University of British Columbia, Vancouver
Department of Computer Science

CPSC 304 Project Cover Page

Milestone #: 2

Date: 02/08/2023

Group Number: 55

Name	Student Number	CS Alias (Userid)	Preferred E-mail Address
Richard Han	50188283	s5v7k	rrhan2002@gmail.com
Clive Yong	34877712	z0e0f	clive.yong.747@gmail.com
Mana Longhenry	43629526	v5w1g	arlonghenry@gmail.com

By typing our names and student numbers in the above table, we certify that the work in the attached assignment was performed solely by those whose names and student IDs are included above. (In the case of Project Milestone 0, the main purpose of this page is for you to let us know your e-mail address, and then let us assign you to a TA for your project supervisor.)

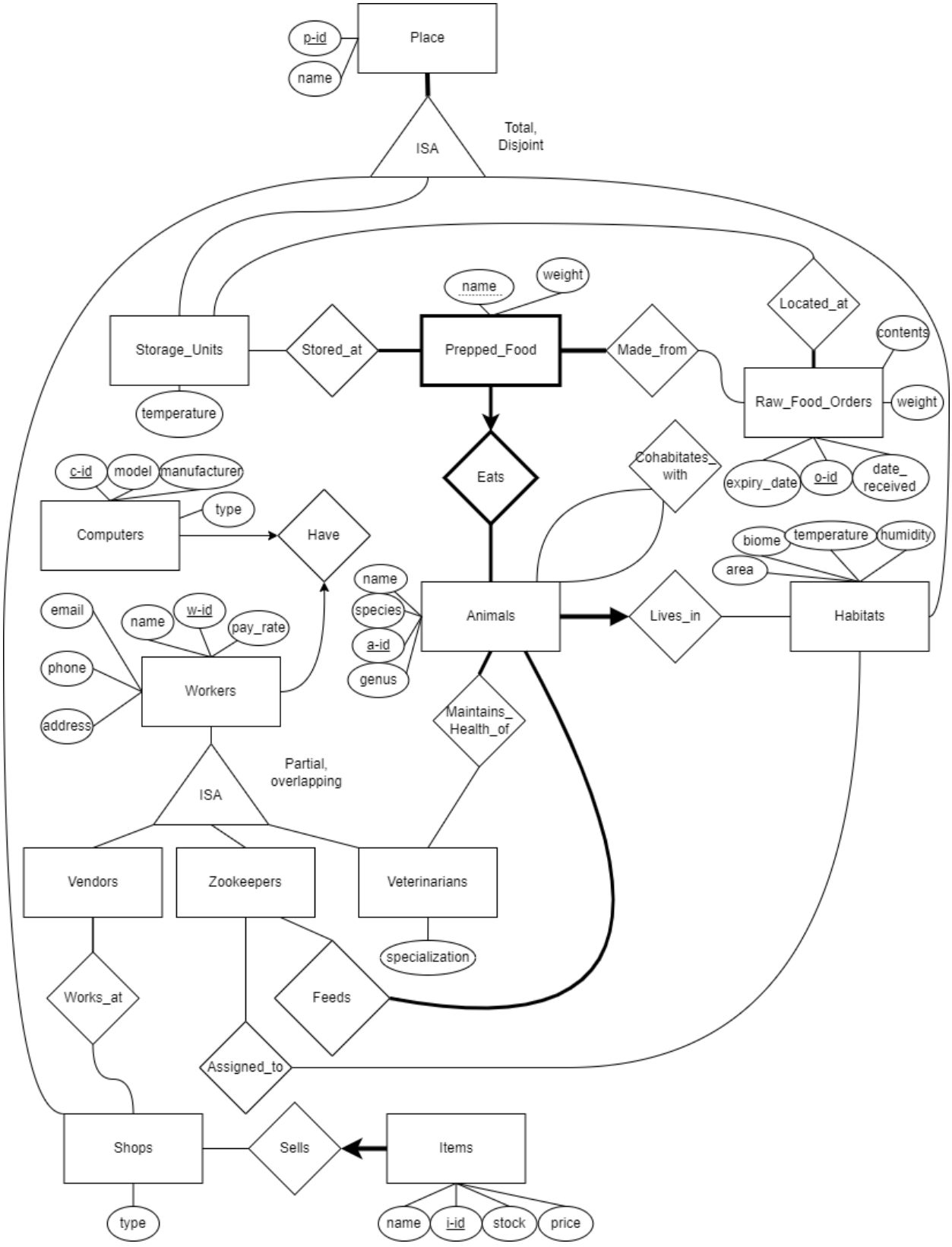
In addition, we indicate that we are fully aware of the rules and consequences of plagiarism, as set forth by the Department of Computer Science and the University of British Columbia

Project Summary:

This application will model a zoo management system to include care of animals, maintenance of habitats, logistics of its workers and also the zoo's shops and their associated merchandise. This application aims to aid a zoo in the management of its workers, property and animals. The application will organize animals, habitats, workers and vendors.

ER Diagram

- made Prepped Food box bold
- Merged Contact Information with Workers
- Added 1-to-1 relationship Computers and Workers
- Added manufacturer attribute to Computers
- Added genus to animals (now can have a new functional dependency)
- Removed h-id from habitats
- Zoo Keepers to Zookeepers
- pluralized vendor and veterinarian
- lower-case for all attributes
- moved weight from prepped food to raw food orders
- in charge of feeding is now feeds
- prepped food has total participation
- items in one to many relationship
- accepted ta suggestions
- added weight to Prepped_Food attribute
- changed amount to stock
- in habitats changed size to area


Place
p-id
name
ISA
Total,
Disjoint
Storage_Units
Stored_at
Prepped_Food
name
weight
Made_from
Located_at
contents
Raw_Food_Orders
weight
temperature
Eats
Cohabitates_
with
expiry_date
o-id
date_
received
c-id
model
manufacturer
Computers
type
Have
biome
temperature
humidity
area
email
name
w-id
pay_rate
phone
address
Workers
name
species
a-id
genus
Animals
Lives_in
Habitats
Maintains_
Health_of
Partial,
overlapping
ISA
Vendors
Zookeepers
Veterinarians
specialization
Works_at
Feeds
Assigned_to
Shops
Sells
Items
type
name
i-id
stock
price

Schema
Underline is primary key, bold is foreign key

Computers(c-id: string, model: string, manufacturer: string, type: string)
model, manufacture, type is not null

Workers(w-id: string, name: string, pay_rate: float, email: string, phone: string, address: string)
name, pay_rate, email, phone, address is not null
email, phone is unique
email is a CK
phone is a CK

Zookeepers(**w-id**: string)

Vendors(**w-id**: string)

Veterinarians(**w-id**: string, specialization: string)

Habitats(p-id: string, name: string, biome: string: area: integer, temperature: integer, humidity: integer)
biome, area is not null

Shops(p-id: string, type: string)
type is not null

Storage_Units(p-id: string, name: string, temperature: integer)
temperature is not null

Items(i-id: string, **p-id**: string, name: string, stock: integer, price: float)
name, stock, price, p-id not null

Animals(a-id: string, **p-id**: string, name: string, species: string, genus: string)
species is not null, genus is not null, p-id is not null

Prepped_Food(**a-id**: string, name: string, weight: float)
p-id, o-id is not null

Raw_Food_Orders(o-id: string, contents: string, weight: integer, date_recieved: date, expiry_date: date)
contents, weight, date_recieved, expiry_date is not null

Have(**c-id**: string, **w-id**: string)
w-id is unique, not null
w-id is candidate key

Works_at(**w-id**: string, **p-id**: string)

Assigned_to(**w-id**: string, **p-id**: string)

Feeds(**w-id**: string, **a-id**: string)

Maintains_Health_of(**w-id**: string, **a-id**: string)

Cohabitates_with(**a-id1**: string, **a-id2**: string)

Made_from(**a-id**: string, **name**: string, **o-id**: string)

Stored_at(**a-id**: string, **name**: string, **p-id**: string)

Located_at(**o-id**: string, **p-id**: string)

Functional Dependencies

Computers
c-id -> model, manufacturer, type
model -> manufacturer, type

Workers
w-id -> name, pay_rate, address, email, phone
email -> w-id, name, pay_rate, address, phone
phone -> w-id, name, pay_rate, address, email

Zookeepers
N/A

Vendors
N/A

Veterinarians
w-id -> specialization

Habitats
p-id -> name, biome, area, temperature, humidity
biome -> temperature, humidity

Shops
p-id -> name, type

Storage_Units
p-id -> name, temperature

Items
i-id -> name, stock, price, p-id

Animals
a-id -> p-id, name, species, genus
Species ⟶ genus

Prepped_Food
a-id, name -> weight

Raw_Food_Orders
o-id -> contents, weight, date_recieved, expiry_date

Have

w-id -> c-id
c-id -> w-id

Works_at
N/A

Assigned_to
N/A

Feeds
N/A

Maintains_Health_of
N/A

Cohabitates_with
N/A

Made_from
N/A

Stored_at
N/A

Located_at
N/A

Normalization

Computers decomposes into Computers1, Computers2 because model -> manufacturer, type violates BCNF:
Computers1(c-id: string, **model**: string), Computers2(model: string, manufacturer: string, type: string)

Workers(w-id: string, name: string, pay_rate: float, email: string, phone: string, address: string)
Zookeepers(**w-id**: string)
Vendors(**w-id**: string)
Veterinarians(**w-id**: string, specialization: string)

Habitats decomposes into Habitats1, Habitats2 because biome -> temperature, humidity violates BCNF:
Habitats1(p-id: string, name: string, **biome**: string, area: integer), Habitats2(biome: string, temperature: integer, humidity: integer)

Shops(p-id: string, type: string)
Storage_Units(p-id: string, name: string, temperature: integer)
Items(i-id: string, **p-id**: string, name: string, stock: integer, price: float)

Animals decomposes into Animals1, Animals2 because species → genus violates BCNF:
Animals1(a-id: string, **p-id**: string, name: string, **species**: string), Animals2(species:string, genus: string)

Prepped_Food(**a-id**: string, name: string, weight: float)
Raw_Food_Orders(o-id: string, contents: string, weight: integer, date_recieved: date, expiry_date: date)
Have(**c-id**: string, **w-id**: string)
Works_at(**w-id**: string, **p-id**: string)
Assigned_to(**w-id**: string, **p-id**: string)
Feeds(**w-id**: string, **a-id**: string)
Maintains_Health_of(**w-id**: string, **a-id**: string)
Cohabitates_with(**a-id1**: string, **a-id2**: string)
Made_from(**a-id**: string, **name**: string, **o-id**: string)
Stored_at(**a-id**: string, **name**: string, **p-id**: string)
Located_at(**o-id**: string, **p-id**: string)

SQL DDL

```
CREATE TABLE Computers2(
        model VARCHAR(20) PRIMARY KEY,
        manufacturer VARCHAR(20) NOT NULL,
        type VARCHAR(20) NOT NULL
        );

CREATE TABLE Computers1(
        c_id VARCHAR(20) PRIMARY KEY,
        model VARCHAR(20) NOT NULL,
        FOREIGN KEY (model) REFERENCES Computers2(model)
);

CREATE TABLE Workers(
        w_id VARCHAR(20) PRIMARY KEY,
        name VARCHAR(20) NOT NULL,
        pay_rate FLOAT NOT NULL,
        address VARCHAR(30) NOT NULL,
        email VARCHAR(30) UNIQUE,
        phone VARCHAR(13) UNIQUE
);

CREATE TABLE Zookeepers(
        w_id VARCHAR(20) PRIMARY KEY,
        FOREIGN KEY (w_id) REFERENCES Workers(w_id)
);

CREATE TABLE Vendors(
        w_id VARCHAR(20) PRIMARY KEY,
        FOREIGN KEY (w_id) REFERENCES Workers(w_id)
);

CREATE TABLE Veterinarians(
        w_id VARCHAR(20) PRIMARY KEY,
        specialization VARCHAR(20),
        FOREIGN KEY (w_id) REFERENCES Workers(w_id)
);


CREATE TABLE Habitats2(
        biome VARCHAR(50) PRIMARY KEY,
        temperature INTEGER,
        humidity INTEGER
);
```

```
CREATE TABLE Habitats1(
        p_id VARCHAR(20) PRIMARY KEY,
        name VARCHAR(20),
        biome VARCHAR(50),
        area INTEGER,
        FOREIGN KEY (biome) REFERENCES Habitats2
);

CREATE TABLE Shops(
        p_id VARCHAR(20) PRIMARY KEY,
        type VARCHAR(20) NOT NULL
);

CREATE TABLE Storage_Units(
        p_id VARCHAR(20) PRIMARY KEY,
        name VARCHAR(20),
        temperature INTEGER NOT NULL
);

CREATE TABLE Items(
        i_id VARCHAR(20) PRIMARY KEY,
        p_id VARCHAR(20) NOT NULL,
        name VARCHAR(30) NOT NULL,
        stock INTEGER NOT NULL,
        price FLOAT NOT NULL,
        FOREIGN KEY (p_id) REFERENCES Shops(p_id)
);

CREATE TABLE Animals2(
        species VARCHAR(20) PRIMARY KEY,
        genus VARCHAR(20)
        );

CREATE TABLE Animals1(
        a_id VARCHAR(20) PRIMARY KEY,
        p_id VARCHAR(20),
        name VARCHAR(30),
        species VARCHAR(20),
        FOREIGN KEY (species) REFERENCES Animals2,
        FOREIGN KEY (p_id) REFERENCES Habitats1(p_id)
);

CREATE TABLE Prepped_Food(
```

```
        a_id VARCHAR(20),
        name VARCHAR(20),
        weight FLOAT,
        PRIMARY KEY (a_id, name),
        FOREIGN KEY (a_id) REFERENCES Animals1(a_id) ON DELETE CASCADE
);

CREATE TABLE Raw_Food_Orders(
        o_id VARCHAR(20) PRIMARY KEY,
        contents VARCHAR(50) NOT NULL,
        weight INTEGER NOT NULL,
        date_received DATE NOT NULL,
        expiry_date DATE NOT NULL
 );

 CREATE TABLE Have(
        c_id VARCHAR(20) PRIMARY KEY,
        w_id VARCHAR(20) UNIQUE NOT NULL ,
        FOREIGN KEY (c_id) REFERENCES Computers1(c_id),
        FOREIGN KEY (w_id) REFERENCES Workers(w_id)
);

CREATE TABLE Works_at(
        w_id VARCHAR(20),
        p_id VARCHAR(20),
        PRIMARY KEY (w_id, p_id),
        FOREIGN KEY (w_id) REFERENCES Workers,
        FOREIGN KEY (p_id) REFERENCES Shops
);

CREATE TABLE Assigned_to(
        w_id VARCHAR(20),
        p_id VARCHAR(20),
        PRIMARY KEY(w_id, p_id),
        FOREIGN KEY (w_id) REFERENCES Zookeepers(w_id),
        FOREIGN KEY (p_id) REFERENCES Habitats1(p_id)
);

CREATE TABLE Feeds(
        w_id VARCHAR(20),
        a_id VARCHAR(20),
        PRIMARY KEY (w_id, a_id),
        FOREIGN KEY (w_id) REFERENCES  Zookeepers(w_id),
        FOREIGN KEY (a_id) REFERENCES Animals1(a_id)
```

```
);

CREATE TABLE Maintains_Health_of(
        w_id VARCHAR(20),
        a_id VARCHAR(20),
        PRIMARY KEY (w_id, a_id),
        FOREIGN KEY (w_id) REFERENCES Veterinarians(w_id),
        FOREIGN KEY (a_id) REFERENCES Animals1(a_id)
);

CREATE TABLE Cohabitates_with(
        a_id1 VARCHAR(20),
        a_id2 VARCHAR(20),
        PRIMARY KEY(a_id1, a_id2),
        FOREIGN KEY (a_id1) REFERENCES Animals1(a_id),
        FOREIGN KEY (a_id2) REFERENCES Animals1(a_id)
);

CREATE TABLE Made_from(
        a_id VARCHAR(20),
        name VARCHAR(30),
        o_id VARCHAR(20),
        PRIMARY KEY(a_id, name, o_id),
        FOREIGN KEY (a_id, name) REFERENCES Prepped_Food(a_id, name),
        FOREIGN KEY (o_id) REFERENCES Raw_Food_Orders(o_id)
);

CREATE TABLE Stored_at(
        a_id VARCHAR(20),
        name VARCHAR(30),
        p_id VARCHAR(20),
        PRIMARY KEY(a_id, name, p_id),
        FOREIGN KEY (a_id, name) REFERENCES Prepped_Food(a_id, name),
        FOREIGN KEY (p_id) REFERENCES Storage_Units(p_id)
);

CREATE TABLE Located_at(
        o_id VARCHAR(20),
        p_id VARCHAR(20),
        PRIMARY KEY(o_id, p_id),
        FOREIGN KEY (o_id) REFERENCES Raw_Food_Orders(o_id),
        FOREIGN KEY (p_id) REFERENCES Storage_Units(p_id)
);
```

Insert Statements

```
INSERT
INTO Computers2(model, manufacturer, type)
VALUES ('Macbook Air 2020', 'Apple', 'Laptop');

INSERT
INTO Computers2(model, manufacturer, type)
VALUES ('Macbook Air 2019', 'Apple', 'Laptop');

INSERT
INTO Computers2(model, manufacturer, type)
VALUES ('Macbook Air 2018', 'Apple', 'Laptop');

INSERT
INTO Computers2(model, manufacturer, type)
VALUES ('Macbook Air 2017', 'Apple', 'Laptop');

INSERT
INTO Computers2(model, manufacturer, type)
VALUES ('Macbook Air 2016', 'Apple', 'Laptop');

INSERT
INTO Computers1(c_id, model)
VALUES ('1', 'Macbook Air 2020');

INSERT
INTO Computers1(c_id, model)
VALUES ('2', 'Macbook Air 2019');

INSERT
INTO Computers1(c_id, model)
VALUES ('3', 'Macbook Air 2018');

INSERT
INTO Computers1(c_id, model)
VALUES ('4', 'Macbook Air 2017');

INSERT
INTO Computers1(c_id, model)
VALUES ('5', 'Macbook Air 2016');
```

```
INSERT
INTO Workers(w_id, name, pay_rate, address, email, phone)
VALUES ('1', 'Clive Yong', 15.40, '1234 noname street', 'cliveyong@domain.com', '6041231234');

INSERT
INTO Workers(w_id, name, pay_rate, address, email, phone)
VALUES ('2', 'Mana Longhenry', 16.00, '4321 thisthing street', 'manalong@domain.com',
'6042468100');

INSERT
INTO Workers(w_id, name, pay_rate, address, email, phone)
VALUES ('3', 'Skye Joe', 12.34, '2468 somewhere way', 'SkyeJoe@domain.com','5852111212');

INSERT
INTO Workers(w_id, name, pay_rate, address, email, phone)
VALUES ('4', 'Bob Way', 15.40, '1234 noname street','bobway@domain.com', '6041231233');

INSERT
INTO Workers(w_id, name, pay_rate, address, email, phone)
VALUES ('5', 'Richard Han', 15.40, '3579 somewhere place',
'richardh@domain.com','6043215678');

INSERT
INTO Zookeepers(w_id)
VALUES ('1');

INSERT
INTO Zookeepers(w_id)
VALUES ('2');

INSERT
INTO Zookeepers(w_id)
VALUES ('3');

INSERT
INTO Zookeepers(w_id)
VALUES ('4');

INSERT
INTO Zookeepers(w_id)
VALUES ('5');

INSERT
INTO Vendors(w_id)
```

```
VALUES ('1');

INSERT
INTO Vendors(w_id)
VALUES ('2');

INSERT
INTO Vendors(w_id)
VALUES ('3');

INSERT
INTO Vendors(w_id)
VALUES ('4');

INSERT
INTO Vendors(w_id)
VALUES ('5');

INSERT
INTO Veterinarians(w_id, specialization)
VALUES ('1', 'large felines');

INSERT
INTO Veterinarians(w_id, specialization)
VALUES ('2', 'small felines');

INSERT
INTO Veterinarians(w_id, specialization)
VALUES ('3', 'birds');

INSERT
INTO Veterinarians(w_id, specialization)
VALUES ('4', 'reptiles');

INSERT
INTO Veterinarians(w_id, specialization)
VALUES ('5', 'bears');

INSERT
INTO Habitats2(biome, temperature, humidity)
VALUES ('Asian Taiga', 32, 70);

INSERT
INTO Habitats2(biome, temperature, humidity)
```

```
VALUES ('African Savanna', 30, 20);

INSERT
INTO Habitats2(biome, temperature, humidity)
VALUES ('Antarctic Tundra', -20, 1);

INSERT
INTO Habitats2(biome, temperature, humidity)
VALUES ('African Grasslands', 28, 25);

INSERT
INTO Habitats2(biome, temperature, humidity)
VALUES ('North American Woodlands', 25, 70);

INSERT
INTO Habitats1(p_id, name, biome, area)
VALUES ('001', 'Tiger Habitat', 'Asian Taiga', 37);

INSERT
INTO Habitats1(p_id, name, biome, area)
VALUES ('002', 'Lion Habitat', 'African Savanna', 27);

INSERT
INTO Habitats1(p_id, name, biome, area)
VALUES ('003', 'Penguin Habitat', 'Antarctic Tundra', 20);

INSERT
INTO Habitats1(p_id, name, biome, area)
VALUES ('004', 'Giraffe Habitat', 'African Grasslands', 40);

INSERT
INTO Habitats1(p_id, name, biome, area)
VALUES ('005', 'Grizzly Bear Habitat', 'North American Woodlands', 40 );

INSERT
INTO Shops(p_id, type)
VALUES ('101', 'Clothing');

INSERT
INTO Shops(p_id, type)
VALUES ('102', 'Drinks');

INSERT
INTO Shops(p_id, type)
```

```
VALUES ('103', 'Stuffed Animals');

INSERT
INTO Shops(p_id, type)
VALUES ('104', 'Balloons');

INSERT
INTO Shops(p_id, type)
VALUES ('105', 'Food');

INSERT
INTO Storage_Units(p_id, name, temperature)
VALUES ('201', 'Unit 1', 2);

INSERT
INTO Storage_Units(p_id, name, temperature)
VALUES ('202', 'Unit 2', -20);

INSERT
INTO Storage_Units(p_id, name, temperature)
VALUES ('203', 'Unit 3', 15);

INSERT
INTO Storage_Units(p_id, name, temperature)
VALUES ('204', 'Unit 4', 15);

INSERT
INTO Storage_Units(p_id, name, temperature)
VALUES ('205', 'Unit 5', 5);

INSERT
INTO Items(i_id, p_id, name, stock, price)
VALUES ('0001', '101', 'T-Shirt', 50, 24.99);

INSERT
INTO Items(i_id, p_id, name, stock, price)
VALUES ('0002', '102', 'Soda Bottle', 20, 3.50);


INSERT
INTO Items(i_id, p_id, name, stock, price)
VALUES ('0003', '103', 'Penguin Stuffie', 10, 14.99);

INSERT
```

```
INTO Items(i_id, p_id, name, stock, price)
VALUES ('0004', '104', 'Bear Balloon', 30, 4.99);

INSERT
INTO Items(i_id, p_id, name, stock, price)
VALUES ('0005', '105', 'Hamburger', 40, 6.99);

INSERT
INTO Animals2(species, genus)
VALUES ('Tiger', 'panthera');

INSERT
INTO Animals2(species, genus)
VALUES ('Lion', 'panthera');

INSERT
INTO Animals2(species, genus)
VALUES ('Emperor Penguin', 'Pinguinus');

INSERT
INTO Animals2(species, genus)
VALUES ('Northern Giraffe', 'Giraffa');

INSERT
INTO Animals2(species, genus)
VALUES ('Grizzly Bear', 'Ursus');

INSERT
INTO Animals1(a_id, p_id, name, species)
VALUES ('1001', '001', 'Stripe', 'Tiger');

INSERT
INTO Animals1(a_id, p_id, name, species)
VALUES ('1002', '002', 'Fluffy', 'Lion');

INSERT
INTO Animals1(a_id, p_id, name, species)
VALUES ('1003', '003', 'Slippy', 'Emperor Penguin');

INSERT
INTO Animals1(a_id, p_id, name, species)
VALUES ('1004', '004', 'Spots', 'Northern Giraffe');

INSERT
```

```
INTO Animals1(a_id, p_id, name, species)
VALUES ('1005', '003', 'Cuddles', 'Grizzly Bear');

INSERT
INTO Prepped_Food(a_id, name, weight)
VALUES ('1001', 'Deer meat', 20.1);

INSERT
INTO Prepped_Food(a_id, name, weight)
VALUES ('1002', 'Raw beef', 10.0);

INSERT
INTO Prepped_Food(a_id, name, weight)
VALUES ('1003', 'Mashed sardines', 15.7);

INSERT
INTO Prepped_Food(a_id, name, weight)
VALUES ('1004', 'Prepared hay', 25.0);

INSERT
INTO Prepped_Food(a_id, name, weight)
VALUES ('1005', 'Cooked moose', 30.2);

INSERT
INTO Raw_Food_Orders(o_id, contents, weight, date_received, expiry_date)
VALUES('1', 'Deer and Moose meat', 50, TO_DATE('2023/01/27', 'yyyy/mm/dd'),
TO_DATE('2023/02/05',  'yyyy/mm/dd'));

INSERT
INTO Raw_Food_Orders(o_id, contents, weight, date_received, expiry_date)
VALUES('2', 'Sardines', 30, TO_DATE('2023/01/15,  'yyyy/mm/dd'), TO_DATE('2023/02/01',
'yyyy/mm/dd'));

INSERT
INTO Raw_Food_Orders(o_id, contents, weight, date_received, expiry_date)
VALUES ('3', 'Hay and fruits', 32, TO_DATE(2023/01/15,  'yyyy/mm/dd'), TO_DATE(2023/02/10,
'yyyy/mm/dd'));

INSERT
INTO Raw_Food_Orders(o_id, contents, weight, date_received, expiry_date)
VALUES ('4', 'Raw beef and chicken', 10, TO_DATE(2023/02/25,  'yyyy/mm/dd'),
TO_DATE(2023/03/09,  'yyyy/mm/dd'));

INSERT
```

```
INTO Raw_Food_Orders(o_id, contents, weight, date_received, expiry_date)
VALUES ('5', 'Pellet food for Giraffes', 100, TO_DATE(2023/01/08,  'yyyy/mm/dd'),
TO_DATE(2023/06/08,  'yyyy/mm/dd'));

INSERT
INTO Have(c_id, w_id)
VALUES ('1', '1');

INSERT
INTO Have(c_id, w_id)
VALUES ('2', '2');

INSERT
INTO Have(c_id, w_id)
VALUES ('3', '3');

INSERT
INTO Have(c_id, w_id)
VALUES ('4', '4');

INSERT
INTO Have(c_id, w_id)
VALUES ('5', '5');

INSERT
INTO Works_at(w_id, p_id)
VALUES ('1', '101');

INSERT
INTO Works_at(w_id, p_id)
VALUES ('2', '102');

INSERT
INTO Works_at(w_id, p_id)
VALUES ('3', '103');

INSERT
INTO Works_at(w_id, p_id)
VALUES ('4', '104');

INSERT
INTO Works_at(w_id, p_id)
VALUES ('5', '105');
```

```
INSERT
INTO Assigned_to(w_id, p_id)
VALUES ('1', '001');

INSERT
INTO Assigned_to(w_id, p_id)
VALUES ('2', '002');

INSERT
INTO Assigned_to(w_id, p_id)
VALUES ('3', '003');

INSERT
INTO Assigned_to(w_id, p_id)
VALUES ('4', '004');

INSERT
INTO Assigned_to(w_id, p_id)
VALUES ('5', '005');

INSERT
INTO Feeds(w_id, a_id)
VALUES ('1', '1001');

INSERT
INTO Feeds(w_id, a_id)
VALUES ('2', '1002');

INSERT
INTO Feeds(w_id, a_id)
VALUES ('3', '1003');

INSERT
INTO Feeds(w_id, a_id)
VALUES ('4', '1004');

INSERT
INTO Feeds(w_id, a_id)
VALUES ('5', '1005');

INSERT
INTO Maintains_Health_of(w_id, a_id)
VALUES ('1', '1001');
```

```
INSERT
INTO Maintains_Health_of(w_id, a_id)
VALUES ('2', '1002');

INSERT
INTO Maintains_Health_of(w_id, a_id)
VALUES ('3', '1003');

INSERT
INTO Maintains_Health_of(w_id, a_id)
VALUES ('4', '1004');

INSERT
INTO Maintains_Health_of(w_id, a_id)
VALUES ('5', '1005');

INSERT
INTO Cohabitates_with(a_id1, a_id2)
VALUES ('1001', '1002');

INSERT
INTO Cohabitates_with(a_id1, a_id2)
VALUES ('1001', '1003');

INSERT
INTO Cohabitates_with(a_id1, a_id2)
VALUES ('1001', '1004');

INSERT
INTO Cohabitates_with(a_id1, a_id2)
VALUES ('1001', '1005');

INSERT
INTO Cohabitates_with(a_id1, a_id2)
VALUES ('1003', '1004');

INSERT
INTO Made_from(a_id, name, o_id)
VALUES ('1001', 'Deer meat', '1');

INSERT
INTO Made_from(a_id, name, o_id)
VALUES ('1002', 'Raw beef', '4');
```

```
INSERT
INTO Made_from(a_id, name, o_id)
VALUES ('1003', 'Mashed sardines', '2');

INSERT
INTO Made_from(a_id, name, o_id)
VALUES ('1004', 'Prepared hay', '3');

INSERT
INTO Made_from(a_id, name, o_id)
VALUES ('1005', 'Cooked moose', '1');

INSERT
INTO Stored_at(a_id, name, p_id)
VALUES ('1001', 'Deer meat', '201');

INSERT
INTO Stored_at(a_id, name, p_id)
VALUES ('1002', 'Raw beef', '201');

INSERT
INTO Stored_at(a_id, name, p_id)
VALUES ('1003', 'Mashed Sardines', '205');

INSERT
INTO Stored_at(a_id, name, p_id)
VALUES ('1004', 'Prepared hay', '203');

INSERT
INTO Stored_at(a_id, name, p_id)
VALUES ('1005', 'Cooked moose', '205');

INSERT
INTO Located_at(o_id, p_id)
VALUES ('1', '201');

INSERT
INTO Located_at(o_id, p_id)
VALUES ('2', '205');

INSERT
INTO Located_at(o_id, p_id)
VALUES ('3', '203');
```

```
INSERT
INTO Located_at(o_id, p_id)
VALUES ('4', '201');

INSERT
INTO Located_at(o_id, p_id)
VALUES ('5', '204');
```

Copy of DDL statements to work in Oracle (as opposed to independent attributes first in order to work in oracle, NOT FOR GRADING):

```
CREATE TABLE Computers2(
        model VARCHAR(20) PRIMARY KEY,
        manufacturer VARCHAR(20) NOT NULL,
        type VARCHAR(20) NOT NULL
        );

CREATE TABLE Computers1(
        c_id VARCHAR(20) PRIMARY KEY,
        model VARCHAR(20) NOT NULL,
        FOREIGN KEY (model) REFERENCES Computers2(model)
);

CREATE TABLE Workers(
        w_id VARCHAR(20) PRIMARY KEY,
        name VARCHAR(20) NOT NULL,
        pay_rate FLOAT NOT NULL,
        address VARCHAR(30) NOT NULL,
        email VARCHAR(30) UNIQUE,
        phone VARCHAR(13) UNIQUE
);

CREATE TABLE Zookeepers(
        w_id VARCHAR(20) PRIMARY KEY,
        FOREIGN KEY (w_id) REFERENCES Workers(w_id)
);

CREATE TABLE Vendors(
        w_id VARCHAR(20) PRIMARY KEY,
        FOREIGN KEY (w_id) REFERENCES Workers(w_id)
);

CREATE TABLE Veterinarians(
        w_id VARCHAR(20) PRIMARY KEY,
        specialization VARCHAR(20),
        FOREIGN KEY (w_id) REFERENCES Workers(w_id)
);


CREATE TABLE Habitats2(
        biome VARCHAR(50) PRIMARY KEY,
        temperature INTEGER,
        humidity INTEGER
```

```
);

CREATE TABLE Habitats1(
        p_id VARCHAR(20) PRIMARY KEY,
        name VARCHAR(20),
        biome VARCHAR(50),
        area INTEGER,
        FOREIGN KEY (biome) REFERENCES Habitats2
);

CREATE TABLE Shops(
        p_id VARCHAR(20) PRIMARY KEY,
        type VARCHAR(20) NOT NULL
);

CREATE TABLE Storage_Units(
        p_id VARCHAR(20) PRIMARY KEY,
        name VARCHAR(20),
        temperature INTEGER NOT NULL
);

CREATE TABLE Items(
        i_id VARCHAR(20) PRIMARY KEY,
        p_id VARCHAR(20) NOT NULL,
        name VARCHAR(30) NOT NULL,
        stock INTEGER NOT NULL,
        price FLOAT NOT NULL,
        FOREIGN KEY (p_id) REFERENCES Shops(p_id)
);

CREATE TABLE Animals2(
        species VARCHAR(20) PRIMARY KEY,
        genus VARCHAR(20)
        );

CREATE TABLE Animals1(
        a_id VARCHAR(20) PRIMARY KEY,
        p_id VARCHAR(20),
        name VARCHAR(30),
        species VARCHAR(20),
        FOREIGN KEY (species) REFERENCES Animals2,
        FOREIGN KEY (p_id) REFERENCES Habitats1(p_id)
);
```

```
CREATE TABLE Prepped_Food(
        a_id VARCHAR(20),
        name VARCHAR(20),
        weight FLOAT,
        PRIMARY KEY (a_id, name),
        FOREIGN KEY (a_id) REFERENCES Animals1(a_id) ON DELETE CASCADE
);

CREATE TABLE Raw_Food_Orders(
        o_id VARCHAR(20) PRIMARY KEY,
        contents VARCHAR(50) NOT NULL,
        weight INTEGER NOT NULL,
        date_received DATE NOT NULL,
        expiry_date DATE NOT NULL
 );

 CREATE TABLE Have(
        c_id VARCHAR(20) PRIMARY KEY,
        w_id VARCHAR(20) UNIQUE NOT NULL ,
        FOREIGN KEY (c_id) REFERENCES Computers1(c_id),
        FOREIGN KEY (w_id) REFERENCES Workers(w_id)
);

CREATE TABLE Works_at(
        w_id VARCHAR(20),
        p_id VARCHAR(20),
        PRIMARY KEY (w_id, p_id),
        FOREIGN KEY (w_id) REFERENCES Workers,
        FOREIGN KEY (p_id) REFERENCES Shops
);

CREATE TABLE Assigned_to(
        w_id VARCHAR(20),
        p_id VARCHAR(20),
        PRIMARY KEY(w_id, p_id),
        FOREIGN KEY (w_id) REFERENCES Zookeepers(w_id),
        FOREIGN KEY (p_id) REFERENCES Habitats1(p_id)
);

CREATE TABLE Feeds(
        w_id VARCHAR(20),
        a_id VARCHAR(20),
        PRIMARY KEY (w_id, a_id),
        FOREIGN KEY (w_id) REFERENCES  Zookeepers(w_id),
```

```
        FOREIGN KEY (a_id) REFERENCES Animals1(a_id)
);

CREATE TABLE Maintains_Health_of(
        w_id VARCHAR(20),
        a_id VARCHAR(20),
        PRIMARY KEY (w_id, a_id),
        FOREIGN KEY (w_id) REFERENCES Veterinarians(w_id),
        FOREIGN KEY (a_id) REFERENCES Animals1(a_id)
);

CREATE TABLE Cohabitates_with(
        a_id1 VARCHAR(20),
        a_id2 VARCHAR(20),
        PRIMARY KEY(a_id1, a_id2),
        FOREIGN KEY (a_id1) REFERENCES Animals1(a_id),
        FOREIGN KEY (a_id2) REFERENCES Animals1(a_id)
);

CREATE TABLE Made_from(
        a_id VARCHAR(20),
        name VARCHAR(30),
        o_id VARCHAR(20),
        PRIMARY KEY(a_id, name, o_id),
        FOREIGN KEY (a_id, name) REFERENCES Prepped_Food(a_id, name),
        FOREIGN KEY (o_id) REFERENCES Raw_Food_Orders(o_id)
);

CREATE TABLE Stored_at(
        a_id VARCHAR(20),
        name VARCHAR(30),
        p_id VARCHAR(20),
        PRIMARY KEY(a_id, name, p_id),
        FOREIGN KEY (a_id, name) REFERENCES Prepped_Food(a_id, name),
        FOREIGN KEY (p_id) REFERENCES Storage_Units(p_id)
);

CREATE TABLE Located_at(
        o_id VARCHAR(20),
        p_id VARCHAR(20),
        PRIMARY KEY(o_id, p_id),
        FOREIGN KEY (o_id) REFERENCES Raw_Food_Orders(o_id),
        FOREIGN KEY (p_id) REFERENCES Storage_Units(p_id)
);
```

workers id: #
computer id: #
habitat id: 0##
shop id: 1##
storage unit: 2##
item id: 0###
animal id: 1###
order id: #