IST659 – Project 2

Project Narrative

I will be making a database on the various plants inside and outside my house. The goal of this database will be to identify the names and details of all of the plants and provide instructions for how to care and propagate each. Because there are many different types of plants with varying levels of tolerance and needs, it is important to understand each characteristic to optimize the plant's health. These characteristics might include the amount of sunlight, water, soil properties, and temperature. Understanding each individual plant will also make caring for the plants faster and more efficient.

The scope for the first part of the project will involve building a conceptual and logical model of the database. The database will eventually be created using SQL and viewed by a database management program. The models will only contain the necessary information required to care for the plants on a superficial level. Detailed information about each plant including pH, flower information, mature size, etc. will be out of scope since it is not necessary for amateur plant care.

Those affected by this database (the stakeholders) will be me, my girlfriend and my roommates. While my girlfriend and I will be the primary caretakers of the plants, it is also important for my housemates to know how to care for them while we are traveling or away.

The final outcome will be a database containing all the necessary means of taking care of our plants. It will show the names, the means of identification, the care requirements, and the method of propagation for each plant. Hopefully this will increase the number of healthy plants, prevent diseases, increase propagation rate and save time and money.

Original Data Dictionary

Entity	Attribute	Properties	Description
Plant	Plant Name	Required, Unique, & Multi-value	The common and Latin name of the plant
	Plant Description	Required & Multi-value	The physical description of the plant (Color, size, leaf shape)
	Plant Details	Required & Multi-value	Non-physical information about the plant (Type of plant, prefers shade)
Propagation	Propagation Type	Required	Type of propagation for plant (cutting, grafting, layering)
	Propagation Instructions		Instructions for how to propagate a plant
Origin	Origin Name	Required & Multi-value	The native region of the plant
	Climate Zone		The zone of the plant's origin
	Climate Type		The type of climate of the plant's origin
Leaf	Shape	Required	The shape of the plant's leaves
	Color	Required	The color of the plant's leaves
	Physical Attributes		Other physical attributes pertaining to the appearance of the leaves
Care Instructions	Sunlight	Required	How much sunlight does the plant require?
	Water	Required	How much water must the plant receive?
	Temperature	Required	What is the ideal range of temperature for the plant?
	Soil	Required	What kind of soil does the plant prefer?

Data Questions

What is the name of the plant?
How much sunlight, water, temperature and soil does each plant require?
How does the plant propagate?
Where should I place the indoor plant within the house?
How often should the sprinkler system be used for the outdoor plants?
How much can I save on my water bill?

Conceptual Model

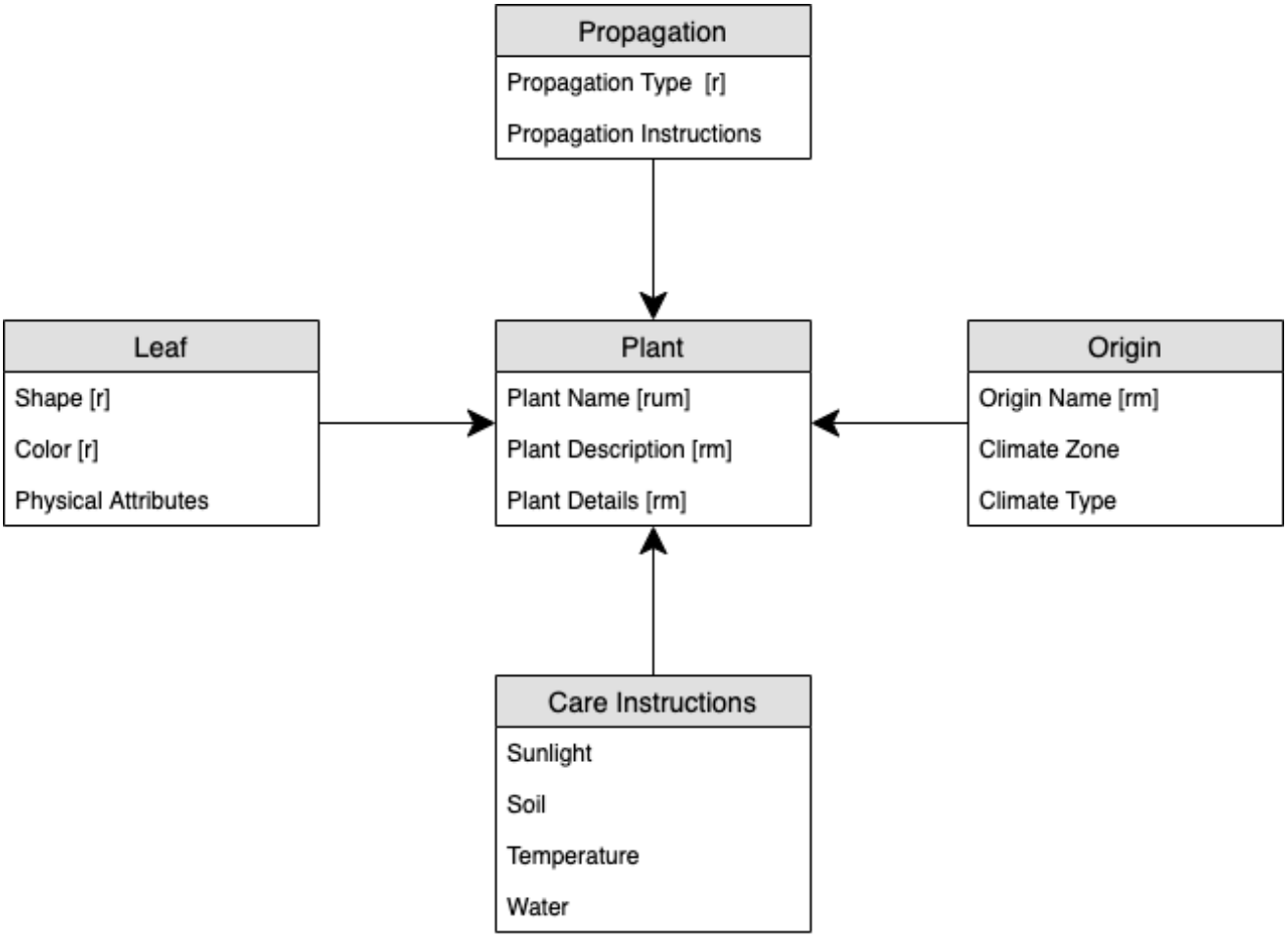
The plant with details describing the name and physical attributes.

The leaf which contains the physical description of the leaves used for identifying the species. Once identified, I can determine details about each plant that will be essential in understanding their needs.

The origin which contains the native region of the plant. This will contain climate data in which the plant originated. This will allow me to determine how similar the plants current conditions are compared to their native habitat.

The propagation which will give details on how to propagate the plant. This will be important for determining how to create multiples of each plant. This will also contain instructions for how to propagate each variety.

The care instructions will contain the details on how to care for the plant. This will be important for optimizing the health of each plant, determining where to place each plant, and conserving water, time and money.



Logical Model

1. Surrogate keys were added to all the tables to provide unique identifiers as auto incremented primary keys.

2. Plant details in the plant table started as a multi-value field but became divided into family, duration, and plant category. Because these values don't rely on each other, they remain independent fields on the table and pass 1NF.

3. Name has been divided from a multi-value field into its own table. The name fields didn't previously pass 1NF because the scientific name can be derived from the common name which is a candidate key.

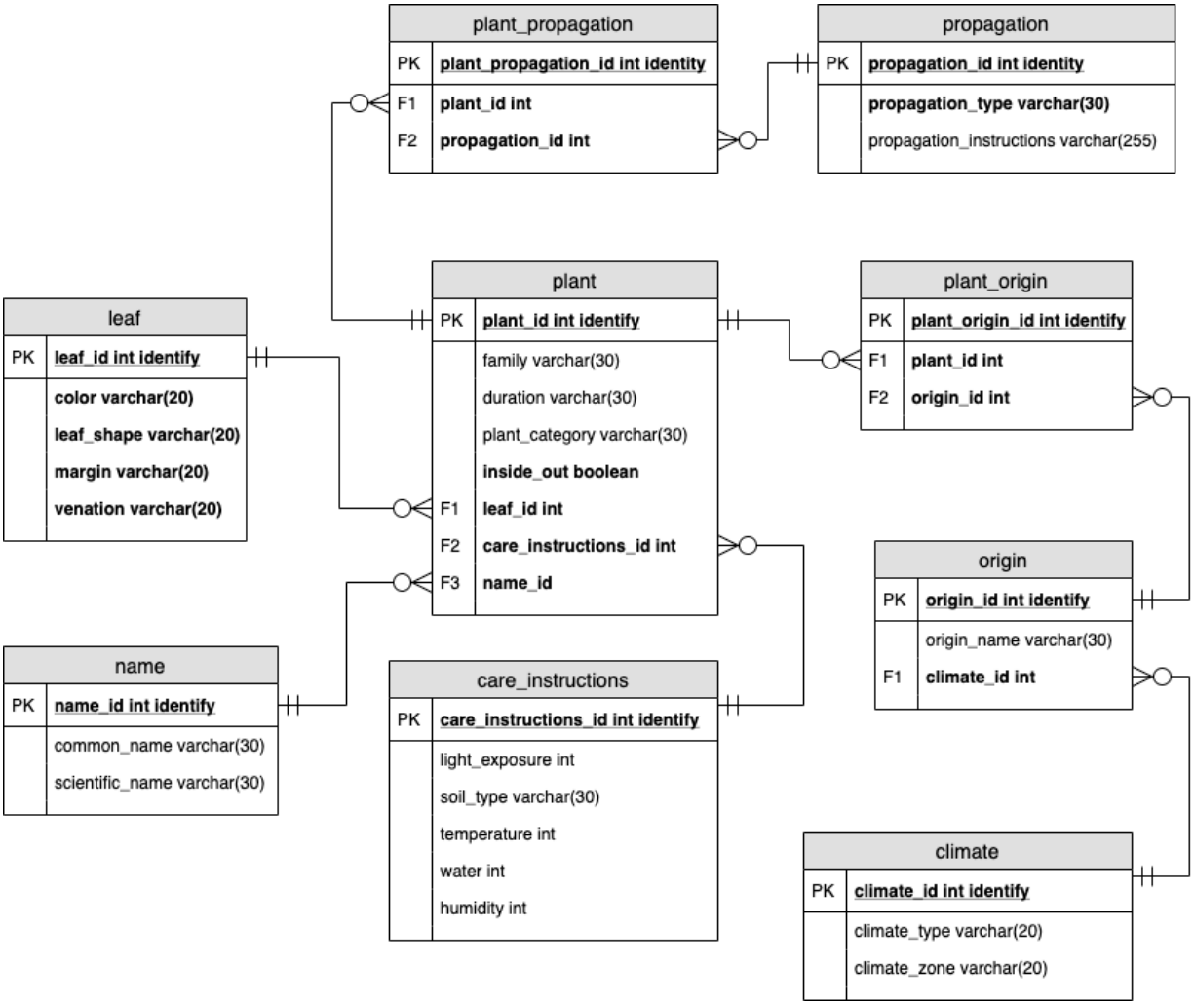
4. The plant origin and plant propagation tables were created to bridge the many-to-many relationships that didn't previously pass 1NF.

5. Foreign keys were added to the plant table to link leaf, name, and care instructions to the plant table in order to pass 2NF and 3NF.

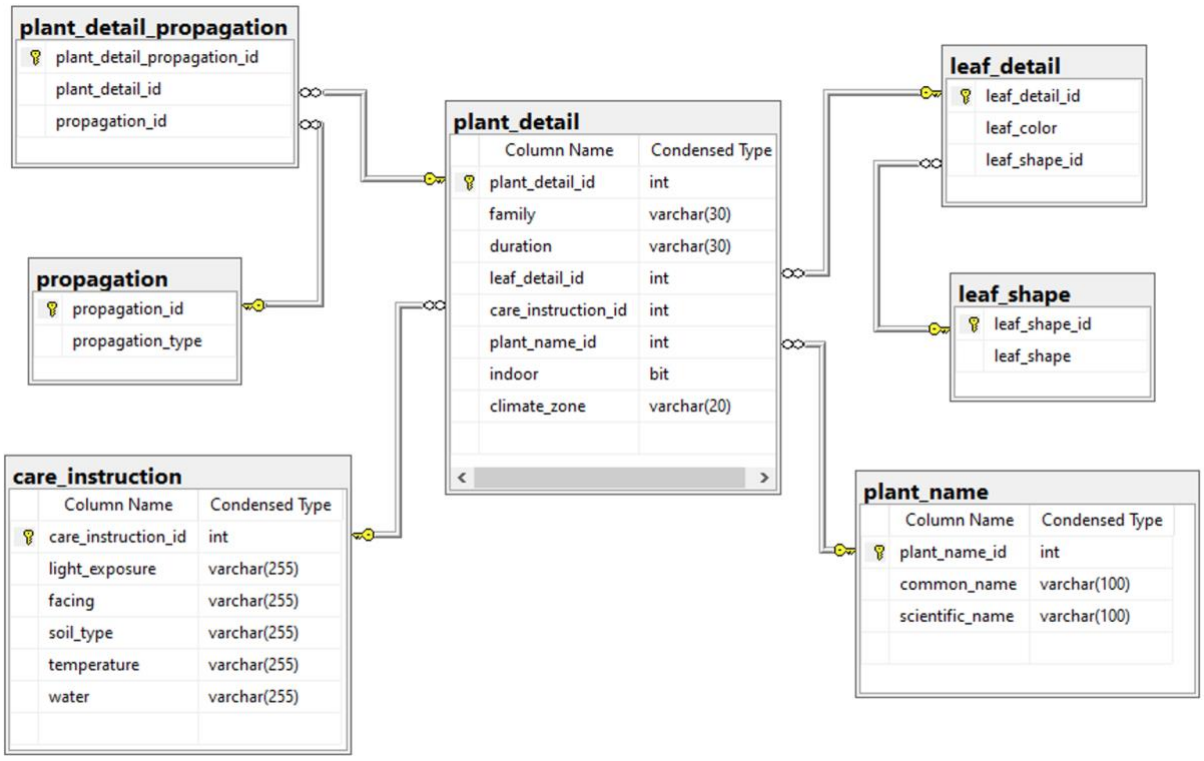
6. A climate table was created out of climate zone which was a candidate key in the origin table. A foreign key was added to the origin table to link the climate table to the origin table.

7. Physical attributes in the leaf table were expanded to margin and venation.

8. First through third normal form for all tables were implemented



Updated Logical Model



After researching the data available for plants, I found that certain information was not as available as I had hoped. The ERD was changed to account for this. The primary changes were:

1. The origin table and plant_origin table were removed because origin_name was more difficult to find than anticipated. The climate zones were added to the plant table.

2. Propagation instructions were removed because the propagation type is sufficient in determining how a plant propagates.

3. The plant table name was changed to plant_details.

4. The leaf margin and venation were not as readily available as I had hoped, so they were removed. The leaf table was changed to leaf_details.

5. A leaf shape table was added that would connect to the leaf detail table.

Physical Database Design

```
-- Create leaf_shape table
If exists (select * from sys.tables where Name = 'leaf_shape')
begin
  drop table leaf_shape
end
create table leaf_shape (
       leaf_shape_id int identity primary key,
       leaf_shape varchar(20),
       constraint u1_leaf_shape unique(leaf_shape)
);

-- Create leaf_detail table
If exists (select * from sys.tables where Name = 'leaf_detail')
begin
  drop table leaf_detail
end
create table leaf_detail (
       leaf_detail_id int identity primary key,
       leaf_color varchar(20),
       leaf_shape_id int references leaf_shape(leaf_shape_id)
       constraint u1_shape_color unique(leaf_shape_id, leaf_color)
);

-- Create plant_name table
If exists (select * from sys.tables where Name = 'plant_name')
begin
  drop table plant_name
end
create table plant_name (
       plant_name_id int identity primary key,
       common varchar(100),
       scientific varchar(100),
       constraint u1_common unique(common),
       constraint u2_scientific unique(scientific)
);

-- Create care_instruction table
If exists (select * from dbo.care_instruction)
begin
  drop table care_instruction
end
create table care_instruction (
       care_instruction_id int identity primary key,
       light_exposure varchar(255),
```

```
	facing varchar(20),
	soil_type varchar(255),
	temperature varchar(20),
	water varchar(255),
);

-- Create plant_detail table
If exists (select * from plant_detail)
begin
  drop table plant_detail
end
create table plant_detail (
	plant_detail_id int identity primary key,
	family varchar(30),
	duration varchar(30),
	category varchar(30),
	inside_out binary,
	leaf_id int references leaf(leaf_id),
	care_instruction_id int references care_instruction(care_instruction_id),
	name_id int references plant_name(plant_name_id)
);

-- Create propagation table
If exists (select * from sys.tables where Name = 'propagation')
begin
  drop table propagation
end
create table propagation (
	propagation_id int identity primary key,
	instructions varchar(255)
);

-- Create plant_detail_propagation table
If exists (select * from sys.tables where Name = 'plant_detail_propagation')
begin
  drop table plant_detail_propagation
end
create table plant_detail_propagation(
	plant_detail_propagation_id int identity primary key,
	plant_detail_id int references plant_detail(plant_detail_id),
	propagation_id int references propagation(propagation_id)
);
```

Data Creation

Most of the information used for the insert statements was taken from online plane websites:

https://plants.ces.ncsu.edu/find_a_plant/
https://garden.org/plants/search/text/?q=Acanthacea
https://en.wikipedia.org/wiki/Aphelandra_squarrosa
https://www.thespruce.com/

```sql
-- Insert values into plant_name table
insert into plant_name (common, scientific)
values
	('Butterfly Palm Tree', 'Dypsis Lutescens'),
	('African Violets', 'Saintpaulia Ionantha'),
	('Black Gold Snake Plant', 'Dracaena Trifasciata'),
	('Sugarvine', 'Cissus Striata'),
	('Marble Queen Pothos', 'Epipremnum Aureum'),
	('Heartleaf Philodendron', 'Philodendron Cordatum'),
	('Boston Fern', 'Nephrolepis Exaltata'),
	('Leopard Plant', 'Farfugium Japonicum'),
	('Hen-and-chickens', 'Jovibarba Heuffelii'),
	('String of Pearls', 'Nacre'),
	('Mexican Fire Cracker', 'Echeveria Setosa'),
	('Little Gem', 'Cremnosedum'),
	('Flapjacks', 'Kalanchoe Thrysiflora'),
	('Powderpuff', 'Pachyveria Exotica'),
	('Zebra Plant', 'Aphelandra Squarrosa'),
	('Echeveria Agavoides', 'Honey Pink Echeveria'),
	('Hens and Chicks', 'Sempervivum Green Wheel'),
	('Star Window', 'Haworthia Cuspidata'),
	('Crinoline Ruffles', 'Echeveria Ruffles'),
	('Nerve Plant', 'Fittonia Gigantea')

-- Insert values into care_instruction table
Insert into care_instruction (light_exposure, facing, soil_type, temperature, water)
values
	('Full Sun to Partial Shade', 'All', 'Rich organic potting soil', '65°F– 75°F',
'Keep moist'),
	('Full Sun to Partial Shade', 'All', 'African violet soil', '65°F– 75°F', 'Keep
moist'),
	('Bright indirect sunlight', 'South - near', 'Rich organic potting soil', '45°F–
77°F', 'Once a week'),
	('Full Sun to Partial Shade', 'All', 'Rich organic potting soil', '60°F– 75°F',
'Allow the soil to dry out a little before watering again'),
	('Partial Shade', 'North', 'Rich organic potting soil','65°F– 75°F', 'Allow the
soil to dry out a little before watering again'),
	('Full Sun to Partial Shade', 'All', 'Succulent soil (3 part compost soil, 2 part
perlite, 1 part sand)', '65°F– 75°F', 'Allow the soil to dry out completely before
watering again'),
	('Partial Shade', 'North', 'Succulent soil (3 part compost soil, 2 part perlite, 1
part sand)', '65°F– 75°F', 'Allow the soil to dry out completely before watering again'),
	('Dappled Sunlight to Partial Shade', 'North or South - near', 'Rich organic
potting soil', '70°F– 80°F', 'Keep moist'),
	('Partial Sun', 'North', 'Succulent soil (3 part compost soil, 2 part perlite, 1
part sand)', '70°F– 90°F', 'Allow the soil to dry out completely before watering again'),
	('Indirect Sun to Partial Shade', 'North or South - near', 'Succulent soil (3 part
compost soil, 2 part perlite, 1 part sand)',    '60°F– 80°F', 'Keep moist'),
	('Partial Shade', 'North', '1 part succulent soil, 1 part african violet soil',
	'65°F– 75°F', 'Allow the soil to dry out a little before watering again')

-- Insert values into propagation table
insert into propagation (propagation_type)
values
	('seed'), ('leaf cuttings'), ('partial leaf cuttings'), ('offsets'), ('stem
cuttings')
```

```sql
-- Insert values into plant_detail_propagation table
insert into plant_detail_propagation(plant_detail_id, propagation_id)
values
	(1, 4), (2, 2), (2, 4), (2, 5), (3, 2), (3, 5), (4, 4), (5, 4), (5, 5), (6, 4),
	(6, 5), (7, 4), (8, 2), (8, 4), (8, 5), (9, 2), (9, 4), (9, 5), (10, 2), (10, 5),
	(11, 2), (11, 5), (12, 2), (12, 5), (13, 2), (13, 5), (14, 2), (14, 4), (15, 2),
	(15, 5), (16, 2), (16, 4), (16, 5), (17, 4), (18, 2), (18, 5), (19, 4), (19, 5),
	(20, 2), (20, 4)

-- Insert values into leaf_shape table
insert into leaf_shape (leaf_description)
values
	('acicular'), ('cordate'), ('deltoid'), ('elliptical'), ('falcate'),
	('hastate'), ('lanceolate'), ('linear'), ('lyrate'), ('obcordate'),
	('oblanceolate'), ('oblong'), ('obovate'), ('orbicular'), ('oval'),
	('ovate'), ('reniform'), ('runcinate'), ('sagittate'), ('spatulate'),
	('pinnate')

-- Insert values into leaf_detail table
insert into leaf_detail (color, leaf_shape_id)
values
	('Green, Yellow, and Gold', 'Pinnate'),
	('Green', 'Oval-Ovate'),
	('Dark Green', 'Obovate'),
	('Green and White', 'Ovate-Lanceolate'),
	('Green', 'Ovate'),
	('Green', 'Pinnate'),
	('Green and Red', 'Obovate'),
	('Lime Green', 'Orbicular'),
	('Green, Grey, and Blue', 'Linear'),
	('Green and Red', 'Oblong'),
	('Green', 'Elliptical - Ovate'),
	('Green, Grey, and Blue', 'Obovate'),
	('Green and White', 'Elliptical - Ovate'),
	('Green and Red', 'Linear'),
	('Lime Green', 'Cordate'),
	('Green, Gold, Yellow, and White', 'Lanceolate - Linear')

-- Insert values into plant_detail table
insert into plant_detail (family, duration, climate_zone, inside_out, leaf_id,
care_instruction_id, name_id)
values
	('Arecaceae', 'Perennial', '9a-11b', 1, 2, 1, 1),
	('Gesneriaceae', 'Perennial', '11-12', 1, 3, 2, 2),
	('Asparagaceae', 'Perennial', '10a-12b', 1, 17, 11, 3),
	('Vitaceae', 'Perennial', '6-9', 1, 4, 3, 4),
	('Araceae', 'Perennial', '10-11', 1, 5, 4, 5),
	('Araceae', 'Perennial', '9-11', 1, 6, 4, 6),
	('Asteraceae', 'Perennial', '10a-13b', 1, 6, 5, 7),
	('Crassulaceae', 'Perennial', '3-11', 1, 8, 6, 9),
	('Asteraceae', 'Perennial', '9-12', 1, 9, 7, 10),
	('Crassulaceae', 'Perennial', '9-12', 1, 10, 6, 11),
	('Crassulaceae', 'Perennial', '8-10', 1, 11, 6, 12),
	('Crassulaceae', 'Perennial', '10-12', 1, 12, 6, 13),
	('Crassulaceae', 'Perennial', '9b', 1, 13, 6, 14),
	('Acanthaceae', 'Perennial', '10a-11', 1, 14, 8, 15),
	('Crassulaceae', 'Perennial', '10a-11b', 1, 13, 6, 16),
	('Crassulaceae', 'Perennial', '4-9', 1, 15, 6, 17),
```

```
        ('Xanthorrhoeaceae', 'Perennial', '9a-11b', 1, 16, 9, 18),
        ('Crassulaceae', 'Perennial', '8-24', 1, 13, 6, 19),
        ('Crassulaceae', 'Perennial', '11', 1, 6, 10, 20),
        ('Xanthorrhoeaceae', 'Perennial', '9a-11b', 1, 16, 9, 21)

-- Create view for all leaf_shape
create view v_leaf_info as
select plant_name.common_name, plant_name.scientific_name, leaf_color, leaf_shape  from
plant_detail
inner join leaf_detail
        on leaf_detail.leaf_detail_id = plant_detail.leaf_detail_id
inner join leaf_shape
        on leaf_shape.leaf_shape_id = leaf_detail.leaf_shape_id
inner join plant_name
        on plant_name.plant_name_id = plant_detail.plant_name_id

-- Create view for care_instructions
create view v_plant_care as
select plant_name.common_name, plant_name.scientific_name,
care_instruction.light_exposure, care_instruction.facing, care_instruction.soil_type,
care_instruction.temperature, care_instruction.water from plant_detail
inner join care_instruction
        on care_instruction.care_instruction_id = plant_detail.care_instruction_id
inner join plant_name
        on plant_name.plant_name_id = plant_detail.plant_name_id
```

Data Manipulation

```
-- Examples of Update
update plant_name
set common_name = 'Hen and Chickens'
where plant_name_id = 9

update plant_detail
set climate_zone = '9a-11'
where plant_detail_id = 1

-- Example of Deletion
delete from plant_detail
where plant_name_id = 13

delete from plant_name
where common_name = 'Flapjacks'
```

Creating update functions is important when new information is learned. Though most of this data was found online, some of these instructions will need to be changed when they are found to not work. A plant might be listed as ideal in south facing windows but thrive more in north facing windows. A stored procedure would ensure that no update function written could disrupt the database.

Deletions will be important when a plant dies or is given away. It will also be important for propagation types that don't work. A plant might be listed as a leaf propagator, but I might find that it can only be propagated through offsets. Using a store procedure would make this process simpler.

Lastly, a stored procedure for inserting data might be important when a new plant is added. Rather than writing another insert command, the name and details of the plant could be added by including them as parameters.

```
-- Update window facing information in plant_detail table
create procedure change_facing (
       @care_instruction_id int,
       @facing varchar(30)
) as
begin
       update care_instruction
       set facing = @facing
       where care_instruction_id = @care_instruction_id
end;

-- Remove plant names from plant_name table
create procedure remove_plant (
       @plant_id int
) as
begin
       delete from plant_name where plant_name_id = @plant_id
end;

-- Add plant names to plant_name table
create procedure add_plant (
       @common varchar(100),
       @scientific varchar(100)
) as
begin
       insert into plant_name
    (common_name, scientific_name)
  values
    (@common, @scientific)
  return @@identity
end;
```

Answering Data Questions

Two views were created to answer the data questions. The first view named v_plant_care combines the plant_detail with the plant_name and care instruction tables. This allows users to view the care instruction for any plant they require. The questions being answered are:

How much sun does this plant require?
How often should I water this plant?
What kind of soil should I use with this plant?
What is the ideal temperature for this plant?
Which direction should this plant be facing?

View for v_plant_care:

```
create view v_plant_care as
select
	dbo.plant_name.common_name,
	dbo.plant_name.scientific_name,
	dbo.care_instruction.light_exposure,
	dbo.care_instruction.facing,
	dbo.care_instruction.soil_type,
	dbo.care_instruction.temperature,
	dbo.care_instruction.water
FROM dbo.plant_detail
inner join dbo.care_instruction
	ON dbo.care_instruction.care_instruction_id = dbo.plant_detail.care_instruction_id
inner join dbo.plant_name
	ON dbo.plant_name.plant_name_id = dbo.plant_detail.plant_name_id
```

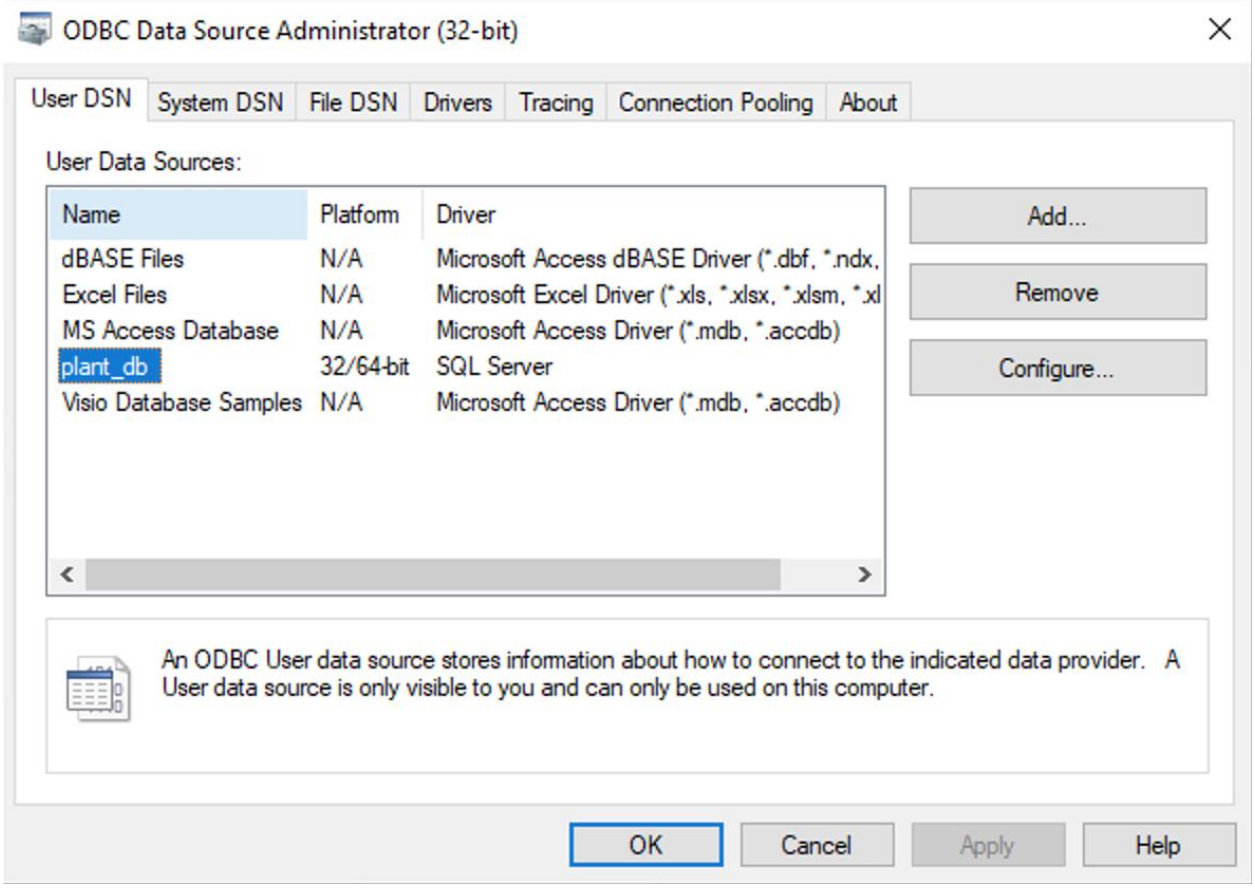
The second view allows the user to lookup the plant name given the leaf color and leaf shape. Though more information would be required to make identifying plants more useful, I was limited in what was easily available online. This view joins the plant_detail table with the leaf_detail table and then join the leaf_shape table to the leaf_detail table. The question being answered with this view is how can I identify the plant in order to take care of it.

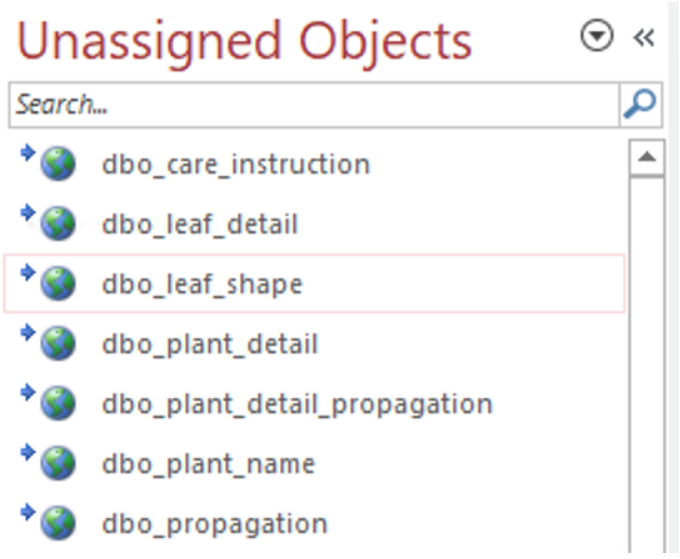
View for v_leaf_info:

```
create view v_leaf_info as
select
	dbo.plant_name.common_name,
	dbo.plant_name.scientific_name,
	dbo.leaf_detail.leaf_color,
	dbo.leaf_shape.leaf_shape
from dbo.plant_detail
inner join dbo.leaf_detail
	ON dbo.leaf_detail.leaf_detail_id = dbo.plant_detail.leaf_detail_id
inner join dbo.leaf_shape
	ON dbo.leaf_shape.leaf_shape_id = dbo.leaf_detail.leaf_shape_id
inner join dbo.plant_name
	ON dbo.plant_name.plant_name_id = dbo.plant_detail.plant_name_id
```

Implementation

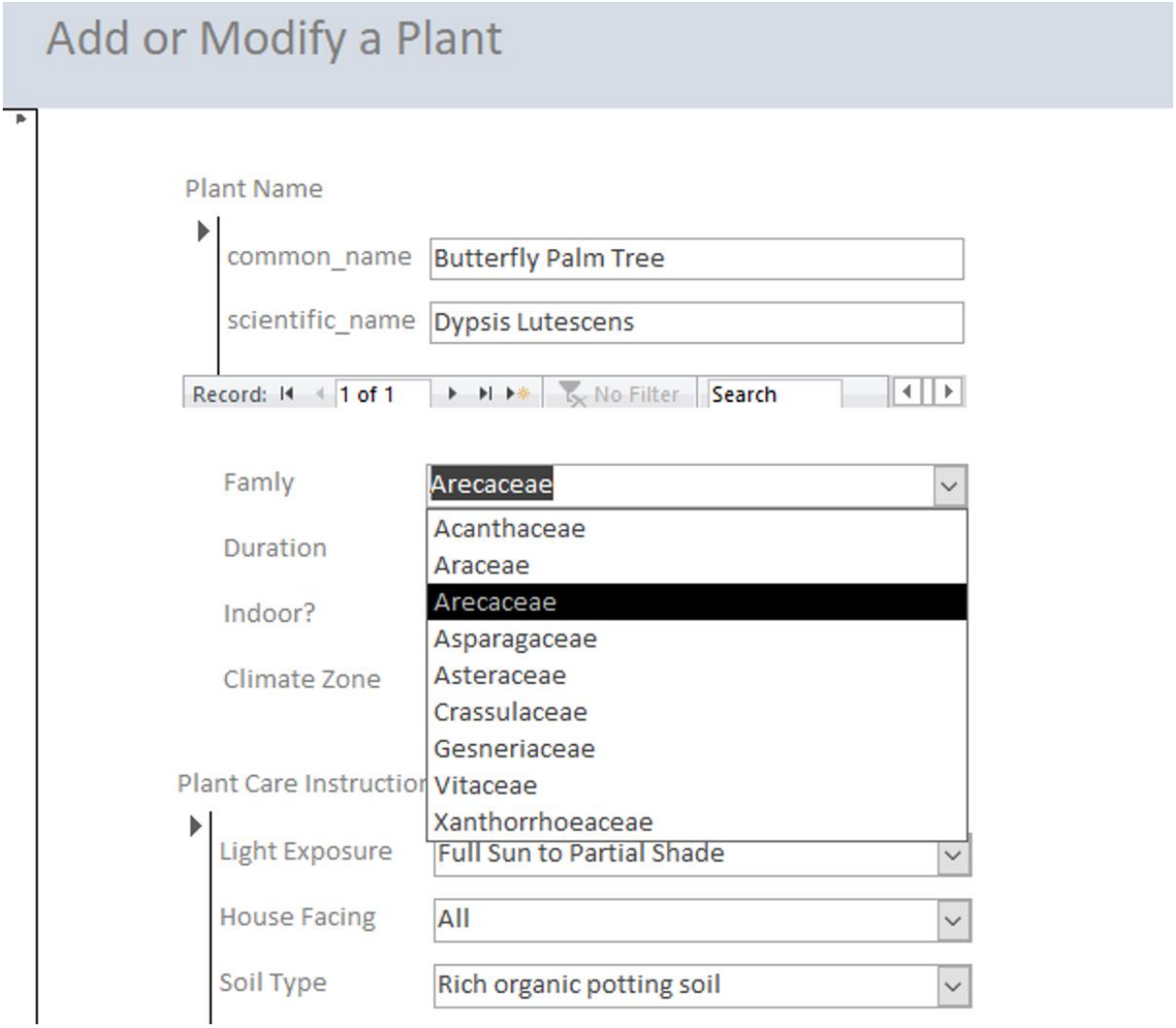
I decided to use Access as the front end for this database. I used ODBC to link the SQL Server database by creating a data source name plant_db.

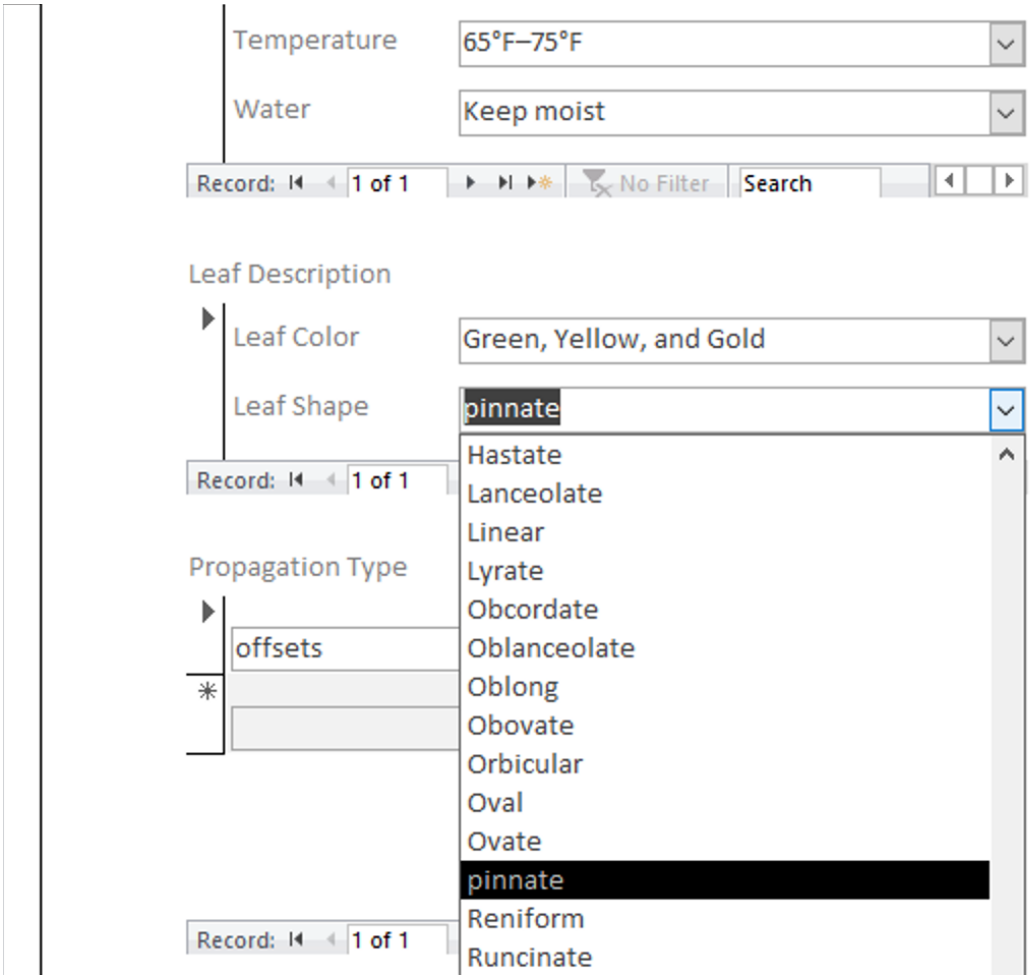




In Access, I created one form and one report to both manipulate and view the data. The form combined all the tables together in order to add or modify data in the database. When a new plant is added, this form can insert that information into the SQL Server database. For each form, I created several queries to return the distinct values listed in the desired column. An example of this query is:

```
SELECT DISTINCT dbo_leaf_detail.leaf_color
FROM dbo_leaf_detail;
```





To create this form, several sub-forms were used. The tables plant_name, care_instructions, leaf_detail, and propagation type were all sub-forms of the plant_detail form. After testing several plant_detail_ids, it appears that this form succeeds in updating and inserting data.

In addition to this form I also created a report to show all the details of each plant listed in a block formation. Though I was not able to create an Access filter, I can search the report for the specific plant I'm looking for. This answers the following business question: What is the name of this plant so I can determine how to care for it?

Plant Report

Common Name	Scientific Name	Family	Zone	Light Exposure	Water Instructions	Leaf Color	Leaf Shape	Propagation Type
African Violets	Saintpaulia Ionantha	Gesneriaceae	11-12	Full Sun to Partial Shade	Keep moist	Green	Oval	leaf cuttings
								offsets
								stem cuttings
Black Gold Snake Plant	Dracaena Trifasciata	Asparagaceae	10a-12b	Partial Sun	Allow the soil to dry out a little before v	Green, Gold, Yellow, and	Linear	stem cuttings
								leaf cuttings
Boston Fern	Nephrolepis Exaltata	Asteraceae	10a-13b	Partial Shade	Allow the soil to dry out a little before v	Green	Ovate	offsets
Butterfly Palm Tree	Dypsis Lutescens	Arecaceae	9a-11b	Full Sun to Partial Shade	Keep moist	Green, Yellow, and Gold	pinnate	offsets
Crinoline Ruffles	Echeveria Ruffles	Crassulaceae	8-24	Full Sun to Partial Shade	Allow the soil to dry out completely bef	Green, Grey, and Blue	Obovate	stem cuttings
								leaf cuttings
Echeveria Agavoides	Honey Pink Echeveria	Crassulaceae	10a-11b	Full Sun to Partial Shade	Allow the soil to dry out completely bef	Green, Grey, and Blue	Obovate	leaf cuttings
								stem cuttings
Flapjacks	Kalanchoe Thrysiflora	Crassulaceae	10-12	Full Sun to Partial Shade	Allow the soil to dry out completely bef	Green	Obovate	stem cuttings
								leaf cuttings
Heartleaf Philodendron	Philodendron Cordatum	Araceae	9-11	Full Sun to Partial Shade	Allow the soil to dry out a little before v	Green	Ovate	offsets
								stem cuttings
Hen-and-chickens	Jovibarba Heuffelii	Crassulaceae	3-11	Full Sun to Partial Shade	Allow the soil to dry out completely bef	Green and Red	Obovate	stem cuttings
								leaf cuttings
								offsets
Hens and Chicks	Sempervivum Green Wheel	Crassulaceae	4-9	Full Sun to Partial Shade	Allow the soil to dry out completely bef	Green and Red	Obovate	leaf cuttings
								offsets

The report joins the plant_detail table with the care_instruction table, the leaf_detail table, the leaf_shape table and the propagation table. Because the form stretches horizontally and cannot be fully capture with a screenshot, I have divided it into two section.

Reflection

What assumptions did you have at the start of your project that changed by the end? Think in terms of both your own problem domain as well as your knowledge of the process.

Initially, I had anticipated finding a dataset that contained most of the information I needed. I had developed the ERD based on easily finding that information. Instead, details like plant margin and venation were more difficult to come by. I ended up changing the ERD based on what information I could find. I had also planned on using leaf information to readily identify the plant, but that process required more details. I was also considering using my outdoor plants, but after the inefficiency of looking up the indoor plants, I decided that it was out of scope.

The next time you do this, what will be different?

For next time I would be more thorough in my research. Rather than looking up the details for each plant, I would scrape the data so it would be more consistent. It may have also been useful to create more reference tables to keep data consistency. Some kind of standardization could be used to ensure that information falls under precise categories. For example, plant color could be [green], [green and red], or [green, red and gold]. These are not well-defined categories. Perhaps a many-to-many table could have been created instead. In terms of implementation, I may try to find alternative form tools to Access. Though Access is simple to use, it is limited in what it can accomplish.

Regardless of whether you go through these steps again, how do you think it will inform your approach to data as an information professional?

I have learned to consider the business questions first before jumping into database design. Though this database was made as a plant lookup, more

extensive business problems require clean, normalized tables. It gives me a perspective of how important databases are for businesses.

Summary

In conclusion, the plant database contains three tools that are useful for plant care. The first is a list of the care requirements: sun exposure, window facing, water frequency and temperature. These give me basic information for keeping a specific plant alive. The second piece of information is the propagation type. When I want to propagate a plant, this information will tell me which techniques are used. If I look up a plant and see that it only contains offsets as a propagation method, I'll know not to waste my time trying to take a leaf clipping. The third tool is the ability to look up a plant in order to find the care information required. If I forget to label something, I should be able to find the plant name by looking up the leaf details. This may not give me the precise plant I'm looking for, but it will narrow down the field. I decided to use the tools covered in the lectures because they contained everything I would require. Access allowed me to not only view the data, but also manipulate it when needed. The result of this project has given me the information to become a better, more informed gardener.