

MySQL Workbench

Jatin

File Edit View Query Database Server Tools Scripting Help

Navigator

PROJECT 2 PROJECT 1 FINAL PROJECT PROJECT 3*

Filter objects

company_db datadigger labwork_1_3 labwork_2_1 labwork_2_2 labwork_2_3 labwork_4_1 labwork_4_2 project_2 project_final Tables Views Stored Procedures Functions student sys

SCHEMAS

-- 3. REQUIRED QUERIES
-- =====
-- Query 1: Perform CRUD Operations on all tables
-- CREATE operations (INSERT) - already done above
--
-- READ operations (SELECT)
• SELECT * FROM Students;
• SELECT * FROM Courses;
• SELECT * FROM Instructors;
• SELECT * FROM Enrollments;
• SELECT * FROM Departments;
--
-- UPDATE operations
• UPDATE Students SET Email = 'john.updated@email.com' WHERE StudentID = 1;
• UPDATE Courses SET Credits = 5 WHERE CourseID = 102;
--
-- DELETE operations (with caution - respecting foreign key constraints)
-- Example: Delete a specific enrollment
• DELETE FROM Enrollments WHERE EnrollmentID = 1;

Administration Schemas

Information

No object selected

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content: |

	StudentID	FirstName	LastName	Email	BirthDate	EnrollmentDate
▶	1	John	Doe	john.doe@email.com	2000-01-15	2022-08-01
▶	2	Jane	Smith	jane.smith@email.com	1999-05-25	2021-08-01
*	NULL	NULL	NULL	NULL	NULL	NULL

Students 1 x Apply Revert

Object Info Session Output

7 Finance headline India reported 6... Search

ENG IN 18:05 22-11-2025

MySQL Workbench

Jatin

File Edit View Query Database Server Tools Scripting Help

Navigator

PROJECT 2 PROJECT 1 FINAL PROJECT PROJECT 3*

SCHEMAS

Filter objects

- company_db
- datadigger
- labwork_1_3
- labwork_2_1
- labwork_2_2
- labwork_2_3
- labwork_4_1
- labwork_4_2
- project_2
- project_final
 - Tables
 - Views
 - Stored Procedures
 - Functions
- student
- sys

-- 3. REQUIRED QUERIES
-- Execute the selected portion of the script or everything, if there is no selection =

-- Query 1: Perform CRUD Operations on all tables
-- CREATE operations (INSERT) - already done above

-- READ operations (SELECT)
• SELECT * FROM Students;
• SELECT * FROM Courses;
• SELECT * FROM Instructors;
• SELECT * FROM Enrollments;
• SELECT * FROM Departments;

-- UPDATE operations
• UPDATE Students SET Email = 'john.updated@email.com' WHERE StudentID = 1;
• UPDATE Courses SET Credits = 5 WHERE CourseID = 102;

-- DELETE operations (with caution - respecting foreign key constraints)
-- Example: Delete a specific enrollment
• DELETE FROM Enrollments WHERE EnrollmentID = 1;

Administration Schemas

Information

No object selected

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content: |

CourseID	CourseName	DepartmentID	Credits
101	Introduction to SQL	1	3
102	Data Structures	2	4
*	HULL	HULL	HULL

Courses 2

Object Info Session Output

7 Finance headline India reported 6... Search

ENG IN 18:05 22-11-2025

MySQL Workbench

Jatin

File Edit View Query Database Server Tools Scripting Help

Navigator

PROJECT 2 PROJECT 1 FINAL PROJECT PROJECT 3*

Filter objects

company_db datadigger labwork_1_3 labwork_2_1 labwork_2_2 labwork_2_3 labwork_4_1 labwork_4_2 project_2 project_final Tables Views Stored Procedures Functions student sys

100 -- 3. REQUIRED QUERIES
101 -- =====
102
103 -- Query 1: Perform CRUD Operations on all tables
104 -- CREATE operations (INSERT) - already done above
105
106 -- READ operations (SELECT)
107 • SELECT * FROM Students;
108 • SELECT * FROM Courses;
109 • SELECT * FROM Instructors;
110 • SELECT * FROM Enrollments;
111 • SELECT * FROM Departments;
112
113 -- UPDATE operations
114 • UPDATE Students SET Email = 'john.updated@email.com' WHERE StudentID = 1;
115 • UPDATE Courses SET Credits = 5 WHERE CourseID = 102;
116
117 -- DELETE operations (with caution - respecting foreign key constraints)
118 -- Example: Delete a specific enrollment
119 • DELETE FROM Enrollments WHERE EnrollmentID = 1;

Administration Schemas

Information

No object selected

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content: |

	EnrollmentID	StudentID	CourseID	EnrollmentDate
▶	1	1	101	2022-08-01
▶	2	2	102	2021-08-01
*	NULL	NULL	NULL	NULL

Enrollments 3

Object Info Session Output

7 Finance headline India reported 6... Search

ENG IN 18:05 22-11-2025

MySQL Workbench

Jatin

File Edit View Query Database Server Tools Scripting Help

Navigator

PROJECT 2 PROJECT 1 FINAL PROJECT PROJECT 3*

Filter objects

company_db
datadigger
labwork_1_3
labwork_2_1
labwork_2_2
labwork_2_3
labwork_4_1
labwork_4_2
project_2
project_final
Tables
Views
Stored Procedures
Functions
student
sys

100 -- 3. REQUIRED QUERIES
101 -- =====
102
103 -- Query 1: Perform CRUD Operations on all tables
104 -- CREATE operations (INSERT) - already done above
105
106 -- READ operations (SELECT)
107 • SELECT * FROM Students;
108 • SELECT * FROM Courses;
109 • SELECT * FROM Instructors;
110 • SELECT * FROM Enrollments;
111 • SELECT * FROM Departments;
112
113 -- UPDATE operations
114 • UPDATE Students SET Email = 'john.updated@email.com' WHERE StudentID = 1;
115 • UPDATE Courses SET Credits = 5 WHERE CourseID = 102;
116
117 -- DELETE operations (with caution - respecting foreign key constraints)
118 -- Example: Delete a specific enrollment
119 • DELETE FROM Enrollments WHERE EnrollmentID = 1;

Administration Schemas

Information

No object selected

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content: |

	DepartmentID	DepartmentName
▶	1	Computer Science
▶	2	Mathematics
*	NULL	NULL

Departments 4

Object Info Session Output

7 Finance headline India reported 6... Search

ENG IN 18:06 22-11-2025

MySQL Workbench

Jatin

File Edit View Query Database Server Tools Scripting Help

Navigator

PROJECT 2 PROJECT 1 FINAL PROJECT PROJECT 3*

Filter objects

company_db
datadigger
labwork_1_3
labwork_2_1
labwork_2_2
labwork_2_3
labwork_4_1
labwork_4_2
project_2
project_final
Tables
Views
Stored Procedures
Functions
student
sys

100 -- 3. REQUIRED QUERIES
101 -- =====
102
103 -- Query 1: Perform CRUD Operations on all tables
104 -- CREATE operations (INSERT) - already done above
105
106 -- READ operations (SELECT)
107 • SELECT * FROM Students;
108 • SELECT * FROM Courses;
109 • SELECT * FROM Instructors;
110 • SELECT * FROM Enrollments;
111 • SELECT * FROM Departments;
112
113 -- UPDATE operations
114 • UPDATE Students SET Email = 'john.updated@email.com' WHERE StudentID = 1;
115 • UPDATE Courses SET Credits = 5 WHERE CourseID = 102;
116
117 -- DELETE operations (with caution - respecting foreign key constraints)
118 -- Example: Delete a specific enrollment
119 • DELETE FROM Enrollments WHERE EnrollmentID = 1;

Administration Schemas

Information

No object selected

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content: |

	DepartmentID	DepartmentName
▶	1	Computer Science
▶	2	Mathematics
*	NULL	NULL

Departments 5

Object Info Session Output

7 Finance headline India reported 6... Search

ENG IN 18:06 22-11-2025

MySQL Workbench

Jatin

File Edit View Query Database Server Tools Scripting Help

Navigator

PROJECT 2 PROJECT 1 FINAL PROJECT PROJECT 3*

Filter objects

company_db
datadigger
labwork_1_3
labwork_2_1
labwork_2_2
labwork_2_3
labwork_4_1
labwork_4_2
project_2
project_final
Tables
Views
Stored Procedures
Functions
student
sys

118 -- Example: Delete a specific enrollment
119 • DELETE FROM Enrollments WHERE EnrollmentID = 1;
120
121 -- =====
122 -- Query 2: Retrieve students who enrolled after 2022
123 -- =====
124 • SELECT StudentID,
125 FirstName,
126 LastName,
127 Email,
128 BirthDate,
129 EnrollmentDate
130 FROM Students
131 WHERE YEAR(EnrollmentDate) > 2022
132 ORDER BY EnrollmentDate;
133
134
135 -- =====
136 -- Query 3: Retrieve courses offered by Mathematics department with limit of 5
137 -- =====

Administration Schemas

Information

No object selected

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

StudentID	FirstName	LastName	Email	BirthDate	EnrollmentDate
*	HULL	HULL	HULL	HULL	HULL

Students 7

Object Info Session Output

7 Finance headline India reported 6... Search

ENG IN 18:07 22-11-2025

MySQL Workbench

Jatin

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

- company_db
- datadigger
- labwork_1_3
- labwork_2_1
- labwork_2_2
- labwork_2_3
- labwork_4_1
- labwork_4_2
- project_2
- project_final
- Tables
- Views
- Stored Procedures
- Functions
- student
- sys

PROJECT 2 PROJECT 1 FINAL PROJECT PROJECT 3*

133 ORDER BY EnrollmentDate;

134

135 -- =====

136 -- Query 3: Retrieve courses offered by Mathematics department with limit of 5

137 -- =====

138 • SELECT

139 c.CourseID,

140 c.CourseName,

141 c.Credits,

142 d.DepartmentName

143 FROM Courses c

144 INNER JOIN Departments d ON c.DepartmentID = d.DepartmentID

145 WHERE d.DepartmentName = 'Mathematics'

146 LIMIT 5;

147

148 -- =====

149 -- Query 4: Get number of students enrolled in each course,

150 -- filtering for courses with more than 5 students

151 -- =====

152 • SELECT

Result Grid | Filter Rows: Export: Wrap Cell Content:

CourseID	CourseName	Credits	DepartmentName
102	Data Structures	5	Mathematics

No object selected

Result 8 x Read Only

Object Info Session Output

7 Finance headline India reported 6...

Search

18:07 22-11-2025 ENG IN

MySQL Workbench

Jatin

File Edit View Query Database Server Tools Scripting Help

Navigator

PROJECT 2 PROJECT 1 FINAL PROJECT PROJECT 3*

Filter objects

company_db datadigger labwork_1_3 labwork_2_1 labwork_2_2 labwork_2_3 labwork_4_1 labwork_4_2 project_2 project_final Tables Views Stored Procedures Functions student sys

Administration Schemas

Information

No object selected

Result Grid Filter Rows: Export: Wrap Cell Content:

Result 9 x Read Only

Object Info Session Output

Finance headline India reported 6... 7

Search

18:07 22-11-2025 ENG IN

```
148 -- -----
149 -- Query 4: Get number of students enrolled in each course,
150 -- filtering for courses with more than 5 students
151 --
152 • SELECT
153     c.CourseID,
154     c.CourseName,
155     COUNT(e.StudentID) AS NumberOfStudents
156     FROM Courses c
157     LEFT JOIN Enrollments e ON c.CourseID = e.CourseID
158     GROUP BY c.CourseID, c.CourseName
159     HAVING COUNT(e.StudentID) > 5
160     ORDER BY NumberOfStudents DESC;
161
162 --
163 -- Query 5: Find students enrolled in both Introduction to SQL and Data Structures
164 --
165 • SELECT
166     s.StudentID,
167     s.FirstName.
```

MySQL Workbench

Jatin

File Edit View Query Database Server Tools Scripting Help

Navigator

PROJECT 2 PROJECT 1 FINAL PROJECT PROJECT 3*

Filter objects

company_db datadigger labwork_1_3 labwork_2_1 labwork_2_2 labwork_2_3 labwork_4_1 labwork_4_2 project_2 project_final Tables Views Stored Procedures Functions student sys

163 -- Query 5: Find students enrolled in both Introduction to SQL and Data Structures
164 -- =====
165 • SELECT s.StudentID,
166 s.FirstName,
167 s.LastName,
168 s.Email
169 FROM Students s
170 WHERE s.StudentID IN (
171 SELECT StudentID FROM Enrollments WHERE CourseID = 101
172)
173 AND s.StudentID IN (
174 SELECT StudentID FROM Enrollments WHERE CourseID = 102
175)
176
177
178 -- Alternative approach using INTERSECT (if supported):
179 -- SELECT StudentID FROM Enrollments WHERE CourseID = 101
180 -- INTERSECT
181 -- SELECT StudentID FROM Enrollments WHERE CourseID = 102;
182

Administration Schemas

Information

No object selected

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

	StudentID	FirstName	LastName	Email
*	HULL	HULL	HULL	HULL

Students 10

Object Info Session Output

Finance headline India reported 6... 7

Search

18:07 ENG IN 22-11-2025

MySQL Workbench

Jatin

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

- company_db
- datadigger
- labwork_1_3
- labwork_2_1
- labwork_2_2
- labwork_2_3
- labwork_4_1
- labwork_4_2
- project_2
- project_final
 - Tables
 - Views
 - Stored Procedures
 - Functions
- student
- sys

Administration Schemas

Information

No object selected

Result Grid | Filter Rows: Export: Wrap Cell Content:

	StudentID	FirstName	LastName	Email
▶	2	Jane	Smith	jane.smith@email.com

Result 11 x Read Only

Object Info Session Output

7 Finance headline India reported 6...

Search

18:07 22-11-2025 ENG IN

```
184 -- Query 6: Find students enrolled in either Introduction to SQL or Data Structures
185 --
186 • SELECT DISTINCT
187     s.StudentID,
188     s.FirstName,
189     s.LastName,
190     s.Email
191 FROM Students s
192 INNER JOIN Enrollments e ON s.StudentID = e.StudentID
193 WHERE e.CourseID IN (101, 102)
194 ORDER BY s.StudentID;
195
196 --
197 -- Query 7: Calculate the average number of credits for all courses
198 --
199 • SELECT
200     AVG(Credits) AS AverageCredits,
201     ROUND(AVG(Credits), 2) AS AverageCreditsRounded
202 FROM Courses;
```

MySQL Workbench

Jatin

File Edit View Query Database Server Tools Scripting Help

Navigator

PROJECT 2 PROJECT 1 FINAL PROJECT PROJECT 3*

Filter objects

company_db datadigger labwork_1_3 labwork_2_1 labwork_2_2 labwork_2_3 labwork_4_1 labwork_4_2 project_2 project_final Tables Views Stored Procedures Functions student sys

Administration Schemas

Information

No object selected

Result Grid | Filter Rows: Export: Wrap Cell Content:

AverageCredits	AverageCreditsRounded
4.0000	4.00

Result 12 x Read Only

Object Info Session Output

7 Finance headline India reported 6...

Search

18:08 22-11-2025 ENG IN

```
193 WHERE e.CourseID IN (101, 102)
194 ORDER BY s.StudentID;
195
196 -----
197 -- Query 7: Calculate the average number of credits for all courses
198 -----
199 • SELECT
200     AVG(Credits) AS AverageCredits,
201     ROUND(AVG(Credits), 2) AS AverageCreditsRounded
202 FROM Courses;
203
204 -----
205 -- Query 8: Find the maximum salary of instructors in Computer Science department
206 -- Note: Salary field not in original schema, using example structure
207 -- If salary data exists, the query would be:
208 -----
209 -- ALTER TABLE Instructors ADD COLUMN Salary DECIMAL(10,2);
210 -- UPDATE Instructors SET Salary = 75000 WHERE InstructorID = 1;
211 -- UPDATE Instructors SET Salary = 80000 WHERE InstructorID = 2;
212
```

MySQL Workbench

Jatin

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

- company_db
- datadigger
- labwork_1_3
- labwork_2_1
- labwork_2_2
- labwork_2_3
- labwork_4_1
- labwork_4_2
- project_2
- project_final
- Tables
- Views
- Stored Procedures
- Functions
- student
- sys

PROJECT 2 PROJECT 1 FINAL PROJECT PROJECT 3*

208 -- -----
209 -- ALTER TABLE Instructors ADD COLUMN Salary DECIMAL(10,2);
210 -- UPDATE Instructors SET Salary = 75000 WHERE InstructorID = 1;
211 -- UPDATE Instructors SET Salary = 80000 WHERE InstructorID = 2;
212
213 • SELECT
214 d.DepartmentName,
215 i.FirstName,
216 i.LastName
217 -- MAX(i.Salary) AS MaxSalary -- Uncomment if Salary field exists
218 FROM Instructors i
219 INNER JOIN Departments d ON i.DepartmentID = d.DepartmentID
220 WHERE d.DepartmentName = 'Computer Science'
221 GROUP BY d.DepartmentName, i.FirstName, i.LastName
222
223 -- -----
224 -- Query 9: Count the number of students enrolled in each department
225 -- -----
226 • SELECT
227 d.DepartmentID,

Administration Schemas

Information

No object selected

Result Grid | Filter Rows: Export: Wrap Cell Content:

DepartmentName	FirstName	LastName
Computer Science	Alice	Johnson

Result 13 x Read Only

Object Info Session Output

7 Finance headline India reported 6...

Search

18:08 22-11-2025 ENG IN

MySQL Workbench

Jatin

File Edit View Query Database Server Tools Scripting Help

Navigator

PROJECT 2 PROJECT 1 FINAL PROJECT PROJECT 3*

Filter objects

company_db datadigger labwork_1_3 labwork_2_1 labwork_2_2 labwork_2_3 labwork_4_1 labwork_4_2 project_2 project_final Tables Views Stored Procedures Functions student sys

220 WHERE d.DepartmentName = 'Computer Science'
221 GROUP BY d.DepartmentName, i.FirstName, i.LastName;
222
223 -- =====
224 -- Query 9: Count the number of students enrolled in each department
225 -- =====
226 • SELECT
227 d.DepartmentID,
228 d.DepartmentName,
229 COUNT(DISTINCT e.StudentID) AS NumberOfStudents
230 FROM Departments d
231 LEFT JOIN Courses c ON d.DepartmentID = c.DepartmentID
232 LEFT JOIN Enrollments e ON c.CourseID = e.CourseID
233 GROUP BY d.DepartmentID, d.DepartmentName
234 ORDER BY NumberOfStudents DESC;
235
236 -- =====
237 -- Query 10: INNER JOIN - Retrieve students and their corresponding courses
238 -- =====
239 • SELECT

Administration Schemas

Information

No object selected

Result Grid | Filter Rows: Export: Wrap Cell Content:

	DepartmentID	DepartmentName	NumberOfStudents
▶	2	Mathematics	1
	1	Computer Science	0

Result 14 x Read Only

Object Info Session Output

7 Finance headline India reported 6... Search

ENG IN 18:08 22-11-2025

MySQL Workbench

Jatin

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

- company_db
- datadigger
- labwork_1_3
- labwork_2_1
- labwork_2_2
- labwork_2_3
- labwork_4_1
- labwork_4_2
- project_2
- project_final
 - Tables
 - Views
 - Stored Procedures
 - Functions
- student
- sys

PROJECT 2 PROJECT 1 FINAL PROJECT PROJECT 3*

232 LEFT JOIN Enrollments e ON c.CourseID = e.CourseID
233 GROUP BY d.DepartmentID, d.DepartmentName
234 ORDER BY NumberOfStudents DESC;
235
236 -- =====
237 -- Query 10: INNER JOIN - Retrieve students and their corresponding courses
238 -- =====
239 • SELECT s.StudentID,
240 s.FirstName,
241 s.LastName,
242 c.CourseID,
243 c.CourseName,
244 e.EnrollmentDate
245 FROM Students s
246 INNER JOIN Enrollments e ON s.StudentID = e.StudentID
247 INNER JOIN Courses c ON e.CourseID = c.CourseID
248 ORDER BY s.StudentID, c.CourseID;
249
250
251 -- =====

Administration Schemas

Information

No object selected

Result Grid | Filter Rows: Export: Wrap Cell Content:

	StudentID	FirstName	LastName	CourseID	CourseName	EnrollmentDate
▶	2	Jane	Smith	102	Data Structures	2021-08-01

Result 15 x Read Only

Object Info Session Output

7 Finance headline India reported 6... Search

ENG IN 18:08 22-11-2025

MySQL Workbench

Jatin

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

- company_db
- datadigger
- labwork_1_3
- labwork_2_1
- labwork_2_2
- labwork_2_3
- labwork_4_1
- labwork_4_2
- project_2
- project_final
 - Tables
 - Views
 - Stored Procedures
 - Functions
- student
- sys

PROJECT 2 PROJECT 1 FINAL PROJECT PROJECT 3*

Don't Limit

```
250
251 -- =====
252 -- Query 11: LEFT JOIN - Retrieve all students and their corresponding courses (if any)
253 -- =====
254 • SELECT
255     s.StudentID,
256     s.FirstName,
257     s.LastName,
258     c.CourseID,
259     c.CourseName,
260     e.EnrollmentDate
261 FROM Students s
262 LEFT JOIN Enrollments e ON s.StudentID = e.StudentID
263 LEFT JOIN Courses c ON e.CourseID = c.CourseID
264 ORDER BY s.StudentID;
265
266 -- =====
267 -- Query 12: Subquery - Find students enrolled in courses with more than 10 students
268 -- =====
269 • SELECT
```

Administration Schemas

Information

No object selected

Result Grid | Filter Rows: Export: Wrap Cell Content:

	StudentID	FirstName	LastName	CourseID	CourseName	EnrollmentDate
▶	1	John	Doe	NULL	NULL	NULL
	2	Jane	Smith	102	Data Structures	2021-08-01

Result 16 x Read Only

Object Info Session Output

7 Finance headline India reported 6...

Search

18:08 22-11-2025 ENG IN

MySQL Workbench

Jatin

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

- company_db
- datadigger
- labwork_1_3
- labwork_2_1
- labwork_2_2
- labwork_2_3
- labwork_4_1
- labwork_4_2
- project_2
- project_final
 - Tables
 - Views
 - Stored Procedures
 - Functions
- student
- sys

PROJECT 2 PROJECT 1 FINAL PROJECT PROJECT 3*

Don't Limit

```
268 -- -----
269 •   SELECT
270     s.StudentID,
271     s.FirstName,
272     s.LastName,
273     s.Email
274   FROM Students s
275   WHERE s.StudentID IN (
276       SELECT e.StudentID
277       FROM Enrollments e
278       WHERE e.CourseID IN (
279           SELECT CourseID
280           FROM Enrollments
281           GROUP BY CourseID
282           HAVING COUNT(StudentID) > 10
283       )
284   )
285   ORDER BY s.StudentID;
```

Administration Schemas

Information

No object selected

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content: |

	StudentID	FirstName	LastName	Email
*	HULL	HULL	HULL	HULL

Students 17

Apply Revert

Object Info Session Output

7 Finance headline India reported 6...

Search

18:08 22-11-2025 ENG IN

MySQL Workbench

Jatin

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

- company_db
- datadigger
- labwork_1_3
- labwork_2_1
- labwork_2_2
- labwork_2_3
- labwork_4_1
- labwork_4_2
- project_2
- project_final
 - Tables
 - Views
 - Stored Procedures
 - Functions
- student
- sys

PROJECT 2 PROJECT 1 FINAL PROJECT PROJECT 3*

286
287 -- =====
288 -- Query 13: Extract the year from EnrollmentDate of students
289 -- =====
290 • SELECT
291 StudentID,
292 FirstName,
293 LastName,
294 EnrollmentDate,
295 YEAR(EnrollmentDate) AS EnrollmentYear,
296 MONTH(EnrollmentDate) AS EnrollmentMonth,
297 DAY(EnrollmentDate) AS EnrollmentDay
298 FROM Students
299 ORDER BY EnrollmentDate;
300
301 -- =====
302 -- Query 14: Concatenate instructor's first and last name
303 -- =====
304 • SELECT
305 InstructorID.

Administration Schemas

Information

No object selected

Result Grid | Filter Rows: Export: Wrap Cell Content:

	StudentID	FirstName	LastName	EnrollmentDate	EnrollmentYear	EnrollmentMonth	EnrollmentDay
▶	2	Jane	Smith	2021-08-01	2021	8	1
	1	John	Doe	2022-08-01	2022	8	1

Result 18 x Read Only

Object Info Session Output

7 Finance headline India reported 6...

Search

18:08 22-11-2025 ENG IN

MySQL Workbench

Jatin

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

- company_db
- datadigger
- labwork_1_3
- labwork_2_1
- labwork_2_2
- labwork_2_3
- labwork_4_1
- labwork_4_2
- project_2
- project_final
 - Tables
 - Views
 - Stored Procedures
 - Functions
- student
- sys

PROJECT 2 PROJECT 1 FINAL PROJECT PROJECT 3*

Don't Limit

```
298 FROM Students
299 ORDER BY EnrollmentDate;
300
301 -- =====
302 -- Query 14: Concatenate instructor's first and last name
303 -- =====
304 • SELECT
305   InstructorID,
306   FirstName,
307   LastName,
308   CONCAT(FirstName, ' ', LastName) AS FullName,
309   Email,
310   DepartmentID
311 FROM Instructors
312 ORDER BY InstructorID;
313
314 -- =====
315 -- Query 15: Calculate the running total of students enrolled in courses
316 -- =====
317 • SELECT
```

Administration Schemas

Information

No object selected

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Read Only

InstructorID	FirstName	LastName	FullName	Email	DepartmentID
1	Alice	Johnson	Alice Johnson	alice.johnson@univ.com	1
2	Bob	Lee	Bob Lee	bob.lee@univ.com	2

Result 19 x

Object Info Session Output

7 Finance headline India reported 6...

Search

18:08 22-11-2025 ENG IN

MySQL Workbench

Jatin

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

- company_db
- datadigger
- labwork_1_3
- labwork_2_1
- labwork_2_2
- labwork_2_3
- labwork_4_1
- labwork_4_2
- project_2
- project_final
 - Tables
 - Views
 - Stored Procedures
 - Functions
- student
- sys

PROJECT 2 PROJECT 1 FINAL PROJECT PROJECT 3*

Don't Limit

```
313
314 -- =====
315 -- Query 15: Calculate the running total of students enrolled in courses
316 -- =====
317 • SELECT
318     c.CourseID,
319     c.CourseName,
320     COUNT(e.StudentID) AS StudentsEnrolled,
321     SUM(COUNT(e.StudentID)) OVER (ORDER BY c.CourseID) AS RunningTotal
322     FROM Courses c
323     LEFT JOIN Enrollments e ON c.CourseID = e.CourseID
324     GROUP BY c.CourseID, c.CourseName
325     ORDER BY c.CourseID;
326
327 -- =====
328 -- Query 16: Label students as 'Senior' or 'Junior' based on enrollment year
329 --           (Senior if enrolled more than 4 years ago, otherwise Junior)
330 -- =====
331 • SELECT
332     StudentID,
```

Administration Schemas

Information

No object selected

Result Grid | Filter Rows: Export: Wrap Cell Content:

	CourseID	CourseName	StudentsEnrolled	RunningTotal
▶	101	Introduction to SQL	0	0
	102	Data Structures	1	1

Result 20 x Read Only

Object Info Session Output

Trending videos Stranger Things...

Search

18:08 22-11-2025 ENG IN

MySQL Workbench

Jatin

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

- company_db
- datadigger
- labwork_1_3
- labwork_2_1
- labwork_2_2
- labwork_2_3
- labwork_4_1
- labwork_4_2
- project_2
- project_final
 - Tables
 - Views
 - Stored Procedures
 - Functions
- student
- sys

PROJECT 2 PROJECT 1 FINAL PROJECT PROJECT 3*

Don't Limit

```
328 -- Query 16: Label students as 'Senior' or 'Junior' based on enrollment year
329 -- (Senior if enrolled more than 4 years ago, otherwise Junior)
330 --
331 • SELECT
332     StudentID,
333     FirstName,
334     LastName,
335     EnrollmentDate,
336     DATEDIFF(CURDATE(), EnrollmentDate) / 365 AS YearsSinceEnrollment,
337     CASE
338         WHEN DATEDIFF(CURDATE(), EnrollmentDate) / 365 > 4 THEN 'Senior'
339         ELSE 'Junior'
340     END AS StudentStatus
341     FROM Students
342     ORDER BY EnrollmentDate;
343 
344 --
345 -- ADDITIONAL USEFUL QUERIES
346 --
347
```

Administration Schemas

Information

No object selected

Result Grid | Filter Rows: Export: Wrap Cell Content:

	StudentID	FirstName	LastName	EnrollmentDate	YearsSinceEnrollment	StudentStatus
▶	2	Jane	Smith	2021-08-01	4.3123	Senior
	1	John	Doe	2022-08-01	3.3123	Junior

Result 21 x Read Only

Object Info Session Output

7 Trending videos Stranger Things...

Search

18:09 22-11-2025 ENG IN

MySQL Workbench

Jatin

File Edit View Query Database Server Tools Scripting Help

Schemas

company_db
datadigger
labwork_1_3
labwork_2_1
labwork_2_2
labwork_2_3
labwork_4_1
labwork_4_2
project_2
project_final
Tables
Views
Stored Procedures
Functions

student
sys

Administration Schemas

No object selected

Information

Object Info Session

PROJECT 2 PROJECT 1 FINAL PROJECT PROJECT 3*

Action Output

#	Time	Action	Message	Duration / Fetch
1	18:03:40	CREATE DATABASE PROJECT_3	1 row(s) affected	0.016 sec
2	18:03:42	USE PROJECT_3	0 row(s) affected	0.016 sec
3	18:03:47	CREATE TABLE Students (StudentID INT PRIMARY KEY AUTO_INCREMENT, FirstName VARCHAR(255) NOT NULL, LastName VARCHAR(255) NOT NULL, Email VARCHAR(255) NOT NULL, BirthDate DATE NOT NULL)	0 row(s) affected	0.047 sec
4	18:03:50	CREATE TABLE Departments (DepartmentID INT PRIMARY KEY AUTO_INCREMENT, DepartmentName VARCHAR(255) NOT NULL)	0 row(s) affected	0.032 sec
5	18:03:54	CREATE TABLE Courses (CourseID INT PRIMARY KEY AUTO_INCREMENT, CourseName VARCHAR(255) NOT NULL, Credits INT NOT NULL)	0 row(s) affected	0.031 sec
6	18:03:59	CREATE TABLE Instructors (InstructorID INT PRIMARY KEY AUTO_INCREMENT, FirstName VARCHAR(255) NOT NULL, LastName VARCHAR(255) NOT NULL, Email VARCHAR(255) NOT NULL, BirthDate DATE NOT NULL)	0 row(s) affected	0.031 sec
7	18:04:03	CREATE TABLE Enrollments (EnrollmentID INT PRIMARY KEY AUTO_INCREMENT, StudentID INT NOT NULL, CourseID INT NOT NULL, EnrollmentDate DATE NOT NULL)	0 row(s) affected	0.047 sec
8	18:04:08	INSERT INTO Departments (DepartmentID, DepartmentName) VALUES (1, 'Computer Science'), (2, 'Mathema...')	2 row(s) affected Records: 2 Duplicates: 0 Warnings: 0	0.016 sec
9	18:04:12	INSERT INTO Students (StudentID, FirstName, LastName, Email, BirthDate, EnrollmentDate) VALUES (1, 'John', 'Doe', 'john.doe@example.com', '1990-01-01', '2022-08-15')	2 row(s) affected Records: 2 Duplicates: 0 Warnings: 0	0.015 sec
10	18:04:16	INSERT INTO Courses (CourseID, CourseName, DepartmentID, Credits) VALUES (101, 'Introduction to SQL', 1, 3)	2 row(s) affected Records: 2 Duplicates: 0 Warnings: 0	0.015 sec
11	18:04:20	INSERT INTO Instructors (InstructorID, FirstName, LastName, Email, DepartmentID) VALUES (1, 'Alice', 'Johnson', 'alice.johnson@example.com', 1)	2 row(s) affected Records: 2 Duplicates: 0 Warnings: 0	0.000 sec
12	18:04:24	INSERT INTO Enrollments (EnrollmentID, StudentID, CourseID, EnrollmentDate) VALUES (1, 1, 101, '2022-08-15')	2 row(s) affected Records: 2 Duplicates: 0 Warnings: 0	0.015 sec
13	18:04:32	SELECT * FROM Students	2 row(s) returned	0.000 sec / 0.000 sec
14	18:05:52	SELECT * FROM Courses	2 row(s) returned	0.000 sec / 0.000 sec
15	18:05:58	SELECT * FROM Enrollments	2 row(s) returned	0.000 sec / 0.000 sec
16	18:06:03	SELECT * FROM Departments	2 row(s) returned	0.000 sec / 0.000 sec
17	18:06:08	SELECT * FROM Departments	2 row(s) returned	0.000 sec / 0.000 sec
18	18:06:15	UPDATE Courses SET Credits = 5 WHERE CourseID = 102	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0	0.000 sec
19	18:06:46	UPDATE Courses SET Credits = 5 WHERE CourseID = 102	0 row(s) affected Rows matched: 1 Changed: 0 Warnings: 0	0.000 sec
20	18:06:51	DELETE FROM Enrollments WHERE EnrollmentID = 1	1 row(s) affected	0.015 sec
21	18:07:05	SELECT StudentID, FirstName, LastName, Email, BirthDate, EnrollmentDate FROM Students	0 row(s) returned	0.000 sec / 0.000 sec
22	18:07:15	SELECT StudentID, FirstName, LastName, Email, BirthDate, EnrollmentDate FROM Students	0 row(s) returned	0.000 sec / 0.000 sec
23	18:07:25	SELECT c.CourseID, c.CourseName, c.Credits, d.DepartmentName FROM Courses c INNER JOIN Departments d ON c.DepartmentID = d.DepartmentID	1 row(s) returned	0.000 sec / 0.000 sec
24	18:07:34	SELECT c.CourseID, c.CourseName, COUNT(e.StudentID) AS NumberOfStudents FROM Courses c LEFT JOIN Enrollments e ON c.CourseID = e.CourseID GROUP BY c.CourseID	0 row(s) returned	0.000 sec / 0.000 sec
25	18:07:45	SELECT s.StudentID, s.FirstName, s.LastName, s.Email FROM Students s WHERE s.StudentID IN (1, 2, 3, 4, 5)	0 row(s) returned	0.000 sec / 0.000 sec
26	18:07:53	SELECT DISTINCT s.StudentID, s.FirstName, s.LastName, s.Email FROM Students s INNER JOIN Enrollments e ON s.StudentID = e.StudentID	1 row(s) returned	0.000 sec / 0.000 sec
27	18:08:00	SELECT AVG(Credits) AS AverageCredits, ROUND(AVG(Credits), 2) AS AverageCreditsRounded FROM Courses	1 row(s) returned	0.000 sec / 0.000 sec
28	18:08:08	SELECT d.DepartmentName, i.FirstName, i.LastName, MAX(i.Salary) AS MaxSalary -- In comment	1 row(s) returned	0.000 sec / 0.000 sec

28°C Clear

Search

System Icons

ENG IN 22-11-2025 18:09

MySQL Workbench

Jatin

File Edit View Query Database Server Tools Scripting Help

Navigator

Output

Action Output

#	Time	Action	Message	Duration / Fetch
10	18:04:16	INSERT INTO Courses (CourseID, CourseName, DepartmentID, Credits) VALUES (101, 'Introduction to SQL', ...)	2 row(s) affected Records: 2 Duplicates: 0 Warnings: 0	0.015 sec
11	18:04:20	INSERT INTO Instructors (InstructorID, FirstName, LastName, Email, DepartmentID) VALUES (1, 'Alice', Johns..., ...)	2 row(s) affected Records: 2 Duplicates: 0 Warnings: 0	0.000 sec
12	18:04:24	INSERT INTO Enrollments (EnrollmentID, StudentID, CourseID, EnrollmentDate) VALUES (1, 1, 101, '2022-08-...')	2 row(s) affected Records: 2 Duplicates: 0 Warnings: 0	0.015 sec
13	18:04:32	SELECT * FROM Students	2 row(s) returned	0.000 sec / 0.000 sec
14	18:05:52	SELECT * FROM Courses	2 row(s) returned	0.000 sec / 0.000 sec
15	18:05:58	SELECT * FROM Enrollments	2 row(s) returned	0.000 sec / 0.000 sec
16	18:06:03	SELECT * FROM Departments	2 row(s) returned	0.000 sec / 0.000 sec
17	18:06:08	SELECT * FROM Departments	2 row(s) returned	0.000 sec / 0.000 sec
18	18:06:15	UPDATE Courses SET Credits = 5 WHERE CourseID = 102	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0	0.000 sec
19	18:06:46	UPDATE Courses SET Credits = 5 WHERE CourseID = 102	0 row(s) affected Rows matched: 1 Changed: 0 Warnings: 0	0.000 sec
20	18:06:51	DELETE FROM Enrollments WHERE EnrollmentID = 1	1 row(s) affected	0.015 sec
21	18:07:05	SELECT StudentID, FirstName, LastName, Email, BirthDate, EnrollmentDate FROM Students ...	0 row(s) returned	0.000 sec / 0.000 sec
22	18:07:15	SELECT StudentID, FirstName, LastName, Email, BirthDate, EnrollmentDate FROM Students ...	0 row(s) returned	0.000 sec / 0.000 sec
23	18:07:25	SELECT c.CourseID, c.CourseName, c.Credits, d.DepartmentName FROM Courses c INNER JOIN ...	1 row(s) returned	0.000 sec / 0.000 sec
24	18:07:34	SELECT c.CourseID, c.CourseName, COUNT(e.StudentID) AS NumberOfStudents FROM Courses c L...	0 row(s) returned	0.000 sec / 0.000 sec
25	18:07:45	SELECT s.StudentID, s.FirstName, s.LastName, s.Email FROM Students s WHERE s.StudentID IN ...	0 row(s) returned	0.000 sec / 0.000 sec
26	18:07:53	SELECT DISTINCT s.StudentID, s.FirstName, s.LastName, s.Email FROM Students s INNER JOIN ...	1 row(s) returned	0.000 sec / 0.000 sec
27	18:08:00	SELECT AVG(Credits) AS AverageCredits, ROUND(AVG(Credits), 2) AS AverageCreditsRounded FROM ...	1 row(s) returned	0.000 sec / 0.000 sec
28	18:08:08	SELECT d.DepartmentName, i.FirstName, i.LastName -- MAX(i.Salary) AS MaxSalary -- Uncomment...;	1 row(s) returned	0.000 sec / 0.000 sec
29	18:08:14	SELECT d.DepartmentID, d.DepartmentName, COUNT(DISTINCT e.StudentID) AS NumberOfStudent...	2 row(s) returned	0.000 sec / 0.000 sec
30	18:08:20	SELECT s.StudentID, s.FirstName, s.LastName, c.CourseID, c.CourseName, e.EnrollmentDate...	1 row(s) returned	0.000 sec / 0.000 sec
31	18:08:27	SELECT s.StudentID, s.FirstName, s.LastName, c.CourseID, c.CourseName, e.EnrollmentDate...	2 row(s) returned	0.000 sec / 0.000 sec
32	18:08:34	SELECT s.StudentID, s.FirstName, s.LastName, s.Email FROM Students s WHERE s.StudentID IN ...	0 row(s) returned	0.000 sec / 0.000 sec
33	18:08:42	SELECT StudentID, FirstName, LastName, EnrollmentDate, YEAR(EnrollmentDate) AS Enrollment...	2 row(s) returned	0.000 sec / 0.000 sec
34	18:08:48	SELECT InstructorID, FirstName, LastName, CONCAT(FirstName, ' ', LastName) AS FullName, E...	2 row(s) returned	0.000 sec / 0.000 sec
35	18:08:54	SELECT c.CourseID, c.CourseName, COUNT(e.StudentID) AS StudentsEnrolled, SUM(COUNT(e.S...)	2 row(s) returned	0.000 sec / 0.000 sec
36	18:09:00	SELECT StudentID, FirstName, LastName, EnrollmentDate, DATEDIFF(CURDATE(), EnrollmentD...	2 row(s) returned	0.000 sec / 0.000 sec

Administration Schemas

No object selected

Information

Object Info Session

28°C Clear

Search

System Icons

ENG IN

18:10 22-11-2025