**Part 1**

1. List the animals (animal names) and the ID of the zoo keeper assigned to them.

SELECT Animal.Aname, handles.zookeepid

FROM Handles, Animal

WHERE handles.animalid = animal.aid;

1. Now repeat the previous query and make sure that the animals without a handler also appear in the answer.

SELECT Animal.Aname, handles.zookeepid

FROM Handles RIGHT OUTER JOIN Animal

ON (handles.animalid = animal.aid);

1. Report, for every zoo keeper name, the total number of hours they spend feeding all animals in their care.

select z.ZName, SUM(a.TimeToFeed)

from Zookeeper z, Handles h, Animal a

WHERE z.ZID = h.ZooKeepID AND h.AnimalID = a.AID

GROUP BY z.ZName;

1. Report every handling assignment (as a list of assignment date, zoo keeper name and animal name). Sort the result of the query by the assignment date in a descending order.

SELECT handles.assigned, zookeeper.zname, animal.aname

FROM Animal,Handles,Zookeeper

WHERE zookeeper.zid = handles.zookeepid

AND handles.animalid = animal.aid

ORDER BY handles.assigned DESC;

1. Find the names of animals that have more than 1 zoo keeper assigned to them.

SELECT Animal.Aname, COUNT(zookeepid) AS ZooKeepNum

FROM Animal, Handles

WHERE Animal.aid = Handles.animalid

GROUP BY Animal.aname

HAVING COUNT(zookeepid) > 1;

1. Find the names of animals that have 0 or 1 (i.e., less than 2) zoo keepers assigned to them.

select AName, COUNT(ZID) as NumZookeepers

from (Animal LEFT JOIN Handles ON AID=AnimalID)

LEFT JOIN ZooKeeper ON ZID = ZookeepID

GROUP BY AName

HAVING COUNT(ZID) <= 1;

* List all combination of animals where the difference between feeding time requirement is within 0.25 hours (e.g., Grizzly bear, 3, Bengal tiger, 2.75). Hint: this will require a self join. Even better if your query avoids listing identical pairs such as (Grizzly bear, 3, Grizzly bear, 3)

-- Tables get aliases in order to perform self-join

-- First condition ensures that the difference is under 0.25

-- ABS is the "absolute value" operator, can be replaced by two conditions

-- 2nd condition ensures that matching listings are excluded

SELECT An1.Aname, An1.TimeToFeed, An2.Aname, An2.TimeToFeed

FROM Animal An1, Animal An2

WHERE ABS(An1.timetofeed - An2.timetofeed) <= 0.25

AND An1.Aid != An2.Aid;

**Part 3**

--1.

SELECT Fname, Minit, Lname

FROM EMPLOYEE

WHERE SUPER\_SSN IN (SELECT SSN

FROM Employee

WHERE FNAME = 'Franklin' AND Minit = 'T'

AND LNAME = 'Wong');

--2.

SELECT Project.Pname, Works\_On.pno, SUM(Hours) TOTAL\_HOURS

FROM Project, Works\_On

WHERE Project.pnumber = Works\_On.pno

GROUP BY Project.pname, Works\_On.pno;

--3.

SELECT Department.Dname, AVG(SALARY) AVERAGE\_SALARY

FROM DEPARTMENT, EMPLOYEE

WHERE dnumber=dno

GROUP BY dname, dno

ORDER BY dno;

--4. Retrieve the average salary of all male employees.

SELECT AVG(Salary)

FROM Employee

WHERE sex = 'F';

--5.

SELECT Dname, COUNT(\*)

FROM Employee, Department

WHERE dno = dnumber

GROUP BY Dname

HAVING AVG(Salary) > 43000;

--6.

SELECT Fname, Minit, Lname

FROM Employee

WHERE Salary >= ((SELECT MAX(Salary)

FROM Employee) - 22000);