

- ❖ This screenshot shows the MySQL command use amazecare; being executed.
- ❖ It confirms that the database named amazecare has been successfully selected, allowing all subsequent queries to interact specifically with this database.

```
mysql> use amazecare;
Database changed
```

- ❖ Here, the show tables; command is used to list all the tables available within the amazecare database.
- ❖ The result displays seven tables: appointment, doctor, medical\_record, patient, prescription, test\_recommendation, and users, confirming that the database schema has been properly set up.

```
mysql> show tables;
+-----+
| Tables_in_amazecare |
+-----+
| appointment          |
| doctor                |
| medical_record        |
| patient               |
| prescription           |
| test_recommendation   |
| users                 |
+-----+
7 rows in set (0.05 sec)
```

- ❖ This screenshot displays the output of select \* from users;, listing all user accounts present in the system.
- ❖ It includes details such as user ID, hashed password, role (like DOCTOR, ADMIN, or PATIENT), username, and email, verifying that multiple user records have been correctly inserted into the users table.

```
mysql> select * from users;
+----+-----+-----+-----+-----+
| id | password | role | username | email |
+----+-----+-----+-----+-----+
| 24 | $2a$10$WoiwFSLirZj0pzTP6G2J4eTB9SUS0aLvva7eZNoJZYmTr50WAE93W | DOCTOR | Stefan | stefan@gmail.com |
| 25 | $2a$10$LCJ52Kx6LQV6TwXQjDawROCBHMGkn5tEfeTWVfzSrRbLnWyd/tYaG | DOCTOR | Klaus | klaus@gmail.com |
| 26 | $2a$10$skXSchdDe5d6sPs/RxBNhu0eC.cgaH0jV.658.ZGz.LHWxSAm15AG | DOCTOR | Elena | elena@gmail.com |
| 27 | $2a$10$7VDR23FMga5L04yB139x10JSCx4ijjI2o5AQoVe/rcgVcvY.T3596K | DOCTOR | Caroline | caroline@gmail.com |
| 28 | $2a$10$7vGN7xa79JeTtAb6Fp01GDue89jcfQ6Jp9g8Lqpe7C0oYsxjFd05xG | DOCTOR | Tyler | tyler@gmail.com |
| 29 | $2a$10$7BsJPm3sqDvpfwPpH8CWMe/IyZfi3g9o1gpU4bGxxNA33iltbxUZS | DOCTOR | Katherine | katherine@gmail.com |
| 30 | $2a$10$abgMvR.s3nQ33KJRRStRduHQHrHYzSTJ7pp/pYTkik70.ylpx1R/e | DOCTOR | Jeremy | jeremy@gmail.com |
| 31 | $2a$10$YLF3NYvaAx9UKEXbtndsNudVrrM0VFuVm0xA0WNK0h36r2dUtaX6C | DOCTOR | Damon | damon@gmail.com |
| 32 | $2a$10$Veu4v17vXhRzZ.c5iDI30RJ5ikIf5ZwBJqortnxUpXtIjrA6V4MK | DOCTOR | Bonnie | bonnie@gmail.com |
| 33 | $2a$10$2igX/pbuasS4tywMG9xBVeSMhCJLZqowueTAXuGAZogKoN011NSwq | DOCTOR | Rose | rose@gmail.com |
| 34 | $2a$10$PijJwuJrrW2K7szMP2NE10.TK1aBgnTQd/L0Q9uCP9uPI3UKzKDNa | DOCTOR | Elijah | elijah@gmail.com |
| 35 | $2a$10$FMHYLXKxL5bfANp3/DIpr.s19IObrsBBY.P6ZIR/nx1r8n904vAAK | DOCTOR | Emily | emily@gmail.com |
| 36 | $2a$10$V.6CJsvFIkmNdcF/QASBz.0ZALhvQMqF7fGPTI0HlhQ9Z142/aU8W | DOCTOR | Gilbert | gilbert@gmail.com |
| 37 | $2a$10$W8WUGFLJoe06Wk8HX1ocuwqgrzInPsWkqN.RIQpwXIGHp9kHRD9S | DOCTOR | Jules | jules@gmail.com |
| 38 | $2a$10$YLORaDUGoMFSrsrw0kChv1.9mS0/zDUgnhu4FMKHxepk/LiPZS6/L. | DOCTOR | Mason | mason@gmail.com |
| 39 | $2a$10$725L2g2ebSz/oIZbzm7xN.reUXLkU8TnPa08I0Y39tV7ATxPh2Bm | DOCTOR | Carroline | carroline@gmail.com |
| 43 | $2a$10$VBQZF6iyheVI04Xp1v17luGcFrXdcuFb8i1Ct0HQ0yOpRgE8V0Qe | PATIENT | Leo | sus200327@gmail.com |
| 48 | $2a$10$R8TCEAPC75CyEbAIJ7N7D0ybe1nbN570oofBY5YKiXC3.hb0ApXQG | ADMIN | sushmitha | sushmitha@gmail.com |
| 54 | $2a$10$XQz6j8LLtXeVtglYpVL4u7aLeDSxepHj2viRmx28rUbZvUv0ER2G | PATIENT | shree | shreests07@gmail.com |
| 55 | $2a$10$7y0zlj9pSbXSkAG.FtKRuuhpC.ca3E/03tgxzwkn5Y0tXRWbz3R3. | DOCTOR | jacob | NULL |
| 56 | $2a$10$.0wN97YxkEeJYp7PBf53NUF0yZcdG13QtNa0B1y1z84LIQg8zbza | PATIENT | surya | NULL |
+----+-----+-----+-----+-----+
21 rows in set (0.00 sec)
```

- ♣ The select \* from doctor; command output is shown here, listing doctor-specific details such as designation, years of experience, name, qualification, specialty, and the associated user ID.
- ♣ This confirms that doctor profiles are accurately stored and linked to their respective user accounts.

```
mysql> select * from doctor;
```

id	designation	experience	name	qualification	specialty	user_id	photo_url
8	Senior Cardiologist	5	Stefan	MD (Cardiology)	Cardiology	24	NULL
9	Consultant Neurologist	14	Klaus	DM (Neurology)	Neurology	25	NULL
10	Skin Specialist	15	Elena	MD (Dermatology)	Dermatology	26	NULL
12	Endocrinologist	6	Tyler	DM (Endocrinology)	Endocrinology	28	NULL
13	Head of Cardiology Department	15	Katherine	MD (Cardiology)	Cardiology	29	NULL
14	Heart Specialist	4	Jeremy	MD (Cardiology)	Cardiology	30	NULL
24	Neuro Specialist	12	Damon	MD (Neurology)	Neurology	31	NULL
25	Neurologist	5	Bonnie	MD (Neurology)	Neurology	32	NULL
26	Cosmetic Dermatologist	19	Rose	MD (Dermatology)	Dermatology	33	NULL
27	Dermatopathologist	11	Elijah	MD (Dermatology)	Dermatology	34	NULL
28	Hormone Specialist	11	Emily	MD (Endocrinology)	Endocrinology	35	NULL
29	Head of Endocrinology Department	19	Gilbert	MD (Endocrinology)	Endocrinology	36	NULL
30	Consultant Nephrologist	9	Jules	MD (Nephrology)	Nephrology	37	NULL
31	Kidney Specialist	10	Mason	MD (Nephrology)	Nephrology	38	NULL
32	Senior Nephrologist	11	Carroline	MD (Nephrology)	Nephrology	39	NULL
36	MD	12	jacab	MD	Cardiology	55	NULL

16 rows in set (0.00 sec)

- ♣ This screenshot shows the patient table using select \* from patient;
- ♣ It presents patient information including contact number, date of birth, full name, gender, health issue, user ID, and email.
- ♣ This verifies that patient data is correctly maintained in the database.

```
mysql> select * from patient;
```

id	contact_number	dob	full_name	gender	health_issue	user_id	email
6	9876545678	2003-12-27	Leo	Male	Heart Attack	43	sus200327@gmail.com
9	999999999999	2025-07-23	surya	Male	Heart attack	56	NULL

2 rows in set (0.00 sec)

- ♣ Here, the select \* from appointment; command displays details of patient appointments, such as appointment date and time, status (e.g., COMPLETED), symptoms, doctor ID, and patient ID.
- ♣ This confirms that the system is successfully capturing and managing appointment records.

```
mysql> select * from appointment;
```

id	appointment_date_time	status	symptoms	doctor_id	patient_id
44	2025-07-22 10:44:00.000000	COMPLETED	Common Cold	24	6

1 row in set (0.00 sec)

- ♣ This screenshot shows the prescription table output from `select * from prescription;`.
- ♣ It includes information like medicine details, linked appointment ID, diagnosis, recommended tests, symptoms, and treatment plan, confirming that prescriptions are stored and connected to relevant appointments.

```
mysql> select * from prescription;
+----+-----+-----+-----+-----+-----+-----+
| id | medicine_details | appointment_id | diagnosis | recommended_tests | symptoms | treatment_plan |
+----+-----+-----+-----+-----+-----+-----+
| 21 | aaaaaaaaaa | 44 | aaaaaa | aaaaaa | common cold | aaaaaa |
+----+-----+-----+-----+-----+-----+-----+
1 row in set (0.01 sec)
```

- ♣ Finally, the `select * from medical_record;` command output displays the medical records table, which holds details like diagnosis, recommended tests, treatment plan, and the associated appointment ID.
- ♣ This ensures that comprehensive patient medical records are maintained and linked to the correct appointments.

```
mysql> select * from medical_record;
+----+-----+-----+-----+-----+
| id | diagnosis | recommended_tests | treatment_plan | appointment_id |
+----+-----+-----+-----+-----+
| 29 | aaaaaa | aaaaaa | aaaaaa | 44 |
+----+-----+-----+-----+-----+
1 row in set (0.00 sec)
```