<u>DBMS Project Part 2</u> :

(a) Relations from the conceptual model:

Clinic (<u>clinicNo</u>,cName,cAddress,cTelephoneNo,staffNo)
PrimaryKey(clinicNo)
ForeignKey(staffNo) refs Staff(staffNo)

Staff (StaffNo,sName,sTelephoneNo,sDOB,sPosition,sSalary,clinicNo)
Primary Key (StaffNo)
Foreign Key (clinicNo) refs Clinic(clinicNo)

Pets (petNo,pName,pDOB,pSpecies,pBreed,pColor,ownerNo,clinicNo)
Primary Key(petNo)
Foreign Key(ownerNo) refs Owner(ownerNo)
Foreign Key(clinicNo) refs Clinic(clinicNo)

Examination (examNo,complaint,description,dateSeen,action,staffNo,petNo)
Primary Key (examNo)
Foreign Key (staffNo) refs Staff(staffNo)
Foreign Key (petNo) refs Pet(petNo)

Owner (OwnerNo,oName,oAddress,oTelephoneNo)
Primary Key(OwnerNo)

(b) Normalization :

The model is currently in UML/1NF due to no repeating entities.

Functional Dependencies:

clinicNo -> cName,cAddress,cTelephoneNo,staffNo

StaffNo -> sName,sTelephoneNo,sDOB,sPosition,sSalary,clinicNo

petNo -> pName,pDOB,pSpecies,pBreed,pColor,ownerNo,clinicNo

examNo -> complaint,description,dateSeen,action,staffNo,petNo

OwnerNo ->oName,oAddress,oTelephoneNo

Partial Dependencies:

No partial dependencies, given that each entity has a single-attributed primary key

Transative Dependencies:

No transative dependencies.

Therefore the table is in 2NF & 3NF.

Final 3NF Model:

Clinic (<u>clinicNo</u>,cName,cAddress,cTelephoneNo,staffNo) ck = pk = {clinicNo}

Staff (<u>staffNo</u>,sName,sTelephoneNo,sDOB,sPosition,sSalary,clinicNo) ck = pk = {staffNo}

Pets (<u>petNo</u>,pName,pDOB,pSpecies,pBreed,pColor,ownerNo,clinicNo) ck = pk = {petNo}

Examination (<u>examNo</u>,complaint,description,dateSeen,action,staffNo,petNo) ck = pk = {examNo}

Owner (<u>ownerNo</u>,oName,oAddress,oTelephoneNo) ck = pk = {ownerNo}

Foreign Keys:

staffNo in Clinc refs Staff(staffNo)

clinicNo in Staff refs Clinic(clinicNo)

ownerNo in Pets refs Owner(ownerNo)

clinicNo in Pets refs Clinic(clinicNo)

staffNo in Examinations refs Staff(staffNo)

petNo in Examinations refs Pet(petNo)

(c) Transactions:
1. List all members of staff that work at a specified clinic address:
   The details of staff members are held in the Staff entity and their associated clinic address are held in the Clinic entity. We can use the Clinc employs Staff relationship to get this information by referencing the clinicNo foreign key in staff to get the clinic address from clinic.
2. What month does the company have the most visits, and which clinic tends to have the most visits in that month?
   This can be performed by aggregating "dateSeen" in the Examinations entity by examNo.
3. Which clinic has the most visits?
   We can first determine how many examinations each pet has undergone through the Pet undergoes Examinations relation. Then, by summarizing using the clinicNo foreign key in the Pets table, we can make clinicNo unique and aggregate the counts of pet visits to identify which clinic has the most visits.
4. What staff have seen pets of a certain breed?
   We can first obtain breed information from the pets entity, through the pets undergo examination relation through the petNo foreign key. Then the examinations entity can filter for the specified breed. Then unique staffNo can be found in the examinations table to determine which staff members are associated with a particular breed.
5. Obtain all owner information,in addition to what clinic they visit.
   We can use the owner owns pet relation to determine what clinic they use, through the clinicNo attribute in Pets. This can be joined to the owner entity, giving us the clinics that our owners visit.

(d) Defining integtrity constraints:
   i.   Primary key contstraints: clincNo in Clinic must not be null. staffNo in Staff must not be null, examNo in Examinations must not be null, ownerNo in Owner must not be null,petNo in pet must not be null. All of these primary keys must also be unique.
   ii.   Foreign Keys: for all foreign keys, you cannot add a row to the entity if that foreign key does not exist in associated parent entity. For example, you cannot add a staff number to clinic if that staffNo does not exist in Staff. Additionally, upon deletion of the attribute in the parent entity, we will set the FK value to null. For instance, if a clinicNo is deleted from the Clinic entity, nulls will be presented for that clincNo in Pets.
   iii. Alternate key contstraints: All telephone numbers are unique
   iv.   Required data :
      -    For a particular row value primary key must have a value inserted, it cannot be null.
      -    All staff members must have an associated position.(no nulls)
   v.   -Attribute Domain constraints:
      Clinic (clinicNo,cName,cAddress,cTelephoneNo,staffNo)

Attribute	Contstraint
clinicNo	2 digit integer
cName	Only letters
cAddress	Open text field
cTelephoneNo	Must be a 10-digit integer
staffNo	5 letter digit

      Staff (staffNo,sName,sTelephoneNo,sDOB,sPosition,sSalary,clinicNo)

Attribute	Constraint
staffNo	5 letter digit
sName	Only letters
sTelephoneNo	Open text field
sDOB	Must be a 10-digit integer
sPosition	Open text field
sSalary	Integer value
clinicNo	2 digit integer

      Pets (petNo,pName,pDOB,pSpecies,pBreed,pColor,ownerNo,clinicNo)

Attribute	Constraint
petNo	10-digit integer
pName	String value
pDOB	Must be in YYYY-MM-DD format
pSpecies	String value
pBreed	String value
pColor	String value
ownerNo	5-digit integer
clinicNo	2-digit integer

Examination (examNo,complaint,description,dateSeen,action,staffNo,petNo)\

Attribute	Constraint
examNo	8-digit integer
complaint	Open text field
description	Open text field
dateSeen	Must be in YYYY-MM-DD format
action	String value
staffNo	5-letter digit
petNo	10-digit integer

Owner (OwnerNo,oName,oAddress,oTelephoneNo)

Attribute	Constraint
ownerNo	5-digit integer
oName	Only letters
oAddress	Open text field
oTelephoneNo	Must be a 10-digit integer

vi.   General Constraints:
- One Staff member cannot manage more than one clinc
- A staff member cannot conduct more than 20 examinations in one day
- A pet cannot have multiple owners

Logical Model Diagram:

Clinic
- clinicNo {PK}
- cName
- cAddress
- cTelephoneNo
- staffNo({FK}

Staff
- staffNo{PK}
- sName
- sTelephoneNo
- sDOB
- sPosition
- sSalary
- clinicNo{FK}

1 —— employs ▶ —— 1..*

0..1 —— ◀ manages —— 1

1 — registers ▼ — 1..*

Pets
- petNo{PK}
- pName
- pDOB
- pSpecies
- pBreed
- pColor
- clinicNo{FK}
- ownerNo{FK}

1 —— undergo ▶ —— 1..*

Examination
- examNo{PK}
- complaint
- description
- dateSeen
- action
- staffNo{FK}
- petNo{FK}

1 — performs ▼ — 1..*

owns ▲ 1..*

1

Owner
- ownerNo{PK}
- oName
- oAddress
- oTelephoneNo