FIT3171 Databases
Week 10 Tutorial
SQL Intermediate - Advanced

FIT Database Teaching Team

Complete the week 10 tutorial activities listed below:

10.1 Class Discussion

10.2 SQL Intermediate - Advanced Questions

FIT3171 2022 S1

FIT3171 Databases

Author: FIT Database Teaching Team

License: Copyright © Monash University, unless otherwise stated. All Rights Reserved.


Learning Objectives:

At the completion of these tutorial activities, you should be able to:
- use SQL aggregate functions (SUM, AVG, COUNT)
- code SQL using GROUP BY Clause
- use SQL clause HAVING
- use a subquery in SQL
- use a correlated subquery in SQL
- consolidate your understanding of subqueries
- use different SQL set operators (INTERSECT, DIFFERENCE, UNION).
- outline the role of VIEWs in the database.
- code SQL OUTER join, recursive relationship join and join between tables that have more than one relationship

Important

Remember before starting any lab activity which involves working with files, first use SQL Developer to pull from FIT GitLab server to ensure your local files and the FIT GitLab server files are in sync. *During this activity, you will be creating a set of SQL scripts, these **MUST** be sent to the FIT GitLab server.*

This week we will continue to use the UNIVERSITY database model:
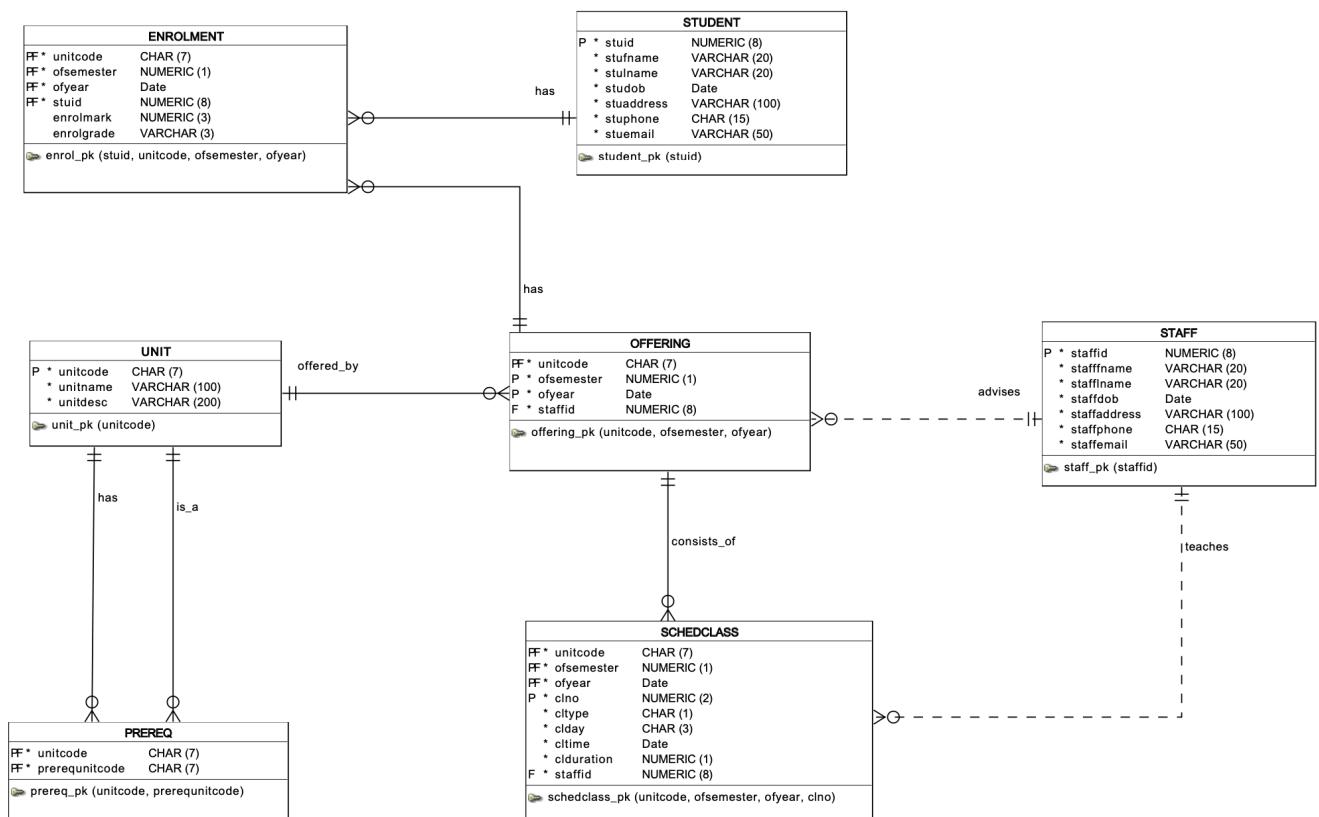
University Data model

10.1 Class Discussion

1. Find the total number of enrolments per semester for each unit in the year 2019. The list should include the unitcode, semester and the total number of enrolment. Order the list in increasing order of enrolment numbers. For units with the same number of enrolments, display them by the unitcode order then by the semester order.

2. Find the oldest student/s in FIT9132. Display the student's id, full name and the date of birth. Sort the list by student id.

3. Assuming that the student name is unique, display Claudette Serman's academic record. Include the unit code, unit name, year, semester, mark and explained_grade in the listing. The Explained Grade column must show Fail for N, Pass for P, Credit for C, Distinction for D and High Distinction for HD. Order the list in increasing order of year, within the same year order the list in increasing order of semester, within the same semester order the list in increasing order of unit code order.

	UNITCODE	UNITNAME	YEAR	OFSEMESTER	ENROLMARK	EXPLAINED_GRADE
1	FIT9132	Introduction to databases	2019	1	56	Pass
2	FIT9136	Algorithms and programming foundations in Python	2019	1	16	Fail
3	FIT9136	Algorithms and programming foundations in Python	2019	2	81	High Distinction
4	FIT9137	Introduction to computer architecture and networks	2019	2	77	Distinction
5	FIT5196	Data wrangling	2020	1	64	Credit

4. Find the total number of prerequisite units for all units. Include in the list the unit code of units that do not have a prerequisite. Order the list in descending order of the number of prerequisite units. If two units have the same number of prerequisite units, order them by the unit code.

	UNITCODE	NO_OF_PREREQ
1	FIT5145	2
2	FIT2094	1
3	FIT3157	1
4	FIT3176	1
5	FIT5196	1
6	FIT1003	0
7	FIT1045	0
8	FIT1050	0
9	FIT9132	0
10	FIT9134	0
11	FIT9136	0
12	FIT9137	0

The sample output is shown for your guidance as to the displayed columns and formats. The data in the database is constantly changing and as a result your actual data displayed (the values in the rows of output) may be different. Therefore, it is important to:
-   check whether your query is correct by manually checking the data in the table/s, and
-   ensuring your query will work for any future possible cases/changes to the data.

10.2 SQL Intermediate - Advanced Questions

Download **week10_sql_inter_adv.sql** from the Week 10 block in Moodle, place this file in your working directory in your App10 folder. Write your answers for question 1 - 19 in the provided area. Test the select statement one by one.

1. Find the maximum mark for FIT9136 in semester 2, 2019.

2. Find the average mark for FIT2094 in semester 2, 2020. Show the average mark with two decimal places. Name the output column as average_mark.

3. List the average mark for each offering of FIT9136. In the listing, include the year and semester number. Sort the result according to the year then the semester.

4. Find the number of students enrolled in FIT1045 in the year 2019, under the following conditions (note two separate selects are required):

    a. Repeat students are counted multiple times in each semester of 2019

    b. Repeat students are only counted once across 2019

5. Find the total number of prerequisite units for FIT5145.

6. Find the total number of prerequisite units for each unit. In the list, include the unit code for which the count is applicable. Order the list by unit code.

7. Find the total number of students whose marks are being withheld (grade is recorded as 'WH') for each unit offered in semester 2 2020. In the listing include the unit code for which the count is applicable. Sort the list by descending order of the total number of students whose marks are being withheld, then by the unit code.

8. For each prerequisite unit, calculate how many times it has been used as a prerequisite (number of times used). In the listing include the prerequisite unit code, the prerequisite unit name and the number of times used. Sort the output by prerequisite unit code.

9. Display the unit code and unit name of units which had at least 2 students who were granted a deferred exam (grade is recorded as 'DEF') in semester 2 2021. Order the list by unit code.

10. Find the unit/s with the highest number of enrolments for each offering in the year 2019. Sort the list by semester then by unit code.

11. Find all students enrolled in FIT3157 in semester 1, 2020 who have scored more than the average mark for FIT3157 in the same offering. Display the students' name and the mark. Sort the list in the order of the mark from the highest to the lowest then in increasing order of student name.

12. Find the number of scheduled classes assigned to each staff member for each semester in 2019. If the number of classes is 2 then this should be labelled as a correct load, more than 2 as an overload and less than 2 as an underload. Include the staff id, staff first name, staff last name, semester, number of scheduled classes and load in the listing. Sort the list by decreasing order of the number of scheduled classes and when the number of classes is the same, sort by increasing order of staff id then by the semester.

	STAFFID	STAFFFNAME	STAFFLNAME	OFSEMESTER	NUMBERCLASSES	LOAD
1	419817	Windham	Ellard	2	6	Overload
2	412994	Gunar	Dutch	1	2	Correct load
3	412994	Gunar	Dutch	2	2	Correct load
4	415448	Sandro	Wethered	1	2	Correct load
5	415448	Sandro	Wethered	2	2	Correct load
6	418454	Lizabeth	Stubbings	2	2	Correct load
7	419421	Trixy	Warner	2	2	Correct load
8	434760	Xena	Epine	1	2	Correct load
9	434760	Xena	Epine	2	2	Correct load
10	436760	Tammi	Soane	1	2	Correct load
11	436760	Tammi	Soane	2	2	Correct load
12	439066	Kennie	Pickin	2	2	Correct load
13	459186	Papageno	Gayton	1	1	Underload
14	459186	Papageno	Gayton	2	1	Underload
15	467165	Deina	MacGarrity	1	1	Underload
16	467165	Deina	MacGarrity	2	1	Underload
17	470313	Gunar	Dutch	1	1	Underload
18	470313	Gunar	Dutch	2	1	Underload
19	475342	Mikol	Kohrt	2	1	Underload
20	475912	Mycah	Preddle	2	1	Underload
21	485533	Worden	Abel	2	1	Underload
22	487823	Benny	Plunket	1	1	Underload
23	487823	Benny	Plunket	2	1	Underload
24	494975	Hildy	Edyson	2	1	Underload
25	497227	Ashleigh	O'Bruen	2	1	Underload
26	498160	Martino	Boram	1	1	Underload
27	498160	Martino	Boram	2	1	Underload

13. Display the unit code and unit name for units that do not have a prerequisite. Order the list in increasing order of unit code. There are many approaches that you can take in writing an SQL statement to answer this query. You can use the SET OPERATORS, OUTER JOIN and a SUBQUERY. Write SQL statements based on *all* three approaches.

	UNITCODE	UNITNAME
1	FIT1003	IT in organisations
2	FIT1045	Algorithms and programming fundamentals in python
3	FIT1050	Web fundamentals
4	FIT9132	Introduction to databases
5	FIT9134	Computer architecture and operating systems
6	FIT9136	Algorithms and programming foundations in Python
7	FIT9137	Introduction to computer architecture and networks

14. List the unit code, semester, number of enrolments and the average mark for each unit offering in 2019. Include offerings without any enrolment in the list. Round the average to 2 digits after the decimal point. If the average result is 'null', display the average as 0.00. All values must be shown with two decimal digits. Order the list in increasing order of average mark, and when the average mark is the same, sort by increasing order of semester then by the unit code.

	UNITCODE	OFSEMESTER	NO_OF_ENROLMENT	AVERAGE_MARK
1	FIT3176	2	0	0.00
2	FIT5196	2	2	57.00
3	FIT9132	1	10	65.20
4	FIT1050	1	10	66.10
5	FIT1050	2	12	68.50
6	FIT9132	2	13	69.31
7	FIT2094	2	9	70.44
8	FIT9137	2	8	71.88
9	FIT5145	2	6	72.00
10	FIT1045	1	10	73.90
11	FIT9136	1	10	74.90
12	FIT3157	2	8	78.25
13	FIT9136	2	11	80.00
14	FIT1045	2	11	83.64

15. List all units offered in semester 2 2019 which do not have any enrolment. Include the unit code, unit name, and the chief examiner's name in the list. Order the list based on the unit code.

	UNITCODE	UNITNAME	CE_NAME
1	FIT3176	Advanced database design	Windham Ellard

16. List the id and full name of students who are enrolled in both 'Introduction to databases' and 'Introduction to computer architecture and networks' (note: both unit names are unique) in semester 1 2020. Order the list by the student id.

	STUID	STUDENT_FULL_NAME
1	19633815	Tessie Rheam
2	20776000	Viviana Brewer

17. Given that the payment rate for a tutorial is $42.85 per hour and the payment rate for a lecture is $75.60 per hour, calculate the weekly payment per type of class for each staff member in semester 1 2020. In the display, include staff id, staff name, type of class (lecture or tutorial), number of classes, number of hours (total duration), and weekly payment (number of hours * payment rate). Order the list by increasing order of staff id and for a given staff id by type of class.

	STAFFID	STAFFNAME	TYPE	NO_OF_CLASSES	TOTAL_HOURS	WEEKLY_PAYMENT
1	412994	Gunar Dutch	Lecture	1	2	$151.20
2	412994	Gunar Dutch	Tutorial	1	2	$85.70
3	415448	Sandro Wethered	Lecture	1	2	$151.20
4	415448	Sandro Wethered	Tutorial	1	2	$85.70
5	418454	Lizabeth Stubbings	Lecture	1	2	$151.20
6	418454	Lizabeth Stubbings	Tutorial	1	2	$85.70
7	419421	Trixy Warner	Lecture	1	2	$151.20
8	419421	Trixy Warner	Tutorial	1	2	$85.70
9	419817	Windham Ellard	Lecture	3	4	$302.40
10	419817	Windham Ellard	Tutorial	3	6	$257.10
11	434760	Xena Epine	Lecture	1	1	$75.60
12	434760	Xena Epine	Tutorial	1	2	$85.70
13	436760	Tammi Soane	Lecture	1	1	$75.60
14	436760	Tammi Soane	Tutorial	1	2	$85.70
15	439066	Kennie Pickin	Lecture	1	1	$75.60
16	439066	Kennie Pickin	Tutorial	1	2	$85.70
17	459186	Papageno Gayton	Tutorial	1	2	$85.70
18	467165	Deina MacGarrity	Tutorial	1	2	$85.70
19	470313	Gunar Dutch	Tutorial	1	2	$85.70
20	475342	Mikol Kohrt	Tutorial	1	2	$85.70
21	475912	Mycah Preddle	Tutorial	1	2	$85.70
22	485533	Worden Abel	Tutorial	1	2	$85.70
23	487823	Benny Plunket	Tutorial	1	2	$85.70
24	494975	Hildy Edyson	Tutorial	1	2	$85.70
25	497227	Ashleigh O'Bruen	Tutorial	1	2	$85.70
26	498160	Martino Boram	Tutorial	1	2	$85.70

18. Given that the payment rate for a tutorial is $42.85 per hour and the payment rate for a lecture is $75.60 per hour, calculate the total weekly payment (the sum of both tutorial and lecture payments) for each staff member in semester 1 2020. In the display, include staff id, staff name, total weekly payment for tutorials, total weekly payment for lectures and the total weekly payment. If the payment is null, show it as $0.00. Order the list by increasing order of staff id.

	STAFFID	STAFFNAME	TUTORIAL_PAYMENT	LECTURE_PAYMENT	TOTAL_WEEKLY_PAYMENT
1	412994	Gunar Dutch	$85.70	$151.20	$236.90
2	415448	Sandro Wethered	$85.70	$151.20	$236.90
3	418454	Lizabeth Stubbings	$85.70	$151.20	$236.90
4	419421	Trixy Warner	$85.70	$151.20	$236.90
5	419817	Windham Ellard	$257.10	$302.40	$559.50
6	434760	Xena Epine	$85.70	$75.60	$161.30
7	436760	Tammi Soane	$85.70	$75.60	$161.30
8	439066	Kennie Pickin	$85.70	$75.60	$161.30
9	459186	Papageno Gayton	$85.70	$0.00	$85.70
10	467165	Deina MacGarrity	$85.70	$0.00	$85.70
11	470313	Gunar Dutch	$85.70	$0.00	$85.70
12	475342	Mikol Kohrt	$85.70	$0.00	$85.70
13	475912	Mycah Preddle	$85.70	$0.00	$85.70
14	485533	Worden Abel	$85.70	$0.00	$85.70
15	487823	Benny Plunket	$85.70	$0.00	$85.70
16	494975	Hildy Edyson	$85.70	$0.00	$85.70
17	497227	Ashleigh O'Bruen	$85.70	$0.00	$85.70
18	498160	Martino Boram	$85.70	$0.00	$85.70

19. Assume that all units are worth 6 credit points each, calculate each student's Weighted Average Mark (WAM) and GPA. Please refer to these Monash websites: https://www.monash.edu/exams/results/wam and https://www.monash.edu/exams/results/gpa for more information about WAM and GPA respectively. Do not include NULL, WH or DEF grade in the calculation.

Calculation example for student 14374036 (Claudette Serman):

	UNITCODE	YEAR	OFSEMESTER	ENROLMARK	ENROLGRADE
1	FIT9132	2019	1	56	P
2	FIT9136	2019	1	16	N
3	FIT9136	2019	2	81	HD
4	FIT9137	2019	2	77	D
5	FIT5196	2020	1	64	C

WAM = (56x6 + 16x6 + 81x6 + 77x6 + 64x6)/(6+6+6+6+6) = 58.80
GPA = (1x6+ 0.3x6 + 4x6 + 3x6 + 2x6)/(6+6+6+6+6) = 2.06

Calculation example for student 23545528 (Benny Plunket):

	UNITCODE	YEAR	OFSEMESTER	ENROLMARK	ENROLGRADE
1	FIT1045	2020	1	53	P
2	FIT1050	2020	1	97	HD
3	FIT2094	2020	2	78	D
4	FIT3157	2020	2	94	HD
5	FIT3176	2021	1	85	HD

WAM = (53x3 + 97x3 + 78x6 + 94x6 + 85 x 6)/(3+3+6+6+6) = 83.00
GPA = (1x6 + 4x6+ 3x6 + 4x6 + 4x6)/(6+6+6+6+6) = 3.20

Include student id, student full name (in a 40 characters wide column headed student_fullname), WAM and GPA in the display. Order the list by descending order of WAM then descending order of GPA. If two students have the same WAM and GPA, order them by their respective id.

Only some data shown:

	STUID	STUDENT_FULLNAME	WAM	GPA
1	12511467	Francyne Rigney	89.17	4.00
2	20648900	Aleda Whistan	86.75	3.75
3	14615430	Siffre Dibdale	85.25	3.50
4	55804738	Odie Portail	84.00	3.50
5	90237362	Flor Pickless	83.00	3.50
6	23545528	Benny Plunket	83.00	3.20
7	17013887	Harv Wethered	81.13	3.40
8	15527149	Jehanna Gheraldi	80.63	3.26
9	40098507	Cyrus Putten	79.50	3.08
10	13390148	Brier Kilgour	79.17	2.75
11	24920425	Mitchell Hilbourne	78.60	2.80
12	18063424	Lynnell Cliburn	78.25	3.40
13	44472707	Perla Broschek	78.20	3.33
14	18841033	Artus Swiffen	77.60	3.00
15	13028303	Herculie Mendus	77.33	3.00
16	21585680	Elihu Fer	77.14	2.50
17	44064793	Zane Roffe	76.25	3.00
18	13453333	Pierrette Moynihan	75.88	3.00
19	19568650	Kary Rayburn	75.63	2.86

Important

You need to get into the habit of establishing this as a standard FIT3171 workflow - pull at the start of your working session, work on the activities you wish to/are able to complete during this session, save the files, add all (stage), commit and then push the changes back to the FIT GitLab server