

STEPS

The screenshot shows the SQL Server Enterprise Manager interface. The left pane displays the 'entri_dsml_d36' database with a tree view showing 'Tables', 'Views', 'Stored Procedures', and 'Functions'. The 'Tables' folder is expanded, showing the 'teachers' table. The right pane shows the SQL query editor with the following code:

```
1 # 1) Create a table named teachers and insert 8 rows
2
3 CREATE TABLE teachers (
4     id INT PRIMARY KEY,
5     name VARCHAR(50),
6     subject VARCHAR(50),
7     experience INT,
8     salary DECIMAL(10,2)
9 );
10
11 INSERT INTO teachers (id, name, subject, experience, salary) VALUES
12 (1, 'John Doe', 'Math', 5, 40000),
13 (2, 'Jane Smith', 'English', 8, 42000),
14 (3, 'Mark Lee', 'Science', 10, 45000);
```

The Output pane shows the results of the query execution:

#	Time	Action	Message
1	21:05:57	CREATE TABLE teachers (id INT PRIMARY KEY, name VARCHAR(50), subj...	0 row(s) affected

The screenshot shows the SQL Server Enterprise Manager interface. The left pane displays the 'entri_dsml_d36' database with a tree view showing 'Tables', 'Views', 'Stored Procedures', and 'Functions'. The 'Tables' folder is expanded, showing the 'teachers' table. The right pane shows the SQL query editor with the following code:

```
8 salary DECIMAL(10,2)
9 );
10
11 INSERT INTO teachers (id, name, subject, experience, salary) VALUES
12 (1, 'John Doe', 'Math', 5, 40000),
13 (2, 'Jane Smith', 'English', 8, 42000),
14 (3, 'Mark Lee', 'Science', 10, 45000),
15 (4, 'Emily Davis', 'History', 2, 35000),
16 (5, 'Paul Brown', 'Math', 12, 50000),
17 (6, 'Anna Johnson', 'English', 4, 37000),
18 (7, 'Michael King', 'Science', 6, 41000),
19 (8, 'Lucy White', 'History', 15, 52000);
20
21 SELECT * FROM teachers;
```

The Output pane shows the results of the query execution:

#	Time	Action	Message	Duration
1	21:05:57	CREATE TABLE teachers (id INT PRIMARY KEY, name VARCHAR(50), subj...	0 row(s) affected	0.000 s
2	21:06:56	INSERT INTO teachers (id, name, subject, experience, salary) VALUES (1, 'John Doe'...	8 row(s) affected Records: 8 Duplicates: 0 Warnings: 0	0.000 s

Query 1 TRIGGERS

Limit to 1000 rows

```

17 (6, 'Anna Johnson', 'English', 4, 37000),
18 (7, 'Michael King', 'Science', 6, 41000),
19 (8, 'Lucy White', 'History', 15, 52000);
20
21 SELECT * FROM teachers;

```

Result Grid

id	name	subject	experience	salary
1	John Doe	Math	5	40000.00
2	Jane Smith	English	8	42000.00
3	Mark Lee	Science	10	45000.00
4	Emily Davis	History	2	35000.00
5	Paul Brown	Math	12	50000.00
6	Anna Johnson	English	4	37000.00

teachers 1 x

Output

Action Output

#	Time	Action	Message
1	21:05:57	CREATE TABLE teachers (id INT PRIMARY KEY, name VARCHAR(50), sub...	0 row(s) affected
2	21:06:56	INSERT INTO teachers (id, name, subject, experience, salary) VALUES (1, 'John Do...	8 row(s) affected Records: 8 Duplicates: 0 Warnings: 0
3	21:07:13	SELECT * FROM teachers LIMIT 0, 1000	8 row(s) returned

Query 1 TRIGGERS

Limit to 1000 rows

```

29 FOR EACH ROW
30 BEGIN
31 IF NEW.salary < 0 THEN
32 SIGNAL SQLSTATE '45000'
33 SET MESSAGE_TEXT = 'salary cannot be negative';
34 END IF;
35 END $$
36
37 DELIMITER ;
38
39
40 # INSERT INTO teachers (id, name, subject, experience, salary) VALUES(9, 'Sibin Francis', 'Physics'
41 # SALARY CANNOT BE NEGATIVE
42

```

Output

Action Output

#	Time	Action	Message
1	21:05:57	CREATE TABLE teachers (id INT PRIMARY KEY, name VARCHAR(50), sub...	0 row(s) affected
2	21:06:56	INSERT INTO teachers (id, name, subject, experience, salary) VALUES (1, 'John Do...	8 row(s) affected Records: 8 Duplicates: 0 Warnings: 0
3	21:07:13	SELECT * FROM teachers LIMIT 0, 1000	8 row(s) returned
4	21:07:42	CREATE TRIGGER before_insert_teacher BEFORE INSERT ON teachers FOR EAC...	0 row(s) affected

Query 1 TRIGGERS

```

40 # INSERT INTO teachers (id, name, subject, experience, salary) VALUES(9, 'Sibin Francis', 'Physics'
41 # SALARY CANNOT BE NEGATIVE
42
43 # 3) Create an after insert trigger named after_insert_teacher that inserts a row with teacher_id,a
44 #timestamp to a table called teacher_log when a new entry gets inserted to the teacher table.
45 #teacher_id -> column of teacher table, action -> the trigger action, timestamp -> time at which t
46
47 CREATE TABLE teacher_log (
48     log_id INT AUTO_INCREMENT PRIMARY KEY,
49     teacher_id INT,
50     action VARCHAR(50),
51     timestamp DATETIME
52 );
53

```

Output

#	Time	Action	Message
2	21:06:56	INSERT INTO teachers (id, name, subject, experience, salary) VALUES (1, John D...	8 row(s) affected Records: 8 Duplicates: 0 Warnings: 0
3	21:07:13	SELECT * FROM teachers LIMIT 0, 1000	8 row(s) returned
4	21:07:42	CREATE TRIGGER before_insert_teacher BEFORE INSERT ON teachers FOR E...	0 row(s) affected
5	21:08:01	CREATE TRIGGER before_insert_teacher BEFORE INSERT ON teachers FOR E...	Error Code: 1359. Trigger already exists
6	21:08:10	CREATE TRIGGER before_insert_teacher BEFORE INSERT ON teachers FOR E...	Error Code: 1359. Trigger already exists
7	21:08:16	CREATE TABLE teacher_log (log_id INT AUTO_INCREMENT PRIMARY KEY, te...	0 row(s) affected

Query 1 TRIGGERS

```

49     teacher_id INT,
50     action VARCHAR(50),
51     timestamp DATETIME
52 );
53
54 DELIMITER $$
55
56 CREATE TRIGGER after_insert_teacher
57 AFTER INSERT ON teachers
58 FOR EACH ROW
59 BEGIN
60     INSERT INTO teacher_log (teacher_id, action, timestamp)
61     VALUES (NEW.id, 'INSERT', NOW());
62 END $$

```

Output

#	Time	Action	Message	Duration /
3	21:07:13	SELECT * FROM teachers LIMIT 0, 1000	8 row(s) returned	0.000 sec /
4	21:07:42	CREATE TRIGGER before_insert_teacher BEFORE INSERT ON teachers FOR E...	0 row(s) affected	0.063 sec
5	21:08:01	CREATE TRIGGER before_insert_teacher BEFORE INSERT ON teachers FOR E...	Error Code: 1359. Trigger already exists	0.016 sec
6	21:08:10	CREATE TRIGGER before_insert_teacher BEFORE INSERT ON teachers FOR E...	Error Code: 1359. Trigger already exists	0.000 sec
7	21:08:16	CREATE TABLE teacher_log (log_id INT AUTO_INCREMENT PRIMARY KEY, te...	0 row(s) affected	0.032 sec
8	21:11:09	CREATE TRIGGER after_insert_teacher AFTER INSERT ON teachers FOR EACH...	0 row(s) affected	0.000 sec

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

HEMAS

Filter objects

entri_dsml_d36

Tables

teacher_log

teachers

Views

Stored Procedures

Functions

product

Tables

Views

Stored Procedures

Functions

sales

Administration

Schemas

No object selected

Query 1

TRIGGERS

Limit to 1000 rows

SQLAdditions

Automatic context disabled. Use the manually get h current caret po toggle autom

```
56 • CREATE TRIGGER after_insert_teacher
57 • ON teachers
58 •
59 •
60 • AFTER INSERT ON teachers FOR EACH ROW
61 • DO BEGIN
62 •   INSERT INTO teacher_log (teacher_id, action, timestamp)
63 •   VALUES (NEW.id, 'INSERT', NOW());
64 • END;
65 •
66 •
67 • INSERT INTO teachers (id, name, subject, experience, salary) VALUES (9, 'Sibin Francis', 'Physics', 7, 85000);
68 •
69 •
```

Output

Action Output

#	Time	Action	Message	
5	21:08:01	CREATE TRIGGER before_insert_teacher BEFORE INSERT ON teachers FOR EACH ROW	Error Code: 1359. Trigger already exists	0
6	21:08:10	CREATE TRIGGER before_insert_teacher BEFORE INSERT ON teachers FOR EACH ROW	Error Code: 1359. Trigger already exists	0
7	21:08:16	CREATE TABLE teacher_log (log_id INT AUTO_INCREMENT PRIMARY KEY, teacher_id INT, action VARCHAR(255), timestamp VARCHAR(255))	0 row(s) affected	0
8	21:11:09	CREATE TRIGGER after_insert_teacher AFTER INSERT ON teachers FOR EACH ROW	0 row(s) affected	0
9	21:11:26	CREATE TRIGGER after_insert_teacher AFTER INSERT ON teachers FOR EACH ROW	Error Code: 1359. Trigger already exists	0
10	21:11:41	INSERT INTO teachers (id, name, subject, experience, salary) VALUES (9, 'Sibin Francis', 'Physics', 7, 85000);	1 row(s) affected	0

Object Info

Session

HEMAS

Filter objects

entri_dsml_d36

Tables

teacher_log

teachers

Views

Stored Procedures

Functions

product

Tables

Views

Stored Procedures

Functions

sales

Administration

Schemas

No object selected

Query 2

teacher_log

Limit to 1000 rows

SQLAdditions

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

```
65 •
66 •
67 • INSERT INTO teachers (id, name, subject, experience, salary) VALUES (9, 'Sibin Francis', 'Physics', 7, 85000);
68 •
69 • SELECT * FROM teacher_log;
```

Result Grid

	log_id	teacher_id	action	timestamp
1	9	9	INSERT	2024-09-26 21:11:41

Output

Action Output

#	Time	Action	Message	Duration
6	21:08:10	CREATE TRIGGER before_insert_teacher BEFORE INSERT ON teachers FOR EACH ROW	Error Code: 1359. Trigger already exists	0.000 s
7	21:08:16	CREATE TABLE teacher_log (log_id INT AUTO_INCREMENT PRIMARY KEY, teacher_id INT, action VARCHAR(255), timestamp VARCHAR(255))	0 row(s) affected	0.032 s
8	21:11:09	CREATE TRIGGER after_insert_teacher AFTER INSERT ON teachers FOR EACH ROW	0 row(s) affected	0.000 s
9	21:11:26	CREATE TRIGGER after_insert_teacher AFTER INSERT ON teachers FOR EACH ROW	Error Code: 1359. Trigger already exists	0.000 s
10	21:11:41	INSERT INTO teachers (id, name, subject, experience, salary) VALUES (9, 'Sibin Francis', 'Physics', 7, 85000);	1 row(s) affected	0.016 s
11	21:12:03	SELECT * FROM teacher_log LIMIT 0. 1000	1 row(s) returned	0.000 s

Object Info

Session

Navigator: HEMAS

Filter objects

- entri_dsmi_d36
 - Tables
 - teacher_log
 - teachers
 - Views
 - Stored Procedures
 - Functions
- product
 - Tables
 - Views
 - Stored Procedures
 - Functions
- sales

Administration Schemas

No object selected

Query 1 TRIGGERS

```

77 FOR EACH ROW
78 BEGIN
79     IF OLD.experience > 10 THEN
80         SIGNAL SQLSTATE '45000'
81         SET MESSAGE_TEXT = 'Cannot delete teacher with more than 10 years of experience';
82     END IF;
83 END $$
84
85 DELIMITER ;
86
87
88 • DELETE FROM teachers WHERE id = 5;
89
90 # 5) Create an after delete trigger that will insert a row to teacher log table when that row is de
  
```

SQLAdditions

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

Context Help Snippets

Output

#	Time	Action	Message	Duration / Fetc
7	21:08:16	CREATE TABLE teacher_log (log_id INT AUTO_INCREMENT PRIMARY KEY, te...	0 row(s) affected	0.032 sec
8	21:11:09	CREATE TRIGGER after_insert_teacher AFTER INSERT ON teachers FOR EACH...	0 row(s) affected	0.000 sec
9	21:11:26	CREATE TRIGGER after_insert_teacher AFTER INSERT ON teachers FOR EACH...	Error Code: 1359. Trigger already exists	0.000 sec
10	21:11:41	INSERT INTO teachers (id, name, subject, experience, salary) VALUES (9, 'Sibin Fr...	1 row(s) affected	0.016 sec
11	21:12:03	SELECT * FROM teacher_log LIMIT 0, 1000	1 row(s) returned	0.000 sec / 0.1
12	21:12:42	CREATE TRIGGER before_delete_teacher BEFORE DELETE ON teachers FOR E...	0 row(s) affected	0.015 sec

Object Info Session

Navigator: CHEMAS

Filter objects

- entri_dsmi_d36
 - Tables
 - teacher_log
 - teachers
 - Views
 - Stored Procedures
 - Functions
- product
 - Tables
 - Views
 - Stored Procedures
 - Functions
- sales

Administration Schemas

No object selected

Query 1 TRIGGERS

```

76 BEFORE DELETE ON teachers
77 FOR EACH ROW
78 BEGIN
79     IF OLD.experience > 10 THEN
80         SIGNAL SQLSTATE '45000'
81         SET MESSAGE_TEXT = 'Cannot delete teacher with more than 10 years of experience';
82     END IF;
83 END $$
84
85 DELIMITER ;
86
87
88 • DELETE FROM teachers WHERE id = 5;
89
  
```

SQLAdditions

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

Context Help Snippets

Output

#	Time	Action	Message	Duration / Fetc
9	21:11:26	CREATE TRIGGER after_insert_teacher AFTER INSERT ON teachers FOR EACH...	Error Code: 1359. Trigger already exists	0.000 sec
10	21:11:41	INSERT INTO teachers (id, name, subject, experience, salary) VALUES (9, 'Sibin Fr...	1 row(s) affected	0.016 sec
11	21:12:03	SELECT * FROM teacher_log LIMIT 0, 1000	1 row(s) returned	0.000 sec / 0.1
12	21:12:42	CREATE TRIGGER before_delete_teacher BEFORE DELETE ON teachers FOR E...	0 row(s) affected	0.015 sec
13	21:12:59	CREATE TRIGGER before_delete_teacher BEFORE DELETE ON teachers FOR E...	Error Code: 1359. Trigger already exists	0.000 sec
14	21:13:03	DELETE FROM teachers WHERE id = 5	Error Code: 1644. Cannot delete teacher with more than 10 years of expe	0.000 sec

Object Info Session

Query 1 TRIGGERS

```

87
88 • DELETE FROM teachers WHERE id = 5;
89
90 # 5) Create an after delete trigger that will insert a row to teacher_log table when that row is de
91
92 DELIMITER $$
93
94 • CREATE TRIGGER after_delete_teacher
95 AFTER DELETE ON teachers
96 FOR EACH ROW
97 BEGIN
98     INSERT INTO teacher_log (teacher_id, action, timestamp)
99     VALUES (OLD.id, 'DELETE', NOW());
100 END$$

```

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

#	Time	Action	Message	Duration / Fetch
10	21:11:41	INSERT INTO teachers (id, name, subject, experience, salary) VALUES (9, 'Sibin Fr...	1 row(s) affected	0.016 sec
11	21:12:03	SELECT * FROM teacher_log LIMIT 0, 1000	1 row(s) returned	0.000 sec / 0.000 se
12	21:12:42	CREATE TRIGGER before_delete_teacher BEFORE DELETE ON teachers FOR E...	0 row(s) affected	0.015 sec
13	21:12:59	CREATE TRIGGER before_delete_teacher BEFORE DELETE ON teachers FOR E...	Error Code: 1359. Trigger already exists	0.000 sec
14	21:13:03	DELETE FROM teachers WHERE id = 5	Error Code: 1644. Cannot delete teacher with more than 10 years of experience	0.016 sec
15	21:13:59	CREATE TRIGGER after_delete_teacher AFTER DELETE ON teachers FOR EAC...	0 row(s) affected	0.016 sec

Query 1 TRIGGERS

```

93
94 • CREATE TRIGGER after_delete_teacher
95 AFTER DELETE ON teachers
96 FOR EACH ROW
97 BEGIN
98     INSERT INTO teacher_log (teacher_id, action, timestamp)
99     VALUES (OLD.id, 'DELETE', NOW());
100 END$$
101
102 DELIMITER ;
103
104
105 • DELETE FROM teachers WHERE id = 7;
106

```

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

#	Time	Action	Message
11	21:12:03	SELECT * FROM teacher_log LIMIT 0, 1000	1 row(s) returned
12	21:12:42	CREATE TRIGGER before_delete_teacher BEFORE DELETE ON teachers FOR E...	0 row(s) affected
13	21:12:59	CREATE TRIGGER before_delete_teacher BEFORE DELETE ON teachers FOR E...	Error Code: 1359. Trigger already exists
14	21:13:03	DELETE FROM teachers WHERE id = 5	Error Code: 1644. Cannot delete teacher with more than 10 years of experience
15	21:13:59	CREATE TRIGGER after_delete_teacher AFTER DELETE ON teachers FOR EAC...	0 row(s) affected
16	21:14:18	DELETE FROM teachers WHERE id = 7	1 row(s) affected

HEMAS

Filter objects

entri_dsml_d36

- Tables
 - teacher_log
 - teachers
- Views
- Stored Procedures
- Functions

product

- Tables
- Views
- Stored Procedures
- Functions

sales

ministration

Schemas

Query 1 TRIGGERS

Limit to 1000 rows

103

104

105 • DELETE FROM teachers WHERE id = 7;

106

107 • SELECT * FROM teacher_log;

Result Grid

log_id	teacher_id	action	timestamp
1	9	INSERT	2024-09-26 21:11:41
2	7	DELETE	2024-09-26 21:14:18
NULL	NULL	NULL	NULL

teacher_log 3

Apply

Revert

Context Help

Shippe

SQLAdditions

Automatic disabled. Use manually g current care toggle au

No object selected

ject Info

Session

Output

Action Output

#	Time	Action	Message
✓ 12	21:12:42	CREATE TRIGGER before_delete_teacher BEFORE DELETE ON teachers FOR E...	0 row(s) affected
✓ 13	21:12:59	CREATE TRIGGER before_delete_teacher BEFORE DELETE ON teachers FOR E...	Error Code: 1359. Trigger already exists
✗ 14	21:13:03	DELETE FROM teachers WHERE id = 5	Error Code: 1644. Cannot delete teacher with more than 10 years of experience
✓ 15	21:13:59	CREATE TRIGGER after_delete_teacher AFTER DELETE ON teachers FOR EAC...	0 row(s) affected
✓ 16	21:14:18	DELETE FROM teachers WHERE id = 7	1 row(s) affected
✓ 17	21:14:36	SELECT * FROM teacher_log LIMIT 0, 1000	2 row(s) returned