**Database Systems Quiz 6**

**U00839259 (syrrmlla)**

1. [**100 pts**] Using the companydb database you generated in the last quiz (Quiz 5), write SQL scripts in the MySQL environment for the following use cases (no screenshots needed):
2. **[10 pts]** Using the IN clause, retrieve only the social security numbers of all employees who work on project numbers 61, 62, 63.

**INPUT SCRIPT:**

select a.ssn from EMPLOYEE as a join WORKS\_ON as b on a.ssn=b.essn where b.Pno in(61,62,63);

**OUTPUT:**

|  |
| --- |
| 111111100 |
| 111111101 |
| 111111102 |
| 111111103 |
| 666666607 |
| 666666610 |
| 666666611 |
| 666666612 |
| 666666613 |
| 222222200 |
| 222222201 |
| 222222202 |
| 222222203 |
| 222222204 |
| 222222205 |
| 666666608 |
| 666666613 |
| 333333300 |
| 333333301 |
| 666666609 |
| 666666613 |
|  |

1. **[10 pts]** Using the IN clause, retrieve the first name, last name and the social security numbers of all employees who work on project numbers 61, 62, 63.

**INPUT SCRIPT:**

select a.fname, a.lname, a.ssn from EMPLOYEE a,WORKS\_ON b where a.ssn=b.essn and b.Pno in(61,62,63);

**OUTPUT:**

|  |  |  |
| --- | --- | --- |
| Jared | James | 111111100 |
| Jon | Jones | 111111101 |
| Justin | Mark | 111111102 |
| Brad | Knight | 111111103 |
| Gerald | Small | 666666607 |
| Naveen | Drew | 666666610 |
| Carl | Reedy | 666666611 |
| Sammy | Hall | 666666612 |
| Red | Bacher | 666666613 |
| Evan | Wallis | 222222200 |
| Josh | Zell | 222222201 |
| Andy | Vile | 222222202 |
| Tom | Brand | 222222203 |
| Jenny | Vos | 222222204 |
| Chris | Carter | 222222205 |
| Arnold | Head | 666666608 |
| Red | Bacher | 666666613 |
| Kim | Grace | 333333300 |
| Jeff | Chase | 333333301 |
| Helga | Pataki | 666666609 |
| Red | Bacher | 666666613 |

1. **[10 pts]** Without using the IN clause, retrieve the first name, last name and the social security numbers of all employees who work on project numbers 61, 62, 63.

**INPUT SCRIPT:**

select a.fname, a.lname, a.ssn from EMPLOYEE as a inner join WORKS\_ON as b on a.ssn=b.essn where b.pno = 61 or b.pno = 62 or b.pno = 63;

**OUTPUT**:

|  |  |  |
| --- | --- | --- |
| Jared | James | 111111100 |
| Jon | Jones | 111111101 |
| Justin | Mark | 111111102 |
| Brad | Knight | 111111103 |
| Gerald | Small | 666666607 |
| Naveen | Drew | 666666610 |
| Carl | Reedy | 666666611 |
| Sammy | Hall | 666666612 |
| Red | Bacher | 666666613 |
| Evan | Wallis | 222222200 |
| Josh | Zell | 222222201 |
| Andy | Vile | 222222202 |
| Tom | Brand | 222222203 |
| Jenny | Vos | 222222204 |
| Chris | Carter | 222222205 |
| Arnold | Head | 666666608 |
| Red | Bacher | 666666613 |
| Kim | Grace | 333333300 |
| Jeff | Chase | 333333301 |
| Helga | Pataki | 666666609 |
| Red | Bacher | 666666613 |
|  |  |  |

1. **[10 pts]** Using a join clause, retrieve all project names and their department names.

**INPUT SCRIPT:**

select b.pname, a.dname from DEPARTMENT as a join PROJECT as b on a.dnumber=b.dnum;

**OUTPUT**:

|  |  |
| --- | --- |
| Computerization | Administration |
| Newbenefits | Administration |
| InkjetPrinters | Hardware |
| LaserPrinters | Hardware |
| Reorganization | Headquarters |
| ProductX | Research |
| ProductY | Research |
| ProductZ | Research |
| OperatingSystems | Software |
| DatabaseSystems | Software |
| Middleware | Software |

1. **[10 pts]** Without using a join clause, retrieve all project names and their department names.

**INPUT SCRIPT:**

select b.pname, a.dname from DEPARTMENT a, PROJECT b where a.dnumber=b.dnum;

**OUTPUT:**

|  |  |
| --- | --- |
| Computerization | Administration |
| Newbenefits | Administration |
| InkjetPrinters | Hardware |
| LaserPrinters | Hardware |
| Reorganization | Headquarters |
| ProductX | Research |
| ProductY | Research |
| ProductZ | Research |
| OperatingSystems | Software |
| DatabaseSystems | Software |
| Middleware | Software |

1. **[10 pts]** Using a join clause, retrieve employees who have dependents.

**INPUT SCRIPT:**

Select a.\* from EMPLOYEE as a join DEPENDENT as b on a.ssn = b.essn;

OUTPUT:

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| John | B | Smith | 123456789 | 1955-01-09 | 731 Fondren, Houston, TX | M | 30000.00 | 333445555 | 5 |
| John | B | Smith | 123456789 | 1955-01-09 | 731 Fondren, Houston, TX | M | 30000.00 | 333445555 | 5 |
| John | B | Smith | 123456789 | 1955-01-09 | 731 Fondren, Houston, TX | M | 30000.00 | 333445555 | 5 |
| Franklin | T | Wong | 333445555 | 1945-12-08 | 638 Voss, Houston, TX | M | 40000.00 | 888665555 | 5 |
| Franklin | T | Wong | 333445555 | 1945-12-08 | 638 Voss, Houston, TX | M | 40000.00 | 888665555 | 5 |
| Franklin | T | Wong | 333445555 | 1945-12-08 | 638 Voss, Houston, TX | M | 40000.00 | 888665555 | 5 |
| Alex | D | Freed | 444444400 | 1950-10-09 | 4333 Pillsbury, Milwaukee, WI | M | 89000.00 | null | 7 |
| Alex | D | Freed | 444444400 | 1950-10-09 | 4333 Pillsbury, Milwaukee, WI | M | 89000.00 | null | 7 |
| Bonnie | S | Bays | 444444401 | 1956-06-19 | 111 Hollow, Milwaukee, WI | F | 70000.00 | 444444400 | 7 |
| Alec | C | Best | 444444402 | 1966-06-18 | 233 Solid, Milwaukee, WI | M | 60000.00 | 444444400 | 7 |
| Jennifer | S | Wallace | 987654321 | 1931-06-20 | 291 Berry, Bellaire, TX | F | 43000.00 | 888665555 | 4 |

1. **[10 pts]** Retrieve all employees (first name, last name) and their dependent names. You should also get those who do not have any dependents. **Hint:** Use left join or right join.

**INPUT SCRIPT:**

Select a.fname, a.lname, b.dependent\_name from EMPLOYEE as a left join DEPENDENT as b on a.ssn = b.essn;

OUTPUT:

45 RECORDS are been retrieved.

1. **[10 pts]** Get the total number of projects in the company.

**INPUT SCRIPT:**

Select count(pnumber) as COUNT from PROJECT;

**OUTPUT:**

11

1. **[10 pts]** For each employee, get the total number of hours worked in the company.

**INPUT SCRIPT:**

Select a.fname,a.lname,a.ssn,sum(b.hours) as TOTAL\_NO\_OF\_HOURS from EMPLOYEE as a join WORKS\_ON as b on a.ssn=b.essn group by b.essn ;

**OUTPUT:**

38 RECORDS have been retrieved.

1. **[10 pts]** For each employee, get the total number of hours worked more than 40 hours in the company.

**INPUT SCRIPT:**

Select a.fname,a.lname,a.ssn,sum(b.hours) as TOTAL\_NO\_OF\_HOURS from EMPLOYEE as a join WORKS\_ON as b on a.ssn=b.essn group by b.essn having sum(b.hours)> 40 ;

**OUTPUT:**

|  |  |  |  |
| --- | --- | --- | --- |
| Josh | Zell | 222222201 | 48.0 |
| Jeff | Chase | 333333301 | 46.0 |
| Nandita | Ball | 555555501 | 44.0 |
| Red | Bacher | 666666613 | 50.0 |

**Bonus Question [10 pts]**

The CEO wants to get a report in which he can see employees with a column "Status" which should include either low-paid, normal-paid or over-paid values. Low-paid is for employees who work under 40 hours. Normal-paid is for employees who worked only 40 regular hours. Over-paid is for employees who worked more than 40 hours. Retrieve employee ssn, first name, last name and status (lower-paid, normal-paid or over-paid) of all employees.

**INPUT SCRIPT:**

SELECT a.fname , a.lname, a.ssn , SUM(b.hours) TOTAL\_HOURS,

CASE WHEN SUM(b.hours) < 40 THEN 'Low-paid'

WHEN SUM(b.hours) = 40 THEN 'Normal-paid'

ELSE 'Over-paid' END AS status

from EMPLOYEE a join works\_on b ON a.ssn = b.essn group by a.ssn;

**OUTPUT:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Jared | James | 111111100 | 40.0 | Normal-paid |
| Jon | Jones | 111111101 | 40.0 | Normal-paid |
| Justin | Mark | 111111102 | 40.0 | Normal-paid |
| Brad | Knight | 111111103 | 40.0 | Normal-paid |
| John | Smith | 123456789 | 40.0 | Normal-paid |
| Evan | Wallis | 222222200 | 40.0 | Normal-paid |
| Josh | Zell | 222222201 | 48.0 | Over-paid |
| Andy | Vile | 222222202 | 40.0 | Normal-paid |
| Tom | Brand | 222222203 | 40.0 | Normal-paid |
| Jenny | Vos | 222222204 | 40.0 | Normal-paid |
| Chris | Carter | 222222205 | 40.0 | Normal-paid |
| Kim | Grace | 333333300 | 40.0 | Normal-paid |
| Jeff | Chase | 333333301 | 46.0 | Over-paid |
| Franklin | Wong | 333445555 | 40.0 | Normal-paid |
| Alex | Freed | 444444400 | 40.0 | Normal-paid |
| Bonnie | Bays | 444444401 | 40.0 | Normal-paid |
| Alec | Best | 444444402 | 40.0 | Normal-paid |
| Sam | Snedden | 444444403 | 40.0 | Normal-paid |
| Joyce | English | 453453453 | 40.0 | Normal-paid |
| John | James | 555555500 | 40.0 | Normal-paid |
| Nandita | Ball | 555555501 | 44.0 | Over-paid |
| Jill | Jarvis | 666666601 | 40.0 | Normal-paid |
| Lyle | Leslie | 666666603 | 40.0 | Normal-paid |
| Billie | King | 666666604 | 40.0 | Normal-paid |
| Jon | Kramer | 666666605 | 40.0 | Normal-paid |
| Ray | King | 666666606 | 40.0 | Normal-paid |
| Gerald | Small | 666666607 | 40.0 | Normal-paid |
| Arnold | Head | 666666608 | 40.0 | Normal-paid |
| Helga | Pataki | 666666609 | 40.0 | Normal-paid |
| Naveen | Drew | 666666610 | 40.0 | Normal-paid |
| Carl | Reedy | 666666611 | 40.0 | Normal-paid |
| Sammy | Hall | 666666612 | 40.0 | Normal-paid |
| Red | Bacher | 666666613 | 50.0 | Over-paid |
| Ramesh | Narayan | 666884444 | 40.0 | Normal-paid |
| James | Borg | 888665555 | 0.0 | Low-paid |
| Jennifer | Wallace | 987654321 | 35.0 | Low-paid |
| Ahmad | Jabbar | 987987987 | 40.0 | Normal-paid |
| Alicia | Zelaya | 999887777 | 40.0 | Normal-paid |