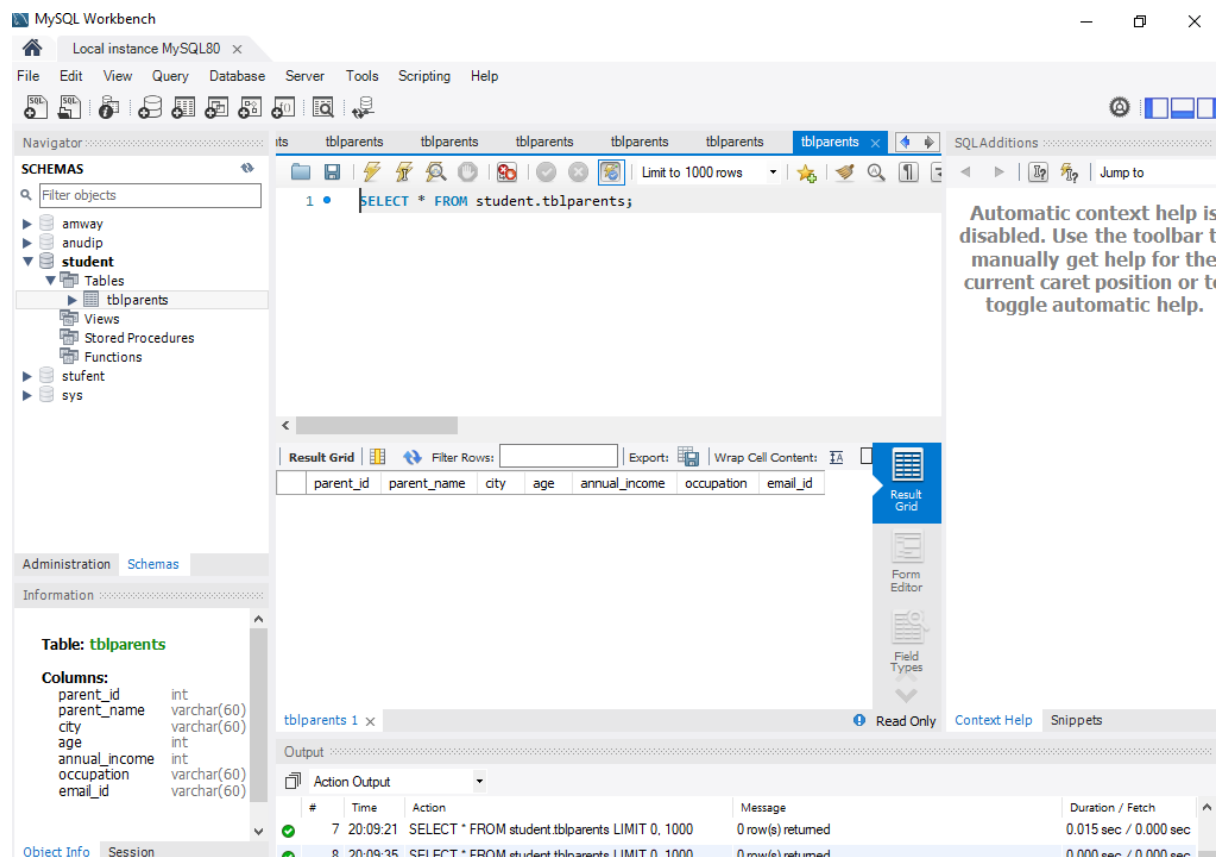


$$);$$


```
SELECT * FROM student.tblparents;
```

```
insert into tblparents values
```

```
(001,'Ravi','Lucknow',40,200000,'Tester','abc@gmail.com'),  
(002,'Rocky','Delhi',45,250000,'Tester','abc@gmail.com'),  
(003,'Yash','NewDelhi',23,300000,'Tester','abc@gmail.com'),  
(004,'Rohan','Mumbai',25,350000,'Tester','abc@gmail.com'),  
(005,'Seema','Newyork',30,400000,'Tester','abc@gmail.com'),  
(006,'Ragav','London',40,450000,'Tester','abc@gmail.com'),  
(007,'Sumit','Lucknow',50,150000,'Tester','abc@gmail.com'),  
(008,'Prakesh','Delhi',45,200000,'Tester','abc@gmail.com'),  
(009,'Titu','Mumbai',56,250000,'Tester','abc@gmail.com'),  
(010,'Pradeep','Lucknow',70,10000,'Tester','abc@gmail.com'),  
(011,'ALOK','Lucknow',50,55000,'Tester','abc@gmail.com'),  
(012,'Deepak','Newyork',51,50000,'Tester','abc@gmail.com'),  
(013,'Karn','Lucknow',53,600000,'Tester','abc@gmail.com'),  
(014,'Ravi','Banglore',40,650000,'Tester','abc@gmail.com'),  
(015,'Vivek','London',54,700000,'Tester','abc@gmail.com'),  
(016,'Vikash','Lucknow',52,76000,'Tester','abc@gmail.com'),  
(017,'Ravi','Lucknow',40,250000,'Tester','abc@gmail.com'),  
(018,'Ravi','Lucknow',21,900000,'Tester','abc@gmail.com'),  
(019,'Ram','Lucknow',40,800000,'Tester','abc@gmail.com'),  
(020,'Sita','NewDelhi',77,850000,'Tester','abc@gmail.com'),  
(021,'Sagar','Ahmedabad',40,950000,'Tester','abc@gmail.com'),  
(022,'Bibek','Chennai',60,150000,'Tester','abc@gmail.com'),  
(023,'Rohan','Nagpur',65,350000,'Tester','abc@gmail.com'),  
(024,'Ravi','Kanpur',55,450000,'Tester','abc@gmail.com'),
```

(025,'Rmash','Lucknow',45,300000,'Tester','abc@gmail.com'),

(026,'Ravi','Delhi',40,250000,'Tester','abc@gmail.com');

The screenshot displays the MySQL Workbench interface. The left sidebar shows the 'HEMAS' database structure with tables like 'tblparents'. The central SQL editor contains a query to insert data into the 'tblparents' table. The 'Result Grid' shows the output of the query, listing 11 rows of parent data. The bottom panel shows the 'Action Output' with a successful insert statement and its execution details.

SQL Query:

```
1 • SELECT * FROM student.tblparents;
2 • insert into tblparents values
3
4     (001,'Ravi','Lucknow',40,200000,'Tester','abc@gmail.com'),
5     (002,'Rocky','Delhi',45,250000,'Tester','abc@gmail.com'),
6     (003,'Yash','NewDelhi',23,300000,'Tester','abc@gmail.com'),
7     (004,'Rohan','Mumbai',25,350000,'Tester','abc@gmail.com'),
8     (005,'Seema','Newyork',30,400000,'Tester','abc@gmail.com'),
9     (006,'Ragav','London',40,450000,'Tester','abc@gmail.com');
```

Result Grid:

	parent_id	parent_name	city	age	annual_income	occupation	email_id
▶	1	Ravi	Luc...	40	200000	Tester	abc@g...
	2	Rocky	Delhi	45	250000	Tester	abc@g...
	3	Yash	Ne...	23	300000	Tester	abc@g...
	4	Rohan	Mu...	25	350000	Tester	abc@g...
	5	Seema	Ne...	30	400000	Tester	abc@g...
	6	Ragav	Lon...	40	450000	Tester	abc@g...
	7	Sumit	Luc...	50	150000	Tester	abc@g...
	8	Prakesh	Delhi	45	200000	Tester	abc@g...
	9	Titu	Mu...	56	250000	Tester	abc@g...
	10	Pradeep	Luc...	70	100000	Tester	abc@g...
	11	MOCK	Luc...	50	55000	Tester	abc@g...

Table: tblparents

Columns:

- parent_id int
- parent_name varchar(60)
- city varchar(60)
- age int
- annual_income int
- occupation varchar(60)
- email_id varchar(60)

Action Output:

#	Time	Action	Message	Duration / Fetch
✓ 10	20:16:20	insert into tblparents values (001,'Ravi','Lucknow',40,...	26 row(s) affected Records: 26 Duplicates: 0 Warning...	0.110 sec

2. `SELECT * FROM student.tblparents;`

`SET SQL_SAFE_UPDATES =0;`

`update tblparents set email_id= 'NA';`

`commit;`

The screenshot shows the MySQL Workbench interface for a 'Local instance MySQL80'. The SQL editor contains the following queries:

```
1 • SELECT * FROM student.tblparents;
2 • SET SQL_SAFE_UPDATES =0;
3 • update tblparents set email_id='NA';
4 • commit;
```

The 'Result Grid' displays the data from the first query, showing 10 rows of parent information. The columns are: parent_id, parent_name, city, age, annual_income, occupation, and email_id.

parent_id	parent_name	city	age	annual_income	occupation	email_id
1	Ravi	Lucknow	40	200000	Tester	NA
2	Rocky	Delhi	45	250000	Tester	NA
3	Yash	NewDelhi	23	300000	Tester	NA
4	Rohan	Mumbai	25	350000	Tester	NA
5	Seema	Newyork	30	400000	Tester	NA
6	Ragav	London	40	450000	Tester	NA
7	Sumit	Lucknow	50	150000	Tester	NA
8	Prakesh	Delhi	45	200000	Tester	NA
9	Titu	Mumbai	56	250000	Tester	NA
10	Pradeep	Lucknow	70	100000	Tester	NA

On the left, the 'Schemas' tab is active, showing the 'student' database with the 'tblparents' table selected. The table's columns and data types are listed:

Table: tblparents

Columns:

- parent_id: int
- parent_name: varchar(60)
- city: varchar(60)
- age: int
- annual_income: int
- occupation: varchar(60)
- email_id: varchar(60)

On the right, a text box states: 'Automatic context help disabled. Use the toolbar manually get help for the current caret position or toggle automatic help'.

3. SELECT * FROM student.tblparents;

select count(parent_id) from tblparents where annual_income>600000;

MySQL Workbench

Local instance MySQL80 x

Edit View Query Database Server Tools Scripting Help

tbldparents tbldparents tbldparents tbldparents - Table tbldparents x

Limit to 1000 rows

1 • SELECT * FROM student.tblparents;

2 • select count(parent_id) from tblparents where annual_income>600000;

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

Result Grid

count(parent_id)
6

Filter Rows: Export: Wrap Cell Content: I A

Table: tblparents

Columns:

parent_id	int
parent_name	varchar(60)
city	varchar(60)
age	int
annual_income	int
occupation	varchar(60)
email_id	varchar(60)

tbldparents 2 Result 3 x

Read Only Context Help Snippets

Output

Action Output

4. SELECT * FROM student.tblparents;

select* from tblparents where city in('Ahmedabad','Delhi','Mumbai','Chennai','Banglore');

The screenshot displays the MySQL Workbench interface for a local instance of MySQL 8.0. The left sidebar shows the 'Schemas' tab with a tree view of databases including 'student'. The main editor window contains two SQL queries:

```
1 • SELECT * FROM student.tblparents;  
2 • select* from tblparents where city in('Ahmedabad','Delhi','Mumbai','Ch
```

The 'Result Grid' shows the results of the first query, displaying 18 rows of parent data. The columns are: parent_id, parent_name, city, age, annual_income, occupation, and email_id.

parent_id	parent_name	city	age	annual_income	occupation	email_id
9	Ravi8	Mumbai	56	25000	Tester	NA
26	Ravi	Delhi	40	25000	Tester	NA
2	Ravi1	Delhi	45	250000	Tester	NA
4	Ravi3	Mumbai	25	350000	Tester	NA
8	Ravi7	Delhi	45	200000	Tester	NA
9	Ravi8	Mumbai	56	25000	Tester	NA
18	Kuldeep	Ahmed...	21	900000	Tester	NA
19	Ravi	Chennai	40	800000	Tester	NA
21	Ravi	Banglore	40	95000	Tester	NA
26	Ravi	Delhi	40	25000	Tester	NA

The 'Output' tab at the bottom shows the execution log with two entries:

#	Time	Action	Message	Duration / Fetch
✓ 29	19:45:59	SELECT * FROM student.tblparents LIMIT 0, 1000	182 row(s) returned	0.000 sec / 0.000 sec
✓ 30	19:45:59	select* from tblparents where city in('Ahmedabad','Delhi'...	33 row(s) returned	0.000 sec / 0.000 sec

A text box on the right side of the interface states: "Automatic context help disabled. Use the toolbar manually get help for th current caret position or toggle automatic help."

5. SELECT * FROM student.tblparents;

select * from tblparents where annual_income<500000 or age>45;

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

Table: tblparents

Columns:

- parent_id int
- parent_name varchar(60)
- city varchar(60)
- age int
- annual_income int
- occupation varchar(60)
- email_id varchar(60)

#	Time	Action	Message	Duration / Fetch
21	20:34:31	SELECT * FROM student.tblparents LIMIT 0, 1000	26 row(s) returned	0.000 sec / 0.000 sec

6. SELECT * FROM student.student;

use student;

```
create table student (  
    student_id int primary key,  
    student_name varchar(20),  
    maths_marks int not null,  
    science_marks int not null,  
    parent_id int not null,  
    foreign key(parent_id) REFERENCES tblparents(parent_id)  
);
```

insert into student values

```
(1,"Himani", 90,89,101),  
(2,"Diksha ",65,72,102),  
(3,"Shivani ",69,55,101),  
(4,"Shiv",45,52,117),  
(5,"Deepak",69,75,119),  
(6,"Chandan",35,45,104),  
(7,"Pooja",78,67,115),  
(8,"Neha",75,49,113),  
(9,"Meher",52,74,111),  
(10,"Manisha",85,75,106);  
  
commit;
```


Local instance MySQL80 x

File Edit View Query Database Server Tools Scripting Help

agents agents employee employee employee student orders tblparents tblparents tblparents tblparents tblparents tblparents tblparents tblp

1 • SELECT * FROM student.student;

student

- Tables
 - student
 - tblparents**
 - Columns
 - Indexes
 - Foreign Keys
 - Triggers
- Views
- Stored Procedures
- Functions

sys

administration Schemas

information

Table: **tblparents**

Columns:

- parent_id int PK
- parent_name varchar(30)
- city varchar(30)
- age int
- annual_income decimal(8,0)
- occupation varchar(25)

Result Grid

student_id	student_name	Maths_marks	science_marks	parent_id
1	Minani	90	89	101
2	Diksha	65	72	102
3	Shivani	69	55	101
4	Shiv	45	52	117
5	Deepak	60	75	119
6	Charidan	35	45	104
7	Pooja	78	67	115
8	Neha	75	49	113
9	Meher	52	74	111
10	Namisha	85	75	106
1000	1000	1000	1000	1000