CS 348 — Assignment 4

Lara Janecka

Fall 2015

---

1 Question 1

```
-- Question: 1
-- Approach: This joins in the course information to get the course name
--     it then joins the schedule information matching on cno, term, and
--      section to prevent duplication
--      finally the query is filtered for rows where the term and room are
--      appropriate values
--      select choses the correct columns to print
select course.cname, course.cno, professor.pname from class
    inner join course
    on class.cno = course.cno
    inner join schedule
    on class.cno = schedule.cno and class.term = schedule.term
        and class.section = schedule.section
    inner join professor
    on class.instructor = professor.eid
    where class.term="F10" AND schedule.room="RCH122"
    ;
1
-- Results:
/*
CNAME                                                CNO         PNAME
---------------------------------------------------  ----------  ----------
Elementary Algorithm Design and Data Abstraction...  CS146       HOLLIS
Computer Architecture                                CS450       BARBARA
Computer Security and Privacy                        CS458       HOLLIS
Advanced Biochemistry                                PHYS442     LILIA
Research Project                                     PHYS451     DEBORA
*/
```

2 Question 2

```
-- Question: 2
-- Approach: This works by joining in the professor table into the class table
--      for all professors in the cs department
--   then the prerequisite table is joined to find the prereqs for a class
--   then the class and professor tables are joined in again to get information
--      for the prerequisite classes such that the prerequisite happened at the
--      same time and is taught by a professor not in the cs department
--   then the table is grouped by relevant information to filter out
--      duplications and is sorted
--   finally the last where function filters for the correct term and the select
--      renames columns to the required values

select class.cno as CNO, professor.pname as CSNAME,
 prereqclass.cno as PrerequisiteCNO, prereqprof.pname as NonCSNAME  from class
    inner join professor
    on class.instructor = professor.eid and professor.dept = "Computer Science"
    inner join prerequisite
    on class.cno = prerequisite.cno
    inner join class prereqclass
    on prereqclass.cno = prerequisite.prereq and prereqclass.term = "S03"
    inner join professor prereqprof
    on prereqclass.instructor = prereqprof.eid
        and prereqprof.dept != "Computer Science"

    where class.term = "S03"
    group by class.cno, professor.pname, prereqclass.cno, prereqprof.pname
    order by class.cno, professor.pname, prereqclass.cno, prereqprof.pname
    ;
/*
-- Results:
CNO         CSNAME      PrerequisiteCNO  NonCSNAME
----------  ----------  ---------------  ----------
CS214       JERALD      PHYS171          DUSTIN
CS348       RODRIGO     MATH217          JESSICA
CS436       RODRIGO     MATH381          PETER
CS446       LINA        PHYS171          DUSTIN
*/
```

3 Question 3

```
-- Question: 3
-- Approach: This works by joining in the professor table into the class table
--       for all professors in the cs department
```

```
--   then the prerequisite table is joined to find the prereqs for a class
--   then the class and professor tables are joined in again to get information
--      for the prerequisite classes such that the prerequisite happened at the
--      same time and is taught by a professor not in the cs department
--   then the table is grouped by relevant information to filter out
--      duplications and is sorted
--   finally the last where function filters for the correct term and the select
--      renames columns to the required values


select professor.pname, professor.dept, class.cno, class.term, class.section,
ROUND(AVG(enrollment.mark),2) as AVG from class
    inner join enrollment
    on class.cno = enrollment.cno and class.term = enrollment.term
    inner join professor
    on class.instructor = professor.eid

    group by class.cno, class.term, class.section
    having AVG(enrollment.mark) > 85 and COUNT(enrollment.cno) > 10

    order by professor.pname, professor.dept, class.cno
    ;
/*
-- Results:
PNAME       DEPT               CNO         TERM        SECTION     AVG
----------  -----------------  ----------  ----------  ----------  ----------
ANN         Computer Science   CS213       F11         2           85.92
BARBARA     Computer Science   CS137       S02         3           85.25
BARBARA     Computer Science   CS213       W13         1           85.46
BARBARA     Computer Science   CS213       W13         3           85.46
BETH        Physics            PHYS457     F01         2           86.64
BETH        Physics            PHYS458     S07         1           85.92
CODY        Computer Science   CS462       S06         2           87.58
DEBORA      Physics            PHYS451     F10         1           86.83
DEBORA      Physics            PHYS478     F01         3           85.31
DUSTIN      Physics            PHYS214     W12         2           86.18
DUSTIN      Physics            PHYS271     W13         3           85.17
DUSTIN      Physics            PHYS312     W12         2           85.46
DUSTIN      Physics            PHYS477     W11         3           86.0
HERMINIA    Physics            PHYS229     S06         2           86.27
HERMINIA    Physics            PHYS312     W12         1           85.46
HOLLIS      Computer Science   CS462       S06         3           87.58
JACKLYN     Physics            PHYS147     W07         2           86.09
JERALD      Computer Science   CS137       S02         1           85.25
JERALD      Computer Science   CS138       F10         2           85.08
JESSICA     Mathematics        MATH226     F12         1           85.55
LILIA       Physics            PHYS189     F11         2           86.92
```

```
LINA        Computer Science  CS350       S03         1           85.42
MORRIS      Physics           PHYS271     W13         1           85.17
MORRIS      Physics           PHYS458     S07         3           85.92
PETER       Mathematics       MATH358     W00         1           86.25
PETER       Mathematics       MATH452     F08         3           86.73
RODRIGO     Computer Science  CS135       S11         2           86.0
RODRIGO     Computer Science  CS350       S03         2           85.42
STUART      Physics           PHYS289     F09         2           85.5
ULYSSES     Physics           PHYS229     S06         3           86.27
ULYSSES     Physics           PHYS312     F11         1           85.08
ULYSSES     Physics           PHYS331     S05         1           86.73
ULYSSES     Physics           PHYS478     F01         2           85.31

*/
```

4 Question 4

```
-- Question: 4
-- Approach: join the enrollment table with itself and then grouped by the
--      original data so that the difference of the student's mark to the
--      average can be computed
--      use a where statement to filter based on the initial mark is greater
--      than 98
-- Assumptions: for the purposes of an average, a class is determined by its cno
--      and term so the average of a class is across all sections when being
--      subtracted from the student's mark


select enrollment.sno, enrollment.cno,
    enrollment.mark -  ROUND(AVG(enrol2.mark), 2) as DIFF
    from enrollment
    left join enrollment enrol2
    on enrollment.cno = enrol2.cno and enrollment.term = enrol2.term

    where enrollment.mark > 98

    group by enrollment.cno, enrollment.term

    order by enrollment.sno, enrollment.cno
    ;


/*
-- Results:
SNO         CNO         DIFF
```

20002939	CS482	17.56
20037890	PHYS255	17.36
20099883	CS488	16.24
20163094	CS247	14.56
20163094	PHYS497	17.33
20193175	CS449	16.0
20304343	MATH285	15.87
20458699	MATH210	14.37
20591079	PHYS420	15.67
20591079	PHYS451	19.13
20602987	CS312	15.12
20602987	CS458	17.96
20652458	PHYS364	12.78
20842491	PHYS484	14.14
20862191	MATH334	17.67
20938960	PHYS285	19.69
20954626	MATH395	16.37
21090840	PHYS193	14.62
21152057	PHYS289	14.5
21210918	CS485	18.74
21267773	PHYS124	20.28
21288103	PHYS451	12.17
21407646	PHYS263	17.2
21433509	PHYS446	13.0
21495049	PHYS458	17.57
21568625	MATH399	15.4
21646983	PHYS361	20.62
21649510	PHYS221	12.33
21670504	PHYS485	10.5
21676293	CS488	17.27
21750774	PHYS483	15.82
21767530	MATH261	19.09
21768045	MATH250	16.57
21818793	CS485	17.18
21821244	MATH491	14.44
21906995	CS432	16.0
21947461	CS316	17.96
21955374	CS484	12.9
21960289	PHYS414	14.36
21993697	MATH226	17.11
22046341	MATH452	12.6
22078262	MATH355	18.03
22215932	MATH217	17.47
22233056	MATH261	17.63
22289300	PHYS388	13.0
22296412	CS486	16.6

22318590	PHYS133	19.21
22468169	MATH479	17.0
22492414	CS314	17.38
22556132	CS489	17.68
22564251	CS476	15.06
22788590	CS487	17.59
23008065	PHYS485	14.33
23010891	MATH382	20.76
23049532	CS312	15.96
23067117	MATH192	18.37
23067117	MATH427	19.57
23193544	CS316	18.06
23214247	CS316	15.93
23240776	PHYS312	17.84
23267930	PHYS271	16.5
23314441	MATH340	17.5
23381814	CS213	18.3
23442335	CS436	17.52
23486913	CS370	17.65
23543938	PHYS460	13.86
23591313	PHYS484	14.62
23749666	CS218	18.45
23759303	PHYS147	12.91
23805142	CS247	14.33
24002271	CS486	17.71
24077663	PHYS368	16.6
24125617	MATH334	16.14
24163544	MATH245	16.12
24169469	MATH358	17.29
24176094	PHYS452	16.6
24236259	PHYS477	18.91
24287823	MATH427	16.9
24294227	CS348	17.82
24410026	CS215	17.6
24596751	MATH217	18.8
24613564	CS338	24.34
24637521	CS314	15.55
24860703	PHYS147	16.73
24885811	PHYS194	17.71
24886621	CS446	12.3
24888192	CS350	16.38
24989644	PHYS193	17.33
25129806	CS137	16.06
25132581	CS476	17.14
25286278	MATH192	18.69
25330788	MATH382	17.25
25365706	CS313	15.67

25664679	MATH424	18.61
25672769	CS436	15.2
25743829	MATH459	14.1
25799060	MATH362	18.24
25800239	CS135	17.16
25827649	PHYS420	11.4
25830348	PHYS254	14.37
25835768	MATH217	18.0
25984438	MATH441	19.6
25987170	PHYS398	20.36
26044854	CS450	15.5
26207553	PHYS478	16.09
26247302	PHYS244	18.25
26256419	MATH383	15.29
26342477	PHYS388	17.75
26523244	CS487	17.82
26570974	CS462	18.71
26607054	MATH358	14.8
26627237	MATH221	15.33
26653529	CS115	19.93
26836481	MATH381	15.86
27007594	PHYS271	15.06
27007594	PHYS317	21.39
27007594	PHYS458	19.56
27168359	PHYS420	17.64
27279702	CS490	16.92
27279702	PHYS290	16.8
27279702	PHYS485	15.91
27292661	MATH285	16.44
27357294	PHYS271	17.75
27386745	MATH192	17.7
27421447	MATH197	18.0
27464811	CS247	14.12
27493259	PHYS457	12.36
27532359	CS230	14.9
27657997	MATH375	14.56
27717039	MATH381	17.5
27722125	CS246	16.0
27729518	CS147	18.22
27729902	MATH382	16.58
27792707	PHYS255	18.11
27902980	PHYS484	18.46
28013643	CS146	15.0
28025345	MATH428	17.17
28192879	PHYS337	17.12
28231389	MATH459	12.3
28234450	MATH261	16.0

```
28471310    PHYS257     19.29
28473457    MATH399     18.83
28513198    MATH226     15.8
28539436    CS240       18.29
28580040    CS458       21.1
28638811    PHYS474     17.37
28921899    PHYS414     13.8
29017604    PHYS455     16.7
29050897    MATH395     17.21
29067979    MATH399     16.06
29119934    CS240       10.75
29171697    MATH197     15.0
29466896    CS312       17.75
29501010    PHYS368     18.57
29507951    MATH452     16.19
29615296    PHYS317     20.0
29643474    CS211       16.14
29808557    PHYS254     19.04
29877646    CS230       13.0
*/
```

5 Question 5

```
-- Question: 5
-- Approach join the enrollment table with the student table based on the
--      student number inorder to get the student name for printing
--      group by the student number to get all classes attended by that student
--      to calculate and filter on the number of classes and minimum mark
-- Assumptions: the student minimum enrollment constraint is not term based
--      (this query checks over all time)

select student.sname, ROUND(AVG(enrollment.mark), 2) as AVG from enrollment
    inner join student
    on enrollment.sno = student.sno

    group by student.sno
    having COUNT(enrollment.sno) >= 5 AND MIN(enrollment.mark) >= 80

    order by student.sname, ROUND(AVG(enrollment.sno),2)
    ;
/*
-- Results:
SNAME        AVG
-----------  ----------
ANITA, STEVE  84.0
```

```
CHRISTIAN, R  86.0
KAREN, BENJA  87.4
SETH, GERMAI  82.33
SUSIE, VIRGI  84.0
*/
```

6 Question 6

```
-- Question: 6
-- Approach: Join the class and enrollment tables to calculate the class count
--      and average, then join in the professor table and group by the
--      professor's id to filter out any professor whose minimum number of
--      students taught is not at least 10. This allows us to return
--      the correct data.

select professor.eid, classdata.cno, classdata.term, classdata.section,
 classdata.AVG
from professor
    inner join (select class.cno, class.term, class.instructor, class.section,
        COUNT(enrollment.sno) as CNT, ROUND(AVG(enrollment.mark), 2) as AVG from
        enrollment
        inner join class
        on class.cno = enrollment.cno

        group by class.cno, class.term) as classdata
    on professor.eid = classdata.instructor

    group by professor.eid
    having MIN(CNT) >= 10
    ;

/*
-- Results:
EID         cno         term        section     AVG
----------  ----------  ----------  ----------  ----------
0           MATH492     F99         2           75.95
2           PHYS243     F01         3           78.85
4           MATH492     F01         1           75.95
5           MATH479     F99         2           76.62
6           CS447       F99         3           78.53
7           PHYS457     S01         3           82.03
8           CS490       W12         2           78.83
9           PHYS488     S09         2           81.72
10          PHYS442     W11         3           79.99
11          PHYS497     W09         2           81.46
```

```
12          CS490       W04         3           78.83
13          CS448       W09         2           81.47
14          MATH492     S00         2           75.95
15          PHYS452     F04         3           80.67
16          CS448       W08         3           81.47
17          CS490       W06         2           78.83
18          MATH492     W08         2           75.95
19          MATH486     F01         2           79.27
20          CS448       W10         1           81.47
21          CS488       F09         3           81.11
23          CS490       W09         3           78.83
24          CS445       F99         2           77.41
28          PHYS497     W11         1           81.46
29          PHYS497     S00         2           81.46
32          PHYS497     S12         2           81.46
33          PHYS497     W05         3           81.46
37          PHYS497     F02         2           81.46
38          PHYS497     S05         3           81.46
39          PHYS497     W12         3           81.46
40          CS490       W11         3           78.83
41          PHYS497     W04         3           81.46
46          PHYS497     W07         1           81.46
48          PHYS485     W03         2           82.03

*/
```