

ASSIGNMENT 2

STUDENT MANAGEMENT SYSTEM

Task1:

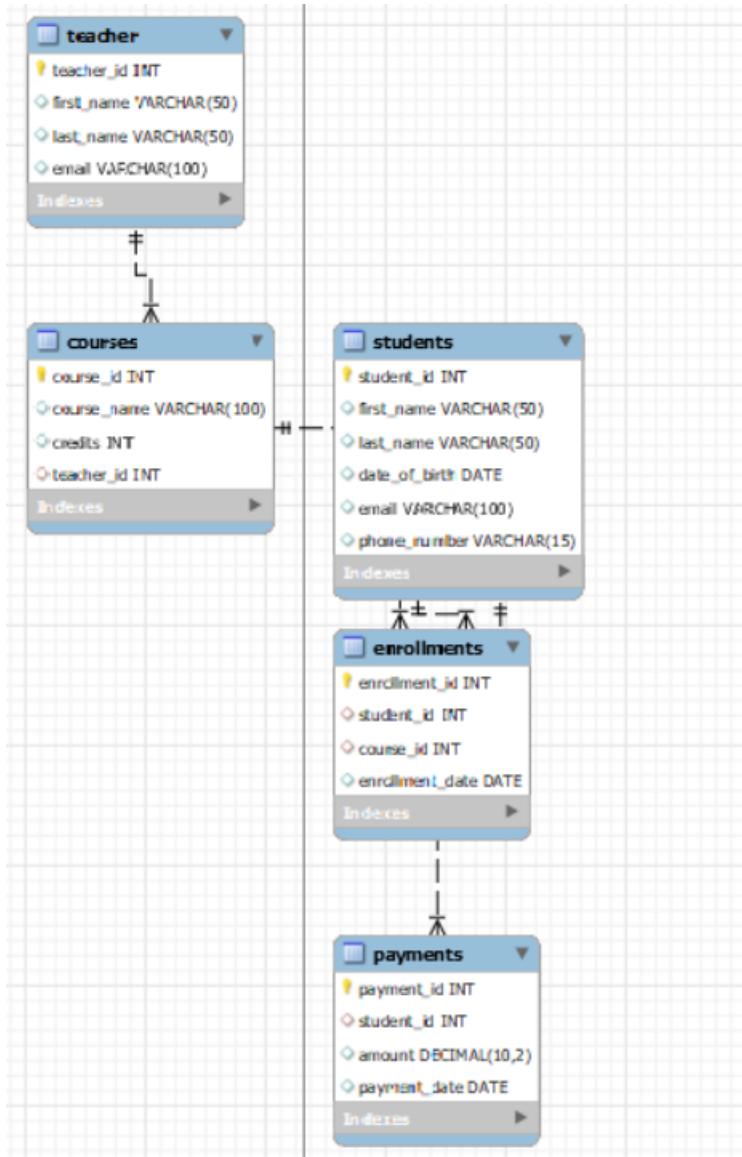
1,2&4.

The screenshot shows a MySQL Workbench interface with a query editor titled "task1.*". The code below is a script to create five tables: Students, Teacher, Courses, Enrollments, and Payments. The code uses MySQL syntax with primary keys, foreign keys, and various data types like INT, VARCHAR, DATE, and DECIMAL.

```
1  /* task1*/
2 • CREATE DATABASE SISDB;
3 • USE SISDB;
4
5 • CREATE TABLE Students (
6     student_id INT PRIMARY KEY,
7     first_name VARCHAR(50),
8     last_name VARCHAR(50),
9     date_of_birth DATE,
10    email VARCHAR(100),
11    phone_number VARCHAR(15)
12 );
13
14 • CREATE TABLE Teacher (
15     teacher_id INT PRIMARY KEY,
16     first_name VARCHAR(50),
17     last_name VARCHAR(50),
18     email VARCHAR(100)
19 );
20
21 • CREATE TABLE Courses (
22     course_id INT PRIMARY KEY,
23     course_name VARCHAR(100),
24     credits INT,
25     teacher_id INT,
26     FOREIGN KEY (teacher_id) REFERENCES Teacher(teacher_id)
27 );
28
29 • CREATE TABLE Enrollments (
30     enrollment_id INT PRIMARY KEY,
31     student_id INT,
32     course_id INT,
33     enrollment_date DATE,
34     FOREIGN KEY (student_id) REFERENCES Students(student_id) ON DELETE CASCADE,
35     FOREIGN KEY (course_id) REFERENCES Courses(course_id)
36 );
37
38 • CREATE TABLE Payments (
39     payment_id INT PRIMARY KEY,
40     student_id INT,
41     amount DECIMAL(10, 2),
42     payment_date DATE,
43     FOREIGN KEY (student_id) REFERENCES Students(student_id) ON DELETE SET NULL
```

Action Output		Message	Duration / Fetch
#	Action		
1	16:15:27 CREATE DATABASE SISDB	1 row(s) affected	0.015 sec
2	16:15:30 USE SISDB	0 row(s) affected	0.000 sec
3	16:15:32 CREATE TABLE Students (student_id INT PRIMARY KEY, first_name VARCHAR(50), last_name VARCHAR(50), date_of_birth DATE, credits INT, teacher_id INT) ENGINE=InnoDB	0 row(s) affected	0.047 sec
4	16:15:35 CREATE TABLE Teacher (teacher_id INT PRIMARY KEY, first_name VARCHAR(50), last_name VARCHAR(50), email VARCHAR(100)) ENGINE=InnoDB	0 row(s) affected	0.344 sec
5	16:15:40 CREATE TABLE Courses (course_id INT PRIMARY KEY, course_name VARCHAR(100), credits INT, teacher_id INT) ENGINE=InnoDB	0 row(s) affected	0.047 sec
6	16:15:44 CREATE TABLE Enrollments (enrollment_id INT PRIMARY KEY, student_id INT, course_id INT, enrollment_date DATE) ENGINE=InnoDB	0 row(s) affected	0.078 sec
7	16:15:47 CREATE TABLE Payments (payment_id INT PRIMARY KEY, student_id INT, amount DECIMAL(10,2), payment_date DATE) ENGINE=InnoDB	0 row(s) affected	0.063 sec

3.ERD Diagram.



5.

```
task1.2* 
INSERT INTO Student VALUES
46  (1, 'David', 'Curran', '1990-01-15', 'davidcurran@email.com', '23514269842'),
47  (2, 'Steve', 'Smith', '1992-05-20', 'jane.smith@email.com', '9876543210'),
48  (3, 'Michael', 'Johnson', '1991-08-10', 'michael.johnson@email.com', '551234567'),
49  (4, 'Sean', 'Williams', '1993-03-25', 'emily.williams@email.com', '1112223333'),
50  (5, 'Daniel', 'Vittori', '1989-11-02', 'daniel.brown@mail.com', '9988877777'),
51  (6, 'Olivia', 'Miller', '1995-04-12', 'olivia.miller@mail.com', '3334455555'),
52  (7, 'Ethan', 'Davis', '1994-09-18', 'ethan.davis@email.com', '6667778888'),
53  (8, 'Ava', 'Jones', '1992-06-30', 'ava.jones@email.com', '4445556666'),
54  (9, 'Logan', 'Paul', '1998-12-05', 'logan.anderson@email.com', '2223334444'),
55  (10, 'Sophia', 'Moore', '1993-02-28', 'sophia.moore@email.com', '8889990000');
56
57
58 ● INSERT INTO Teacher VALUES
59  (1, 'Professor', 'Smith', 'prof.smith@email.com'),
60  (2, 'Dr.', 'Johnson', 'dr.johnson@email.com'),
61  (3, 'Ms.', 'Williams', 'ms.williams@email.com'),
62  (4, 'Mr.', 'Davis', 'mr.davis@email.com'),
63  (5, 'Professor', 'Moore', 'prof.moore@email.com'),
64  (6, 'Dr.', 'Anderson', 'dr.anderson@email.com'),
65  (7, 'Mrs.', 'Brown', 'mrs.brown@email.com'),
66  (8, 'Ms.', 'Miller', 'ms.miller@email.com'),
67  (9, 'Mr.', 'Jones', 'mr.jones@email.com'),
68  (10, 'Mrs.', 'Doe', 'mrs.doe@email.com');
69
70
71 ● INSERT INTO Courses VALUES
72  (101, 'Introduction to Computer Science', 3, 1),
73  (102, 'Mathematics for Engineers', 4, 2),
74  (103, 'History of Art', 3, 3),
75  (104, 'Physics for Beginners', 4, 1),
76  (105, 'Business Ethics', 3, 2),
77  (106, 'Literature and Society', 3, 3),
78  (107, 'Chemistry Fundamentals', 4, 2),
79  (108, 'Psychology 101', 3, 3),
80  (109, 'Data Structures', 4, 1),
81  (110, 'Introduction to Marketing', 3, 2);
82
```

```
task1.2* 
INSERT INTO Enrollments VALUES
82  (1, 1, 101, '2023-01-01'),
83  (2, 2, 102, '2023-04-02'),
84  (3, 3, 103, '2023-03-03'),
85  (4, 4, 104, '2023-02-04'),
86  (5, 5, 105, '2023-10-05'),
87  (6, 6, 106, '2023-09-06'),
88  (7, 7, 107, '2023-02-07'),
89  (8, 8, 108, '2023-04-08'),
90  (9, 9, 109, '2023-01-09'),
91  (10, 10, 110, '2023-03-10'),
92  (11, 1, 107, '2023-08-05'),
93  (12, 8, 104, '2023-05-12');
94
95
96 ● INSERT INTO Payments VALUES
97  (1, 1, 500.00, '2023-01-01'),
98  (2, 2, 750.00, '2023-04-02'),
99  (3, 3, 600.00, '2023-03-03'),
100  (4, 4, 800.00, '2023-02-04'),
101  (5, 5, 550.00, '2023-10-05'),
102  (6, 6, 700.00, '2023-09-06'),
103  (7, 7, 850.00, '2023-02-07'),
104  (8, 8, 600.00, '2023-04-08'),
105  (9, 9, 700.00, '2023-01-09'),
106  (10, 10, 500.00, '2023-03-10'),
107  (11, 1, 500.00, '2023-08-01'),
108  (12, 8, 600.00, '2023-05-12');
109
```

Output			Message	Duration / Fetch
#	Time	Action		
1	16:23:19	INSERT INTO Students VALUES (1, 'David', 'Curran', '1990-01-15', 'davidcurran@email.com', '23514269842'), (2, ..., 10 row(s) affected Records: 10 Duplicates: 0 Warnings: 0		0.015 sec
2	16:23:31	INSERT INTO Teacher VALUES (1, 'Professor', 'Smith', 'prof.smith@email.com'), (2, 'Dr.', 'Johnson', 'dr.johnson@...', 10 row(s) affected Records: 10 Duplicates: 0 Warnings: 0		0.047 sec
3	16:23:34	INSERT INTO Courses VALUES (101, 'Introduction to Computer Science', 3, 1), (102, 'Mathematics for Engineers...', 4, 2), (103, 'History of Art', 3, 3), (104, 'Physics for Beginners', 4, 1), (105, 'Business Ethics', 3, 2), (106, 'Literature and Society', 3, 3), (107, 'Chemistry Fundamentals', 4, 2), (108, 'Psychology 101', 3, 3), (109, 'Data Structures', 4, 1), (110, 'Introduction to Marketing', 3, 2); 10 row(s) affected Records: 10 Duplicates: 0 Warnings: 0		0.016 sec
4	16:23:38	INSERT INTO Enrollments VALUES (1, 1, 101, '2023-01-01'), (2, 2, 102, '2023-04-02'), (3, 3, 103, '2023-03-03'), (4, 4, 104, '2023-02-04'), (5, 5, 105, '2023-10-05'), (6, 6, 106, '2023-09-06'), (7, 7, 107, '2023-02-07'), (8, 8, 108, '2023-04-08'), (9, 9, 109, '2023-01-09'), (10, 10, 110, '2023-03-10'), (11, 1, 107, '2023-08-05'), (12, 8, 104, '2023-05-12'); 12 row(s) affected Records: 12 Duplicates: 0 Warnings: 0		0.016 sec
5	16:23:42	INSERT INTO Payments VALUES (1, 1, 500.00, '2023-01-01'), (2, 2, 750.00, '2023-04-02'), (3, 3, 600.00, '2023-03-03'), (4, 4, 800.00, '2023-02-04'), (5, 5, 550.00, '2023-10-05'), (6, 6, 700.00, '2023-09-06'), (7, 7, 850.00, '2023-02-07'), (8, 8, 600.00, '2023-04-08'), (9, 9, 700.00, '2023-01-09'), (10, 10, 500.00, '2023-03-10'), (11, 1, 500.00, '2023-08-01'), (12, 8, 600.00, '2023-05-12'); 12 row(s) affected Records: 12 Duplicates: 0 Warnings: 0		0.000 sec

TASK2:

1.

The screenshot shows a SQL editor window titled "task1.2". The code pane contains the following SQL statement:

```
112  /*TASK-2*/
113
114
115 • INSERT INTO Students (student_id, first_name, last_name, date_of_birth, email, phone_number)
116   VALUES (11, 'John', 'Doe', '1995-08-15', 'john.doe@example.com', '1234567890');
117
118
119
```

The output pane shows the execution results:

#	Time	Action	Message	Duration / Fetch
1	16:25:24	INSERT INTO Students (student_id, first_name, last_name, date_of_birth, email, phone_number) VALUES (11, 'Jo...')	1 row(s) affected	0.015 sec

2.

The screenshot shows a SQL editor window titled "task1.2". The code pane contains the following SQL statement:

```
117
118  /*2*/
119 • INSERT INTO Enrollments (enrollment_id, student_id, course_id, enrollment_date)
120   VALUES (13,7, 109, '2023-12-10');
121
122
```

The output pane shows the execution results:

#	Time	Action	Message	Duration / Fetch
1	16:27:30	INSERT INTO Enrollments (enrollment_id, student_id, course_id, enrollment_date) VALUES (13,7, 109, '2023-12-1...')	1 row(s) affected	0.000 sec

3.

The screenshot shows a SQL editor window titled "task1.2". The code pane contains the following SQL statement:

```
126
127  /*3*/
128 • UPDATE Teacher
129   SET email = 'new.email@example.com'
130   WHERE teacher_id = 1;
131
```

The output pane shows the execution results:

#	Time	Action	Message	Duration / Fetch
1	16:28:22	UPDATE Teacher SET email = 'new.email@example.com' WHERE teacher_id = 1;	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0	0.000 sec

4.

A screenshot of a SQL IDE interface titled "task1.2". The main pane shows a sequence of numbered lines of SQL code. Lines 132 through 139 are blank. Line 140 starts with a comment /*4*/. Line 141 contains a DELETE statement: "DELETE FROM Enrollments WHERE student_id = 1 AND course_id = 101;". Lines 142 and 143 are blank. The status bar at the bottom right indicates "Duration / Fetch 0.016 sec".

```
132
133
134
135 /*4*/
136 • DELETE FROM Enrollments
137 WHERE student_id = 1 AND course_id = 101;
138
139
140
141
142
143 /*5*/
144 • UPDATE Courses
145 SET teacher_id = 2
146 WHERE course_id = 105;
```

5.

A screenshot of a SQL IDE interface titled "task1.2". The main pane shows a sequence of numbered lines of SQL code. Lines 140 through 147 are blank. Line 148 starts with a comment /*5*/. Line 149 contains an UPDATE statement: "UPDATE Courses SET teacher_id = 2 WHERE course_id = 105;". Lines 150 and 151 are blank. The status bar at the bottom right indicates "Duration / Fetch 0.000 sec".

```
140
141
142
143 /*5*/
144 • UPDATE Courses
145 SET teacher_id = 2
146 WHERE course_id = 105;
147
148
149
150
151
```

6.

A screenshot of a SQL IDE interface titled "task1.2". The code editor shows the following SQL statement:

```
150  
151  
152 /*6*/  
153 • DELETE FROM Students  
154 WHERE student_id = 4;  
155  
156  
157
```

The output window displays the results of the execution:

Action Output	Time	Action	Message	Duration / Fetch
1 16:31:27	DELETE FROM Students WHERE student_id = 4		1 row(s) affected	0.000 sec

7.

A screenshot of a SQL IDE interface titled "task1.2". The code editor shows the following SQL statement:

```
157  
158  
159 /*7*/  
160 • UPDATE Payments  
161 SET amount = 1500.00  
162 WHERE payment_id = 1;  
163  
164
```

The output window displays the results of the execution:

Action Output	Time	Action	Message	Duration / Fetch
1 16:32:12	UPDATE Payments SET amount = 1500.00 WHERE payment_id = 1		1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0	0.000 sec

TASK3:

1.

The screenshot shows the MySQL Workbench interface with a query editor titled 'task3'. The SQL code is as follows:

```
1  /*Task-3*/
2  /*1*/
3  • SELECT s.student_id, s.first_name, s.last_name, SUM(p.amount) AS total_payments
4  FROM Students s
5  INNER JOIN Payments p ON s.student_id = p.student_id
6  GROUP BY s.student_id, s.first_name, s.last_name;
7
```

The results grid displays the following data:

student_id	first_name	last_name	total_payments
1	David	Curran	2000.00
2	Steve	Smith	750.00
3	Michael	Johnson	600.00
5	Daniel	Vittori	550.00
6	Olivia	Miller	700.00
7	Ethan	Davis	850.00
8	Ava	Jones	1200.00
9	Logan	Paul	700.00
10	Sophia	Moore	500.00

The output pane shows the executed query and its duration:

```
1 16:33:20 SELECT s.student_id, s.first_name, s.last_name, SUM(p.amount) AS total_payments FROM Students s INNER J... 9 row(s) returned
```

Duration / Fetch: 0.000 sec / 0.000 sec

2.

The screenshot shows the MySQL Workbench interface with a query editor titled 'task3'. The SQL code is as follows:

```
7
8  /*2*/
9  • SELECT c.course_id, c.course_name, COUNT(e.student_id) AS enrolled_students_count
10 FROM Courses c
11 LEFT JOIN Enrollments e ON c.course_id = e.course_id
12 GROUP BY c.course_id, c.course_name;
13
```

The results grid displays the following data:

course_id	course_name	enrolled_students_count
101	Introduction to Computer Science	0
102	Mathematics for Engineers	1
103	History of Art	1
104	Physics for Beginners	1
105	Business Ethics	1
106	Literature and Society	1
107	Chemistry Fundamentals	2
108	Psychology 101	1
109	Data Structures	2

The output pane shows the executed query and its duration:

```
1 16:34:42 SELECT c.course_id, c.course_name, COUNT(e.student_id) AS enrolled_students_count FROM Courses c LEF... 10 row(s) returned
```

Duration / Fetch: 0.000 sec / 0.000 sec

3.

```
task3
13
14  /*3*/
15 •  SELECT s.first_name, s.last_name
16  FROM Students s
17  LEFT JOIN Enrollments e ON s.student_id = e.student_id
18  WHERE e.enrollment_id IS NULL;
19
```

first_name	last_name
John	Doe

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Result Grid | Form Editor | Read Only | Context Help | Snippets | Output | Action Output | # | Time | Action | Message | Duration / Fetch | 1 16:36:17 SELECT s.first_name, s.last_name FROM Students s LEFT JOIN Enrollments e ON s.student_id = e.student_id ... 1 row(s) returned | 0.000 sec / 0.000 sec

4.

```
task3
19
20  /*4*/
21 •  SELECT s.first_name, s.last_name, c.course_name
22  FROM Students s
23  JOIN Enrollments e ON s.student_id = e.student_id
24  JOIN Courses c ON e.course_id = c.course_id;
25
```

first_name	last_name	course_name
David	Curran	Chemistry Fundamentals
Steve	Smith	Mathematics for Engineers
Michael	Johnson	History of Art
Daniel	Vittori	Business Ethics
Olivia	Miller	Literature and Society
Ethan	Davis	Chemistry Fundamentals
Ethan	Davis	Data Structures
Ava	Jones	Psychology 101
Ava	Jones	Physics for Beginners

Result 5 | Filter Rows: | Export: | Wrap Cell Content: | Result Grid | Form Editor | Read Only | Context Help | Snippets | Output | Action Output | # | Time | Action | Message | Duration / Fetch | 1 16:36:54 SELECT s.first_name, s.last_name, c.course_name FROM Students s JOIN Enrollments e ON s.student_id = e.st... 11 row(s) returned | 0.000 sec / 0.000 sec

5.

The screenshot shows a SQL query editor window titled "task3". The query is:

```
26
27  /*5*/
28 • SELECT t.first_name, t.last_name, c.course_name
29   FROM Teacher t
30 JOIN Courses c ON t.teacher_id = c.teacher_id;
31
32
```

The results grid displays the following data:

	first_name	last_name	course_name
▶	Professor	Smith	Introduction to Computer Science
▶	Professor	Smith	Physics for Beginners
▶	Professor	Smith	Data Structures
▶	Dr.	Johnson	Mathematics for Engineers
▶	Dr.	Johnson	Business Ethics
▶	Dr.	Johnson	Chemistry Fundamentals
▶	Dr.	Johnson	Introduction to Marketing
▶	Ms.	Williams	History of Art
▶	Ms.	Williams	Literature and Society

The output pane shows the following log entry:

#	Time	Action	Message	Duration / Fetch
1	16:37:53	SELECT t.first_name, t.last_name, c.course_name FROM Teacher t JOIN Courses c ON t.teacher_id = c.teacher_id;	10 row(s) returned	0.000 sec / 0.000 sec

6.

The screenshot shows a SQL query editor window titled "task3". The query is:

```
33
34  /*6*/
35 • SELECT s.first_name, s.last_name, e.enrollment_date
36   FROM Students s
37 JOIN Enrollments e ON s.student_id = e.student_id;
38
39
```

The results grid displays the following data:

	first_name	last_name	enrollment_date
▶	David	Curran	2023-08-05
▶	Steve	Smith	2023-04-02
▶	Michael	Johnson	2023-03-03
▶	Daniel	Vittor	2023-10-05
▶	Olivia	Miller	2023-09-06
▶	Ethan	Davis	2023-02-07
▶	Ethan	Davis	2023-12-10
▶	Ava	Jones	2023-04-08
▶	Ava	Jones	2023-05-12

The output pane shows the following log entry:

#	Time	Action	Message	Duration / Fetch
1	16:38:28	SELECT s.first_name, s.last_name, e.enrollment_date FROM Students s JOIN Enrollments e ON s.student_id = e.student_id;	11 row(s) returned	0.000 sec / 0.000 sec

7.

The screenshot shows the SQL Server Management Studio interface with a query window titled 'task3'. The query is:

```
39
40  /*7*/
41 •  SELECT s.first_name, s.last_name
42  FROM Students s
43  LEFT JOIN Payments p ON s.student_id = p.student_id
44  WHERE p.payment_id IS NULL;
45
```

The results grid shows one row:

first_name	last_name
John	Doe

The output pane shows two actions:

#	Time	Action	Message	Duration / Fetch
1	16:38:28	SELECT s.first_name, s.last_name, e.enrollment_date FROM Students s JOIN Enrollments e ON s.student_id = e...	11 row(s) returned	0.000 sec / 0.000 sec
2	16:39:11	SELECT s.first_name, s.last_name FROM Students s LEFT JOIN Payments p ON s.student_id = p.student_id W...	1 row(s) returned	0.000 sec / 0.000 sec

8.

The screenshot shows the SQL Server Management Studio interface with a query window titled 'task3'. The query is:

```
45
46  /*8*/
47 •  SELECT c.course_id, c.course_name
48  FROM Courses c
49  LEFT JOIN Enrollments e ON c.course_id = e.course_id
50  WHERE e.enrollment_id IS NULL;
51
```

The results grid shows one row:

course_id	course_name
101	Introduction to Computer Science

The output pane shows one action:

#	Time	Action	Message	Duration / Fetch
1	16:39:52	SELECT c.course_id, c.course_name FROM Courses c LEFT JOIN Enrollments e ON c.course_id = e.course_id ...	1 row(s) returned	0.000 sec / 0.000 sec

9.

The screenshot shows the SQL Server Management Studio interface with a query window titled 'task3'. The code is as follows:

```
51
52  /*9*/
53 •  SELECT e1.student_id, count(e1.student_id) AS no_of_enrollments
54  FROM Enrollments e1
55  JOIN Enrollments e2 ON e1.student_id = e2.student_id AND e1.enrollment_id <> e2.enrollment_id
56  GROUP BY e1.student_id HAVING COUNT(DISTINCT e2.course_id) > 1;
```

The result grid shows two rows:

student_id	no_of_enrollments
7	2
8	2

The output pane shows the execution details:

#	Time	Action	Message	Duration / Fetch
1	16:40:38	SELECT e1.student_id, count(e1.student_id) AS no_of_enrollments FROM Enrollments e1 JOIN Enrollments e2 ...	2 row(s) returned	0.000 sec / 0.000 sec

10.

The screenshot shows the SQL Server Management Studio interface with a query window titled 'task3'. The code is as follows:

```
58
59  /*10*/
60 •  SELECT t.teacher_id, t.first_name, t.last_name
61  FROM Teacher t
62  LEFT JOIN Courses c ON t.teacher_id = c.teacher_id
63  WHERE c.course_id IS NULL;
```

The result grid shows ten rows:

teacher_id	first_name	last_name
4	Mr.	Davis
5	Professor	Moore
6	Dr.	Anderson
7	Mrs.	Brown
8	Ms.	Miller
9	Mr.	Jones
10	Mrs.	Doe

The output pane shows the execution details:

#	Time	Action	Message	Duration / Fetch
1	16:41:10	SELECT t.teacher_id, t.first_name, t.last_name FROM Teacher t LEFT JOIN Courses c ON t.teacher_id = c.teac...	7 row(s) returned	0.000 sec / 0.000 sec