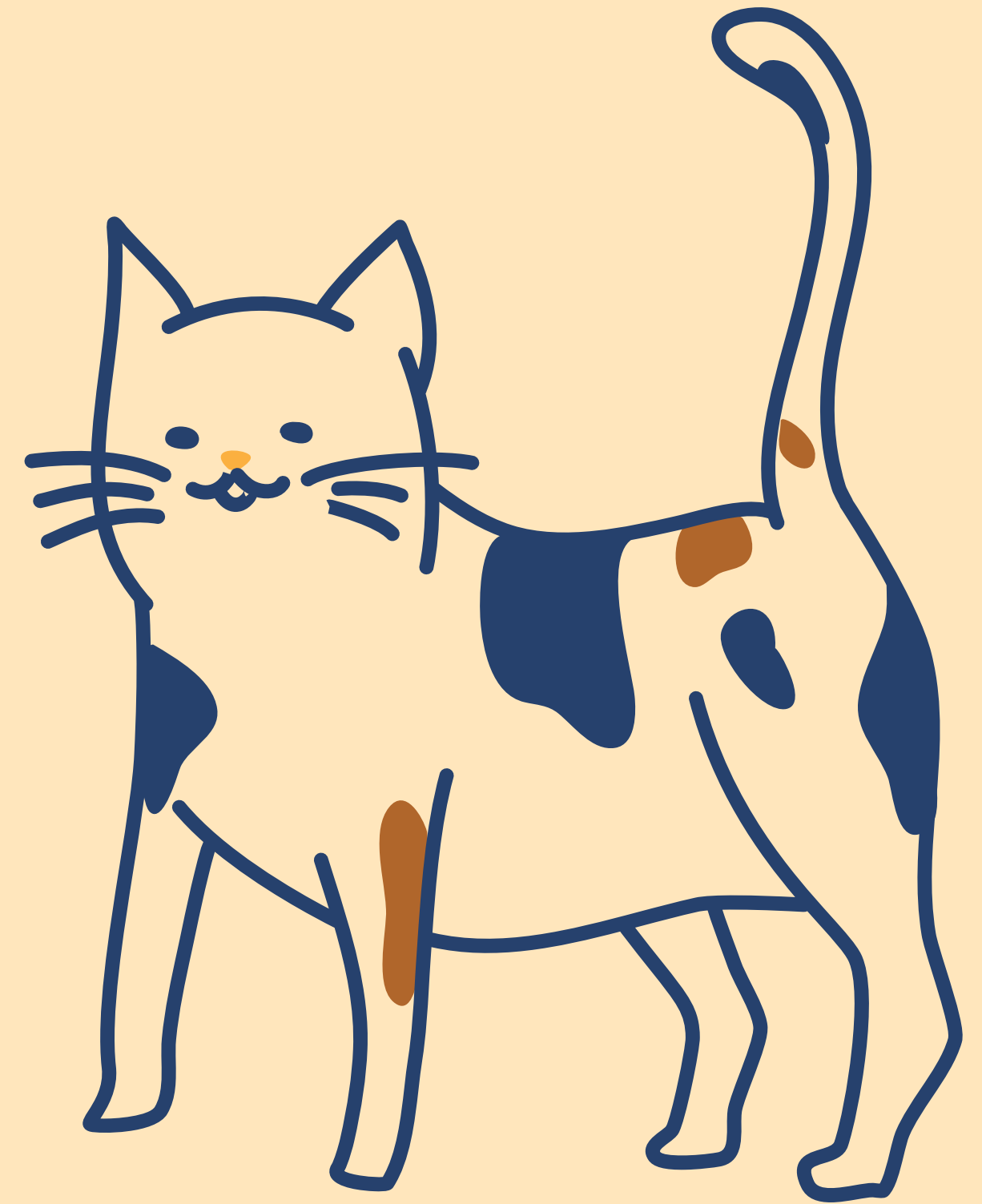


Data Base Project

SECTION: SE3
CCCS 215

GROUP MEMBERS & ID

- Refal Najmi 2006271
- Asool Rajab 2005623
- Mai Alqurashi 2008060
- Joury Alsulimani 2006121



Problem Description

MANY PET OWNERS STRUGGLE TO FIND PLACES TO HOST THEIR PETS. WHENEVER A PET OWNER IS EXPOSED TO ANY CIRCUMSTANCE, THEY ARE FORCED TO LEAVE THEIR PETS ON THE STREET OR PUT IT UP FOR ADOPTION. WE COLLECTIVELY DECIDED TO CREATE A RELIABLE PET HOSPITALITY APPLICATION THAT ALLOWS PET OWNERS TO DROP THEIR PETS AT. THE APPLICATION STRIVES TO PROVIDE A SAFE AND SECURE ENVIRONMENT FOR THE PETS THAT INCORPORATES MANY SERVICES INCLUDING PET KENNELS, BOARDING, DAY-CARE, GROOMING, AND A VETERINARY CLINIC. WE HAVE CONSIDERED THAT THE APPLICATION IS THE LINK BETWEEN PET OWNERS AND TENANTS FOR EASE AND FLEXIBILITY OF HANDLING. OUR IDEA IS LIKELY TO BE THE IDEAL SOLUTION TO THIS PROBLEM, GIVEN THE STUDIES AND STATISTICS WE HAVE DONE

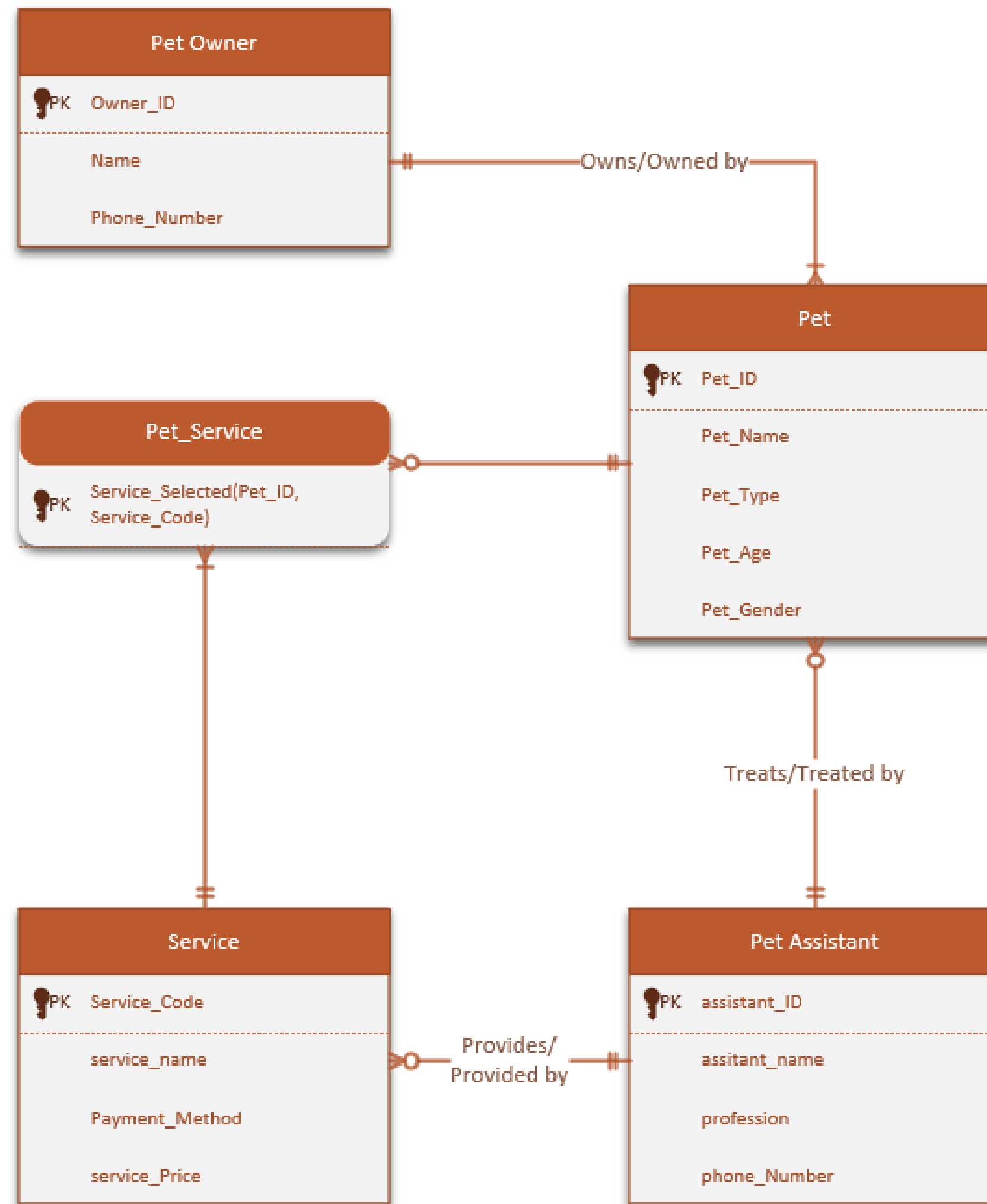


Business Rule

- In a pet hospitality application, a pet owner can have at least one pet or more, yet each pet must be owned by only one owner.
- A pet is treated by only one assistant always and forever.
- A pet assistant can treat zero, one, or many pets.
- One pet assistant can provide zero, one, or more services.
- A service is managed by only one pet assistant.
- A pet can have one or more services.
- A service is provided to at least one or many pets



ER



Diagram

Relations



Pet Owner

<u>Owner_ID</u>	Owner_Name	Phone_Number
-----------------	------------	--------------

Pet

<u>Pet_ID</u>	Pet_Name	Pet_Type	Pet_Age	Pet_Gender	<u>Owner_ID</u>	<u>Assistant_ID</u>
---------------	----------	----------	---------	------------	-----------------	---------------------

Pet Service

<u>Pet_ID</u>	<u>Service_Code</u>
---------------	---------------------

Service

<u>Service_Code</u>	Service_Name	Payment_Method	Service_Price	<u>Assistant_ID</u>
---------------------	--------------	----------------	---------------	---------------------

Pet Assistant

<u>Assistant_ID</u>	Asisstant_Name	Profession	Phone_Number
---------------------	----------------	------------	--------------

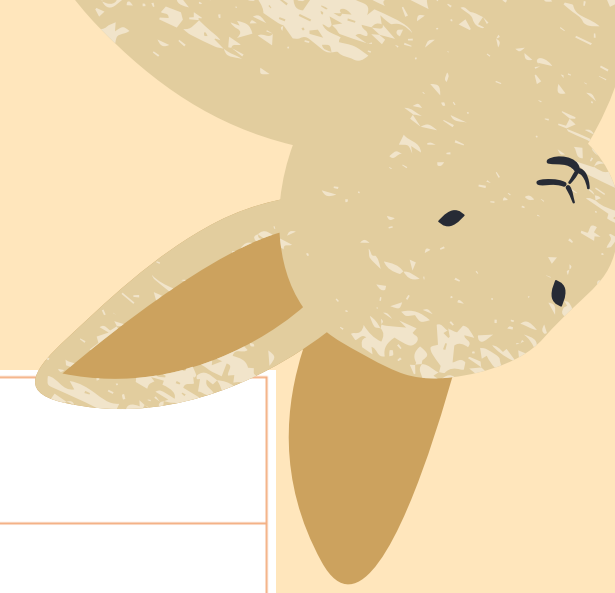
Normalization

- Pet Owner (Owner_ID, name, Phone_Number)
In 1NF, 2NF and 3NF
- Pet (Pet_ID, Pet_Name, Pet_Type, Pet_Age, Pet_Gender, Owner_ID, Assistant_ID)
In 1NF, 2NF and 3NF
- Pet Service (Pet_ID, Service_Code)
In 1NF, 2NF and 3NF
- Service (Service_Code, Service_Name, Payment_Method, Service_Price, Assistant_ID)
In 1NF, 2NF and 3NF
- Disclaimer: Service_Name and Service_Price do not have transitive dependency because the price differs from one pet to another
- Pet Assistant (Assistant_ID, Assistant_Name, Profession, Phone_Number)
In 1NF, 2NF and 3NF

All Relations are in the 3NF as you can see:

- there is no repeating group
- all of them are full dependency
- no transitive

Normalized Tables



Pet Owner

Column name	Data type Size	Constraint
Owner_ID	Number(7)	Primary key
Name	Varchar(12)	Not null
Phone_number	Number(10)	Not null

Assistant

Column name	Datatype Size	Constraint
Assistant_ID	Number(7)	primary key
Assistant_name	Varchar2(12)	Not null
Profession	varchar2(25)	Not null
Phone_number	number(10)	Not null



Pet

Column name	Datatype Size	Constraint
Pet_ID	Number(7)	primary key
Pet_Name	Varchar2(12)	Not null
Pet_Type	Varchar2(25)	Not null
Pet_Age	Varchar2(15)	Not null
Pet_Gender	Varchar2(12)	Not null
Owner_ID	Number(7)	Foreign key
Assistant_ID	Number(7)	Foreign key

Service

Column name	Datatype Size	Constraint
Service_code	Number(7)	Primary key
Service_name	Varchar2(25)	Not null
Payment_method	Varchar2(14)	Not null
Service_price	Number(3)	Not null
Assistant_ID	Number(7)	Foreign key



Functional dependencies

- Owner_ID -----> Name, Phone_Number
- Pet_ID -----> Pet_Name, Pet_Type, Pet_Age, Pet_gender, Owner_ID (FK), Assistant_ID (FK)
- Assistant_ID -----> Assistant_name, Profession, Phone_Number
- Service_Code -----> Service_Name, Payment_Method, Service_Price, Assistant ID (FK)
- Pet Service -----> (Pet ID (FK), Service Code (FK)) (PK)



The coding Phase



```

31 --Creating Pet Table--
32 Create table Pet(
33 Pet_ID Number(7) Primary key,
34 Pet_Name varchar2(12) not null,
35 Pet_Type varchar2(25) not null,
36 Pet_Age varchar2(15) not null,
37 Pet_Gender varchar2(12) not null,
38 Owner_ID Number(7),
39 Assistant_ID Number(7),
40 Foreign Key (Owner_ID) references Pet_Owner(Owner_ID),
41 Foreign Key (Assistant_ID) references Assistant(Assistant_ID));
42 --Inserting Values to Pet table--
43 Insert into Pet values (4002358,'Mex','German Shepard,Dog','1YR,2MONTHS','Male',2005623,3006734);
44 Insert into Pet values (4008123,'Sparks','Sherazi,Cat','9MONTHS','Female',2005624,3002345);
45 Insert into Pet values (4009087,'Mimi','Ragdoll,Cat','4MONTHS','Female',2005625,3007256);
46 Insert into Pet values (4001238,'Baily','Holland lop,Rabbit','2MONTHS','Female',2005626,3007249);
47 Insert into Pet values (4009375,'Maui','Poodle,Dog','1YR,7MONTHS','Male',2005682,3003219);
48
49 select * from Pet;|
50

```

Creating The Pet Table and inserting the data

PET_ID	PET_NAME	PET_TYPE	PET_AGE	PET_GENDER	OWNER_ID	ASSISTANT_ID
4002358	Mex	German Shepard,Dog	1YR,2MONTHS	Male	2005623	3006734
4008123	Sparks	Sherazi,Cat	9MONTHS	Female	2005624	3002345
4009087	Mimi	Ragdoll,Cat	4MONTHS	Female	2005625	3007256
4001238	Baily	Holland lop,Rabbit	2MONTHS	Female	2005626	3007249
4009375	Maui	Poodle,Dog	1YR,7MONTHS	Male	2005682	3003219

[Download CSV](#)

5 rows selected.

```

16 --Creating Assistant Table--
17 Create table Assistant(
18 Assistant_ID Number(7) Primary key,
19 Assistant_Name Varchar2(12) not null,
20 Profession Varchar2(25),
21 Phone_Number number(10) not null);
22 --Inserting Values to Assitant table--
23 insert into Assistant values (3006734,'Analyn','Pet Radiochemist',0552340273);
24 insert into Assistant values (3002345,'Felix','Animal Technician',050803219);
25 insert into Assistant values (3007256,'Jana','Pet Host',0532182467);
26 insert into Assistant values (3007249,'John','Profeesional Pet Groomer',0572315423);
27 insert into Assistant values (3003219,'Sarah','Pet Hydrotherapist',055762349);
28
29 select * from Assistant;

```

Creating The assistant Table and inserting the data

ASSISTANT_ID	ASSISTANT_NAME	PROFESSION	PHONE_NUMBER
3006734	Analyn	Pet Radiochemist	552340273
3002345	Felix	Animal Technician	50803219
3007256	Jana	Pet Host	532182467
3007249	John	Profeesional Pet Groomer	572315423
3003219	Sarah	Pet Hydrotherapist	55762349

Download CSV

5 rows selected.

```

51 -- Creating Service Table --
52 Create table Service(
53 Service_Code number(7) primary key,
54 Service_Name Varchar2(25) not null,
55 Payment_Method Varchar2(14) not null,
56 Service_Price number(3) not null,
57 Assistant_ID Number(7),
58 Foreign Key (Assistant_ID) references Assistant(Assistant_ID));
59 --Inserting Values to Service table--
60 Insert into Service values (7006321,'Pet sitting','Cash',75,3006734);
61 Insert into Service values (7008793,'Pet Training','Credit',100,3002345);
62 Insert into Service values (7006132,'Pet Nails clipping','Credit',20,3007256);
63 Insert into Service values (7009060,'Pet full grooming','Credit',250,3007249);
64 Insert into Service values (7008939,'Pet Spraying','Cash',800,3003219);
65
66 select * from Service;
67

```

Creating The Service Table and inserting the data

SERVICE_CODE	SERVICE_NAME	PAYMENT_METHOD	SERVICE_PRICE	ASSISTANT_ID
7006321	Pet sitting	Cash	75	3006734
7008793	Pet Training	Credit	100	3002345
7006132	Pet Nails clipping	Credit	20	3007256
7009060	Pet full grooming	Credit	250	3007249
7008939	Pet Spraying	Cash	800	3003219

Download CSV

5 rows selected.

```

68
69 --Cretig Pet_Service table--
70 Create table Pet_Service(
71 Pet_ID Number(7),
72 Service_Code number(7),
73 Constraint Pet_Service_PK Primary key (Pet_ID, Service_Code),
74 Foreign Key (Pet_ID) references Pet(Pet_ID),
75 Foreign Key (Service_Code) references service(Service_Code));/*Composite PK*/
76 --Inserting Values to Service table--
77 Insert into Pet_Service values (4002358,7006321);
78 Insert into Pet_Service values (4008123,7008793);
79 Insert into Pet_Service values (4009087,7006132);
80 Insert into Pet_Service values (4001238,7009060);
81 Insert into Pet_Service values (4009375,7008939);
82
83 select * from Pet_Service;
84

```

Creating The Pet_Service Table and inserting the data

1 row(s) inserted.

PET_ID	SERVICE_CODE
4001238	7009060
4002358	7006321
4008123	7008793
4009087	7006132
4009375	7008939

[Download CSV](#)

5 rows selected.

```

1  --Creating Pet_Owner Table--
2  Create table Pet_Owner(
3  Owner_ID Number(7) Primary key,
4  Owner_Name Varchar2(12) not null,
5  Phone_Number number(10) not null);
6  --Inserting Values to Pet_Owner table--
7  Insert into Pet_Owner values (2005623,'Asool' ,0551240219);
8  Insert into Pet_Owner values (2005624,'Joury' ,0551398721);
9  Insert into Pet_Owner values (2005625,'Refal' ,0551398642);
10 Insert into Pet_Owner values (2005626, 'Mai' ,0551397221);
11 Insert into Pet_Owner values (2005682,'Shereen',0551392921);
12
13 select * from Pet_Owner;
14

```

Creting The Pet_Owner Table and inserting the data

OWNER_ID	OWNER_NAME	PHONE_NUMBER
2005623	Asool	551240219
2005624	Joury	551398721
2005625	Refal	551398642
2005626	Mai	551397221
2005682	Shereen	551392921

Download CSV

5 rows selected.

Queries

SQL Worksheet

```
84
85 --first query--
86 select Payment_Method, count(*)
87 from Service
88 group by Payment_Method
89 order by Payment_Method;|
90
```

PAYMENT_METHOD	COUNT(*)
Cash	2
Credit	3

[Download CSV](#)

2 rows selected.

SQL Worksheet

```
90
91 --seconed query--
92 select Pet.Pet_ID, Pet.Pet_Name, Assistant.Assistant_ID
93 FROM Pet, Assistant
94 WHERE Pet.Assistant_ID = Assistant.Assistant_ID
95 ORDER BY Pet_Name;|
96
```

PET_ID	PET_NAME	ASSISTANT_ID
4001238	Baily	3007249
4009375	Maui	3003219
4002358	Mex	3006734
4009087	Mimi	3007256
4008123	Sparks	3002345

[Download CSV](#)

5 rows selected.

Queries

SQL Worksheet

```
95 ORDER BY Fee_Name;  
96  
97 --third query--  
98 select Service_Name, Service_Price  
99 From service  
100 where Service_Price>100;  
101
```

SERVICE_NAME	SERVICE_PRICE
Pet full grooming	250
Pet Spraying	800

[Download CSV](#)
2 rows selected.

SQL Worksheet

```
101  
102 --fourth query--  
103 select MIN(Service_Price) As MINIMUM_PRICE,  
104 MAX(Service_Price) As MAXIMUM_PRICE,  
105 AVG(Service_Price) As AVERAGE  
106 FROM service;  
107
```

MINIMUM_PRICE	MAXIMUM_PRICE	AVERAGE
20	800	249

[Download CSV](#)

Queries

SQL Worksheet

```
107
108 --fifth query--
109 select Pet_Name, Pet_Gender
110 from Pet
111 where Pet_Gender IN ('Female')
112 order by Pet_Name;
113 |
```

PET_NAME	PET_GENDER
Baily	Female
Mimi	Female
Sparks	Female

[Download CSV](#)

3 rows selected.

SQL Worksheet

```
113
114 --sixth query--
115 select Service_Name, Service_Price
116 from service
117 where Service_Price >
118 (select Service_Price
119 FROM service
120 where Service_Name = 'Pet sitting');|
121
```

SERVICE_NAME	SERVICE_PRICE
Pet Training	100
Pet full grooming	250
Pet Spraying	800

[Download CSV](#)

3 rows selected.

Procedures

```
121
122 -- first procedure--
123 create or replace procedure Service_Price_Update (Update_Service_Code in Service.Service_Code%type , New_Service_Price in Service.Service_Price%type) As
124 begin
125 update Service set Service_Price = New_Service_Price
126 where Service_Code = Update_Service_Code;
127 commit;
128 DBMS_OUTPUT.PUT_LINE('THE PRICE OF SERVICE NUMBER ' || Update_Service_Code );
129 DBMS_OUTPUT.PUT_LINE('Has changed, and here is the new price ' || New_Service_Price );
130 end Service_Price_Update;
131
132 execute Service_Price_Update (7006321,150);
133 select * from Service;
134
```

Statement processed.

THE PRICE OF SERVICE NUMBER 7006321

Has changed, and here is the new price 150

SERVICE_CODE	SERVICE_NAME	PAYMENT_METHOD	SERVICE_PRICE	ASSISTANT_ID
7006321	Pet sitting	Cash	150	3006734
7008793	Pet Training	Credit	100	3002345
7006132	Pet Nails clipping	Credit	20	3007256
7009060	Pet full grooming	Credit	250	3007249
7008939	Pet Spraying	Cash	800	3003219

[Download CSV](#)

5 rows selected.

Procedures

```
134
135 --second procedure--
136 CREATE OR REPLACE PROCEDURE AddPetOwner(nonwer_id Pet_Owner.Owner_ID%TYPE, nowner_name Pet_Owner.Owner_Name%TYPE, nphone_number Pet_Owner.Phone_Number%TYPE)
137 AS
138 BEGIN
139 insert into Pet_Owner(Owner_ID, Owner_Name, Phone_Number)
140 VALUES (nonwer_id, nowner_name, nphone_number);
141 END AddPetOwner;
142
143
144 EXEC AddPetOwner(2885741, 'Joud', 0547878252);
145 select * from Pet_Owner;
146
```

Statement processed.

OWNER_ID	OWNER_NAME	PHONE_NUMBER
2005623	Asool	551240219
2005624	Joury	551398721
2005625	Refal	551398642
2005626	Mai	551397221
2005682	Shereen	551392921
2885741	Joud	547878252

[Download CSV](#)

6 rows selected.

Procedures

```
147  --third procedur--
148  CREATE PROCEDURE Gender_owner(p_gender Pet.Pet_Gender%TYPE)
149  AS
150  CURSOR executive IS
151  select Owner_Name from Pet, Pet_Owner where Pet.Owner_ID = Pet_Owner.Owner_ID
152  AND pet_gender = p_gender;
153  BEGIN
154  FOR curs IN executive LOOP
155  dbms_output.put_line('Owner: ' || curs.Owner_Name);
156  END LOOP;
157  END Gender_owner;
158
159  exec Gender_owner('Female');
160
```

Statement processed.

Owner: Joury

Owner: Refal

Owner: Mai