

Create and use the database :-

```

1  -- create the database
2
3 •  create database da_ds;
4
5
6  -- use to the database
7
8 •  use da_ds;
9
10

```

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

Context Help Snippets

Output Action Output

#	Time	Action	Message	Duration / Fetch
1	08:01:58	use da_ds	0 row(s) affected	0.016 sec

Create table students :-

```

16  -- create the table students
17
18 • CREATE TABLE students (
19     student_id INT PRIMARY KEY,
20     student_name VARCHAR(100) NOT NULL,
21     email VARCHAR(100) UNIQUE,
22     join_date DATE
23 )
24
25
26  -- Insert the values in the student table
27
28 • insert into students(student_id,student_name,email,join_date) values
29     (1,"angel","angel@gmail.com","2025-08-15"),
30     (2,"raghav","raghav@gmail.com","2025-08-17"),
31     (3,"prashant","prashant@gmail.com","2025-08-19"),
32     (4,"sidhaanth","sidhaanth@gmail.com","2025-08-29");

```

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Contents: | Result Grid | Form Editor | Context Help Snippets

student_id	student_name	email	join date
1	angel	angel@gmail.com	2025-08-15
2	raghav	raghav@gmail.com	2025-08-17
3	prashant	prashant@gmail.com	2025-08-19
4	sidhaanth	sidhaanth@gmail.com	2025-08-29

students 1 x

Output Action Output

#	Time	Action	Message	Duration / Fetch
1	08:01:58	use da_ds	0 row(s) affected	0.016 sec
2	08:05:05	select * from students LIMIT 0,1000	4 row(s) returned	0.031 sec / 0.000 sec

Create table courses :-

```

40  -- create the table courses
41
42 • CREATE TABLE courses(
43     course_id INT PRIMARY KEY,
44     course_name VARCHAR(100) NOT NULL,
45     duration_months INT,
46     fees DECIMAL(8,2)
47 );
48
49
50  -- insert the values in the courses table
51
52 • insert into courses(course_id,course_name,duration_months,fees) values
53     (11,"da",3,10000),
54     (12,"ds",4,15000),
55     (13,"full stack",6,20000),
56     (14,"reactjs",5,12000),

```

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Contents: | Result Grid | Form Editor | Context Help Snippets

course_id	course_name	duration_months	fees
11	da	3	10000.00
12	ds	4	15000.00
13	full stack	6	20000.00
14	reactjs	5	12000.00
15	digital marketing	3	15000.00
16	robotics	6	50000.00

courses 2 x

Output Action Output

#	Time	Action	Message	Duration / Fetch
2	08:05:05	select * from students LIMIT 0,1000	4 row(s) returned	0.031 sec / 0.000 sec
3	08:08:08	select * from courses LIMIT 0,1000	7 row(s) returned	0.000 sec / 0.000 sec

Create table enrollment :-

```

65 -- create the table enrollment
66
67
68 • CREATE TABLE enrollments (
69     enrollment_id INT PRIMARY KEY,
70     student_id INT,
71     course_id INT,
72     enrollment_date DATE,
73     FOREIGN KEY (student_id) REFERENCES students(student_id),
74     FOREIGN KEY (course_id) REFERENCES courses(course_id)
75 );
76
77
78 -- insert the value into the enrollment table
79
80 • insert into enrollments(enrollment_id, student_id, course_id, enrollment_date) values

```

Automatic context help disabled. Use the toolbar manually get help for the current caret position or toggle automatic help.

Result Grid

enrollment_id	student_id	course_id	enrollment_date
101	1	12	2016-01-15
102	2	15	2016-01-20
103	3	14	2016-01-15
104	4	15	2016-01-15
105	1	13	2016-01-15
106	2	15	2016-01-15

enrollments 3 ×

Action Output

#	Time	Action	Message	Duration / Fetch
3	08:08:08	select * from courses LIMIT 0, 1000	7 row(s) returned	0.000 sec / 0.000 sec
4	08:10:20	select * from enrollments LIMIT 0, 1000	12 row(s) returned	0.000 sec / 0.000 sec

Show all the tables in the da_ds database :-

```

100 -- show all the table
101
102 • show tables;
103
104

```

Result Grid

Tables_in_da_ds
courses
employee
enrollments
student_mark
students

Result 4 ×

Action Output

#	Time	Action	Message	Duration / Fetch
4	08:10:20	select * from enrollments LIMIT 0, 1000	12 row(s) returned	0.000 sec / 0.000 sec
5	08:11:53	show tables	5 row(s) returned	0.015 sec / 0.000 sec

join the two tables students and enrollments :-

```

104
105 -- join the two tables students and enrollments
106
107 • select * from students as s
108 join enrollments as e on s.student_id = e.student_id;
109
110

```

Result Grid

student_id	student_name	email	join_date	enrollment_id	student_id	course_id	enrollment_date
1	angel	angel@gmail.com	2025-08-15	101	1	12	2016-01-15
1	angel	angel@gmail.com	2025-08-15	105	1	13	2016-01-15
1	angel	angel@gmail.com	2025-08-15	109	1	14	2016-01-15
1	angel	angel@gmail.com	2025-08-15	111	1	15	2016-01-15
2	raghav	raghav@gmail.com	2025-08-17	102	2	15	2016-01-20
2	raghav	raghav@gmail.com	2025-08-17	106	2	15	2016-01-15

Result 5 ×

Action Output

#	Time	Action	Message	Duration / Fetch
5	08:11:53	show tables	5 row(s) returned	0.015 sec / 0.000 sec
6	08:16:59	select * from students as s join enrollments as e on s.student_id = e.student_id LIMIT 0, 1000	12 row(s) returned	0.000 sec / 0.000 sec

join the table only the where student id is 1 :-

```
111 -- join the table only the where student id is 1
112
113 • select * from students as s
114   join enrollments as e on s.student_id = e.student_id
115   where s.student_id = 1;
116
```

Result Grid | Filter Rows: Export: Wrap Cell Content: Result Grid | Read Only | Context Help | Snippets

student_id	student_name	email	join_date	enrollment_id	student_id	course_id	enrollment_date
1	angel	angel@gmail.com	2025-08-15	101	1	12	2016-01-15
1	angel	angel@gmail.com	2025-08-15	105	1	13	2016-01-15
1	angel	angel@gmail.com	2025-08-15	109	1	14	2016-01-15
1	angel	angel@gmail.com	2025-08-15	111	1	15	2016-01-15

Action Output

#	Time	Action	Message	Duration / Fetch
6	08:16:59	select * from students as s join enrollments as e on s.student_id = e.student_id LIMIT 0, 1000	12 row(s) returned	0.000 sec / 0.000 sec
7	08:19:48	select * from students as s join enrollments as e on s.student_id = e.student_id where s.student_id = 1 LIMIT 0, ... 4 row(s) returned		0.016 sec / 0.000 sec

count the student where the student id is 1 :-

```
111
112 -- count the student where the student id is 1
113
114 • select count(*) from students as s
115   join enrollments as e on s.student_id = e.student_id
116   where s.student_id = 1;
```

Result Grid | Filter Rows: Export: Wrap Cell Content: Result Grid | Read Only | Context Help | Snippets

count(*)
4

Action Output

#	Time	Action	Message	Duration / Fetch
7	08:19:48	select * from students as s join enrollments as e on s.student_id = e.student_id where s.student_id = 1 LIMIT 0, ... 4 row(s) returned		0.016 sec / 0.000 sec
8	08:22:27	select count(*) from students as s join enrollments as e on s.student_id = e.student_id where s.student_id = 1 U... 1 row(s) returned		0.000 sec / 0.000 sec

total courses get by student id 1 :-

```
124
125 -- total courses get by student id 1
126
127 • select count(*) as total_courses from students as s
128   join enrollments as e on s.student_id = e.student_id
129   where s.student_id = 1;
130
131
```

Result Grid | Filter Rows: Export: Wrap Cell Content: Result Grid | Read Only | Context Help | Snippets

total_courses
4

Action Output

#	Time	Action	Message	Duration / Fetch
8	08:22:27	select count(*) from students as s join enrollments as e on s.student_id = e.student_id where s.student_id = 1 Li... 1 row(s) returned		0.000 sec / 0.000 sec
9	08:25:51	select count(*) as total_courses from students as s join enrollments as e on s.student_id = e.student_id where s... 1 row(s) returned		0.000 sec / 0.000 sec

check the which column are in the courses table :-

```
132 -- check the which column are in the courses table
133
134 • select * from courses;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content: | Result Grid | Form Editor | Context Help | Snippets

course_id	course_name	duration_months	fees
11	da	3	10000.00
12	ds	4	15000.00
13	full stack	6	20000.00
14	reactjs	5	12000.00
15	digital marketing	3	15000.00
16	robotics	6	50000.00

courses 9 ×

Output:

Action Output

#	Time	Action	Message	Duration / Fetch
9	08:25:51	select count(*) as total_courses from students as s join enrollments as e on s.student_id = e.student_id where s...	1 row(s) returned	0.000 sec / 0.000 sec
10	08:27:58	select * from courses LIMIT 0, 1000	7 row(s) returned	0.000 sec / 0.000 sec

total fees of the course id 12 :-

```
136
137 -- total fees of the course id 12
138
139 • select sum(fees) as total_fees from courses as c
140 join enrollments as e on c.course_id = e.course_id
141 where e.course_id=12;
142
143
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Result Grid | Form Editor | Context Help | Snippets

total_fees
30000.00

Result 10 ×

Action Output

#	Time	Action	Message	Duration / Fetch
10	08:27:58	select * from courses LIMIT 0, 1000	7 row(s) returned	0.000 sec / 0.000 sec
11	08:32:14	select sum(fees) as total_fees from courses as c join enrollments as e on c.course_id = e.course_id where e.co...	1 row(s) returned	0.016 sec / 0.000 sec

select 3 column course id , course name , sum(fees) of top 3 highest fees :-

```
145
146 • select e.course_id, c.course_name, sum(fees) from courses as c
147 join enrollments as e on c.course_id = e.course_id
148 group by e.course_id
149 order by sum(fees) desc
150 limit 3;
151
152
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Fetch Rows: | Result Grid | Form Editor | Context Help | Snippets

course_id	course_name	sum(fees)
15	digital marketing	60000.00
13	full stack	60000.00
12	ds	30000.00

Result 11 ×

Action Output

#	Time	Action	Message	Duration / Fetch
11	08:32:14	select sum(fees) as total_fees from courses as c join enrollments as e on c.course_id = e.course_id where e.co...	1 row(s) returned	0.016 sec / 0.000 sec
12	08:36:10	select e.course_id, c.course_name, sum(fees) from courses as c join enrollments as e on c.course_id = e.course_id ...	3 row(s) returned	0.000 sec / 0.000 sec

select 3 column course id , course name , sum(fees) of highest fees to lowest fees :-

```
160 -- select 3 column course id , course name , sum(fees) of highest fees to lowest fees
161
162 • select e.course_id, c.course_name, sum(fees) from courses as c
163 join enrollments as e on c.course_id = e.course_id
164 group by e.course_id, c.course_name
165 order by sum(fees) desc
166
167
```

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

course_id	course_name	sum(fees)
13	full stack	60000.00
15	digital marketing	60000.00
12	ds	30000.00
14	reactjs	24000.00
11	da	10000.00

Result 13 × Read Only Context Help Snippets

Action Output

#	Time	Action	Message	Duration / Fetch
14	08:40:49	select e.course_id, c.course_name, sum(fees) from courses as c join enrollments as e on c.course_id = e.course_id group by e.course_id, c.course_name order by sum(fees) desc	5 row(s) returned	0.015 sec / 0.000 sec
15	08:44:47	select e.course_id, c.course_name, sum(fees) from courses as c left join enrollments as e on c.course_id = e.course_id group by e.course_id, c.course_name order by sum(fees) desc	5 row(s) returned	0.000 sec / 0.000 sec

Left Join select 3 column course id , course name , sum(fees) of highest fees to lowest fees :-

```
167 -- Left Join select 3 column course id , course name , sum(fees) of highest fees to lowest fees
168
169
170 • select e.course_id, c.course_name, sum(fees) from courses as c
171 left join enrollments as e on c.course_id = e.course_id
172 group by e.course_id, c.course_name
173 order by sum(fees) desc
174
```

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

course_id	course_name	sum(fees)
NULL	Quantum Physics	150000.00
13	full stack	60000.00
15	digital marketing	60000.00
NULL	robotics	50000.00
12	ds	30000.00
14	reactjs	24000.00

Result 14 × Read Only Context Help Snippets

Action Output

#	Time	Action	Message	Duration / Fetch
15	08:44:47	select e.course_id, c.course_name, sum(fees) from courses as c join enrollments as e on c.course_id = e.course_id group by e.course_id, c.course_name order by sum(fees) desc	5 row(s) returned	0.000 sec / 0.000 sec
16	08:46:10	select e.course_id, c.course_name, sum(fees) from courses as c left join enrollments as e on c.course_id = e.course_id group by e.course_id, c.course_name order by sum(fees) desc	7 row(s) returned	0.000 sec / 0.000 sec