```

Data Output Explain Messages Notifications

	Name character varying (20)	season character varying (10)	ratings double precision	Address text
1	Golden	All	4.9	Golden Tem...
2	Taj	All	4.9	Dharmapuri, ...

7) View all the restaurants that serve “only veg” food at a particular location.

--Relational Algebra:

result -> Π (“Name”, phone, foodtype, rating, CONCAT (r.address,', ', l.city, ', ', l.state, ' - ', r.pincode) -> Address) (σ foodtype=”Veg” AND l.city=”Ahmedabad” (ρ (r, restaurant) $\bowtie_{<r.pincode = l.pincode>}$ ρ (l, location)))

--SQL Query:

SELECT r."Name", r.phone, r.foodType, r.ratings,

CONCAT (r.address,', ', l.city, ', ', l.state, ' - ', r.pincode) AS "Address"

FROM restaurant AS r JOIN location AS l ON r.pincode = l.pincode

where r.foodType = 'VEG' AND l.city = 'Ahmedabad';

--Output:

Query Editor

```
1  SET search_path TO Tourism_Management_System;
2
3  --View all the restaurants that serve “only veg” food at a particular location.
4  SELECT r."Name", r.phone, r.foodType, r.ratings,
5  CONCAT (r.address,', ', l.city, ', ', l.state, ' - ', r.pincode) AS "Address"
6  FROM restaurant AS r JOIN location AS l ON r.pincode = l.pincode
7  where r.foodType = 'VEG' AND l.city = 'Ahmedabad';
8
```

Data Output | Explain | Messages | Query History | Notifications

	Name character varying (50)	phone numeric (10)	foodtype character varying	ratings double precision	Address text
1	Huber & Holly	9889855455	VEG	2.5	Shreekunj Mandapam, Beside Golden Tulip Bunglows & Tulip Citadel, Manekbaug, Ambavadi, Ahmedabad, Gujarat - 38...
2	Cryo Lab	9876543210	VEG	3	Ground Floor, Arjun Avenue, Opposite Samartheshwar Mahadev, Law Garden, Ahmedabad, Gujarat - 380027

8) View all the restaurants that have “Chinese” cuisine included in their menu.

--Relational Algebra:

r1 -> ρ (l, location) $\bowtie_{<l.pincode = r.pincode>}$ ρ (r, restaurant) $\bowtie_{<r.rid = rc.rid>}$ ρ (rc, restaurant_cuisines)

result -> $\prod$ (“Name”, phone, foodtype, ratings, cuisines, CONCAT (r.address,', ', l.city, ', ', l.state, ' - ', r.pincode) -> Address) (σ cuisines=”Chinese”(r1))

--SQL Query:

SELECT r."Name", r.phone, r.foodType, r.ratings, rc.cuisines,

CONCAT (r.address,', ', l.city, ', ', l.state, ' - ', r.pincode) AS "Address"

FROM restaurant AS r JOIN location AS l ON r.pincode = l.pincode

JOIN restaurant_cuisines AS rc ON r.rid = rc.rid WHERE rc.cuisines = 'Chinese';

--Output:

Query Editor Query History

```
1  set search_path to tourism_management_system;
2  SELECT r."Name", r.phone, r.foodType, r.ratings, rc.cuisines,
3  CONCAT (r.address,', ', l.city, ', ', l.state, ' - ', r.pincode) AS "Address"
4  FROM restaurant AS r JOIN location AS l ON r.pincode = l.pincode
5  JOIN restaurant_cuisines AS rc ON r.rid = rc.rid WHERE rc.cuisines = 'Chinese';
6
7
8
```

Data Output Explain Messages Notifications

	Name character varying (50)	phone numeric (10)	foodtype character varying (20)	ratings double precision	cuisines character varying (20)	Address text
1	Jahanpanah	9898456721	VEG	4.5	Chinese	E 23, Shoppi...
2	Huber &	9889855455	VEG	2.5	Chinese	Shreekunj M...
3	Tandoor	7954215885	NON-VEG	4	Chinese	17/33, Maha...
4	Three Dots	7878252364	NON-VEG	3.5	Chinese	840/1,100 F...
5	ECHOES	9465853246	BOTH	3	Chinese	44, 4th B Cr...

9) Retrieve all the hotels that are situated at location “xyz”.

--Relational Algebra:

result -> $\prod$ (“Name”, phone, foodtype, ratings, cuisines, CONCAT (h.address,', ', l.city,', ', l.state,' - ', h.pincode) -> Address) (σ city=”Ahmedabad” (ρ (h, hotel) $\bowtie_{<h.pincode = l.pincode>}$ ρ (l, location)))

--SQL Query:

```
SELECT h."Name", h.phone, h.foodType, h.ratings,
CONCAT (h.address,', ', l.city,', ', l.state,' - ', h.pincode) AS "Address"
FROM hotel AS h JOIN location AS l ON h.pincode = l.pincode WHERE  l.city = 'Ahmedabad';
```

--Output:

Query Editor | Query History

```
1  set search_path to tourism_management_system;
2  SELECT h."Name", h.phone, h.foodType, h.ratings,
3  CONCAT (h.address,', ', l.city,', ', l.state,' - ', h.pincode) AS "Address"
4  FROM hotel AS h JOIN location AS l ON h.pincode = l.pincode WHERE  l.city = 'Ahmedabad';
5
6
7
8
```

Data Output | Explain | Messages | Notifications

	Name character varying (50)	phone numeric (10)	foodtype character varying (20)	ratings double precision	Address text
1	Ahmedabad	9878456512	BOTH	4.5	Vastrapur,Ah...
2	Hyatt	8794561251	BOTH	4.8	Opp Ahmed...
3	The Metropole	7889451575	BOTH	4.2	Near R.T.O. ...

10) Retrieve list of hotels that are providing "xyz" services.

--Relational Algebra:

r1 -> ρ (l, location) $\bowtie_{\text{<l.pincode = h.pincode>}}$ ρ (h, hotel) $\bowtie_{\text{<h.hotelid = hs.hotelid>}}$ ρ (hs, hotel_services)

result -> $\prod$ ("Name", phone, foodtype, ratings, services, CONCAT (h.address,', ', l.city,', ', l.state,' - ', h.pincode) -> Address) (σ servies="Gym"(r1))

--SQL Query:

SELECT h."Name", h.phone, h.foodType, h.ratings,hs.services,

CONCAT (h.address,', ', l.city,', ', l.state,' - ', h.pincode) AS "Address"
FROM hotel AS h
JOIN location AS l ON h.pincode = l.pincode
JOIN hotel_services AS hs ON h.hotelid = hs.hotelid
where hs.services = 'Gym';

--Output:

Query Editor | Query History

```
39
40  SELECT h."Name", h.phone, h.foodType, h.ratings,hs.services,
41  CONCAT (h.address,', ', l.city,', ', l.state,' - ', h.pincode) AS "Address"
42  FROM hotel AS h
43  JOIN location AS l ON h.pincode = l.pincode
44  JOIN hotel_services AS hs ON h.hotelid = hs.hotelid
45  where hs.services = 'Gym';
46
47
48
49
50
```

Notifications | Explain | Data Output | Messages

	Name character varying (50)	phone numeric (10)	foodtype character varying (20)	ratings double precision	services character varying (50)	Address text
1	Hotel Thomas Villa	9855004767	VEG	3.5	Gym	Simsa Villag...
2	Punjab Sindh Awas	7858495615	BOTH	3	Gym	Main Market...
3	Jaisalkot - A Luxury Boutiqu...	9116010801	VEG	4.6	Gym	Kuldhara Tu...
4	Statue of Unity Tent City 1	9797949494	VEG	4.5	Gym	Sardar Sarov...
5	Aurick Hotel	6658498756	NON-VEG	3	Gym	15th Cross, ...

11) Retrieve the hotel with highest user ratings

--Relational Algebra:

r1 -> $\mathcal{F}_{MAX\ (ratings)}$ (hotel)

r2 -> $_{>}$ ρ (h, hotel) $\bowtie_{<h.pincode = l.pincode>}$ ρ (l, location)

result -> $\prod$ ("Name", phone, foodtype, ratings, services, CONCAT (h.address,', ', l.city,', ', l.state,' - ', h.pincode) -> Address) (σ ratings IN (r1)="Gym"(r2))

--SQL Query:

```
SELECT h."Name", h.phone, h.foodType, h.ratings,
CONCAT (h.address,', ', l.city,', ', l.state,' - ', h.pincode) AS "Address"
FROM hotel AS h JOIN location AS l ON h.pincode = l.pincode
where h.ratings IN (SELECT max(ratings) from hotel);
```

--Output:

Query Editor | Query History

```
1  set search_path to tourism_management_system;
2  SELECT h."Name", h.phone, h.foodType, h.ratings,
3  CONCAT (h.address,', ', l.city,', ', l.state,' - ', h.pincode) AS "Address"
4  FROM hotel AS h JOIN location AS l ON h.pincode = l.pincode
5  where h.ratings IN (SELECT max(ratings) from hotel);
6
7
8
```

Data Output | Explain | Messages | Notifications

	Name character varying (50)	phone numeric (10)	foodtype character varying (20)	ratings double precision	Address text
1	Hyatt	8794561251	BOTH	4.8	Opp Ahmed...

12) Retrieve list of hotels sorted according to their user ratings.

--Relational Algebra:

r1-> ρ (h, hotel) $\bowtie_{<h.pincode = l.pincode>}$ ρ (l, location)

result -> $\prod_{(\text{“Name”, phone, foodtype, ratings, services, CONCAT (h.address,', ', l.city,', ', l.state,' - ', h.pincode) -> Address})}$ (σ ORDER BY DESC $_{h.ratings}$ (r1))

--SQL Query:

```
SELECT h."Name", h.phone, h.foodType, h.ratings,
```

CONCAT (h.address,', ', l.city,', ', l.state,' - ', h.pincode) AS "Address"

FROM hotel AS h JOIN location AS l ON h.pincode = l.pincode

ORDER BY h.ratings DESC;

--Output:

Query Editor Query History

```
51
52  SELECT h."Name", h.phone, h.foodType, h.ratings,
53  CONCAT (h.address,', ', l.city,', ', l.state,' - ', h.pincode) AS "Address"
54  FROM hotel AS h JOIN location AS l ON h.pincode = l.pincode
55  ORDER BY h.ratings DESC;
56
57
58
```

Notifications Explain Data Output Messages

	Name character varying (50)	phone numeric (10)	foodtype character varying (20)	ratings double precision	Address text
1	Hyatt Ahmedabad	8794561251	BOTH	4.8	Opp Ahmed...
2	ITC Grand Central, A Luxury ...	8978152345	BOTH	4.6	287, Dr, Dr B...
3	Jaisalkot - A Luxury Boutiqu...	9116010801	VEG	4.6	Kuldhara Tu...
4	ITC Mughal, A Luxury Collec...	5624021700	BOTH	4.5	Itc Mughal, ...
5	Central Hotel	9894517223	BOTH	4.5	Ashirwad En...
6	Ahmedabad Inn	9878456512	BOTH	4.5	Vastrapur, A...
7	Statue of Unity Tent City 1	9797949494	VEG	4.5	Sardar Saro...
8	The Imperial Palace	7894556218	VEG	4.4	Dr Yagnik Rd...
9	The Metropole Hotel	7889451575	BOTH	4.2	Near R.T.O. ...
10	Cosset-Comfort And Cusine	7859485625	BOTH	4.2	Mumbai Pun...
11	Meritas Picaddle Resort	9689521111	BOTH	4.2	Plot No. 13...
12	Hotel Levana	8989456512	NON-VEG	4.1	6, VS Marg, ...
13	Ahdoos Hotel	7889464612	VEG	4.1	Residency r...
14	The Fern Residency	2876225200	VEG	4.1	Talala Road, ...
15	Astoria Hotel	9726549956	BOTH	4	J. Tata Road...
16	Billberry Hotel	9865476521	NON-VEG	4	Ghat No. 17,...
17	Ramada by Wyndham Amrit...	1835025555	BOTH	3.9	117 Hall Baz...
18	Hotel Kashi	9878451532	VEG	3.8	No.74-A, Ka...
19	The White Hotels	6868626585	VEG	3.5	181/1, Oppo...
20	Hotel Thomas Villa	9855004767	VEG	3.5	Simsa Villag...
21	Anmol Hotel Pvt.Ltd	9898565844	VEG	3.5	8180 Street ...
22	Aurick Hotel	6658498756	NON-VEG	3	15th Cross, ...
23	BB Palace-A Boutique Hotel	8655855687	VEG	3	2638-2642 ...
24	Punjab Sindh Awas	7858495615	BOTH	3	Main Market...
25	Chouki Dhani	9465813587	VEG	3	Near All Indi...
26	Hotel Sai International	7889456124	VEG	2.9	Chandpur In...

13) View list of hotel rooms starting from the Lowest Price to Highest Price.

--Relational Algebra:

r1 -> ρ (h, hotel) $\bowtie_{<h.hotelid = r.hotelid>}$ ρ (r, room)

result -> $\prod_{(h.\text{“Name”->Hotel\_Name,r.room->Room\_Number,r.Type->Room\_Type,r.beds->No\_of\_beds,r.capacity->capacity,r.rate->price,r.status->Current\_Status})}$ (σ ORDER $BY_{r.rate,h.\text{“name”},r.room\_no}$ (r1))

--SQL Query:

```
SELECT h."Name" As "Hotel_Name", r.room_no AS "Room_Number", r."Type" AS
"Room_Type",
r.beds AS "No_of_Beds", r.capacity AS "Capacity", r.rate AS "Price", r.status As
"Current_Status"
FROM hotel AS h JOIN room AS r ON h.hotelid = r.hotelid
ORDER BY r.rate,h."Name", r.room_no;
```

--Output:

Query Editor | Query History

```
57  SELECT h."Name" As "Hotel_Name", r.room_no AS "Room_Number", r."Type" AS "Room_Type",
58  r.beds AS "No_of_Beds", r.capacity AS "Capacity", r.rate AS "Price", r.status As "Current_Status"
59  FROM hotel AS h JOIN room AS r ON h.hotelid = r.hotelid
60  ORDER BY r.rate,h."Name", r.room_no;
61
62
```

Notifications | Explain | Data Output | Messages

	Hotel_Name character varying (50)	Room_Number numeric (3)	Room_Type character varying (6)	No_of_Beds integer	Capacity integer	Price double precision	Current_Status character varying (15)
1	Ahdoos Hotel	2	AC	1	2	1500	AVAILABLE
2	Ahmedabad Inn	2	AC	1	2	1500	AVAILABLE
3	Astoria Hotel	2	AC	1	2	1500	AVAILABLE
4	Aurick Hotel	5	NON-AC	1	2	1500	AVAILABLE
5	BB Palace-A Boutique Hotel	2	AC	1	2	1500	AVAILABLE
6	Billberry Hotel	5	NON-AC	1	2	1500	AVAILABLE
7	Central Hotel	5	NON-AC	1	2	1500	AVAILABLE
8	Chouki Dhani	2	AC	1	2	1500	AVAILABLE
9	Cosset-Comfort And Cusine	2	AC	1	2	1500	AVAILABLE
10	Hotel Kashi	5	NON-AC	1	2	1500	AVAILABLE
11	Hotel Levana	5	NON-AC	1	2	1500	AVAILABLE
12	Hotel Sai International	2	AC	1	2	1500	AVAILABLE
13	Hotel Thomas Villa	10	NON-AC	1	2	1500	AVAILABLE
14	ITC Grand Central, A Luxury ...	5	NON-AC	1	2	1500	AVAILABLE
15	ITC Mughal, A Luxury Collec...	5	NON-AC	1	2	1500	AVAILABLE
16	Jaisalkot - A Luxury Boutiqu...	5	NON-AC	1	2	1500	AVAILABLE
17	Meritas Picaddle Resort	5	NON-AC	1	2	1500	AVAILABLE
18	Punjab Sindh Awas	2	AC	1	2	1500	AVAILABLE
19	Ramada by Wyndham Amrit...	10	NON-AC	1	2	1500	AVAILABLE
20	The Fern Residency	5	NON-AC	1	2	1500	AVAILABLE

14) Retrieve list of hotel rooms that have “Cable TV” facility at a particular location.

--Relational Algebra:

r1 -> ρ (h, hotel) $\bowtie_{<h.pincode=l.pincode>}$ ρ (l, location) $\bowtie_{<h.hotelid= r.hotelid>}$ ρ (r, room)

$\bowtie_{<\ h.hotelid=rf.hotelid\ and\ r.room\_no=rf.room\_no\ >}$ ρ (rf, room_facilities)

Result -> Π (h. “Name”->Hotel_Name,r.room->Room_Number,r.Type->Room_Type,r.beds->No_of_beds,r.capacity->capacity,

r.rate->price,r.status->Current_Status,rf.facility,l.city) (σ (rf.facility='Cable TV' and l.city = 'Amritsar')

ORDER BY$_{r.rate,h.\text{"name"},r.room_no}$ (r1))

--SQL Query:

```
SELECT h."Name" As "Hotel_Name",r.room_no AS "Room_Number", r."Type" AS "Room_Type",
r.beds AS "No_of_Beds", r.capacity AS "Capacity", r.rate AS "Price", r.status As "Current_Status", rf.facility, l.city
FROM hotel AS h JOIN location AS l ON h.pincode = l.pincode
JOIN room AS r ON h.hotelid = r.hotelid
JOIN room_facilities AS rf ON (h.hotelid=rf.hotelid and r.room_no=rf.room_no)
WHERE rf.facility='Cable TV' and l.city = 'Amritsar'
ORDER BY h."Name", r.room_no;
```

--Output:

```
1  set search_path to tourism_management_system;
2  SELECT h."Name" As "Hotel_Name",r.room_no AS "Room_Number", r."Type" AS "Room_Type",
3  r.beds AS "No_of_Beds", r.capacity AS "Capacity", r.rate AS "Price", r.status As "Current_Status", rf.fac
4  FROM hotel AS h JOIN location AS l ON h.pincode = l.pincode
5  JOIN room AS r ON h.hotelid = r.hotelid
6  JOIN room_facilities AS rf ON (h.hotelid=rf.hotelid and r.room_no=rf.roomno)
7  WHERE rf.facility='Cable TV' and l.city = 'Amritsar'
8  ORDER BY h."Name", r.room_no;
```

Data Output | Explain | Messages | Notifications

	Hotel_Name character varying (50)	Room_Number numeric (3)	Room_Type character varying (6)	No_of_Beds integer	Capacity integer	Price double precision	Current_Status character varying (15)	facility character varying (50)	city character varyi
1	Ramada by	1	AC	2	2	3000	AVAILABLE	Cable TV	Amritsar

15) Retrieve all the packages associated with a particular guide. (admin)

--Relational Algebra:

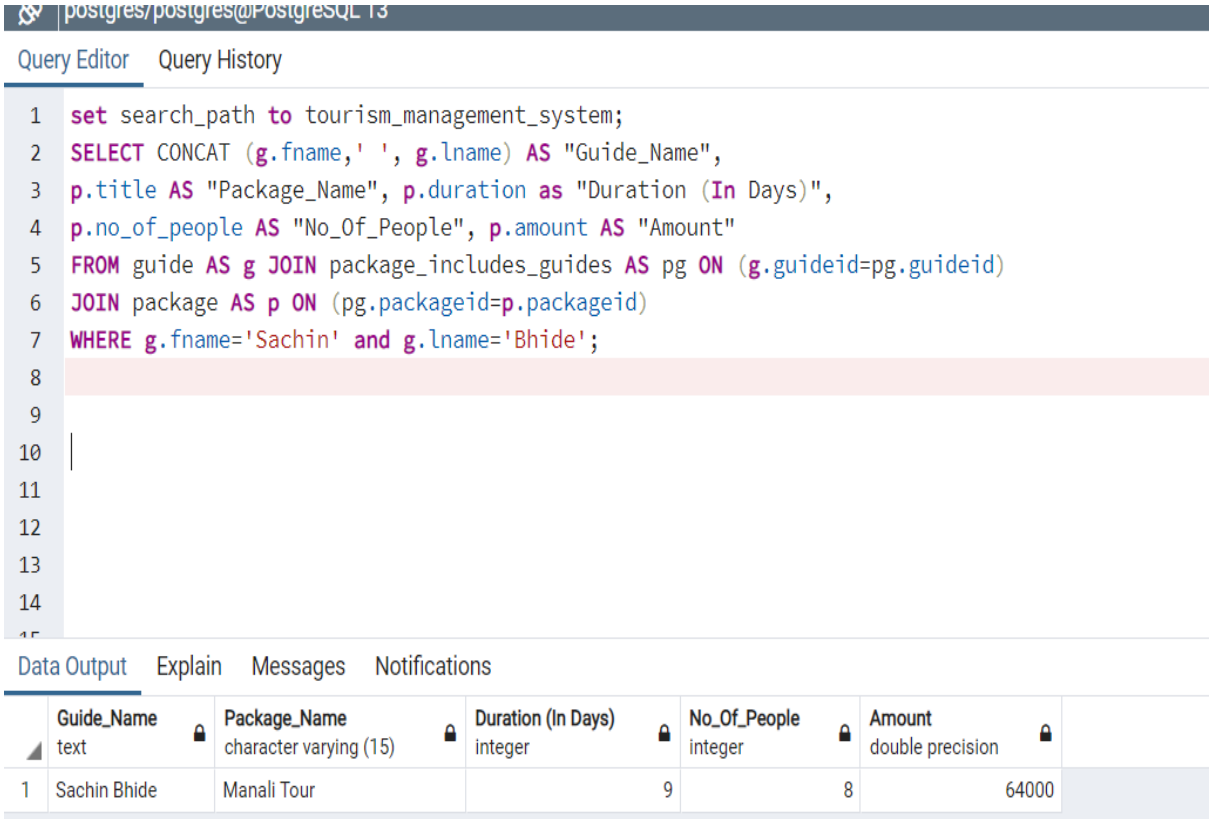
r1 -> ρ (g, guide) $\bowtie_{<g.GuideAadharNo = pg.GuideAadharNo>}$ ρ (pg, guideid) $\bowtie_{<pg.packageid = p.packageid>}$ ρ (p, package)

result -> Π (CONCAT (g.fname,' ', g.lname) -> Guide_Name, p.title -> package_name, p.duration -> Duration(in days), p.no_of_people -> No_of_people, p.amount -> Amount)(σ g.fname='Sachin' and g.lname='Bhide'(r1))

--SQL Query:

```
SELECT CONCAT (g.fname,' ', g.lname) AS "Guide_Name",
p.title AS "Package_Name", p.duration as "Duration (In Days)",
p.no_of_people AS "No_Of_People", p.amount AS "Amount"
FROM guide AS g JOIN package_includes_guides AS pg
ON (g.GuideAadharNo = pg.GuideAadharNo)
JOIN package AS p ON (pg.packageid=p.packageid)
WHERE g.fname='Sachin' and g.lname='Bhide';
```

--Output:



16) Retrieve the list of all package associated with a particular hotel.

--Relational Algebra:

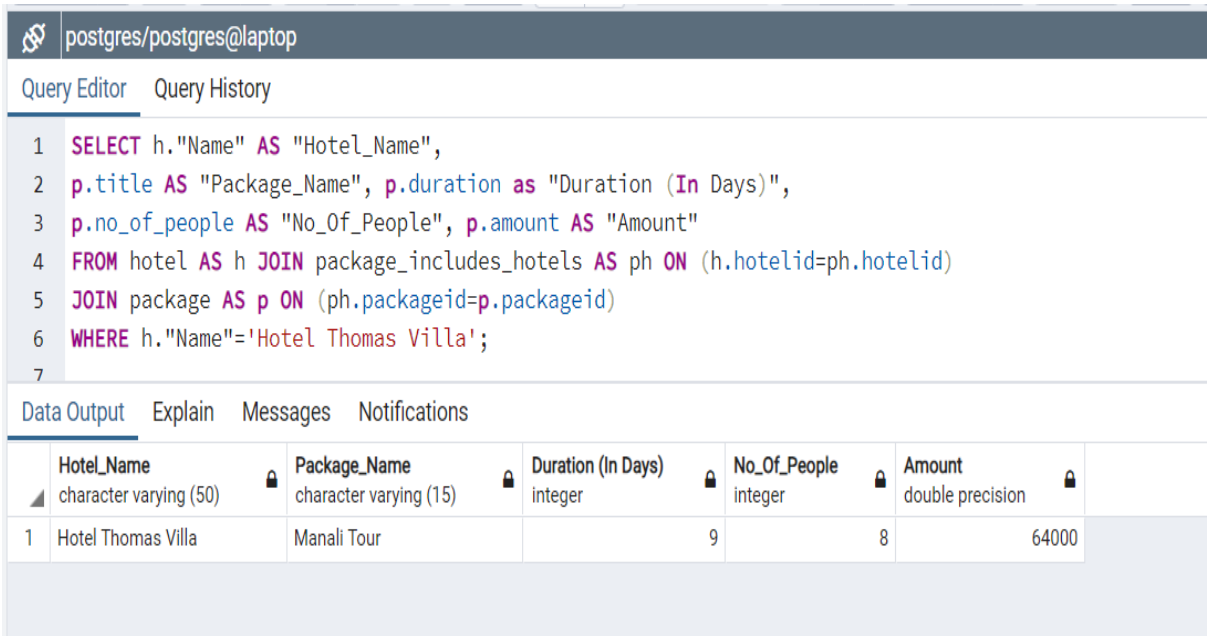
r1 -> ρ (h, hotel) $\bowtie_{<h.hotelid = ph.hotelid>}$ ρ (ph, package_includes_hotels) $\bowtie_{<pg.packageid = p.packageid>}$ ρ (p, package)

result -> Π (h."Name" -> Hotel_Name, p.title -> package_name, p.duration -> Duration(in days), p.no_of_people -> No_of_people, p.amount -> Amount)(σ h."Name"='Hotel Thomas Villa' (r1))

--SQL Query:

```
SELECT h."Name" AS "Hotel_Name",
p.title AS "Package_Name", p.duration as "Duration (In Days)",
p.no_of_people AS "No_Of_People", p.amount AS "Amount"
FROM hotel AS h JOIN package_includes_hotels AS ph ON (h.hotelid=ph.hotelid)
JOIN package AS p ON (ph.packageid=p.packageid)
WHERE h."Name"='Hotel Thomas Villa';
```

--Output:



17) Retrieve all the packages which include "xyz" spots.

--Relational Algebra:

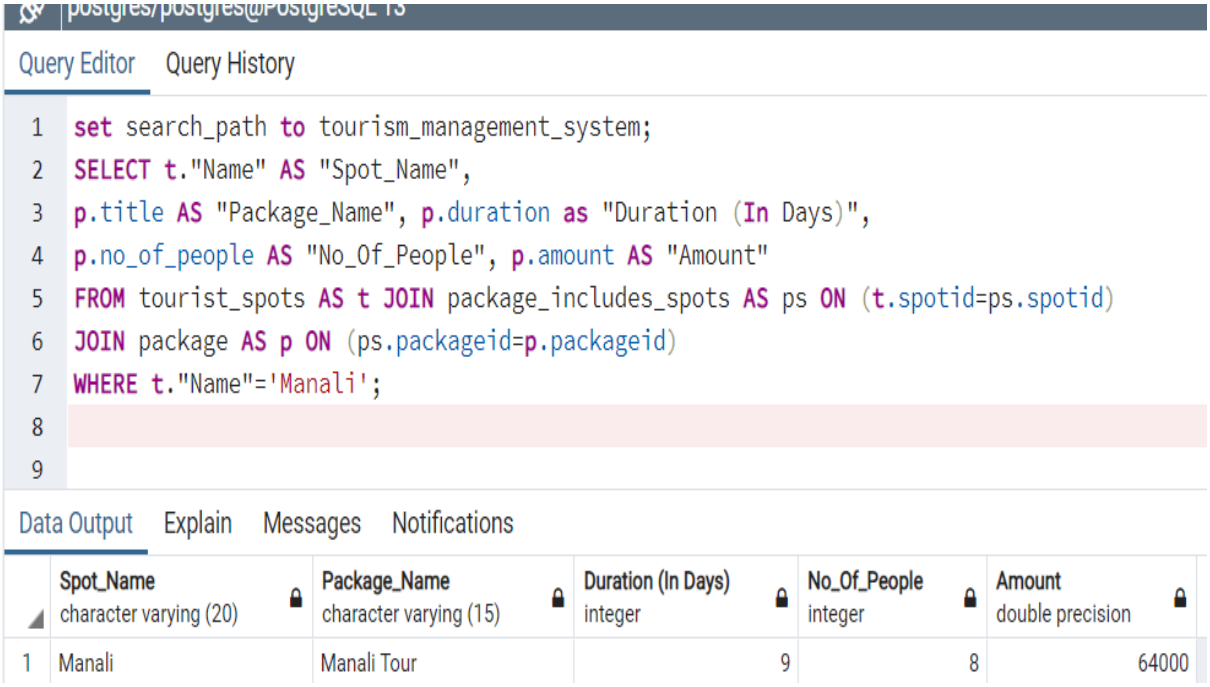
r1 -> ρ (t, tourist_spots) $\bowtie_{<t.spotid = ps.spotid>}$ ρ (ps, package_includes_spots) $\bowtie_{<ps.packageid = p.packageid>}$ ρ (p, package)

result -> $\prod_{(t."Name" \text{ -> Spot\_Name, p.title -> package\_name, p.duration -> Duration(in days), p.no\_of\_people -> No\_of\_people, p.amount -> Amount})}$($\sigma_{t."Name"='Manali'}$ (r1))

--SQL Query:

```
SELECT t."Name" AS "Spot_Name",
p.title AS "Package_Name", p.duration as "Duration (In Days)",
p.no_of_people AS "No_Of_People", p.amount AS "Amount"
FROM tourist_spots AS t JOIN package_includes_spots AS ps ON (t.spotid=ps.spotid)
JOIN package AS p ON (ps.packageid=p.packageid)
WHERE t."Name"='Manali';
```

--Output:



18) Best tourist place to visit in “xyz” season.

--Relational Algebra:

r1 -> ρ (ts, tourist_spots) $\bowtie_{<ts.pincode = l.pincode>}$ ρ (l, location)

result -> Π (t.”Name” -> Spot_Name, ts.season -> Season, ts.ratings -> Ratings,

CONCAT (ts.address,', ',l.city,', ',l.state, ' - ',ts.pincode) ->Address)) (σ ts.season "='Winter' (r1))

--SQL Query:

```
SELECT ts."Name" AS "Spot_Name", ts.season AS "Season", ts.ratings AS "Ratings",
CONCAT (ts.address,', ',l.city,', ',l.state, ' - ',ts.pincode) AS "Address"
FROM tourist_spots AS ts JOIN "location" AS l
ON ts.pincode=l.pincode
WHERE ts.season='Winter';
```

--Output:

Query Editor | Query History

```
1  set search_path to tourism_management_system;
2  SELECT ts."Name" AS "Spot_Name", ts.season AS "Season", ts.ratings AS "Ratings",
3  CONCAT (ts.address,', ',l.city,', ',l.state, ' - ',ts.pincode) AS "Address"
4  FROM tourist_spots AS ts JOIN "location" AS l
5  ON ts.pincode=l.pincode
6  WHERE ts.season='Winter';
7
8
9
```

Data Output | Explain | Messages | Notifications

	Spot_Name character varying (20)	Season character varying (10)	Ratings double precision	Address text
1	Manali	Winter	4.7	Rotang Pass...
2	Srinagar	Winter	4.7	Srinagar,J & ...
3	Dashashwamedh	Winter	4.6	Dashashwa...
4	Somnath	Winter	4.3	Somnath M...

19) Name and address of hotels which provides rooms between specific price range.

--Relational Algebra:

r1 -> ρ (h, hotel) $\bowtie_{<h.pincode=l.pincode>}$ ρ (l, location) $\bowtie_{<h.hotelid= r.hotelid>}$ ρ (r, room)

LEFT $\bowtie_{< h.hotelid=rf.hotelid and r.room\_no=rf.room\_no >}$ ρ (rf, room_facilities)

Result -> $\prod$ (h. "Name"->Hotel_Name,r.room->Room_Number,r.Type->Room_Type,r.beds->No_of_beds,r.capacity->capacity,

r.rate->price,r.status->Current_Status,rf.facility, CONCAT (h.address,', ', l.city,', ', l.state,' - ', h.pincode) -> Address)

(σ (r.rate BETWEEN 1500 and 2000) ORDER BY$_{r.rate,h."name",r.room_no}$ (r1))

--SQL Query:

```
SELECT h."Name" As "Hotel_Name", r.room_no AS "Room_Number", r."Type" AS
"Room_Type",
r.beds AS "No_of_Beds", r.capacity AS "Capacity", r.rate AS "Price", r.status As
"Current_Status",
rf.facility, CONCAT(h.address,', ', l.city,', ', l.state,' - ', h.pincode) AS "Address"
FROM hotel AS h JOIN location AS l ON h.pincode = l.pincode
JOIN room AS r ON h.hotelid = r.hotelid
LEFT JOIN room_facilities AS rf ON (h.hotelid=rf.hotelid and r.room_no=rf.room_no)
WHERE r.rate between 1500 and 2000
ORDER BY r.rate, h."Name", r.room_no;
```

--Output:

Query Editor | Query History | Scratch Pad

```
1  set search_path to tourism_management_system;
2  SELECT h."Name" As "Hotel_Name", r.room_no AS "Room_Number", r."Type" AS "Room_Type",
3  r.beds AS "No_of_Beds", r.capacity AS "Capacity", r.rate AS "Price", r.status As "Current_Status",
4  rf.facility, CONCAT(h.address,', ', l.city,', ', l.state,' - ', h.pincode) AS "Address"
5  FROM hotel AS h JOIN location AS l ON h.pincode = l.pincode
6  JOIN room AS r ON h.hotelid = r.hotelid
7  LEFT JOIN room_facilities AS rf ON (h.hotelid=rf.hotelid and r.room_no=rf.roomno)
8  WHERE r.rate between 1500 and 2000
9  ORDER BY r.rate, h."Name", r.room_no;
10
```

Data Output | Explain | Messages | Notifications

	Hotel_Name character varying (50)	Room_Number numeric (3)	Room_Type character varying (6)	No_of_Beds integer	Capacity integer	Price double precision	Current_Status character varying (15)	facility character varying (50)	Address text
1	Ahdoos	2	HEATER	1	2	1500	AVAILABLE	[null]	Residency r...
2	Ahmedabad	2	AC	1	2	1500	AVAILABLE	[null]	Vastrapur,Ah...
3	Astoria Hotel	2	AC	1	2	1500	AVAILABLE	[null]	J. Tata Road...
4	Aurick Hotel	5	NONAC	1	2	1500	AVAILABLE	[null]	15th Cross, ...
5	BB Palace-A	2	AC	1	2	1500	AVAILABLE	[null]	2638-2642 ...
6	Billberry Hotel	5	NONAC	1	2	1500	AVAILABLE	[null]	Ghat No. 17,...
7	Central Hotel	5	NONAC	1	2	1500	AVAILABLE	[null]	Ashirwad En...
8	Chouki Dhani	2	AC	1	2	1500	AVAILABLE	[null]	Near All Indi...
9	Cosset-Comfort	2	AC	1	2	1500	AVAILABLE	[null]	Mumbai Pun...
10	Hotel Sai	2	AC	1	2	1500	AVAILABLE	[null]	Chandpur In...

20) Retrieve list of all the guides which are not associated with any active packages.

--Relational Algebra:

r1-> $\Pi_{(pg.GuideAadharNo)}$ ($\sigma_{(p.isActive='TRUE')}$ (ρ (pg, package_include_guide) $\bowtie_{<pg.package.id=p.packageid>}$ ρ (p,package)))

r2 -> ρ (g,guide) $\bowtie_{<g.pincode=l.pincode>}$ ρ (l, location)

result-> $\Pi_{(CONCAT(g.fname,' ',g.lname)->Guide\_Name,g.email,g.phone,g.age,g.gender,(g.address,',',l.city,',',l.state,'-',g.pincode)->Address)}$ ($\sigma_{(g.GuideAadharNo\ NOT\ IN\ (r1))}$ (r2))

--SQL Query:

```
SELECT CONCAT (g.fname,' ', g.lname) AS "Guide_Name", g.email, g.phone, g.age, g.gender,
CONCAT (g.address,', ',l.city,', ',l,state,' - ',g.pincode) AS "Address"
FROM guide AS g JOIN "location" AS l ON (g.pincode=l.pincode)
WHERE g.GuideAadharNo NOT IN
(SELECT GuideAadharNo from package_includes_guides AS pg
JOIN (Select * from package where isActive='TRUE') AS p ON (pg.packageid=p.packageid));
```

--Output:

Query Editor Query History

```
1  set search_path to tourism_management_system;
2  SELECT CONCAT (g.fname,' ', g.lname) AS "Guide_Name", g.email, g.phone, g.age, g.gender,
3  CONCAT (g.address,', ',l.city,', ',l,state,' - ',g.pincode) AS "Address"
4  FROM guide AS g JOIN "location" AS l ON (g.pincode=l.pincode)
5  WHERE g.guideid NOT IN
6  (SELECT guideid from package_includes_guides AS pg
7  JOIN (Select * from package where isActive='TRUE') AS p ON (pg.packageid=p.packageid));
8
9
10
```

Data Output Explain Messages Notifications

	Guide_Name text	email character varying (20)	phone numeric (10)	age integer	gender character (1)	Address text
1	Param Singh	psingh@gmail.com	6645789155	28	M	Gill Medical ...
2	Pooran Singh	theps4@gmail.com	9977884455	40	M	Dargah Hom...
3	Shanker Desai	shivd88@gmail.com	7984561534	56	M	Shree Muniv...
4	Karan Thakker	kt14@gmail.com	9988451601	30	M	Abhay Ghat, ...
5	Akshar Patel	akpatel45@gmail.com	7845561255	26	M	Aarogya Van...
6	Ganesh Gaitonde	gg0007@gmail.com	9988990007	35	M	128 ,pragati ...

21) List all the hotels that have room availability from "this date" to "last date" at a particular location.

--Relational Algebra:

r1 -> ρ (b, booking) RIGHT $\bowtie_{<b.bid = bfh.bid>}$ ρ (bfh, booking_for_hotel)

r2 -> $\prod_{(hotelid, room\_no)}$ (σ tripstart_date NOT BETWEEN '2020-12-30' AND '2020-12-31' AND tripend_date NOT BETWEEN '2020-12-30' AND '2020-12-31') AND r.status = 'AVAILABLE' AND l.city = 'Mumbai' GROUP BY h.hotelid (r1))

r3 -> ρ (h, hotel) $\bowtie_{<h.pincode = l.pincode>}$ ρ (l, location) $\bowtie_{<h.hotelid = r.hotelid>}$ ρ (r, room)

r4result -> $\prod_{(h."Name", \mathcal{F}COUNT(r.room\_no) \text{ -> } rooms\_count)}$ (σ (r.hotelid, r.room_no) NOT IN (r2) AND r.status = 'AVAILABLE' AND l.city = 'Mumbai' (r3))

--SQL Query:

```
SELECT h."Name", COUNT(r.room_no) AS rooms_count FROM
hotel AS h JOIN
```

location AS l ON h.pincode = l.pincode JOIN

room AS r ON h.hotelid = r.hotelid WHERE (r.hotelid, r.room_no) NOT IN

(SELECT hotelid, room_no FROM

booking AS b RIGHT JOIN booking_for_hotel AS bfh ON bfh.bid = b.bid

WHERE tripstart_date NOT BETWEEN '2020-12-30' AND '2020-12-31'

AND tripend_date NOT BETWEEN '2020-12-30' AND '2020-12-31')

AND r.status = 'AVAILABLE' AND l.city = 'Mumbai' GROUP BY h.hotelid;

--Output:

```
32
33  SELECT h."Name", COUNT(r.room_no) AS rooms_count FROM
34  hotel AS h JOIN
35  location AS l ON h.pincode = l.pincode JOIN
36  room AS r ON h.hotelid = r.hotelid WHERE (r.hotelid, r.room_no) NOT IN
37  (SELECT hotelid, room_no FROM
38  booking AS b RIGHT JOIN booking_for_hotel AS bfh ON bfh.bid = b.bid
39  WHERE tripstart_date NOT BETWEEN CAST('2020-12-30' AS DATE) AND CAST('2020-12-31' AS DATE)
40  AND tripend_date NOT BETWEEN CAST('2020-12-30' AS DATE) AND CAST('2020-12-31' AS DATE))
41  AND r.status = 'AVAILABLE' AND l.city = 'Mumbai' GROUP  BY h.hotelid;
```

Data Output | Explain | Messages | Notifications

	Name character varying (50)	rooms_count bigint
1	Astoria Hotel	4
2	Central Hotel	5
3	ITC Grand Central, A Luxury Collection Hotel	5

22) Number of rooms available at a particular hotel right now.

--Relational Algebra:

r1 -> ρ (h, hotel) $\bowtie_{\text{<h.hotelid = r.hotelid>}}$ ρ (r, $\mathcal{F}_{\text{COUNT(*)}}$ (room))

result -> $\sigma_{\text{h.”Name” = ‘Ahdoos Hotel’ AND r.status = ‘AVAILABLE’}}$ (r1)

--SQL Query:

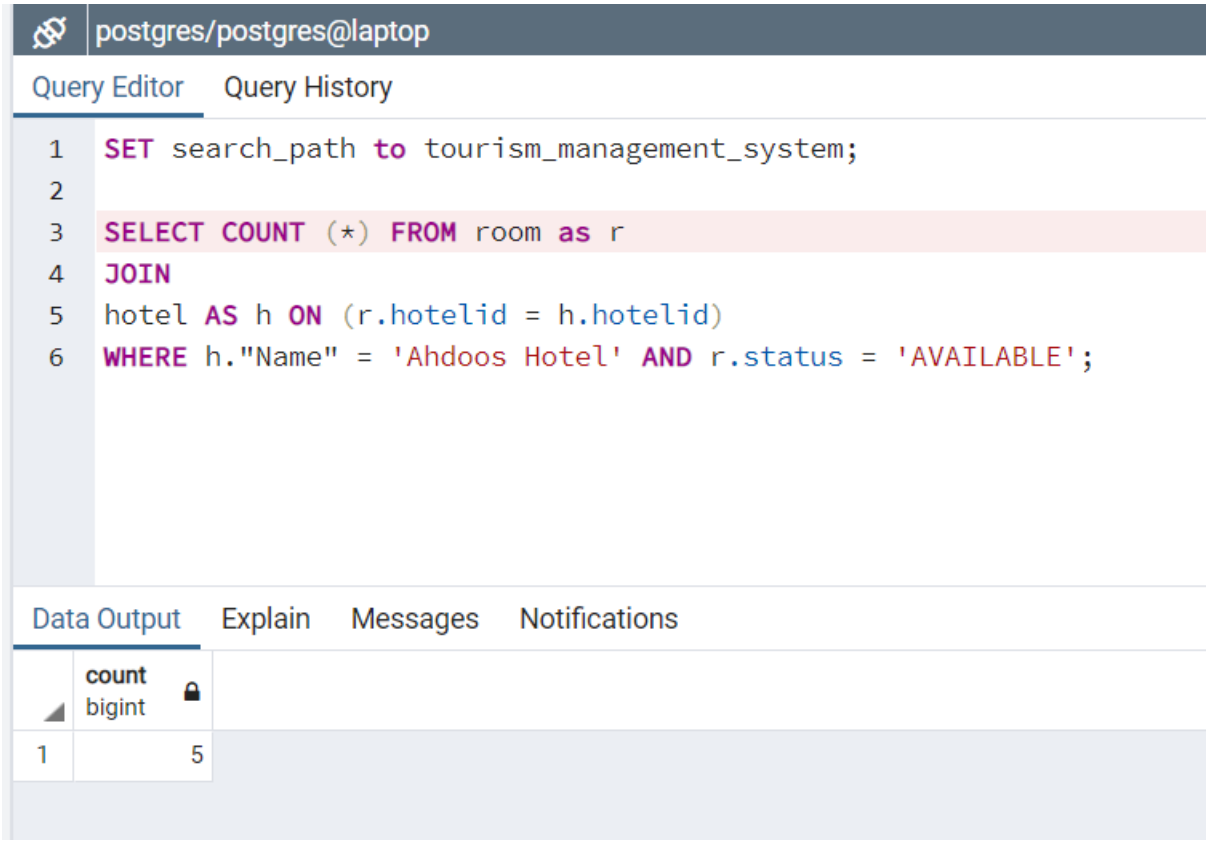
SELECT COUNT (*) FROM room as r

JOIN

hotel AS h ON (r.hotelid = h.hotelid)

WHERE h."Name" = 'Ahdoos Hotel' AND r.status = 'AVAILABLE';

--Output:



23) Retrieve all the previous bookings of user. (both)

--Relational Algebra:

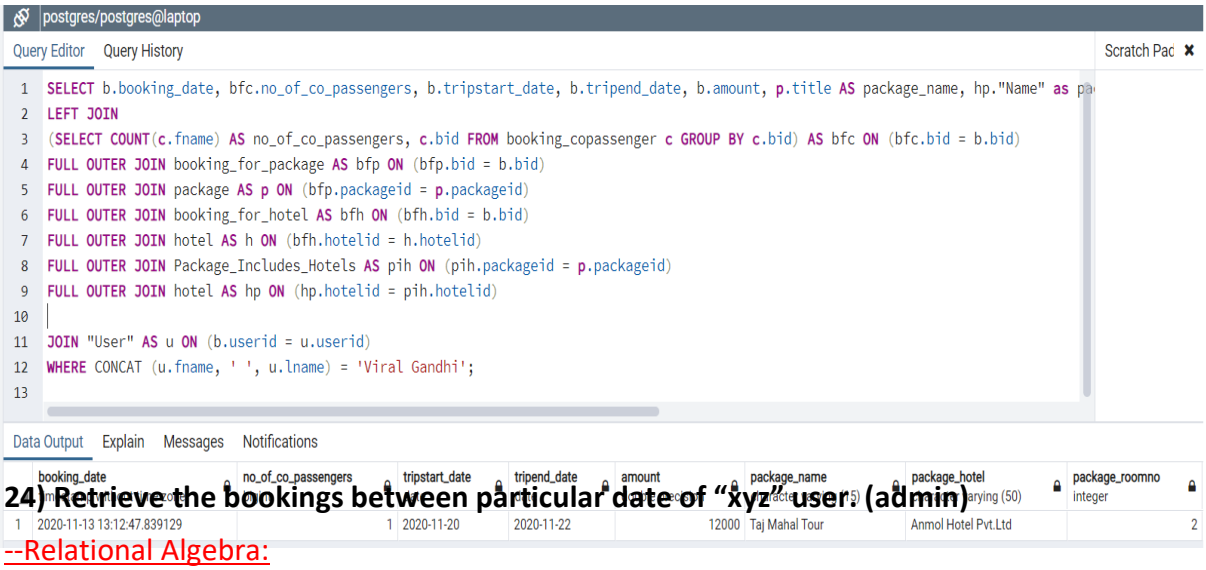
r1 -> ρ (b, booking) LEFT $\bowtie_{<b.bid = bfc.bid>}$ ρ (bfc, bid $\mathcal{F}_{COUNT(CoPassID) -> no\_of\_co\_passengers}$ (booking_copassenger)) FULL OUTER $\bowtie_{<b.bid = bfp.bid>}$ ρ (bfp, booking_for_package) FULL OUTER $\bowtie_{<bfp.packageid = p.packageid>}$ ρ (p, package) FULL OUTER $\bowtie_{<b.bid = bfh.bid>}$ ρ (bfh, booking_for_hotel) FULL OUTER $\bowtie_{<bfh.hotelid = h.hotelid>}$ ρ (h, hotel) FULL OUTER $\bowtie_{<h.hotelid = pih.hotelid>}$ ρ (pih, package_includes_hotels) FULL OUTER $\bowtie_{<pih.hotelid = hp.hotelid>}$ ρ (hp, hotel) $\bowtie_{<b.UserAadharNo = u.UserAadharNo>}$ ρ (u,"User")

result -> $\Pi_{(b.booking\_date, bfc.no\_of\_co\_passengers, b.tripstart\_date, b.tripend\_date, b.amount, p.title -> package\_name, hp."Name" -> package\_hotel, pih.room\_no -> package\_roomno, h."Name" -> hotel\_name, bfh.room\_no)}$ (σ CONCAT (u.fname,' ',u.lname) = 'Viral Gandhi' (r1))

--SQL Query:

```
SELECT b.booking_date, bfc.no_of_co_passengers, b.tripstart_date, b.tripend_date,
b.amount, p.title AS package_name, hp."Name" as package_hotel, pih.room_no
package_roomno,h."Name" AS hotel_name, bfh.room_no  FROM booking as b
LEFT JOIN
(SELECT COUNT(c.CoPassID) AS no_of_co_passengers, c.bid FROM booking_copassenger c
GROUP BY c.bid) AS bfc ON (bfc.bid = b.bid)
FULL OUTER JOIN booking_for_package AS bfp ON (bfp.bid = b.bid)
FULL OUTER JOIN package AS p ON (bfp.packageid = p.packageid)
FULL OUTER JOIN booking_for_hotel AS bfh ON (bfh.bid = b.bid)
FULL OUTER JOIN hotel AS h ON (bfh.hotelid = h.hotelid)
FULL OUTER JOIN Package_Includes_Hotels AS pih ON (pih.packageid = p.packageid)
FULL OUTER JOIN hotel AS hp ON (hp.hotelid = pih.hotelid)
JOIN "User" AS u ON (b.UserAadharNo = u.UserAadharNo)
WHERE CONCAT (u.fname, ' ', u.lname) = 'Viral Gandhi';
```

--Output:



24) Retrieve the bookings between particular date of "xyz" user. (admin)

--Relational Algebra:

r1 -> ρ (b, booking) LEFT $\bowtie_{<b.bid = bfc.bid>}$ ρ (bfc, bid $\mathcal{F}_{COUNT(CoPassID) -> no\_of\_co\_passengers}$ (booking_copassenger)) FULL OUTER $\bowtie_{<b.bid = bfp.bid>}$ ρ (bfp, booking_for_package) FULL OUTER $\bowtie_{<bfp.packageid = p.packageid>}$ ρ (p, package) FULL OUTER $\bowtie_{<b.bid = bfh.bid>}$ ρ (bfh, booking_for_hotel) FULL OUTER $\bowtie_{<bfh.hotelid = h.hotelid>}$ ρ (h, hotel) FULL OUTER $\bowtie_{<h.hotelid = pih.hotelid>}$ ρ (pih, package_includes_hotels) FULL OUTER $\bowtie_{<pih.hotelid = hp.hotelid>}$ ρ (hp, hotel) $\bowtie_{<b.UserAadharNo = u.UserAadharNo>}$ ρ (u,"User")

result -> ∏ (b.booking_date, bfc.no_of_co_passengers, b.tripstart_date, b.tripend_date, b.amount, p.title -> package_name, hp.”Name” -> package_hotel, pih.RoomNo -> package_roomno, h.”Name” -> hotel_name, bfh.roomno) (σ CONCAT (u.fname,’ ‘,u.lname) = ‘Viral Gandhi’ and b.booking_date < '2020-11-30' AND b.booking_date >= '2020-11-20' (r1))

--SQL Query:

```
SELECT CONCAT (u.fname,' ',u.lname), b.booking_date, bfc.no_copassengers,
b.tripstart_date, b.tripend_date,
b.amount, p.title AS package_name,hp."Name" as package_hotel, pih.room_no
package_roomno, h."Name" AS hotel_name, bfh.room_no
FROM booking as b
LEFT JOIN
(SELECT COUNT(c.CoPassID) AS no_copassengers, c.bid FROM booking_copassenger c
GROUP BY c.bid) AS bfc
ON (bfc.bid = b.bid)
FULL OUTER JOIN
booking_for_package AS bfp ON (bfp.bid = b.bid)
FULL OUTER JOIN package AS p ON (bfp.packageid = p.packageid)
FULL OUTER JOIN
booking_for_hotel AS bfh ON (bfh.bid = b.bid)
FULL OUTER JOIN hotel AS h ON (bfh.hotelid = h.hotelid)
FULL OUTER JOIN Package_Includes_Hotels AS pih ON (pih.packageid = p.packageid)
FULL OUTER JOIN hotel AS hp ON (hp.hotelid = pih.hotelid)
JOIN
"User" AS u ON (b.UserAadharNo = u.UserAadharNo) WHERE CONCAT(u.fname, ' ', u.lname)
= 'Viral Gandhi'
AND b.booking_date < '2020-11-30' AND b.booking_date >= '2020-11-20';
```

--Output:

postgres/postgres@laptop

Query Editor | Query History | Scratch Pad

```
SELECT CONCAT(u.fname,' ',u.lname), b.booking_date, bfc.no_copassengers, b.tripstart_date, b.tripend_date,
b.amount, p.title AS package_name,hp."Name" as package_hotel, pih.RoomNo package_roomno, h."Name" AS hotel_name, bfh.roomno
FROM booking as b
LEFT JOIN
(SELECT COUNT(c.fname) AS no_copassengers, c.bid FROM booking_copassenger c GROUP BY c.bid) AS bfc
ON (bfc.bid = b.bid)
FULL OUTER JOIN
booking_for_package AS bfp ON (bfp.bid = b.bid)
FULL OUTER JOIN package AS p ON (bfp.packageid = p.packageid)
FULL OUTER JOIN
booking_for_hotel AS bfh ON (bfh.bid = b.bid)
FULL OUTER JOIN hotel AS h ON (bfh.hotelid = h.hotelid)
FULL OUTER JOIN Package_Includes_Hotels AS pih ON (pih.packageid = p.packageid)
FULL OUTER JOIN hotel AS hp ON (hp.hotelid = pih.hotelid)
JOIN
"User" AS u ON (b.userid = u.userid) WHERE CONCAT(u.fname, ' ', u.lname) = 'Viral Gandhi'
AND b.booking_date < '2020-11-14' AND b.booking_date >= '2020-11-09';
```

Data Output | Explain | Messages | Notifications

	concat text	booking_date timestamp without time zone	no_copassengers bigint	tripstart_date date	tripend_date date	amount double precision	package_name character varying (15)	package_hotel character varying (50)	package_roomno integer
1	Viral Gandhi	2020-11-13 13:12:47.839129	1	2020-11-20	2020-11-22	12000	Taj Mahal Tour	Anmol Hotel Pvt.Ltd	2

25) Retrieve all the bookings between particular date. (admin)

--Relational Algebra:

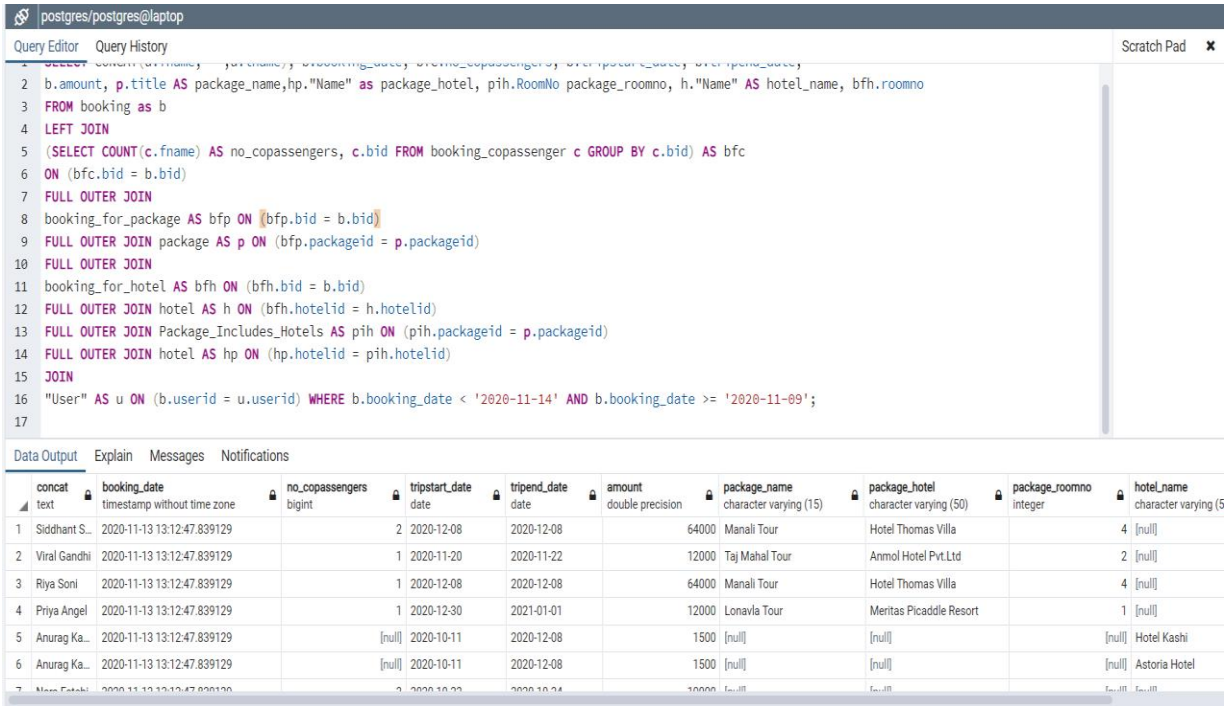
r1 -> ρ (b, booking) LEFT $\bowtie_{<b.bid = bfc.bid>}$ ρ (bfc, bid $\mathcal{F}$ COUNT(CoPassID) -> no_of_co_passengers (booking_copassenger)) FULL OUTER $\bowtie_{<b.bid = bfp.bid>}$ ρ (bfp, booking_for_package) FULL OUTER $\bowtie_{<bfp.packageid = p.packageid>}$ ρ (p, package) FULL OUTER $\bowtie_{<b.bid = bfh.bid>}$ ρ (bfh, booking_for_hotel) FULL OUTER $\bowtie_{<bfh.hotelid = h.hotelid>}$ ρ (h, hotel) FULL OUTER $\bowtie_{<h.hotelid = pih.hotelid>}$ ρ (pih, package_includes_hotels) FULL OUTER $\bowtie_{<pih.hotelid = hp.hotelid>}$ ρ (hp, hotel) $\bowtie_{<b.UserAadharNo = u.UserAadharNo>}$ ρ (u,"User")

result -> Π (b.booking_date, bfc.no_of_co_passengers, b.tripstart_date, b.tripend_date, b.amount, p.title -> package_name, hp."Name" -> package_hotel, pih.room_no -> package_roomno, h."Name" -> hotel_name, bfh.room_no) (σ b.booking_date < '2020-11-28' AND b.booking_date >= '2020-11-26' (r1))

--SQL Query:

```
SELECT CONCAT(u.fname,' ',u.lname), b.booking_date, bfc.no_copassengers,
b.tripstart_date, b.tripend_date,

b.amount, p.title AS package_name,hp."Name" as package_hotel, pih.room_no
package_roomno, h."Name" AS hotel_name, bfh.room_no

FROM booking as b

LEFT JOIN

(SELECT COUNT(c.CoPassID) AS no_copassengers, c.bid FROM booking_copassenger c
GROUP BY c.bid) AS bfc

ON (bfc.bid = b.bid)

FULL OUTER JOIN

booking_for_package AS bfp ON (bfp.bid = b.bid)

FULL OUTER JOIN package AS p ON (bfp.packageid = p.packageid)

FULL OUTER JOIN

booking_for_hotel AS bfh ON (bfh.bid = b.bid)

FULL OUTER JOIN hotel AS h ON (bfh.hotelid = h.hotelid)

FULL OUTER JOIN Package_Includes_Hotels AS pih ON (pih.packageid = p.packageid)

FULL OUTER JOIN hotel AS hp ON (hp.hotelid = pih.hotelid)

JOIN

"User" AS u ON (b.UserAadharNo = u.UserAadharNo) WHERE b.booking_date < '2020-11-28'
AND b.booking_date >= '2020-11-26';
```

--Output:



26) Retrieve all the details of user of “xyz” hotel room. (admin)

--Relational Algebra:

r1 -> ρ (r,room) $\bowtie_{< r.hotelid = h.hotelid >} \rho$ (h,hotel)

r2-> ($\sigma_{h."Name" = 'Hotel Thomas' and r.room\_no = 5}$ (r1))

r3-> ρ (bfh,booking_for_hotel) $\bowtie_{<r2.hotelid = bfh.hotelid >}$ (r2)

r4-> ρ (b,booking) $\bowtie_{<r3.bid = b.bid >}$ (r3)

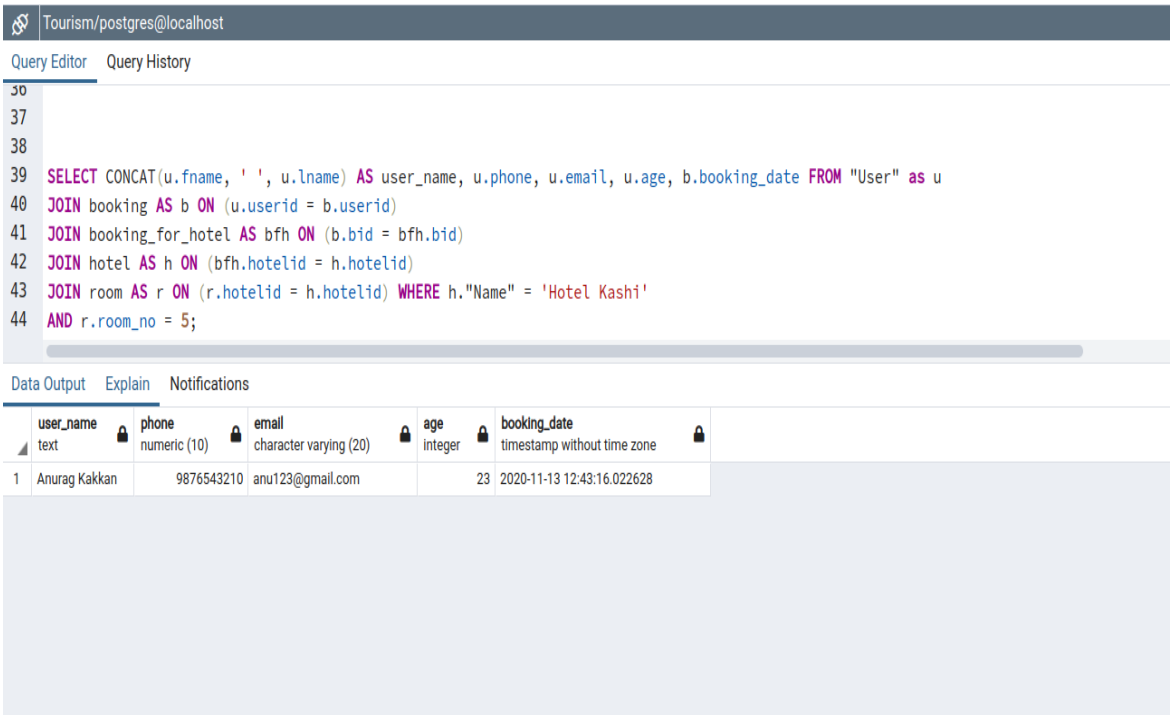
r5-> ρ (u,User) $\bowtie_{< u.UserAadharNo = r4.UserAadharNo >}$ (r4)

result-> $\Pi_{(CONCAT(u.fname, ' ', u.lname)->user\_name,\ u.phone, u.email, u.age, b.booking\_date)}$ (r5)

--SQL Query:

```
SELECT CONCAT(u.fname, ' ', u.lname) AS user_name, u.phone, u.email, u.age,
b.booking_date FROM "User" as u
JOIN booking AS b ON (u.UserAadharNo = b.UserAadharNo)
JOIN booking_for_hotel AS bfh ON (b.bid = bfh.bid)
JOIN hotel AS h ON (bfh.hotelid = h.hotelid)
JOIN room AS r ON (r.hotelid = h.hotelid) WHERE h."Name" = 'Hotel Kashi'
AND r.room_no = 5;
```

--Output:



27) Retrieve all the user booking details of “xyz” hotel. (admin)

--Relational Algebra:

r1-> ρ (bfh, booking_for_hotel) $\bowtie_{<\text{bfh.hotelid = h.hotelid}>} \rho$ (h,hotel)

r2-> $\sigma_{\text{h."Name" = 'Hotel Kashi'}}$(r1)

r3-> ρ (b,booking) $\bowtie_{<\text{b.bid=r2.bid}>}$(r2)

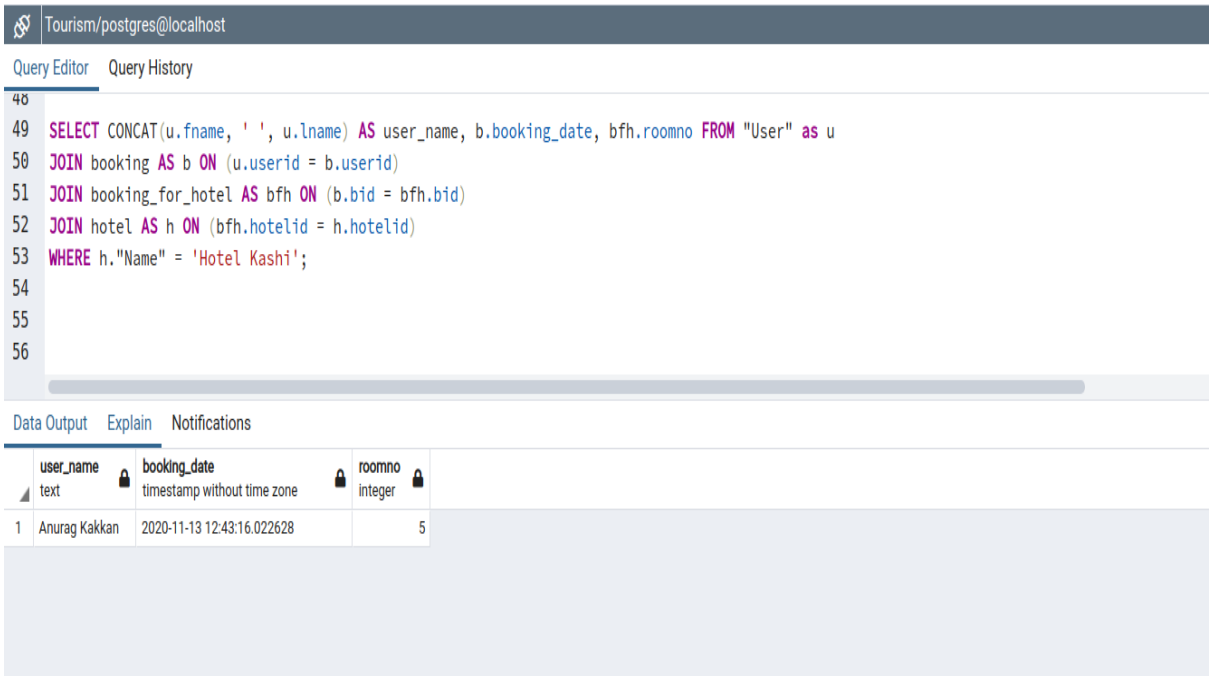
r4-> ρ (u,user) $\bowtie_{<\text{u.UserAadharNo=r3.UserAadharNo}>}$(r3)

$\prod_{\text{(CONCAT(u.fname, ' ', u.lname)->user_name, b.booking_date, bfh.room_no)}}$(r4)

--SQL Query:

```
SELECT CONCAT(u.fname, ' ', u.lname) AS user_name, b.booking_date, bfh.room_no FROM
"User" as u
JOIN booking AS b ON (u.UserAadharNo = b.UserAadharNo)
JOIN booking_for_hotel AS bfh ON (b.bid = bfh.bid)
JOIN hotel AS h ON (bfh.hotelid = h.hotelid)
WHERE h."Name" = 'Hotel Kashi';
```

--Output:



28) Give details of co-passenger with “xyz” user with dates. (admin)

--Relational Algebra:

r1-> ρ (bc, booking_copassenger) $\bowtie_{<\text{bc.UserAadharNo = c.UserAadharNo AND bc.CoPassID = c.CoPassID}>}$

ρ (c,copassanger)

r2-> $\sigma_{\text{CONCAT(u.fname, ' ', u.lname) = 'Viral Gandhi'}}(r1)$

r3-> ρ (b,booking) $\bowtie_{<b.bid=r2.bid>}(r2)$

r4-> ρ (u,user) $\bowtie_{<u.UserAadharNo=r3.UserAadharNo>}(r3)$

$\prod_{(\text{ CONCAT(u.fname, ' ', u.lname) AS user_name, b.booking_date,}}$

$\text{CONCAT(c.fname, ' ', c.lname) as copassenger, c.phone, c.gender, c.age)}(r4)$

--SQL Query:

```
SELECT CONCAT(u.fname, ' ', u.lname) AS user_name, b.booking_date,
CONCAT(c.fname, ' ', c.lname) as copassenger, c.phone, c.gender, c.age
FROM "User" as u
JOIN booking AS b ON (u.UserAadharNo = b.UserAadharNo)
JOIN booking_copassenger AS bc ON (bc.bid = b.bid)
JOIN copassenger AS c ON (bc.UserAadharNo = c.UserAadharNo AND bc.CoPassID =
c.CoPassID)
WHERE CONCAT(u.fname, ' ', u.lname) = 'Viral Gandhi';
```

--Output:

postgres/postgres@PostgreSQL 13

Query Editor | Query History

```
1  set search_path to tourism_management_system;
2  SELECT CONCAT(u.fname, ' ', u.lname) AS user_name, b.booking_date,
3  CONCAT(c.fname, ' ', c.lname) as copassenger, c.phone, c.gender, c.age
4  FROM "User" as u
5  JOIN booking AS b ON (u.userid = b.userid)
6  JOIN booking_copassenger AS bc ON (bc.bid = b.bid)
7  JOIN copassenger AS c ON (bc.userid = c.uid AND bc.fname = c.fname AND bc.lname = c.lname)
8  WHERE CONCAT(u.fname, ' ', u.lname) = 'Viral Gandhi';
9
10
11
```

Data Output | Explain | Messages | Notifications

	user_name text	booking_date timestamp without time zone	copassenger text	phone numeric (12)	gender character (1)	age integer
1	Viral Gandhi	2020-10-16 10:52:52.192482	Raju Japani	6765462659	M	26