* 1. List all employees, i.e. all tuples in the EMPLOYEE relation.

SELECT \* FROM employee

* 1. List the name of all departments, i.e. the NAME attribute for all tuples in the DEPT relation.

SELECT name FROM dept

* 1. What parts are not in store, i.e. QOH=0? (QOH = Quantity On Hand).

SELECT name FROM `parts` WHERE qoh=0

* 1. Which employees have a salary between 10000 and 12000 (inclusive)?

SELECT name FROM `employee` WHERE salary BETWEEN 10000 and 12000

* 1. Retrieve all items sold in the department 49 with their name, price, and price increased by10%.

SELECT name,price, price+price\*0.10 as "Increased Price" from item where number in (select item from sale WHERE item in (SELECT number from item where dept=49))

* 1. Which employees have a family name starting with “S”? Retrieve their names, numbers and salaries.

select name,salary,number from employee where SUBSTRING\_INDEX(name, ',' , -1) like's%'

* 1. What are the names and weights of all parts delivered by a supplier called “DEC”? Formulate this query using a subquery in the where-clause.

SELECT name,weight FROM `parts` WHERE number in (SELECT part from supply WHERE supplier=(SELECT number from supplier where name='DEC'))

* 1. Formulate the same query as above, but without a subquery.

SELECT p.name,p.weight FROM parts p INNER JOIN supply s on s.part = p.number INNER join supplier sp on sp.number=s.supplier WHERE sp.name='DEC'

* 1. Retrieve the name and the color of all parts that are heavier than a black tape drive. Formulate this query using a subquery in the where-clause. (The SQL query should not contain the weight as a constant.)

SELECT name,weight,color FROM parts where weight>(select weight FROM parts where name='tape drive')

* 1. Formulate the same query as above, but without a subquery. (The query should not contain the weight as a constant.)

select p.name, p.color ,p.weight from parts p, parts pt where pt.name = 'tape drive' AND p.weight >pt.weight;

* 1. What is the average salary of all the employees whose manager is the employee with number 199?

SELECT AVG(salary) FROM `employee` WHERE manager=199

* 1. For each supplier retrieve its name and the number of different items it supplies.

select sp.name, count(i.number) as 'items' from supplier sp, item i where i.supplier=sp.number GROUPBY sp.name

* 1. For each supplier in Massachusetts (“Mass”) retrieve the total weight of all the arts delivered by the supplier.

SELECT sup.name, SUM(p.weight\*sp.quantity) weight FROM parts p, supplier sup, supply sp, city c WHERE p.number = sp.part AND sup.number = sp.supplier AND sup.city = c.name AND c.state = 'Mass' GROUP BY sup.name

* 1. Insert data about a new supplier on your choice in the supplier table. Note that the city column is a foreign key to the city table, i.e. the supplier city must already exist or be inserted in advance to the city table.

INSERT INTO `supplier`(`number`, `name`, `city`) VALUES (111,'Manual','New York')

* 1. All departments in store number 8 showed good sales figures last year! Give the managers of these departments 5% raise of their salaries. Retrieve the information about these managers by a query before and after the update statement to verify that the data has been updated.

BEGIN

SELECT name,salary from employee where number in (SELECT manager from dept where store=stor);

UPDATE employee SET salary = (salary+salary\*0.05) where number in (SELECT manager from dept where store= stor);

SELECT name,salary from employee where number in (SELECT manager from dept where store= stor);

END

Execution results of routine `UpdateEmployeeSalary`

|  |  |
| --- | --- |
| **name** | **salary** |
| Hayes,Evelyn | 10100 |
| Smythe,Carol | 9050 |
| Evans,Michael | 5000 |

|  |  |
| --- | --- |
| **name** | **salary** |
| Hayes,Evelyn | 10605 |
| Smythe,Carol | 9503 |
| Evans,Michael | 5250 |