












18 • `Select * from students;`

19

Result Grid   Filter Rows: <input type="text"/> Edit:    Export/Import:   Wrap						
	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




32 • `select * from Courses;`

33

Result Grid   Filter Rows: <input type="text"/> Edit:  				
	CourseID	CourseName	DepartmentID	Credits
▶	101	Introduction to SQL	1	3
	102	Data Structures	2	4
•	NULL	NULL	NULL	NULL

47 • `select * from emp;`



48

Result Grid   Filter Rows: <input type="text"/> Edit: 				
	EnrolmentID	StudentID	CourseID	EnrollmentDate
▶	1	1	101	2022-08-01
	2	2	102	2021-08-01
•	NULL	NULL	NULL	NULL

60

61 • `Select * from departments;`

62

Result Grid		 Filter Rows: <input data-bbox="504 353 531 358" type="text"/>
	DepartmentID	DepartmentName
▶	1	Computer Science
	2	Mathematics
●	NULL	NULL

68 • `UPDATE Students`

69 `SET Email = 'john.new@email.com'`

70 `WHERE StudentID = 1;`

71 • `Select * from students;`

Result Grid

Filter Rows:

Edit:

Export/Import:

StudentID	FirstName	LastName	Email	BirthDate	EnrollmentDate
1	John	Doe	john.new@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

72

73 • `DELETE FROM emp`

74 `WHERE EnrolmentID = 2;`

75 • `Select * from emp;`

Result Grid		Filter Rows:	Edit:
EnrolmentID	StudentID	CourseID	EnrollmentDate
1	1	101	2022-08-01
NULL	NULL	NULL	NULL

```

76
77 • SELECT *
78 FROM Students
79 WHERE EnrollmentDate > '2022-01-01';
80

```

Result Grid			Filter Rows:	<input type="text"/>	Edit:				Export/Import:		
	StudentID	FirstName	LastName	Email	BirthDate	EnrollmentDate					
▶	1	John	Doe	john.new@email.com	2000-01-15	2022-08-01					
•	NULL	NULL	NULL	NULL	NULL	NULL					

```

81
82 • SELECT c.CourseID, c.CourseName
83 FROM Courses c
84 JOIN Departments d ON c.DepartmentID = d.DepartmentID
85 WHERE d.DepartmentName = 'Mathematics'
86 LIMIT 5;

```

Result Grid

Filter Rows:

Export:

Wrap Cell Content:

	CourseID	CourseName
	102	Data Structures

```

87
88 • SELECT
89     c.CourseName,
90     COUNT(e.StudentID) AS TotalStudents
91 FROM emp e
92 JOIN Courses c ON e.CourseID = c.CourseID
93 GROUP BY c.CourseName
94 HAVING COUNT(e.StudentID) > 0;
95

```

Result Grid			Filter Rows:	Export:	Wrap
	CourseName	TotalStudents			
▶	Introduction to SQL	1			

```

102 HAVING COUNT(DISTINCT c.CourseName) = 4;
103
104 • SELECT DISTINCT s.StudentID, s.FirstName, s.LastName
105 FROM Students s
106 JOIN emp e ON s.StudentID = e.StudentID
107 JOIN Courses c ON e.CourseID = c.CourseID
108 WHERE c.CourseName IN ('Introduction to SQL', 'Data Structures');
109

```

Result Grid			Filter Rows:	Export:	Wrap Cell Content:
	StudentID	FirstName	LastName		
▶	1	John	Doe		

```

102 HAVING COUNT(DISTINCT c.CourseName) = 4;
103
104 • SELECT DISTINCT s.StudentID, s.FirstName, s.LastName
105 FROM Students s
106 JOIN emp e ON s.StudentID = e.StudentID
107 JOIN Courses c ON e.CourseID = c.CourseID
108 WHERE c.CourseName IN ('Introduction to SQL', 'Data Structures');
109

```

Result Grid			Filter Rows:	Export:	Wrap Cell Content:
	StudentID	FirstName	LastName		
▶	1	John	Doe		

```

102     HAVING COUNT(DISTINCT c.CourseName) = 4;
103
104 • SELECT DISTINCT s.StudentID, s.FirstName, s.LastName
105     FROM Students s
106     JOIN emp e ON s.StudentID = e.StudentID
107     JOIN Courses c ON e.CourseID = c.CourseID
108     WHERE c.CourseName IN ('Introduction to SQL', 'Data Structures');
109

```

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

	StudentID	FirstName	LastName
▶	1	John	Doe

```

109
110 • SELECT AVG(Credits) AS AverageCredits
111     FROM Courses;
112

```

Result Grid |  Filter Rows: | Export:  | Wrap Ce

	AverageCredits
▶	3.5000

```

109
110 • SELECT AVG(Credits) AS AverageCredits
111     FROM Courses;
112

```

Result Grid |  Filter Rows: | Export:  | Wrap Ce




	AverageCredits
▶	3.5000

109

110 • `SELECT AVG(Credits) AS AverageCredits`

111 `FROM Courses;`

112

Result Grid   Filter Rows: <input type="text"/> Export:  Wrap Cell	
AverageCredits	
3.5000	

18

19 • `SELECT`

20 `d.DepartmentName,`




21 `COUNT(DISTINCT e.StudentID) AS TotalStudents`

22 `FROM emp e`

23 `JOIN Courses c ON e.CourseID = c.CourseID`

24 `JOIN Departments d ON c.DepartmentID = d.DepartmentID`

25 `GROUP BY d.DepartmentName;`

Result Grid   Filter Rows: <input type="text"/> Export:  Wrap Cell Content:	
DepartmentName	TotalStudents
Computer Science	1

18

19 • `SELECT`

20 `d.DepartmentName,`




21 `COUNT(DISTINCT e.StudentID) AS TotalStudents`

22 `FROM emp e`

23 `JOIN Courses c ON e.CourseID = c.CourseID`

24 `JOIN Departments d ON c.DepartmentID = d.DepartmentID`

25 `GROUP BY d.DepartmentName;`

Result Grid   Filter Rows: <input type="text"/> Export:  Wrap Cell Content:	
DepartmentName	TotalStudents
Computer Science	1

18

```
19 • SELECT
20     d.DepartmentName,
21     COUNT(DISTINCT e.StudentID) AS TotalStudents
22 FROM emp e
23 JOIN Courses c ON e.CourseID = c.CourseID
24 JOIN Departments d ON c.DepartmentID = d.DepartmentID
25 GROUP BY d.DepartmentName;
```

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

DepartmentName	TotalStudents
Computer Science	1

18

```
19 • SELECT
20     d.DepartmentName,
21     COUNT(DISTINCT e.StudentID) AS TotalStudents
22 FROM emp e
23 JOIN Courses c ON e.CourseID = c.CourseID
24 JOIN Departments d ON c.DepartmentID = d.DepartmentID
25 GROUP BY d.DepartmentName;
```

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

DepartmentName	TotalStudents
Computer Science	1

126

```
127 • SELECT
128     s.FirstName,
129     s.LastName,
130     c.CourseName
131 FROM emp e
132 INNER JOIN Students s ON e.StudentID = s.StudentID
133 INNER JOIN Courses c ON e.CourseID = c.CourseID;
134
```

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

	FirstName	LastName	CourseName
▶	John	Doe	Introduction to SQL

126



```
127 • SELECT
128     s.FirstName,
129     s.LastName,
130     c.CourseName
131 FROM emp e
132 INNER JOIN Students s ON e.StudentID = s.StudentID
133 INNER JOIN Courses c ON e.CourseID = c.CourseID;
134
```

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

	FirstName	LastName	CourseName
▶	John	Doe	Introduction to SQL

126

```
127 • SELECT
128     s.FirstName,
129     s.LastName,
130     c.CourseName
131 FROM emp e
132 INNER JOIN Students s ON e.StudentID = s.StudentID
133 INNER JOIN Courses c ON e.CourseID = c.CourseID;
134
```

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

	FirstName	LastName	CourseName
▶	John	Doe	Introduction to SQL

134

```
135 • SELECT
136     s.FirstName,
137     s.LastName,
138     c.CourseName
139 FROM Students s
140 LEFT JOIN emp e ON s.StudentID = e.StudentID
141 LEFT JOIN Courses c ON e.CourseID = c.CourseID;
```

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

	FirstName	LastName	CourseName
▶	John	Doe	Introduction to SQL
	Jane	Smith	NULL

```

134
135 • SELECT
136     s.FirstName,
137     s.LastName,
138     c.CourseName
139 FROM Students s
140 LEFT JOIN emp e ON s.StudentID = e.StudentID
141 LEFT JOIN Courses c ON e.CourseID = c.CourseID;

```

Result Grid				Filter Rows:	Export:	Wrap Cell Content
	FirstName	LastName	CourseName			
▶	John	Doe	Introduction to SQL			
	Jane	Smith	NULL			




```

134
135 • SELECT
136     s.FirstName,
137     s.LastName,
138     c.CourseName
139 FROM Students s
140 LEFT JOIN emp e ON s.StudentID = e.StudentID
141 LEFT JOIN Courses c ON e.CourseID = c.CourseID;

```




Result Grid				Filter Rows:	Export:	Wrap Cell Content
	FirstName	LastName	CourseName			
▶	John	Doe	Introduction to SQL			
	Jane	Smith	NULL			

```
152
153 • SELECT
154     StudentID,
155     FirstName,
156     LastName,
157     YEAR(EnrollmentDate) AS EnrollmentYear
158 FROM Students;
```

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

	StudentID	FirstName	LastName	EnrollmentYear
1	1	John	Doe	2022
2	2	Jane	Smith	2021

```
152
153 • SELECT
154     StudentID,
155     FirstName,
156     LastName,
157     YEAR(EnrollmentDate) AS EnrollmentYear
158 FROM Students;
```



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

	StudentID	FirstName	LastName	EnrollmentYear
1	1	John	Doe	2022
2	2	Jane	Smith	2021

```

152
153 • SELECT
154     StudentID,
155     FirstName,
156     LastName,
157     YEAR(EnrollmentDate) AS EnrollmentYear
158 FROM Students;

```





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

	StudentID	FirstName	LastName	EnrollmentYear
1	1	John	Doe	2022
2	2	Jane	Smith	2021

```

160 • SELECT
161     CourseID,
162     COUNT(StudentID) AS StudentsPerCourse,
163     SUM(COUNT(StudentID)) OVER (ORDER BY CourseID) AS RunningTotalStudents
164 FROM emp
165 GROUP BY CourseID;

```

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

	CourseID	StudentsPerCourse	RunningTotalStudents
101	101	1	1

```

57 • SELECT
58     StudentID,
59     FirstName,
60     LastName,
61     EnrollmentDate,
62     CASE
63         WHEN EnrollmentDate <= CURDATE() - INTERVAL 4 YEAR
64         THEN 'Senior'
65         ELSE 'Junior'
66     END AS StudentLevel
67 FROM Students;
68

```

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

StudentID	FirstName	LastName	EnrollmentDate	StudentLevel
1	John	Doe	2022-08-01	Junior
2	Jane	Smith	2021-08-01	Senior

```

57 • SELECT
58     StudentID,
59     FirstName,
60     LastName,
61     EnrollmentDate,
62     CASE
63         WHEN EnrollmentDate <= CURDATE() - INTERVAL 4 YEAR
64         THEN 'Senior'
65         ELSE 'Junior'
66     END AS StudentLevel
67 FROM Students;
68

```

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

StudentID	FirstName	LastName	EnrollmentDate	StudentLevel
1	John	Doe	2022-08-01	Junior
2	Jane	Smith	2021-08-01	Senior

```

57 • SELECT
58     StudentID,
59     FirstName,
60     LastName,
61     EnrollmentDate,
62     CASE
63         WHEN EnrollmentDate <= CURDATE() - INTERVAL 4 YEAR
64         THEN 'Senior'
65         ELSE 'Junior'
66     END AS StudentLevel
67 FROM Students;
68

```

StudentID	FirstName	LastName	EnrollmentDate	StudentLevel
1	John	Doe	2022-08-01	Junior
2	Jane	Smith	2021-08-01	Senior