Databases Project Part 2

- Nicey and Sal

Develop a logical data model based on the following requirements:

a. Derive relations from the conceptual model.

Staff(staffNo, staffName, staffAddress, staffTeleNo, staffDOB, position, salary, clinicNo) **Primary Key** staffNo **Alternate Key** staffTeleNo **Foreign Key** clinicNo **references** Clinic(clinicNo)	**Clinic**(clinicNo, clinicName, clinicAddress, clinicTeleNo, managerStaffNo) **Primary Key** clinicNo **Alternate Key** clinicTeleNo **Foreign Key** managerStaffNo **references** Staff(staffNo)
Owner(ownerNo, ownerName, ownerAddress, ownerTeleNo) **Primary Key** ownerNo **Alternate Key** ownerTeleNo **Foreign Key** N/A	**Examination**(examNo, chiefComplaint, description, dateSeen, actionsTaken, petNo, staffNo) **Primary Key** examNo **Alternate Key** N/A **Foreign Key** petNo **references** Pet(petNo) **Foreign Key** staffNo **references** Staff(staffNo)
Pet(petNo, petName, petDOB, petSpecies, petBreed, petColor, ownerNo, clinicNo) **Primary Key** petNo **Alternate Key** N/A **Foreign Key** ownerNo **references** Owner(ownerNo) **Foreign Key** clinicNo **references** Clinic(clinicNo)	

Assumptions:

- All telephone numbers are unique
- A staff can only work at one clinic
- Position doesn't depend on salary, or vice-versa, hence we assume that 2 people in the same position can have different salaries -> this means that there is no transitive dependency

b. Validate the logical model using normalization to 3NF.

When normalizing the above tables, we realized that petSpecies depends on petBreeds which is a transitive dependency. As petNo -> petBreed -> petSpecies, and as petSpecies depends on petBreed which is a non-prime attribute it is not in 3NF. To make it 3NF we can add a separate table for breeds and species. All the other tables are in 3NF so no need to change them.

Staff(staffNo, staffName, staffAddress, staffTeleNo, staffDOB, position, salary, clinicNo) **Primary Key** staffNo **Alternate Key** staffTeleNo **Foreign Key** clinicNo **references** Clinic(clinicNo)	**Clinic**(clinicNo, clinicName, clinicAddress, clinicTeleNo, managerStaffNo) **Primary Key** clinicNo **Alternate Key** clinicTeleNo **Foreign Key** managerStaffNo **references** Staff(staffNo)

Owner(ownerNo, ownerName, ownerAddress, ownerTeleNo) **Primary Key** ownerNo **Alternate Key** ownerTeleNo **Foreign Key** N/A	**Examination**(examNo, chiefComplaint, description, dateSeen, actionsTaken, petNo, staffNo) **Primary Key** examNo **Foreign Key** petNo **references** Pet(petNo) **Foreign Key** staffNo **references** Staff(staffNo)
Pet(petNo, petName, petDOB, petBreed, petColor, ownerNo, clinicNo) **Primary Key** petNo **Foreign Key** ownerNo **references** Owner(ownerNo) **Foreign Key** clinicNo **references** Clinic(clinicNo) **Foreign Key** petBreed **references** Breed(petBread)	**Breed**(petBreed, petSpecies) **Primary Key** petBreed

c. Validate the logical model against 5 user transactions. (Note: These will be then implemented in 3c).

Transaction 1: **Register a New Pet Owner and their Pet(s) at a clinic**

- Insert a new record into the Owner Table. Collect the owner's details (ownerName, ownerAddress, owenerTeleNo)
- Insert one or more records into the Pet table, linking them to the owner via the foreign key owerNo. Collect the pet's details (petName, petDOB, petBreed, petcolor) and the clinicNo will be the clinic the pet is registered at.
- Constraints check:
    - Ensure that the pet can only be registered into 1 owner.
    - Check if the foreign keys owerNo and clinicNo are correctly referenced.

Transaction 2: Schedule an Examination for a Pet

- Insert a new record into the Examination table. Collect the examination details (examNo, dateSeen, petNo, staffNo)
- Constraints check:
    - Check if the foreign keys staffNo and petNo are correctly referenced.

Transaction 3: **Update a Pet's medical records after an examination**

- Updating the existing record in the Examination table, link this with the examNo. Update the examination details (chiefComplaint, description, actionsTaken)
- Constraints check:
    - Check if the examNo exists.

Transaction 4: **Retrieve a list of all Pets owned by a specific owner**

- Query the Pet table for records where ownerNo matches and then for all those entries find the pet details
- Constraints check:
    - Ensure that the owerNo exists.

Transaction 5: **Assign a Manager to the Clinic**

- Update the clinic table to get the managerStaffNo with the staffNo of the staff member that is assigned as manager.
- Change the position if needed.
- Constraints check:
    - Ensure that the staff member is not already managing another clinic.
    - Check if the foreign key managerStaffNo is correctly referenced.

d. Define integrity constraints:

i. Primary key constraints.

- The primary keys must be unique.
- The primary keys are as follows: Staff(staffNo), Clinic(clinicNo), Owner(ownerNo), Pet(petNo), Examination(examNo), Breed(petBreed)

ii. Referential integrity/Foreign key constraints.

- **Staff**:
    - clinicNo references Clinic(clinicNo)
        - ON UPDATE CASCADE ON DELETE NO ACTION
- **Clinic**:
    - managerStaffNo references Staff(staffNo)
        - ON UPDATE CASCADE ON DELETE SET NULL
- **Pet**:
    - ownerNo references Owner(ownerNo)
        - ON UPDATE CASCADE ON DELETE CASCADE
    - clinicNo references Clinic(clinicNo)
        - ON UPDATE CASCADE ON DELETE NO ACTION
    - petBreed references Breed(petBreed)
        - ON UPDATE CASCADE ON DELETE NO ACTION
- **Examination**:
    - petNo references Pet(petNo)
        - ON UPDATE CASCADE ON DELETE CASCADE
    - staffNo references Staff(staffNo)
        - ON UPDATE CASCADE ON DELETE SET NULL

iii. Alternate key constraints (if any).

- Staff(staffTeleNo), Clinic(clinicTeleNo), Owner(ownerTeleNo) are all alternate keys assuming that the telephone numbers are unique

iv. Required data.

- The owner name, address must be required data
- Pet name, species is also required to easily identify them
- Staff names and addresses are also required

v. Attribute domain constraints.

- Telephone numbers(staffTeleNo, clinicTeleNo, ownerTeleNo) must be 10 digits
- Staff(salary) must be a positive integer and > 0

vi. General constraints (if any).

- staffDOB, petDOB and dateSeen must be valid dates and cannot be in the future
- petBreed must correspond to the petSpecies (logically)
- Each examination should have a chief complaint that is written in company standards

e. Generate the E-R diagram for the logical level (contains FKs as attributes).