CECS 323

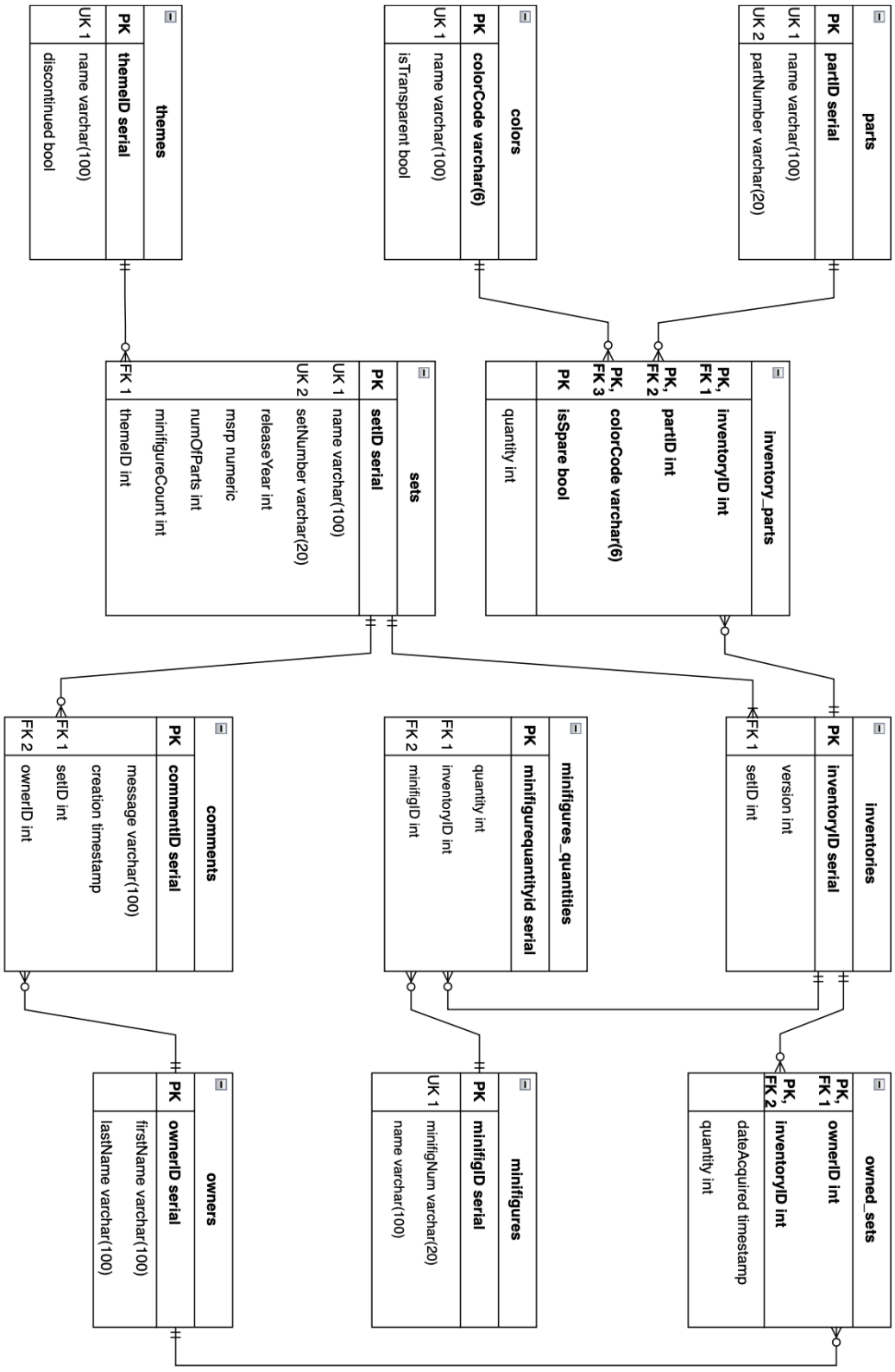
Section 6 / 7909

- Project 2: Plain AFOL -

Tasos Lilis
Christ Nguyen

April 18th 2024

- ER Diagram -


parts
PK partID serial
UK 1 name varchar(100)
UK 2 partNumber varchar(20)
colors
PK colorCode varchar(6)
UK 1 name varchar(100)
isTransparent bool
themes
PK themeID serial
UK 1 name varchar(100)
discontinued bool
inventory_parts
PK, FK 1 inventoryID int
PK, FK 2 partID int
PK, FK 3 colorCode varchar(6)
PK isSpare bool
quantity int
sets
PK setID serial
UK 1 name varchar(100)
UK 2 setNumber varchar(20)
releaseYear int
msrp numeric
numOfParts int
minifigureCount int
FK 1 themeID int
inventories
PK inventoryID serial
version int
FK 1 setID int
minifigures_quantities
PK minifigurequantityid serial
quantity int
FK 1 inventoryID int
FK 2 minifigID int
minifigures
PK minifigID serial
UK 1 minifigNum varchar(20)
name varchar(100)
owned_sets
PK, FK 1 ownerID int
PK, FK 2 inventoryID int
dateAcquired timestamp
quantity int
comments
PK commentID serial
message varchar(100)
creation timestamp
FK 1 setID int
FK 2 ownerID int
owners
PK ownerID serial
firstName varchar(100)
lastName varchar(100)

- Insert -

colors:

```
insert into proj2.colors (colorcode, name, istransparent)
values  ('05131D', 'Black', false),
        ('FFFFFF', 'White', false),
        ('E4CD9E', 'Tan', false),
        ('A5A9B4', 'Metallic', false);
```

comments:

```
insert into proj2.comments (commentid, message, creation, setid, ownerid)
values  (1, 'Great set!', '2024-04-18 18:00:00.000000', 2, 1),
        (2, 'Cool figures!', '2024-04-18 18:30:00.000000', 2, 1);
```

inventories:

```
insert into proj2.inventories (inventoryid, version, setid)
values  (1, 1, 1),
        (2, 2, 1),
        (3, 1, 2);
```

inventory_parts:

```
insert into proj2.inventory_parts (inventoryid, partid, colorcode, quantity, isspare)
values  (1, 1, '05131D', 2, false),
        (1, 1, 'FFFFFF', 2, false),
        (1, 2, 'A5A9B4', 1, false),
        (2, 1, '05131D', 2, false),
        (2, 1, 'FFFFFF', 2, false),
        (2, 2, 'FFFFFF', 1, false),
        (3, 1, 'FFFFFF', 1, false),
        (3, 2, 'FFFFFF', 1, false),
        (3, 3, '05131D', 1, false);
```

minifigure_quantities:

```
insert into proj2.minifigure_quantities (minifigurequantityid, quantity, inventoryid,
minifigureid)
values  (1, 1, 1, 1),
        (2, 2, 1, 2),
        (3, 1, 2, 1),
        (4, 2, 2, 2),
        (5, 1, 3, 3),
        (6, 1, 3, 2);
```

minifigures:

```
insert into proj2.minifigures (minifigureid, minifigurenumber, name)
values  (1, 'fig-004422', 'Princess Leia, White Robe, Skirk'),
        (2, 'fig-003503', 'R2-D2, Flat Silver Dome, Dark Blue Print with Dark Pink
Dots'),
        (3, 'fig-013246', 'Nova - Black Top, Sand Blue Pants'),
        (4, 'sw0465a', 'Yoda - NY I Heart Torso, White Hair (TRU Times Square 2013
Exclusive)');
```

owned_sets:

```
insert into proj2.owned_sets (ownerid, inventoryid, dateacquired, quantity)
values  (1, 3, '2024-04-18 20:00:00.000000', 1),
        (1, 2, '2024-04-18 20:00:00.000000', 2);
```

owners:

```
insert into proj2.owners (ownerid, firstname, lastname)
values  (1, 'Neal', 'Terrell');
```

parts:

```
insert into proj2.parts (partid, name, partnumber)
values  (1, 'Brick 1 x 4 ', '3010'),
        (2, 'Weapon Lightsaber Hilt with Bottom Ring', '64567'),
        (3, 'Plate 4 x 6', '3032');
```

sets:

```
insert into proj2.sets (setid, name, setnumber, releaseyear, msrp, numofparts,
themeid, minifigurecount)
values  (1, 'Luke Skywalker''s X-wing Fighter', '75301', 2021, 49.99, 474, 1, 4),
        (2, 'Nova''s Room', '41755', 2023, 19.99, 179, 2, 2);
```

themes:

```
insert into proj2.themes (themeid, name, discontinued)
values  (1, 'Star Wars', false),
        (2, 'Friends', false);
```

- Select -

1. Select the names of all Minifigures that are included in at least one Inventory. [Be wary of duplicates!]

```
select distinct name
from minifigures
inner join minifigure_quantities on
    minifigures.minifigureid = minifigure_quantities.minifigureid;
```

2. Select the names of all Minifigures that are never included in an Inventory.

```
select name
from minifigures
left outer join minifigure_quantities on
    minifigures.minifigureid = minifigure_quantities.minifigureid
where minifigure_quantities.minifigureid is null;
```

3. Determine the total number of parts in the owned collection of the user named Neal.

```
select sum(inventory_parts.quantity * owned_sets.quantity) as totalParts
from inventory_parts
inner join inventories on inventory_parts.inventoryid = inventories.inventoryid
inner join owned_sets on inventories.inventoryid = owned_sets.inventoryid
inner join owners on owned_sets.ownerid = owners.ownerid
where owners.firstname = 'Neal';
```

4. Select the name of the User whose collection of sets is worth the most, i.e., with the maximum total MSRP of all sets owned by that user.

```
select firstname, lastname
from owners
inner join owned_sets on owners.ownerid = owned_sets.ownerid
inner join inventories on owned_sets.inventoryid = inventories.inventoryid
inner join sets on inventories.setid = sets.setid
group by firstname, lastname
order by sum (msrp * quantity) desc
fetch first 1 row only;
```

5. In a single query, count the number of "play" and "investment" sets. An "investment" set has an MSRP greater than $50, and is either from the Lego Star Wars theme, or was released at least 5 years ago (from today); otherwise it is a "play" set. Your query should produce two rows of output, with status and count columns.

```
select
   case when
      (
      msrp > 50 and
      (themeid = 1 or releaseyear <= extract(year from current_date) - 5)
      )
      then 'Investment'
      else 'Play'
   end as status,
   count(*) as count
from sets
group by status;
```