CS532 : Project 1

**Name** : Shriram Suryawanshi (**CS532**)

**B-Number**: B00734421 (Spring 2018)

**Name** : Vinen Furtado (**CS532**)

**B-Number**: B00717015 (Spring 2018)

1. **Find the dept\_code, course# and title of each course that was offered in the Spring semester of 2018. In the output, dept\_code and course# of each course should be concatenated under a new column header course\_id.**

SELECT concat(co.dept\_code, co.course#) AS course\_id, title FROM courses co , classes cl

WHERE co.course# = cl.course# AND cl.year = 2018 AND cl.semester='Spring';

|  |  |
| --- | --- |
| **COURSE\_ID** | **TITLE** |
| CS432 | database systems |
| Math221 | calculus I |
| CS532 | database systems |

1. **Find the first name of each student who has taken at least one CS course and at least one math course.**

SELECT firstname FROM students

WHERE sid IN

((SELECT sid FROM enrollments en JOIN classes cl ON en.classid = cl.classid AND cl.dept\_code = 'Math')

INTERSECT

(SELECT sid FROM enrollments en JOIN classes cl ON en.classid = cl.classid AND cl.dept\_code = 'CS'));

|  |
| --- |
| **FIRSTNAME** |
| Anne |

1. **Find the dept\_code and course# of each course that was not offered in 2017.**

SELECT dept\_code, course# FROM courses

MINUS

(SELECT dept\_code, course# FROM classes WHERE year = 2017);

|  |  |
| --- | --- |
| **DEPT** | **COURSE#** |
| BIOL | 425 |
| CS | 532 |
| CS | 552 |
| Math | 221 |

1. **Find the sid, firstname and GPA of each undergraduate student who has received an A for at least one course he/she has taken.**

SELECT DISTINCT st.sid, st.firstname, st.gpa FROM students st, enrollments en

WHERE status != 'graduate' AND st.sid = en.sid AND en.lgrade = 'A';

|  |  |  |
| --- | --- | --- |
| **SID** | **FIRSTNAME** | **GPA** |
| B003 | Tracy | 4 |
| B001 | Anne | 3.17 |
| B007 | Becky | 4 |

1. **Find the firstname of each student who has never received an A for any course he/she has taken. If you write a nested query, make sure the subquery is uncorrelated.**

SELECT firstname FROM students

WHERE sid in

((SELECT st.sid FROM students st, enrollments en WHERE st.sid = en.sid)

MINUS

(SELECT st.sid FROM students st, enrollments en WHERE st.sid = en.sid AND en.lgrade = 'A'));

|  |
| --- |
| **FIRSTNAME** |
| Terry |
| Barbara |
| Jack |

1. **Find the sid and firstname of each student who has received an A for every course he/she has taken. Count only classes for which he/she received a non-null grade. GPA information is not permitted to be used in this query.**

SELECT sid, firstname FROM students

WHERE sid in

((SELECT sid FROM enrollments WHERE lgrade = 'A' AND lgrade IS NOT NULL)

MINUS

(SELECT e.sid FROM enrollments e WHERE e.sid = sid AND e.lgrade != 'A' AND e.lgrade IS NOT NULL));

|  |  |
| --- | --- |
| **SID** | **FIRSTNAME** |
| B003 | Tracy |
| B006 | Terry |
| B007 | Becky |

1. **Find the dept\_code and course# of each course that has been offered the most number of times (each record in the classes table corresponds to a course offering). Note that it is possible that more than one course may satisfy this query condition; in this case, all such courses should be retrieved.**

SELECT dept\_code, course# FROM

(SELECT dept\_code, course#, COUNT(\*) AS count1 FROM classes GROUP BY dept\_code, course#)

WHERE count1 IN (SELECT MAX(COUNT(\*)) AS count1 FROM classes GROUP BY dept\_code, course#);

|  |  |
| --- | --- |
| **DEPT** | **COURSE#** |
| CS | 432 |
| Math | 314 |

1. **Find the classid, dept\_code and course# of each class offered in Spring 2018 that is not full and for each such class, also list the number of seats available (computed by limit – class\_size) under the header “seats\_available’.**

SELECT classid, dept\_code, course#, (limit - class\_size) AS seats\_available FROM classes

WHERE year = 2018 AND semester = 'Spring' AND limit != class\_size;

|  |  |  |  |
| --- | --- | --- | --- |
| **CLASS** | **DEPT** | **COURSE#** | **SEATS\_AVAILABLE** |
| c0001 | CS | 432 | 1 |
| c0006 | CS | 532 | 1 |

1. **Find every student (all attributes are needed) who has taken more than 4 classes.**

SELECT st.\* FROM students st

JOIN (SELECT sid FROM enrollments GROUP BY sid HAVING COUNT(classid) > 4) en ON en.sid = st.sid**;**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **SID** | **FIRSTNAME** | **LASTNAME** | **STATUS** | **GPA** | **EMAIL** |
| B001 | Anne | Broder | junior | 3.17 | broder@bu.edu |

1. **Find every class (all attributes are needed) that is offered by the CS department in the Spring semester of 2018 and has less than 3 students enrolled. For this query, you are not allowed to use the size information from the classes table.**

SELECT cl.\* FROM classes cl JOIN (SELECT classid FROM enrollments GROUP BY classid HAVING COUNT(classid) < 3) en

ON en.classid = cl.classid WHERE cl.dept\_code = 'CS' AND cl.year = 2018 AND cl.semester ='Spring';

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **CLASS** | **DEPT** | **COURSE#** | **SECT#** | **YEAR** | **SEMEST** | **LIMIT** | **CLASS\_SIZE** |
| c0001 | CS | 432 | 1 | 2018 | Spring | 35 | 34 |

1. **Find the sid and first name of every student who has taken all 400-level CS courses. Here we are referring to courses, not classes.**
2. **A**
3. **A**
4. **A**
5. **A**
6. **A**
7. **A**
8. **A**
9. **A**
10. **A**