1. write a SQL query to find those employees who receive a higher salary than the employee with ID 163. Return first name, last name.

select first\_name,last\_name from employees

where salary > (select salary from employees where employee\_id = 163);

|  |  |
| --- | --- |
| Steven | King |
| Neena | Kochhar |
| Lex | De Haan |
| Nancy | Greenberg |
| Den | Raphaely |
| John | Russell |
| Karen | Partners |
| Alberto | Errazuriz |
| Gerald | Cambrault |
| Eleni | Zlotkey |
| Peter | Tucker |
| Janette | King |
| Clara | Vishney |
| Lisa | Ozer |
| Harrison | Bloom |
| Tayler | Fox |
| Ellen | Abel |
| Michael | Hartstein |
| Hermann | Baer |
| Shelley | Higgins |

2. From the following table, write a SQL query to find out which employees have the same designation as the employee whose ID is 169. Return first name, last name, department ID and job ID.

select first\_name,last\_name,department\_ID,job\_id from employees

where job\_id = (select job\_id from employees where employee\_id = 169);

|  |  |  |  |
| --- | --- | --- | --- |
| Peter | Tucker | 80 | SA\_REP |
| David | Bernstein | 80 | SA\_REP |
| Peter | Hall | 80 | SA\_REP |
| Christopher | Olsen | 80 | SA\_REP |
| Nanette | Cambrault | 80 | SA\_REP |
| Oliver | Tuvault | 80 | SA\_REP |
| Janette | King | 80 | SA\_REP |
| Patrick | Sully | 80 | SA\_REP |
| Allan | McEwen | 80 | SA\_REP |
| Lindsey | Smith | 80 | SA\_REP |
| Louise | Doran | 80 | SA\_REP |
| Sarath | Sewall | 80 | SA\_REP |
| Clara | Vishney | 80 | SA\_REP |
| Danielle | Greene | 80 | SA\_REP |
| Mattea | Marvins | 8 | SA\_REP |
| David | Lee | 80 | SA\_REP |
| Sundar | Ande | 80 | SA\_REP |
| Amit | Banda | 80 | SA\_REP |
| Lisa | Ozer | 80 | SA\_REP |
| Harrison | Bloom | 80 | SA\_REP |
| Tayler | Fox | 80 | SA\_REP |
| William | Smith | 80 | SA\_REP |
| Elizabeth | Bates | 80 | SA\_REP |
| Sundita | Kumar | 80 | SA\_REP |
| Ellen | Abel | 80 | SA\_REP |
| Alyssa | Hutton | 80 | SA\_REP |
| Jonathon | Taylor | 80 | SA\_REP |
| Jack | Livingston | 80 | SA\_REP |
| Kimberely | Grant | 0 | SA\_REP |
| Charles | Johnson | 80 | SA\_REP |

**3.** From the following table, write a SQL query to find those employees whose salary matches the lowest salary of any of the departments. Return first name, last name and department ID.

select first\_name,last\_name,department\_ID,job\_id from employees

where salary = (SELECT MIN(salary) AS lowest\_salary

FROM employees where department\_id = 20

GROUP BY department\_id);

|  |  |  |  |
| --- | --- | --- | --- |
| Bruce | Ernst | 60 | IT\_PROG |
| Pat | Fay | 20 | MK\_REP |

4. From the following table, write a SQL query to find those employees who earn more than the average salary. Return employee ID, first name, last name.

select employee\_id,first\_name,last\_name from employees

where salary > (select avg(salary) from employees);

|  |  |  |
| --- | --- | --- |
| 100 | Steven | King |
| 101 | Neena | Kochhar |
| 102 | Lex | De Haan |
| 103 | Alexander | Hunold |
| 108 | Nancy | Greenberg |
| 109 | Daniel | Faviet |
| 110 | John | Chen |
| 111 | Ismael | Sciarra |
| 112 | Jose Manuel | Urman |
| 113 | Luis | Popp |
| 114 | Den | Raphaely |
| 120 | Matthew | Weiss |
| 121 | Adam | Fripp |
| 122 | Payam | Kaufling |
| 123 | Shanta | Vollman |
| 145 | John | Russell |
| 146 | Karen | Partners |
| 147 | Alberto | Errazuriz |
| 148 | Gerald | Cambrault |
| 149 | Eleni | Zlotkey |
| 150 | Peter | Tucker |
| 151 | David | Bernstein |
| 152 | Peter | Hall |
| 153 | Christopher | Olsen |
| 154 | Nanette | Cambrault |
| 155 | Oliver | Tuvault |
| 156 | Janette | King |
| 157 | Patrick | Sully |
| 158 | Allan | McEwen |
| 159 | Lindsey | Smith |
| 160 | Louise | Doran |
| 161 | Sarath | Sewall |
| 162 | Clara | Vishney |
| 163 | Danielle | Greene |
| 164 | Mattea | Marvins |
| 165 | David | Lee |
| 168 | Lisa | Ozer |
| 169 | Harrison | Bloom |
| 170 | Tayler | Fox |
| 171 | William | Smith |
| 172 | Elizabeth | Bates |
| 174 | Ellen | Abel |
| 175 | Alyssa | Hutton |
| 176 | Jonathon | Taylor |
| 177 | Jack | Livingston |
| 178 | Kimberely | Grant |
| 201 | Michael | Hartstein |
| 203 | Susan | Mavris |
| 204 | Hermann | Baer |
| 205 | Shelley | Higgins |
| 206 | William | Gietz |

PART II

* List all courses that Alice (student\_id 1) is taking.

select course\_name from courses where studentId = 1;

|  |
| --- |
| Math101 |
| Science101 |

* List the names of students who are taking "Math 101."

select name from student s,courses c where c.studentID = s.studentID and course\_name = 'Math101';

|  |
| --- |
| Alice |
| Bob |
| David |

* Find the names of students who are older than Bob (student\_id 2).

select name from student where age>(select age from student where studentId = 2);

David

* List all students who are taking the same courses as Alice (student\_id 1).

select distinct name from student s,courses c

where c.course\_id in (select course\_id from courses where studentId =1);ü List all students who are not taking any course.

|  |
| --- |
| Alice |
| Bob |
| Charlie |
| David |
| Emily |

* Find the courses that have the most students enrolled.

select course\_name,count(studentId) as count from courses group by course\_name limit 2

|  |  |
| --- | --- |
| Math101 | 3 |
| Science101 | 3 |

* List the names of students who are taking the same courses as David (student\_id 4) but are younger than him.

select course\_name,name from courses c, student s

where c.studentId = s.studentId and course\_name in (select course\_name from courses where studentId = 4) and

age < (select age from student s where s.studentId = 4)

|  |  |
| --- | --- |
| Math101 | Alice |
| Math101 | Bob |

* Find the average age of students who are taking "Science 101."

select avg(s.age) from student s join courses c

where c.studentId = s.studentId and c.course\_name = 'Science101'

group by course\_name;

20.0000

* List the students who are taking all available courses.

select \* from student , courses where student.studentId = courses.studentId

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 1 | Alice | 20 | 101 | Math101 | 1 |
| 1 | Alice | 20 | 102 | Science101 | 1 |
| 2 | Bob | 22 | 103 | Math101 | 2 |
| 2 | Bob | 22 | 104 | History101 | 2 |
| 3 | Charlie | 21 | 105 | Science101 | 3 |
| 4 | David | 23 | 106 | English101 | 4 |
| 4 | David | 23 | 107 | Math101 | 4 |
| 5 | Emily | 19 | 108 | Science101 | 5 |
| 5 | Emily | 19 | 109 | History101 | 5 |

* List the students who are taking at least one course but are not taking "Math 101."

SELECT s.name

FROM student s

WHERE s.studentId NOT IN (

SELECT s.studentId

FROM student s

JOIN courses c ON s.studentId = c.studentId

WHERE c.course\_name = 'Math101'

)

|  |
| --- |
| Charlie |
| Emily |

* List the students who are taking more courses than Bob (student\_id 2).

select student.studentId,student.name from student join

(select c.studentId, count(c.course\_id) as cnt

from courses c group by c.studentId) student\_courses on

student.studentId = student\_courses.studentId

where student\_courses.cnt > (select count(course\_id) from courses where studentId = 2)

0 rows

* Find the students who are taking courses with names containing the word "Science."

select studentId,name from student where studentId in

(select studentId from courses where course\_Name like 'Science%')

|  |  |
| --- | --- |
| 1 | Alice |
| 3 | Charlie |
| 5 | Emily |
|  |  |

* List the courses that have more than two students enrolled.

select student.studentId,student.name from student join

(select c.studentId, count(c.course\_id) as cnt

from courses c group by c.studentId) student\_courses on

student.studentId = student\_courses.studentId

where student\_courses.cnt > 2

0 rows

* Find the names of students who are not taking "History 101."

select studentId,name from student where studentId not in

(select studentId from courses where course\_Name like 'History%')

|  |  |
| --- | --- |
| 1 | Alice |
| 3 | Charlie |
| 4 | David |
|  |  |

* List the courses that are not taken by any student.

select course\_id,course\_name from courses

where studentId not in (select studentID from student )

0 rows

* Find the names of students who are taking the same courses as Alice (student\_id 1) and Bob (student\_id 2).

select course\_id,course\_name from courses

where studentId not in (select studentID from student )

|  |  |
| --- | --- |
| 1 | Alice |
| 2 | Bob |
|  |  |

* List the courses that have the highest average age among students taking them.

(select c.studentId, c.course\_name, avg(c.course\_id) as a

from courses c group by c.studentId,c.course\_name order by a desc limit 1)

|  |  |  |
| --- | --- | --- |
| 5 | History101 | 109.0000 |

* Find the students who are taking the same courses as David (student\_id 4) but are not older than him.

select course\_name,name from courses c, student s

where c.studentId = s.studentId and course\_name in (select course\_name from courses where studentId = 4) and

age < (select age from student s where s.studentId = 4)

|  |  |
| --- | --- |
| Math101 | Alice |
| Math101 | Bob |

* List the courses that are taken by students aged 20 or older.

select course\_id,course\_name from courses

where studentId in (select studentID from student where age>20)

|  |  |
| --- | --- |
| 103 | Math101 |
| 104 | History101 |
| 105 | Science101 |
| 106 | English101 |
| 107 | Math101 |

* Find the names of students who are taking at least one course with the word "Math" in the course name and are younger than 22 years old.

select \* from courses

where studentId in (select studentID from student where age<22)

|  |  |  |
| --- | --- | --- |
| 101 | Math101 | 1 |
| 102 | Science101 | 1 |
| 105 | Science101 | 3 |
| 108 | Science101 | 5 |
| 109 | History101 | 5 |