

عنوان المشروع: SQL 3

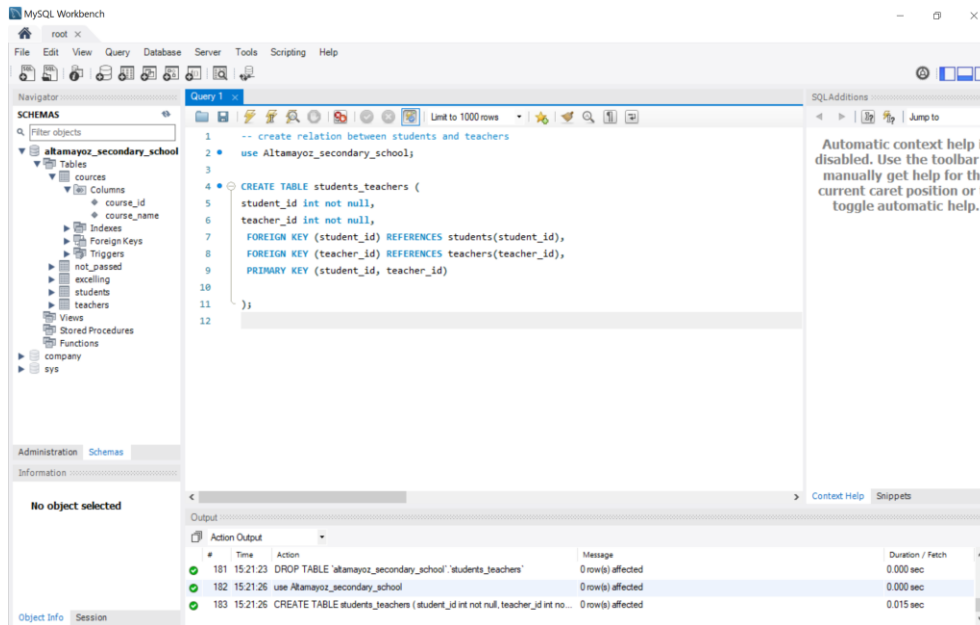
اسم المتدربة: البندري القصيمي

اسم المجموعة: القدية

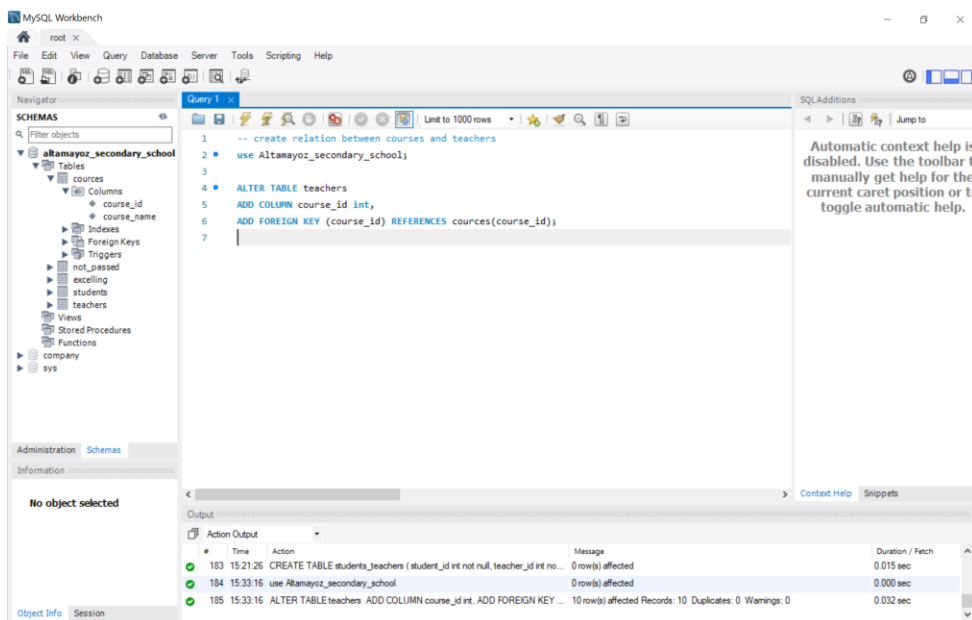
اسم المشرفة: نوره عبدالله

وصف المشروع: مشروع مكمل لمشروع دورة SQL2

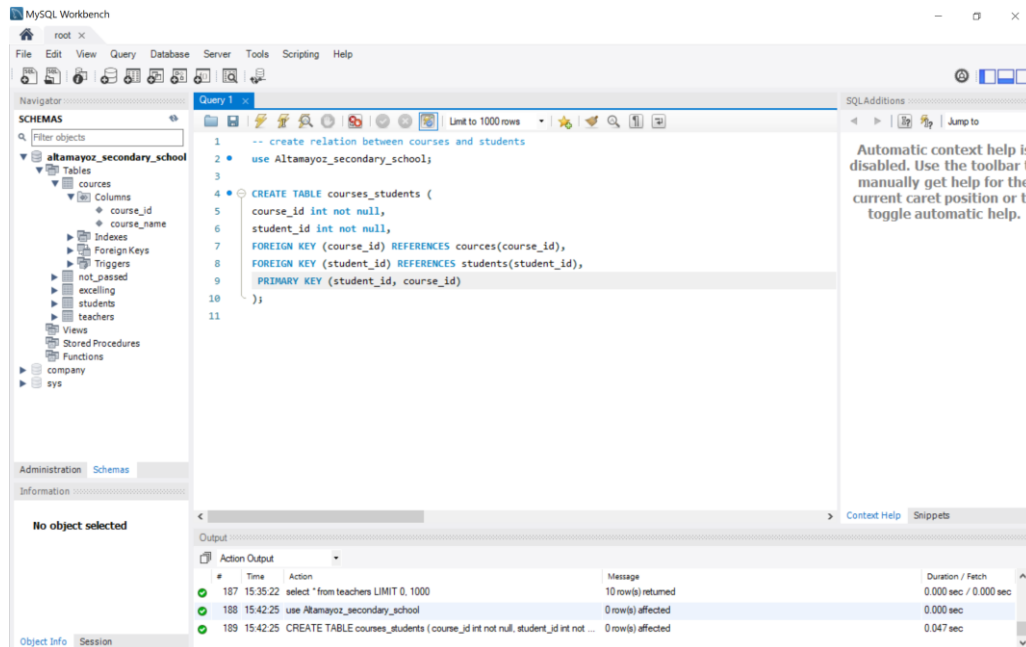
- إنشاء علاقة بين جدول المعلمين والطلاب (بحيث أن المعلم يدرّس أكثر من طالب، والطالب يقوم بتدريسه أكثر من معلم)



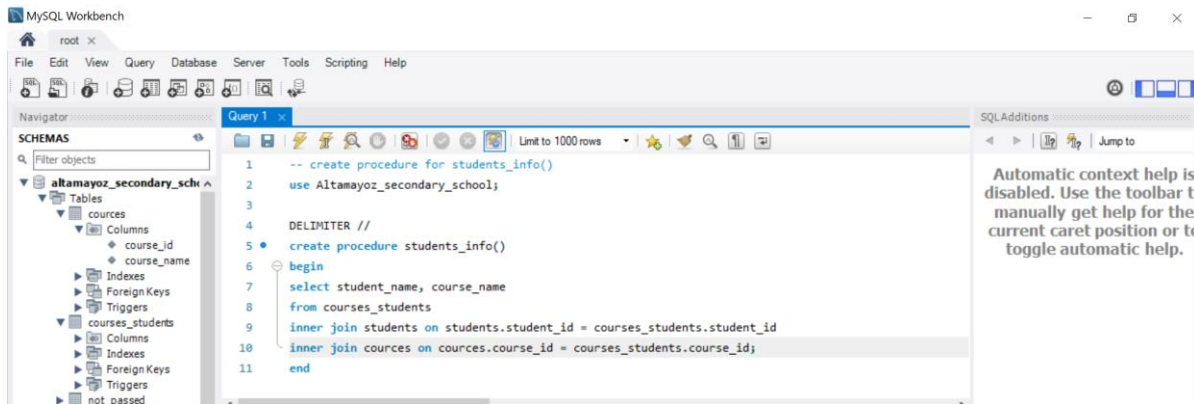
- إنشاء علاقة بين جدول المواد والمعلمين (بحيث أن المعلم يقوم بتدريس مادة واحدة فقط، والمادة يقوم بتدريسها أكثر من معلم).



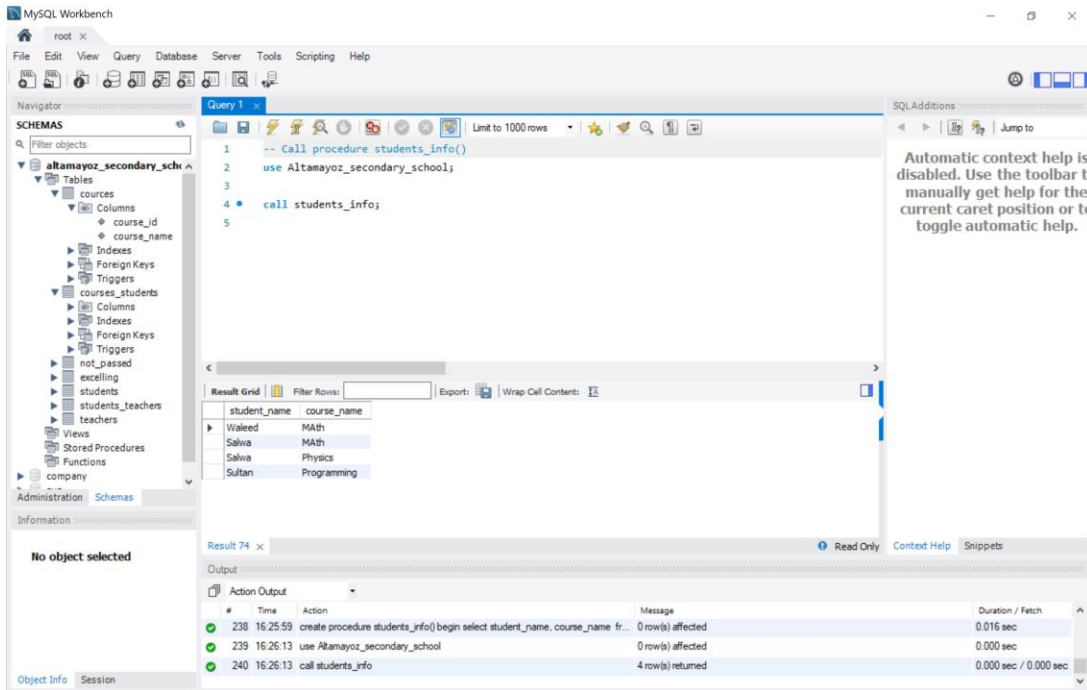
- إنشاء علاقة بين جدول المواد والطلاب (بحيث أن الطالب يدرس أكثر مادة، والمادة يدرسها أكثر من طالب).



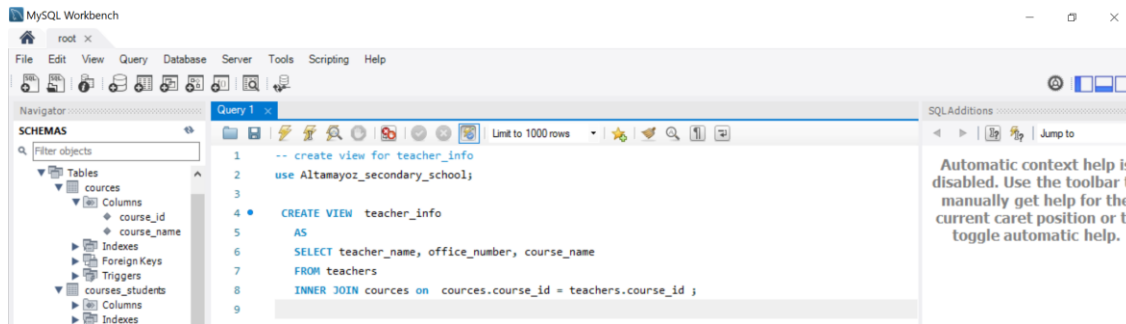
- قم بإنشاء Procedure باسم student_info يعرض اسماء الطلاب و المواد يحتوي على جميع البيانات المشتركة بين جدول المواد و الطلاب



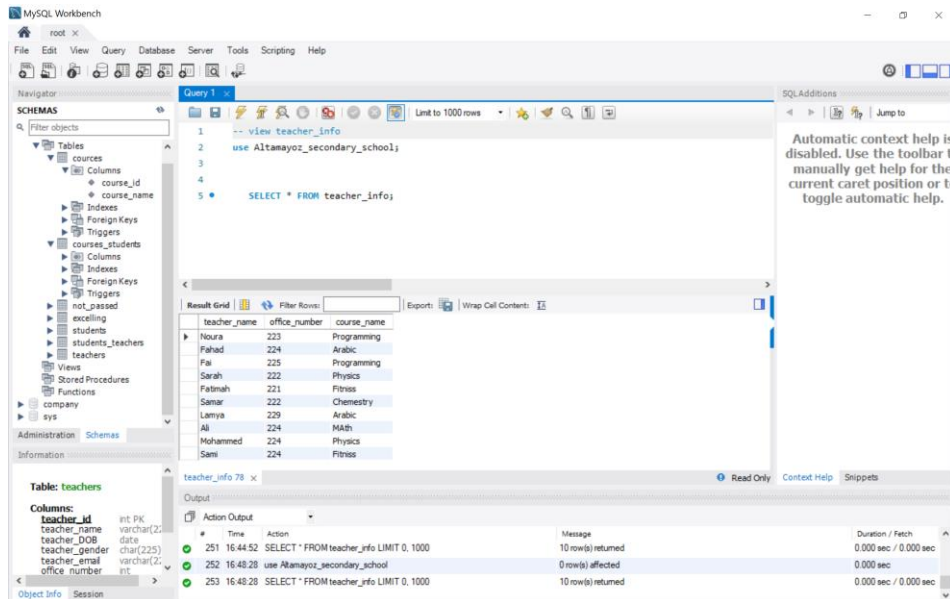
○ قم باستدعائها.



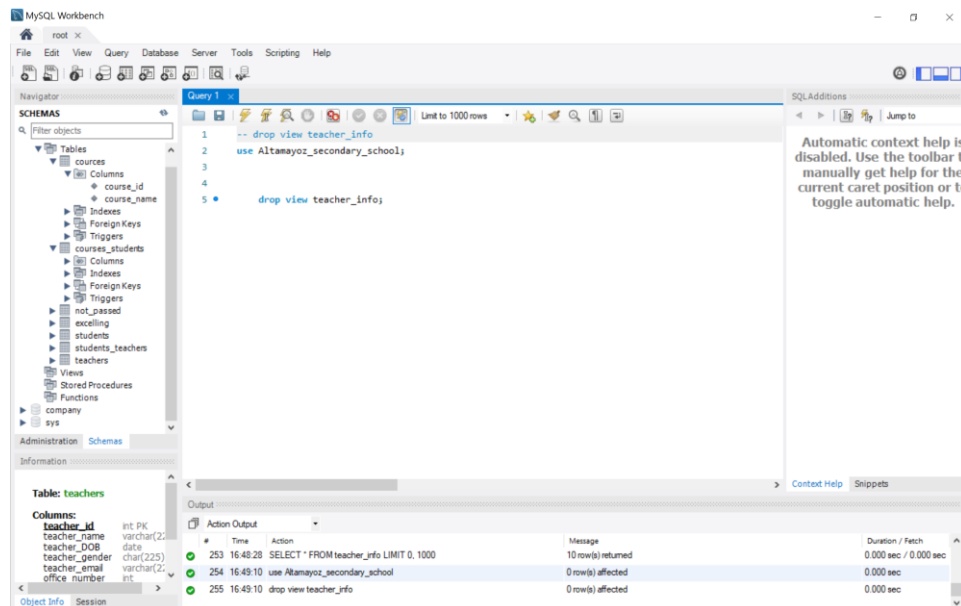
● قم بإنشاء view باسم teacher_info يحتوي على اسم المعلم و رقم المكتب و اسم المادة التي يتم تدريسها.



○ قم بعرض view



○ قم بحذف view



- قم بإنشاء index للبحث باستخدام أسماء الطلاب ابجدياً.

The screenshot shows the MySQL Workbench interface. In the left sidebar, the 'Schemas' panel is open, showing a tree view of the database structure. The 'students' table is selected under the 'Altamayoz_secondary_school' schema. The 'Columns' panel for the 'students' table is visible, showing columns like student_id, student_name, student_DOB, etc. The main query editor contains the following SQL code:

```

1  -- create search index
2  use Altamayoz_secondary_school;
3
4
5  CREATE INDEX students_search
6  ON students(student_name ASC);

```

The 'Output' panel at the bottom shows the execution results of the queries:

#	Time	Action	Message	Duration / Fetch
255	16:49:10	drop view teacher_info	0 row(s) affected	0.000 sec
256	16:51:11	use Altamayoz_secondary_school	0 row(s) affected	0.000 sec
257	16:51:11	CREATE INDEX students_search ON students(student_name ASC)	30 row(s) affected Records: 30 Duplicates: 0 Warnings: 0	0.016 sec

- قم بعرض index

The screenshot shows the MySQL Workbench interface. In the left sidebar, the 'Schemas' panel is open, showing a tree view of the database structure. The 'students' table is selected under the 'Altamayoz_secondary_school' schema. The 'Columns' panel for the 'students' table is visible, showing columns like student_id, student_name, student_DOB, etc. The main query editor contains the following SQL code:

```

1  -- show search index
2  use Altamayoz_secondary_school;
3
4
5  show index from students;
6

```

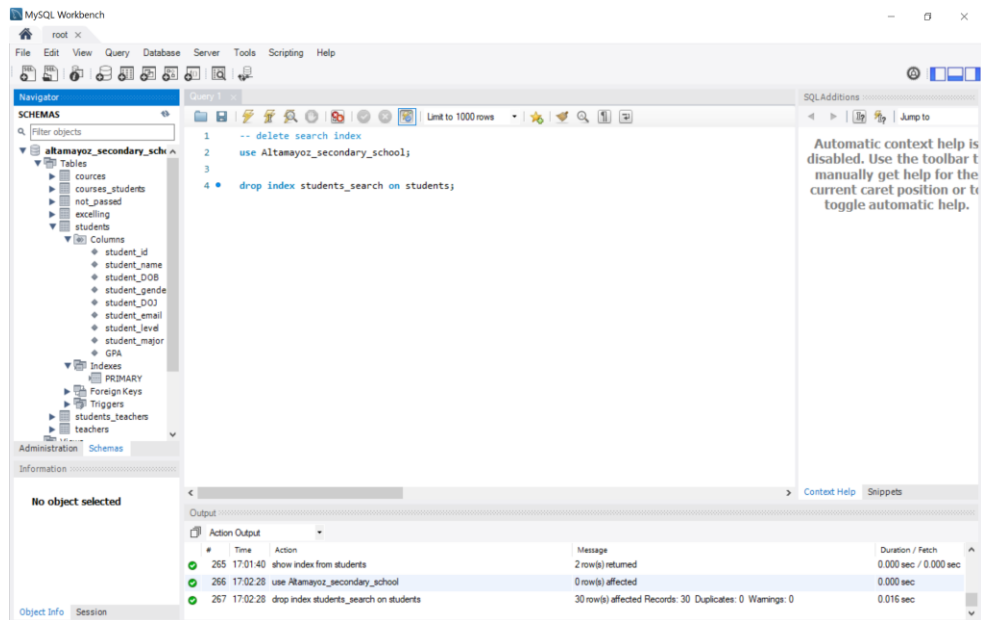
The 'Result Grid' panel shows the output of the 'show index from students' query:

Table	Non_unique	Key_name	Seq_in_index	Column_name	Collation	Cardinality	Sub_part	Packed	Null	Index_type
students	0	PRIMARY	1	student_id	A	30				BTREE
students	1	students_search	1	student_name	A				YES	BTREE

The 'Output' panel at the bottom shows the execution results of the queries:

#	Time	Action	Message	Duration / Fetch
263	16:54:32	show index from students	2 row(s) returned	0.000 sec / 0.000 sec
264	17:01:40	use Altamayoz_secondary_school	0 row(s) affected	0.000 sec
265	17:01:40	show index from students	2 row(s) returned	0.000 sec / 0.000 sec

○ قم بحذف index



Source code:

```
-- create relation between students and teachers

use Altamayoz_secondary_school;

CREATE TABLE students_teachers (

student_id int not null,

teacher_id int not null,

FOREIGN KEY (student_id) REFERENCES students(student_id),

FOREIGN KEY (teacher_id) REFERENCES teachers(teacher_id)

PRIMARY KEY (student_id, teacher_id)

);


-- create relation between courses and teachers

use Altamayoz_secondary_school;

ALTER TABLE teachers

ADD COLUMN course_id int,

ADD FOREIGN KEY (course_id) REFERENCES courses(course_id);


-- create relation between courses and teachers

use Altamayoz_secondary_school;

CREATE TABLE courses_teachers (

course_id int not null,

student_id int not null,

Constraint FOREIGN KEY (course_id) REFERENCES students(course_id),

Constraint FOREIGN KEY (student_id) REFERENCES teachers(student_id)

PRIMARY KEY (student_id, course_id)

);
```



```
--create procedure
use Altamayoz_secondary_school;
DELIMITER //
create procedure students_info()
begin
select student_name, course_name
from courses_students
inner join students on students.student_id = courses_students.student_id
inner join courses on courses.course_id = courses_students.course_id;
end
```

```
call students_info;
```

```
--Create view
CREATE VIEW teacher_info
AS
SELECT teacher_name, office_number, course_name
FROM teachers
INNER JOIN courses on courses.course_id = teachers.course_id;
SELECT * FROM teacher_info;
```

```
drop view teacher_info;
```

```
-- create search index
use Altamayoz_secondary_school;
```

```
CREATE INDEX students_search
ON students(student_name);
```

show index from students;

drop index students_search on students;