



```
171 -- Insert new appointment
172 • INSERT INTO Appointments (appointment_id, patient_id, doctor_id, appointment_date, status)
173   VALUES (11, 11, 11, CURDATE(), 'Scheduled');
174
175 • SELECT * FROM Appointments;
176
```

Result Grid		Filter Rows:		Edit:		Export/Import:		Wrap Cell Content:	
	appointment_id	patient_id	doctor_id	appointment_date	status				
▶	1	1	1	2024-03-01	Completed				
	2	2	2	2024-03-02	Scheduled				
	3	3	3	2024-02-20	Completed				
	4	4	4	2024-02-15	Cancelled				
	5	5	5	2024-01-10	Completed				
	6	6	6	2024-03-03	Scheduled				
	7	7	7	2024-02-01	Completed				
	8	8	8	2024-01-20	Completed				
	9	9	9	2024-03-05	Scheduled				
	10	10	10	2024-02-10	Completed				
	11	11	11	2026-02-25	Scheduled				
*	NULL	NULL	NULL	NULL	NULL				

```
177 • UPDATE Patients  
178     SET address = 'Maninagar, Ahmedabad'  
179     WHERE patient_id = 1;  
180 • SELECT * FROM Patients;
```

```
191     # 2.----- WHERE,HAVING,LIMIT -----  
192  
193 •  SELECT * FROM Patients WHERE registration date >= DATE SUB(CURDATE(), INTERVAL 1 YEAR);
```

Result Grid				Filter Rows:		Edit:		Export/Import:		Wrap Cell Content:	
	patient_id	name	dob	gender	phone_number	email	address	registration_date			
▶	13	Rohit Kumar	1999-04-15	Male	9898989898	rohit@gmail.com	Ahmedabad	2026-02-25			
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL			

```
195 # HAVING
196 • SELECT patient_id, SUM(amount) AS total_paid FROM Billing GROUP BY patient_id ORDER BY total_paid DESC LIMIT 5;
---
```

Result Grid		Filter Rows:	Export:	Wrap Cell Content:	Fetch rows:	
	patient_id	total_paid				
▶	2	1500.00				
▶	10	1400.00				
▶	7	1300.00				
▶	1	1200.00				
▶	4	1100.00				

```
198 # LIMIT
```

```
199 • SELECT * FROM Doctors WHERE consultation_fee > 1000;
---
```

Result Grid							Filter Rows:	Edit:	Export/Import:	Wrap Cell Content:	
	doctor_id	name	specialization	phone_number	email	consultation_fee	experience_years				
▶	1	Dr. Mehul	Cardiology	9000000001	mehul@hosp.com	1200.00	12				
▶	2	Dr. Rakesh	Neurology	9000000002	rakesh@hosp.com	1500.00	10				
▶	4	Dr. Kunal	Orthopedic	9000000004	kunal@hosp.com	1100.00	14				
▶	7	Dr. Alok	Cardiology	9000000007	alok@hosp.com	1300.00	16				
▶	10	Dr. Neeraj	Neurology	9000000010	neeraj@hosp.com	1400.00	11				
▶	11	Dr. Suresh	Cardiology	9000000011	suresh@hospital.com	1800.00	9				
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL				

```
201 # 3.----- AND,OR,NOT -----
```

```
202
```

```
203 # AND
```

```
204 • SELECT * FROM Appointments WHERE status='Scheduled' AND doctor_id=3;
---
```

Result Grid					Filter Rows:	Edit:	Export/Import:	Wrap Cell Content:		
	appointment_id	patient_id	doctor_id	appointment_date	status					
*	NULL	NULL	NULL	NULL	NULL					

```
206 # OR
```

```
207 • SELECT * FROM Doctors WHERE specialization='Cardiology' OR specialization='Neurology';
---
```

Result Grid							Filter Rows:	Edit:	Export/Import:	Wrap Cell Content:	
	doctor_id	name	specialization	phone_number	email	consultation_fee	experience_years				
▶	1	Dr. Mehul	Cardiology	9000000001	mehul@hosp.com	1200.00	12				
▶	2	Dr. Rakesh	Neurology	9000000002	rakesh@hosp.com	1500.00	10				
▶	7	Dr. Alok	Cardiology	9000000007	alok@hosp.com	1300.00	16				
▶	10	Dr. Neeraj	Neurology	9000000010	neeraj@hosp.com	1400.00	11				
▶	11	Dr. Suresh	Cardiology	9000000011	suresh@hospital.com	1800.00	9				
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL				

```

209 # NOT
210 • SELECT * FROM Patients WHERE patient_id NOT IN (SELECT patient_id FROM Appointments WHERE appointment_date >= DATE_S
...

```

patient_id	name	dob	gender	phone_number	email	address	registration_date
1	Amit Shah	1998-02-10	Male	9876543210	amit@gmail.com	Maninagar, Ahmedabad	2024-01-10
2	Neha Patel	1999-05-12	Female	9876543211	neha@gmail.com	Surat	2024-02-15
3	Rahul Mehta	1997-03-21	Male	9876543212	rahul@gmail.com	Vadodara	2024-03-10
4	Priya Desai	2000-07-01	Female	9876543213	priya@gmail.com	Rajkot	2024-01-25
5	Karan Joshi	1995-08-11	Male	9876543214	karan@gmail.com	Bhavnagar	2023-12-20
6	Riya Sharma	2001-11-23	Female	9876543215	riya@gmail.com	Delhi	2024-02-05
7	Arjun Verma	1996-06-19	Male	9876543216	arjun@gmail.com	Jaipur	2023-11-15
8	Sneha Rao	1999-09-09	Female	9876543217	sneha@gmail.com	Mumbai	2024-03-01
9	Vikram Singh	1994-04-18	Male	9876543218	vikram@gmail.com	Indore	2024-01-05
10	Anjali Gupta	2002-12-02	Female	9876543219	anjali@gmail.com	Pune	2024-02-20
12	Test User	2000-01-01	Male	9999999999	test@gmail.com	Test City	2024-03-01
13	Rohit Kumar	1999-04-15	Male	9898989898	rohit@gmail.com	Ahmedabad	2026-02-25
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL

```
# 4.----- ORDER BY , GROUP BY -----
```

```
213
```

```
214 • SELECT * FROM Doctors ORDER BY specialization;
```

doctor_id	name	specialization	phone_number	email	consultation_fee	experience_years
1	Dr. Mehul	Cardiology	9000000001	mehul@hosp.com	1200.00	12
7	Dr. Alok	Cardiology	9000000007	alok@hosp.com	1300.00	16
11	Dr. Suresh	Cardiology	9000000011	suresh@hospital.com	1800.00	9
3	Dr. Nisha	Dermatology	9000000003	nisha@hosp.com	800.00	6
6	Dr. Sanjay	ENT	9000000006	sanjay@hosp.com	600.00	5
9	Dr. Aman	General	9000000009	aman@hosp.com	500.00	3
8	Dr. Pooja	Gynecology	9000000008	pooja@hosp.com	900.00	7
2	Dr. Rakesh	Neurology	9000000002	rakesh@hosp.com	1500.00	10
10	Dr. Neeraj	Neurology	9000000010	neeraj@hosp.com	1400.00	11
4	Dr. Kunal	Orthopedic	9000000004	kunal@hosp.com	1100.00	14
5	Dr. Rina	Pediatrics	9000000005	rina@hosp.com	700.00	8
*	NULL	NULL	NULL	NULL	NULL	NULL

```
215 • SELECT doctor_id, COUNT(*) AS total_patients FROM Appointments GROUP BY doctor_id;
```

```
216 • SELECT d.department_name, SUM(b.amount) FROM Billing b
```

doctor_id	total_patients
1	1
2	1
3	1
4	1
5	1
6	1
7	1
8	1
9	1
10	1
11	1

```

218      # GROUP BY
219 •   SELECT d.department_name, SUM(b.amount) FROM Billing b
220      JOIN Appointments a ON b.appointment_id=a.appointment_id
221      JOIN Doctor_Department dd ON a.doctor_id=dd.doctor_id
222      JOIN Departments d ON dd.department_id=d.department_id
223      GROUP BY d.department_name;

```

Result Grid | Filter Rows: Export: Wrap Cell Content:

	department_name	SUM(b.amount)
▶	Cardiology	2500.00
	Neurology	2900.00
	Dermatology	800.00
	Orthopedic	1100.00
	Pediatrics	700.00
	ENT	600.00
	Gynecology	900.00
	General	500.00

```
225      # 5.----- AGGREGATE FUNCTIONS -----
```

```
226
```

```
227 •   SELECT SUM(amount) AS total_revenue FROM Billing WHERE payment_status='Paid';
```

Result Grid | Filter Rows: Export: Wrap Cell Content:

	total_revenue
▶	6300.00

```
!28 •   SELECT doctor_id, COUNT(*) AS visits FROM Appointments GROUP BY doctor_id ORDER BY visits DESC LIMIT 1;
```

Result Grid | Filter Rows: Export: Wrap Cell Content: Fetch rows:

doctor_id	visits
1	1

```
229 •   SELECT AVG(consultation_fee) FROM Doctors;
```

Result Grid | Filter Rows: Export: Wrap Cell Content:

AVG(consultation_fee)
1072.727273

```

1      # 7.----- JOINS -----
2
3      # ----INNER JOIN----
4 •  SELECT d.doctor_id, d.name AS doctor_name, dep.department_name
5      FROM Doctors d
6      INNER JOIN Doctor_Department dd ON d.doctor_id = dd.doctor_id
7      INNER JOIN Departments dep ON dd.department_id = dep.department_id;
8

```

Ult Grid | Filter Rows:  Export: Wrap Cell Content:

doctor_id	doctor_name	department_name
1	Dr. Mehul	Cardiology
7	Dr. Alok	Cardiology
2	Dr. Rakesh	Neurology
10	Dr. Neeraj	Neurology
3	Dr. Nisha	Dermatology
4	Dr. Kunal	Orthopedic
5	Dr. Rina	Pediatrics
6	Dr. Sanjay	ENT
8	Dr. Pooja	Gynecology
9	Dr. Aman	General

```

239      # -----LEFT JOIN-----
240 •  SELECT p.patient_id, p.name, a.appointment_date, a.status
241      FROM Patients p
242      LEFT JOIN Appointments a
243      ON p.patient_id = a.patient_id
244      WHERE a.status = 'Completed';

```

Result Grid | Filter Rows:  Export: Wrap Cell Content:

	patient_id	name	appointment_date	status
▶	1	Amit Shah	2024-03-01	Completed
▶	3	Rahul Mehta	2024-02-20	Completed
▶	5	Karan Joshi	2024-01-10	Completed
▶	7	Arjun Verma	2024-02-01	Completed
▶	8	Sneha Rao	2024-01-20	Completed
▶	10	Anjali Gupta	2024-02-10	Completed

```

246      # ----- RIGHT JOIN-----
247 •   SELECT a.appointment_id, a.appointment_date, b.payment_status
248     FROM Billing b
249     RIGHT JOIN Appointments a
250    ON b.appointment_id = a.appointment_id
251   WHERE b.appointment_id IS NULL;

```

Result Grid			
	appointment_id	appointment_date	payment_status
▶	11	2026-02-25	NULL

```

253      #-----FULL OUTER JOIN-----
254
255 •   SELECT p.patient_id, p.name
256     FROM Patients p
257     LEFT JOIN Appointments a
258    ON p.patient_id = a.patient_id
259   WHERE a.appointment_id IS NULL;

```

Result Grid		
	patient_id	name
▶	12	Test User
	13	Rohit Kumar

```

---+
266      # 8.----- SUBQUERIES -----
267 •   SELECT d.doctor_id, d.name AS doctor_name, COUNT(a.appointment_id) AS total_appointments
268     FROM Appointments a
269     JOIN Doctors d
270    ON a.doctor_id = d.doctor_id
271    GROUP BY d.doctor_id, d.name
272   HAVING COUNT(a.appointment_id) > 5;

```

Result Grid			
	doctor_id	doctor_name	total_appointments
▶	1	Dr. Mehul	6

```

273
e 274 •   SELECT patient_id FROM Billing GROUP BY patient_id ORDER BY SUM(amount) DESC LIMIT 1;
e

```

Result Grid		
	patient_id	
▶	2	

```
275 • SELECT * FROM Appointments WHERE doctor_id IN (SELECT doctor_id FROM Doctors WHERE specialization='Dermatology');
```

Result Grid				
appointment_id	patient_id	doctor_id	appointment_date	status
3	3	3	2024-02-20	Completed
NULL	NULL	NULL	NULL	NULL

Result Grid

```
277 # 9.----- DATE FUNCTIONS -----
```

```
278
```

```
279 • SELECT MONTH(appointment_date), COUNT(*) FROM Appointments GROUP BY MONTH(appointment_date);
```

```
280
```

Result Grid	
MONTH(appointment_date)	COUNT(*)
3	9
2	5
1	2

```
---
```

```
281 • SELECT DATEDIFF('2024-03-10','2024-03-01') AS stay_days;
```

```
282
```

Result Grid	
stay_days	
9	

```
283 • SELECT DATE_FORMAT(treatment_date,'%d-%m-%Y') FROM Medical_Records;
```

```
284
```

Result Grid	
DATE_FORMAT(treatment_date,'%d-%m-%Y')	
01-03-2024	
02-03-2024	
20-02-2024	
15-02-2024	
10-01-2024	
03-03-2024	
01-02-2024	
20-01-2024	
05-03-2024	
10-02-2024	

```
285      # 10.----- STRING FUNCTIONS -----
286
287 •   SELECT UPPER(name) FROM Patients;
```

Result Grid		Filter Rows:	Export:	Wrap Cell Content:
	UPPER(name)			
▶	AMIT SHAH			
	NEHA PATEL			
	RAHUL MEHTA			
	PRIYA DESAI			
	KARAN JOSHI			
	RIYA SHARMA			
	ARJUN VERMA			
	SNEHA RAO			
	VIKRAM SINGH			
	ANJALI GUPTA			
	TEST USER			
	TEST USER			
	ROHIT KUMAR			
...				

```
289 •   SELECT TRIM(name) FROM Doctors;
```

Result Grid		Filter Rows:	Export:	Wrap Cell Content:
	TRIM(name)			
▶	Dr. Mehul			
	Dr. Rakesh			
	Dr. Nisha			
	Dr. Kunal			
	Dr. Rina			
	Dr. Sanjay			
	Dr. Alok			
	Dr. Pooja			
	Dr. Aman			
	Dr. Neeraj			
	Dr. Suresh			

```
291 •   SELECT IFNULL(phone_number,'Not Available') FROM Patients;
```

Result Grid		Filter Rows:	Export:	Wrap Cell Content:
	IFNULL(phone_number,'Not Available')			
▶	9876543210			
	9876543211			
	9876543212			
	9876543213			
	9876543214			
	9876543215			
	9876543216			
	9876543217			
	9876543218			
	9876543219			
	9999999999			
	9999999999			
	9898989898			

```
293     # 11.----- WINDOW FUNCTIONS -----
```

```
294
```

```
295 •   SELECT doctor_id, COUNT(*) AS total_visits,  
296      RANK() OVER (ORDER BY COUNT(*) DESC) AS rank_no  
297      FROM Appointments GROUP BY doctor_id;
```

```
298
```

Result Grid			Filter Rows:	Export:	Wrap Cell Content:
	doctor_id	total_visits	rank_no		
▶	1	6	1		
	2	1	2		
	3	1	2		
	4	1	2		
	5	1	2		
	6	1	2		
	7	1	2		
	8	1	2		
	9	1	2		
	10	1	2		
	11	1	2		

```
99 • SELECT payment_date, SUM(amount) OVER (ORDER BY payment_date) AS running_revenue FROM Billing;
```

payment_date	running_revenue
NULL	3700.00
2024-01-10	4400.00
2024-01-20	5300.00
2024-02-01	6600.00
2024-02-10	8000.00
2024-02-20	8800.00
2024-03-01	10000.00

```
301 • SELECT appointment_date, COUNT(*) OVER (ORDER BY appointment_date) AS running_appointments FROM Appointments;
```

appointment_date	running_appointments
2024-01-10	1
2024-01-20	2
2024-02-01	3
2024-02-10	4
2024-02-15	5
2024-02-20	6
2024-03-01	7
2024-03-02	8
2024-03-03	9
2024-03-05	10
2024-03-06	11
2024-03-07	12
2024-03-08	13
2024-03-09	14
2024-03-10	15
2026-02-25	16

```
303      # 12.----- CASE EXPRESSION -----  
304  
305 •  SELECT p.patient_id,  
306   CASE  
307     WHEN COUNT(m.record_id) > 5 THEN 'High'  
308     WHEN COUNT(m.record_id) BETWEEN 3 AND 5 THEN 'Medium'  
309     ELSE 'Low'  
310   END AS Patient_Risk_Level  
311   FROM Patients p  
312   LEFT JOIN Medical_Records m ON p.patient_id=m.patient_id  
313   GROUP BY p.patient_id;  
314
```

Result Grid		
	patient_id	Patient_Risk_Level
▶	1	Low
	2	Low
	3	Low
	4	Low
	5	Low
	6	Low
	7	Low
	8	Low
	9	Low
	10	Low
	11	Low
	12	Low
	13	Low