

Student ID Reference

MySQL Workbench

Migration x Test2 x

File Edit View Query Database Server Tools Scripting Help

Navigator Query 1 student grades Limit to 1000 rows

SCHEMAS Filter

1 • SELECT * FROM databaseproject.student;

Result Grid Filter Rows: Edit: Export/Import: Wrap Cell Contents:

student_id	f_name	l_name	major
1	John	Doe	Computer Science
2	Jane	Smith	Mathematics
3	David	Brown	Physics
4	Emily	Johnson	Biology
5	Michael	Williams	Chemistry
6	Sophia	Jones	History
7	Daniel	Brown	Biology
8	Olivia	Davis	Chemistry
9	William	Miller	History
10	Emma	Wilson	Biology
11	Alexan...	Martinez	Chemistry
12	Mia	Taylor	History
13	Quincy	Quinn	Computer Science
* NULL	NULL	NULL	NULL

student 1 x Apply Revert Context Help Snippets

Output

SQLAdditions

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

Course ID Reference

MySQL Workbench

Migration x Test2 x

File Edit View Query Database Server Tools Scripting Help

Navigator Query 1 student grades assignment course Limit to 1000 rows

SCHEMAS Filter objects

blueberryapple databaseproject Tables assignment course department distribution enrollment grades student Administration Schemas

Table: course

Columns: course_id int PK name varchar course_name varchar semester int department_id int course_number int year int

Result Grid Filter Rows: Edit: Export/Import: Wrap Cell Contents:

course_id	name	course_name	semester	department_id	course_number	year
101	CS101	Introduction to Computer Science	Fall	1	101	2023
102	MATH201	Calculus I	Spring	2	201	2024
103	PHY301	Mechanics	Fall	3	301	2023
104	BIO101	Introduction to Biology	Fall	4	101	2023
105	CHEM201	Organic Chemistry	Spring	5	201	2024
106	HIST301	World History	Fall	6	301	2023
* NULL	NULL	NULL	NULL	NULL	NULL	NULL

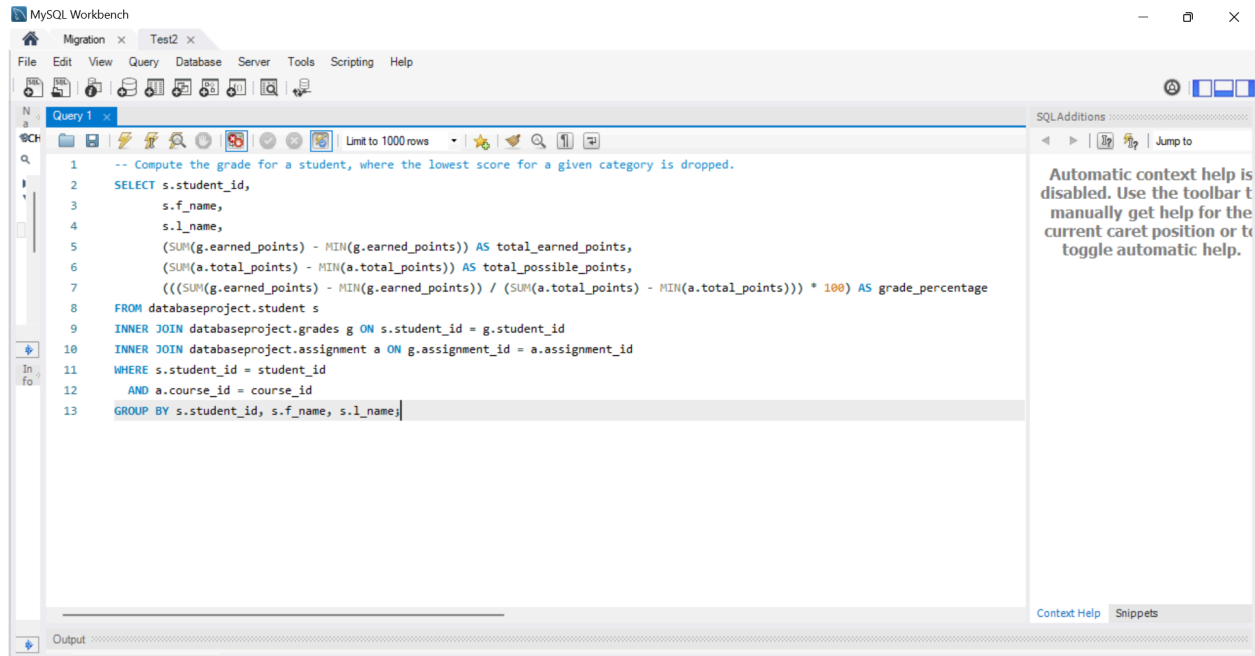
course 1 x Apply Revert Context Help Snippets

Output

SQLAdditions

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

SQL CODE TO BE EXECUTED

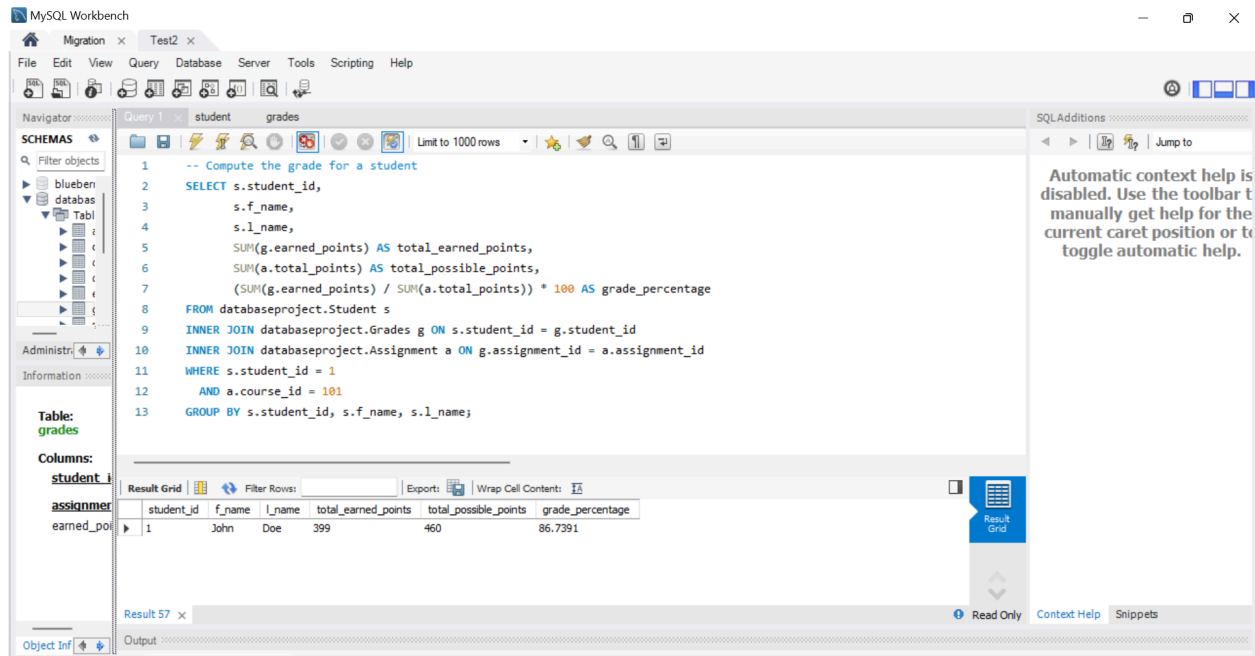


The screenshot shows the MySQL Workbench interface with a query editor. The query is designed to calculate the grade percentage for a student, dropping the lowest score. The SQL code is as follows:

```
1 -- Compute the grade for a student, where the lowest score for a given category is dropped.
2 SELECT s.student_id,
3        s.f_name,
4        s.l_name,
5        (SUM(g.earned_points) - MIN(g.earned_points)) AS total_earned_points,
6        (SUM(a.total_points) - MIN(a.total_points)) AS total_possible_points,
7        (((SUM(g.earned_points) - MIN(g.earned_points)) / (SUM(a.total_points) - MIN(a.total_points))) * 100) AS grade_percentage
8 FROM databaseproject.student s
9 INNER JOIN databaseproject.grades g ON s.student_id = g.student_id
10 INNER JOIN databaseproject.assignment a ON g.assignment_id = a.assignment_id
11 WHERE s.student_id = student_id
12 AND a.course_id = course_id
13 GROUP BY s.student_id, s.f_name, s.l_name;
```

On the right side, there is a help message: "Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help."

TEST CASE #1



The screenshot shows the MySQL Workbench interface after executing the query. The query editor displays the same SQL code as in the previous screenshot. The output grid shows the results of the query:

student_id	f_name	l_name	total_earned_points	total_possible_points	grade_percentage
1	John	Doe	399	460	86.7391

On the left side, the Navigator pane shows the database structure, including the 'student' and 'grades' tables. The 'Table: grades' is selected, and its columns are listed: student_id, f_name, l_name, total_earned_points, total_possible_points, and grade_percentage. The 'Result Grid' tab is active, showing the query results. The status bar at the bottom indicates 'Result 57 x' and 'Read Only'.

TEST CASE #1 RESULT (after SQL code execution)

MySQL Workbench interface showing the execution of a SQL query. The query is designed to calculate the grade percentage for a student based on their total earned points and total possible points, considering the minimum values for each category.

```
1 -- Compute the grade for a student, where the lowest score for a given category is dropped.
2 SELECT s.student_id,
3        s.f_name,
4        s.l_name,
5        (SUM(g.earned_points) - MIN(g.earned_points)) AS total_earned_points,
6        (SUM(a.total_points) - MIN(a.total_points)) AS total_possible_points,
7        ((SUM(g.earned_points) - MIN(g.earned_points)) / (SUM(a.total_points) - MIN(a.total_points))) * 100 AS grade_percentage
8 FROM databaseproject.student s
9 INNER JOIN databaseproject.grades g ON s.student_id = g.student_id
10 INNER JOIN databaseproject.assignment a ON g.assignment_id = a.assignment_id
11 WHERE s.student_id = 1
12 AND a.course_id = 101
13 GROUP BY s.student_id, s.f_name, s.l_name;
```

The result grid displays the following data:

student_id	f_name	l_name	total_earned_points	total_possible_points	grade_percentage
1	John	Doe	397	450	88.2222

TEST CASE #2

MySQL Workbench interface showing the execution of a SQL query. The query is designed to calculate the grade percentage for a student based on their total earned points and total possible points, considering the minimum values for each category.

```
1 -- Compute the grade for a student
2 SELECT s.student_id,
3        s.f_name,
4        s.l_name,
5        SUM(g.earned_points) AS total_earned_points,
6        SUM(a.total_points) AS total_possible_points,
7        (SUM(g.earned_points) / SUM(a.total_points)) * 100 AS grade_percentage
8 FROM databaseproject.Student s
9 INNER JOIN databaseproject.Grades g ON s.student_id = g.student_id
10 INNER JOIN databaseproject.Assignment a ON g.assignment_id = a.assignment_id
11 WHERE s.student_id = 2
12 AND a.course_id = 102
13 GROUP BY s.student_id, s.f_name, s.l_name;
```

The result grid displays the following data:

student_id	f_name	l_name	total_earned_points	total_possible_points	grade_percentage
2	Jane	Smith	74	300	24.6667

TEST CASE #2 RESULT (after SQL code execution)

MySQL Workbench

Migration x Test2 x

File Edit View Query Database Server Tools Scripting Help

Query 1 x

```
1 -- Compute the grade for a student, where the lowest score for a given category is dropped.
2 SELECT s.student_id,
3        s.f_name,
4        s.l_name,
5        (SUM(g.earned_points) - MIN(g.earned_points)) AS total_earned_points,
6        (SUM(a.total_points) - MIN(a.total_points)) AS total_possible_points,
7        (((SUM(g.earned_points) - MIN(g.earned_points)) / (SUM(a.total_points) - MIN(a.total_points))) * 100) AS grade_percentage
8 FROM databaseproject.student s
9 INNER JOIN databaseproject.grades g ON s.student_id = g.student_id
10 INNER JOIN databaseproject.assignment a ON g.assignment_id = a.assignment_id
11 WHERE s.student_id = 2
12 AND a.course_id = 102
13 GROUP BY s.student_id, s.f_name, s.l_name;
```

Result Grid

student_id	f_name	l_name	total_earned_points	total_possible_points	grade_percentage
2	Jane	Smith	62	250	24.8000

Result 61 x

Output

SQLAdditions

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

TEST CASE #3

MySQL Workbench

Migration x Test2 x

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

blueben

databases

Table

Administr

Information

Table: grades

Columns:

student_id

assignment_id

earned_points

Query 1 x

```
1 -- Compute the grade for a student
2 SELECT s.student_id,
3        s.f_name,
4        s.l_name,
5        SUM(g.earned_points) AS total_earned_points,
6        SUM(a.total_points) AS total_possible_points,
7        (SUM(g.earned_points) / SUM(a.total_points)) * 100 AS grade_percentage
8 FROM databaseproject.Student s
9 INNER JOIN databaseproject.Grades g ON s.student_id = g.student_id
10 INNER JOIN databaseproject.Assignment a ON g.assignment_id = a.assignment_id
11 WHERE s.student_id = 3
12 AND a.course_id = 103
13 GROUP BY s.student_id, s.f_name, s.l_name;
```

Result Grid

student_id	f_name	l_name	total_earned_points	total_possible_points	grade_percentage
3	David	Brown	130	400	32.5000

Result 59 x

Output

SQLAdditions

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

TEST CASE #3 RESULT (after SQL code execution)

The screenshot shows the MySQL Workbench interface. The main window displays a SQL query in the 'Query 1' tab. The query is designed to calculate the grade percentage for a student, where the lowest score for a given category is dropped. The query uses a subquery to calculate the total earned points and total possible points, then calculates the grade percentage based on these values. The query is executed, and the result is displayed in the 'Result Grid' tab. The result shows a single row for student ID 3, with a grade percentage of 34.2857.

MySQL Workbench

Migration x Test2 x

File Edit View Query Database Server Tools Scripting Help

Query 1 x

Limit to 1000 rows

```
1 -- Compute the grade for a student, where the lowest score for a given category is dropped.
2 SELECT s.student_id,
3        s.f_name,
4        s.l_name,
5        (SUM(g.earned_points) - MIN(g.earned_points)) AS total_earned_points,
6        (SUM(a.total_points) - MIN(a.total_points)) AS total_possible_points,
7        (((SUM(g.earned_points) - MIN(g.earned_points)) / (SUM(a.total_points) - MIN(a.total_points))) * 100) AS grade_percentage
8 FROM databaseproject.student s
9 INNER JOIN databaseproject.grades g ON s.student_id = g.student_id
10 INNER JOIN databaseproject.assignment a ON g.assignment_id = a.assignment_id
11 WHERE s.student_id = 3
12 AND a.course_id = 103
13 GROUP BY s.student_id, s.f_name, s.l_name;
```

SQLAdditions

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: | Export: | Wrap Cell Content: |

student_id	f_name	l_name	total_earned_points	total_possible_points	grade_percentage
3	David	Brown	120	350	34.2857

Result 62 x

Context Help Snippets

Output