Team GLASTA's *Fantastic Furniture*: Normalization

Team Info

Team Name:	Team GLASTA	
Project Name:	*Fantastic Furniture*	
Participants:	Timothy Gibson	tgibson1@csustan.edu
	Alexander Altman	aaltman@csustan.edu
	Schuyler Davis	sdavis20@csustan.edu

Initial Relations

```
1  create domain posreal as double precision
2  check         (value > 0.0);
3
4  create domain posint as integer
5  check         (value > 0);
6
7  -- all measures in this type are in inches
8  create type dimensions as (length_ posreal,
9                             width   posreal,
10                            height  posreal);
11
12 create table Supplier(supplierID  varchar(10),
13                       name_       nchar varying(50)
14                                   not null,
15                       phone       varchar(12),
16                       address     nchar varying(100),
17                       country     char(2),
18                       website     nchar varying(50),
19                       primary key (supplierID));
20
21 create table Designer(designerID  varchar(10),
22                       name_       nchar varying(50)
23                                   not null,
24                       phone       varchar(12),
25                       address     nchar varying(100),
26                       country     char(2),
27                       website     nchar varying(50),
28                       designFocus nchar varying(100),
29                       primary key (designerID));
30
31 create table Set_(setID         varchar(10),
```

```
                    name_          nchar varying(50)
                                   not null,
                    catalogYear    numeric(4,0),
                    catalogNumber integer
                                   not null,
                    style_         nchar varying(30),
                    primary key    (setID));

create table Model(modelNumber varchar(10),
                     name_       nchar varying(50)
                                 not null,
                     material    nchar varying(30),
                     upholstery  nchar varying(30),
                     durability  nchar varying(30),
                     color       nchar varying(30),
                     primary key (modelNumber));

create table Item(sku          varchar(10),
                    dimensions  dimensions
                                not null, -- anti-redundancy constraint
                    condition   nchar varying(30),
                    weightLimit posreal, -- in pounds of weight
                    primary key (sku));

create table DistributionCenter(centerID    varchar(10),
                                  name_       nchar varying(50)
                                              not null,
                                  phone       varchar(12),
                                  address     nchar varying(100),
                                  country     char(2),
                                  website     nchar varying(50),
                                  primary key (centerID));

create table make(supplierID  varchar(10),
                    designerID  varchar(10),
                    setID       varchar(10),
                    primary key (supplierID,
                                 designerID,
                                 setID),
                    foreign key (supplierID)
                                references Supplier,
                    foreign key (designerID)
                                references Designer,
                    foreign key (setID)
                                references Set_);
```

```
77
78  create table contains_(setID       varchar(10),
79                         modelNumber varchar(10),
80                         count_      posint,
81                         primary key (setID,
82                                      modelNumber),
83                         foreign key (setID)
84                                     references Set_,
85                         foreign key (modelNumber)
86                                     references Model);
87
88  create table describes(modelNumber varchar(10)
89                                     not null,
90                         sku         varchar(10),
91                         primary key (sku),
92                         foreign key (modelNumber)
93                                     references Model,
94                         foreign key (sku)
95                                     references Item);
96
97  create table canOrderFrom(centerID    varchar(10),
98                            supplierID  varchar(10),
99                            leadTime    double precision, -- in days
100                           primary key (centerID,
101                                        supplierID),
102                           foreign key (centerID)
103                                       references DistributionCenter,
104                           foreign key (supplierID)
105                                       references Supplier,
106                           check       (leadTime >= 0.0));
107
108 create table stocks(centerID    varchar(10)
109                                 not null,
110                     sku         varchar(10),
111                     primary key (sku),
112                     foreign key (centerID)
113                                 references DistributionCenter,
114                     foreign key (sku)
115                                 references Item);
116
117 create table Chair(sku          varchar(10),
118                    numberOfLegs posint,
119                    hasCushion   boolean,
120                    hasArms      boolean,
121                    backHeight   posreal, -- in inches
```

```
                    seatHeight   posreal, -- in inches
                    primary key  (sku),
                    foreign key  (sku)
                                 references Item);

create table Table_(sku           varchar(10),
                     numberOfLegs  posint,
                     numberOfSeats posint,
                     shape         nchar varying(30),
                     primary key   (sku),
                     foreign key   (sku)
                                   references Item);

create table Desk(sku             varchar(10),
                  angle           double precision, -- in degrees, possibly negative
                  numberOfDrawers posint,
                  primary key     (sku),
                  foreign key     (sku)
                                  references Item,
                  check              (angle > -360.0
                                  and angle < 360.0));

create table Stool(sku          varchar(10),
                    numberOfLegs posint,
                    hasCushion   boolean,
                    hasSwivel    boolean,
                    primary key  (sku),
                    foreign key  (sku)
                                 references Item);

create table Cabinet(sku                 varchar(10),
                      numberOfCompartments posint,
                      capacity             nchar varying(30),
                      primary key          (sku),
                      foreign key          (sku)
                                           references Item);

create table Bedframe(sku         varchar(10),
                       size_      nchar varying(30),
                       depth_     double precision, -- in inches, possibly negative
                       primary key (sku),
                       foreign key (sku)
                                   references Item);

create table features_Feature(modelNumber varchar(10),
```

```
                        description nchar varying(50),
                        count_      posint,
                        primary key (modelNumber,
                                     description),
                        foreign key (modelNumber)
                                    references Model);
```

Functional Dependencies

$$
\begin{array}{rcll}
\texttt{supplierID} & \longrightarrow & \texttt{name\_} & \left.\vphantom{x}\right\} \\
\texttt{supplierID} & \longrightarrow & \texttt{phone} & \\
\texttt{supplierID} & \longrightarrow & \texttt{address} & \left.\vphantom{x}\right\}\texttt{Supplier} \\
\texttt{supplierID} & \longrightarrow & \texttt{country} & \\
\texttt{supplierID} & \longrightarrow & \texttt{website} & \left.\vphantom{x}\right\} \\
\texttt{designerID} & \longrightarrow & \texttt{name\_} & \left.\vphantom{x}\right\} \\
\texttt{designerID} & \longrightarrow & \texttt{phone} & \\
\texttt{designerID} & \longrightarrow & \texttt{address} & \left.\vphantom{x}\right\}\texttt{Designer} \\
\texttt{designerID} & \longrightarrow & \texttt{country} & \\
\texttt{designerID} & \longrightarrow & \texttt{website} & \\
\texttt{designerID} & \longrightarrow & \texttt{designFocus} & \left.\vphantom{x}\right\} \\
\texttt{setID} & \longrightarrow & \texttt{name\_} & \left.\vphantom{x}\right\} \\
\texttt{setID} & \longrightarrow & \texttt{catalogYear} & \\
\texttt{setID} & \longrightarrow & \texttt{catalogNumber} & \left.\vphantom{x}\right\}\texttt{Set\_} \\
\texttt{setID} & \longrightarrow & \texttt{style\_} & \left.\vphantom{x}\right\} \\
\texttt{modelNumber} & \longrightarrow & \texttt{name\_} & \left.\vphantom{x}\right\} \\
\texttt{modelNumber} & \longrightarrow & \texttt{material} & \\
\texttt{modelNumber} & \longrightarrow & \texttt{upholstery} & \left.\vphantom{x}\right\}\texttt{Model} \\
\texttt{modelNumber} & \longrightarrow & \texttt{durability} & \\
\texttt{modelNumber} & \longrightarrow & \texttt{color} & \left.\vphantom{x}\right\} \\
\texttt{sku} & \longrightarrow & \texttt{name\_} & \left.\vphantom{x}\right\} \\
\texttt{sku} & \longrightarrow & \texttt{dimensions.length\_} & \\
\texttt{sku} & \longrightarrow & \texttt{dimensions.width} & \left.\vphantom{x}\right\}\texttt{Item} \\
\texttt{sku} & \longrightarrow & \texttt{dimensions.height} & \\
\texttt{sku} & \longrightarrow & \texttt{condition} & \\
\texttt{sku} & \longrightarrow & \texttt{weightLimit} & \left.\vphantom{x}\right\} \\
\texttt{centerID} & \longrightarrow & \texttt{name\_} & \left.\vphantom{x}\right\} \\
\texttt{centerID} & \longrightarrow & \texttt{phone} & \\
\texttt{centerID} & \longrightarrow & \texttt{address} & \left.\vphantom{x}\right\}\texttt{DistributionCenter} \\
\texttt{centerID} & \longrightarrow & \texttt{country} & \\
\texttt{centerID} & \longrightarrow & \texttt{website} & \left.\vphantom{x}\right\}
\end{array}
$$

Note: the `make` *relation has no nontrivial functional dependencies.*

$$
\begin{array}{rcll}
\texttt{\{setID, modelNumber\}} & \longrightarrow & \texttt{count\_} & \left.\vphantom{x}\right\}\texttt{contains\_} \\
\texttt{sku} & \longrightarrow & \texttt{modelNumber} & \left.\vphantom{x}\right\}\texttt{describes} \\
\texttt{\{centerID, supplierID\}} & \longrightarrow & \texttt{leadTime} & \left.\vphantom{x}\right\}\texttt{canOrderFrom}
\end{array}
$$

sku	⟶	centerID	} stocks
sku	⟶	numberOfLegs	} Chair
sku	⟶	hasCushion	
sku	⟶	hasArms	
sku	⟶	backHeight	
sku	⟶	seatHeight	
sku	⟶	numberOfLegs	} Table_
sku	⟶	numberOfSeats	
sku	⟶	shape	
sku	⟶	angle	} Desk
sku	⟶	numberOfDrawers	
sku	⟶	numberOfLegs	} Stool
sku	⟶	hasCushion	
sku	⟶	hasSwivel	
sku	⟶	numberOfCompartments	} Cabinet
sku	⟶	capacity	
sku	⟶	size_	} Bedframe
sku	⟶	depth_	
{modelNumber, description}	⟶	count_	} features_Feature

Functional Dependency Notes

One might think, at first, that several attributes of `Supplier` (such as `website` and `phone`) should be candidate keys by virtue of uniquely determining the primary key `supplierID`. However, consider the following scenario: a supplier is an independent carpenter living in a country whose laws (for whatever reason) disallow any one single business from selling both chairs and bedframes. This carpenter produces both types of items, but, because of the laws of his home country, he has to run two separate businesses from the legal perspective. The inevitable result of us getting furniture from this carpenter is two `Supplier` tuples with (necessarily) distinct `supplierID`s but where *every other attribute is identical*! Therefore, by constructed counterexample, `Supplier`'s only candidate key is its primary key `supplierID`; very similar reasoning applies to `DistributionCenter` and `Designer` as well.

Normalized Relations

```
1  create domain posreal as double precision
2  check         (value > 0.0);
3
4  create domain posint as integer
5  check         (value > 0);
6
7  create table Supplier(supplierID  varchar(10),
8                        name_       nchar varying(50)
9                                    not null,
10                       phone       varchar(12),
```

```
                        address     nchar varying(100),
                        country     char(2),
                        website     nchar varying(50),
                        primary key (supplierID));

create table Designer(designerID  varchar(10),
                        name_       nchar varying(50)
                                    not null,
                        phone       varchar(12),
                        address     nchar varying(100),
                        country     char(2),
                        website     nchar varying(50),
                        designFocus nchar varying(100),
                        primary key (designerID));

create table Set_(setID         varchar(10),
                    name_         nchar varying(50)
                                  not null,
                    catalogYear   numeric(4,0),
                    catalogNumber integer
                                  not null,
                    style_        nchar varying(30),
                    primary key   (setID));

create table Model(modelNumber varchar(10),
                     name_       nchar varying(50)
                                 not null,
                     material    nchar varying(30),
                     upholstery  nchar varying(30),
                     durability  nchar varying(30),
                     color       nchar varying(30),
                     primary key (modelNumber));

create table Item(sku         varchar(10),
                    length_     posreal, -- in inches
                    width       posreal, -- in inches
                    height      posreal, -- in inches
                    condition   nchar varying(30),
                    weightLimit posreal, -- in pounds of weight
                    primary key (sku));

create table DistributionCenter(centerID    varchar(10),
                                  name_       nchar varying(50)
                                              not null,
                                  phone       varchar(12),
```

```
                                address     nchar varying(100),
                                country     char(2),
                                website     nchar varying(50),
                                primary key (centerID));

create table make(supplierID  varchar(10),
                  designerID  varchar(10),
                  setID       varchar(10),
                  primary key (supplierID,
                               designerID,
                               setID),
                  foreign key (supplierID)
                               references Supplier,
                  foreign key (designerID)
                               references Designer,
                  foreign key (setID)
                               references Set_);

create table contains_(setID       varchar(10),
                       modelNumber varchar(10),
                       count_      posint,
                       primary key (setID,
                                    modelNumber),
                       foreign key (setID)
                                    references Set_,
                       foreign key (modelNumber)
                                    references Model);

create table describes(modelNumber varchar(10)
                                    not null,
                       sku         varchar(10),
                       primary key (sku),
                       foreign key (modelNumber)
                                    references Model,
                       foreign key (sku)
                                    references Item);

create table canOrderFrom(centerID    varchar(10),
                          supplierID  varchar(10),
                          leadTime    double precision, -- in days
                          primary key (centerID,
                                       supplierID),
                          foreign key (centerID)
                                       references DistributionCenter,
                          foreign key (supplierID)
```

```
                                    references Supplier,
                          check     (leadTime >= 0.0));

create table stocks(centerID    varchar(10)
                                not null,
                    sku         varchar(10),
                    primary key (sku),
                    foreign key (centerID)
                                references DistributionCenter,
                    foreign key (sku)
                                references Item);

create table Chair(sku          varchar(10),
                   numberOfLegs posint,
                   hasCushion   boolean,
                   hasArms      boolean,
                   backHeight   posreal, -- in inches
                   seatHeight   posreal, -- in inches
                   primary key  (sku),
                   foreign key  (sku)
                                references Item);

create table Table_(sku           varchar(10),
                    numberOfLegs  posint,
                    numberOfSeats posint,
                    shape         nchar varying(30),
                    primary key   (sku),
                    foreign key   (sku)
                                  references Item);

create table Desk(sku             varchar(10),
                  angle           double precision, -- in degrees, possibly negative
                  numberOfDrawers posint,
                  primary key     (sku),
                  foreign key     (sku)
                                  references Item,
                  check              (angle > -360.0
                                  and angle < 360.0));

create table Stool(sku          varchar(10),
                   numberOfLegs posint,
                   hasCushion   boolean,
                   hasSwivel    boolean,
                   primary key  (sku),
                   foreign key  (sku)
```

```
                                references Item);

create table Cabinet(sku                 varchar(10),
                     numberOfCompartments posint,
                     capacity            nchar varying(30),
                     primary key         (sku),
                     foreign key         (sku)
                                         references Item);

create table Bedframe(sku         varchar(10),
                      size_       nchar varying(30),
                      depth_      double precision, -- in inches, possibly negative
                      primary key (sku),
                      foreign key (sku)
                                  references Item);

create table features_Feature(modelNumber varchar(10),
                              description nchar varying(50),
                              count_      posint,
                              primary key (modelNumber,
                                           description),
                              foreign key (modelNumber)
                                          references Model);
```

Normal Forms

`Supplier` `Supplier` is in BCNF. Each functional dependency in the relation points back to `supplierID`. Every other attribute of the relation is dependent on `supplierID` and as such is a member of the candidate key. Also, every attribute of the `Supplier` relation is functionally determined by `supplierID`, making it the superkey.

`Designer` `Designer` is in BCNF. `designerID` determines every attribute of the `Designer` relation. This effectively makes `designerID` the superkey for the `Designer` relation.

`Set_` The `Set_` relation is in BCNF. `Set_` has a candidate key that is `setID`. Each attribute of `Set_` is unique to a specific `setID`. This indicates that each attribute is determined by that setID. This makes setID the superkey for `Set_`. From this, it lends that `Set_` is in BCNF because that for every non-trivial functional dependency, `setID` is the superkey.

`Model` `Model` is in BCNF. For every attribute of the `Model` relation, they are functionally determined by the `modelNumber`, meaning that they are only determined by one specific `modelNumber`. This makes `modelNumber` the superkey for `Model`.

`Item` The `Item` relation is in BCNF. The attribute `sku` is a single identifier for each individual item. Each item has one `name`, `length_`, `width`, `height`, and `weightLimit`. Each of these attributes are dependent on one `sku`. This makes `sku` the superkey for `Item`. Since all attributes are dependent on a single `sku` then `Item` is in BCNF.

In our original SQL, the `Item` relation used a type (`dimensions`) that was created specifically for that relation in its table. That type's fields have now been defined inline inside of the table so that it complies with the requirements of BCNF.

`DistributionCenter` `DistributionCenter` is in BCNF. Each `name_`, `phone`, `address`, `country`, and `website` is specific to one `centerID`. This makes `centerID` the superkey for the `DistributionCenter` relation. Since each attribute is only populated by one tuple and the superkey determines every attribute, then the `DistributionCenter` relation is in BCNF.

`make` The `make` relation is in BCNF, since it only contains trivial functional dependencies.

`contains_` The `contains_` relation is in BCNF. The `contains_` relation has a primary key of {`setID`, `modelNumber`}. These two together effectively become the superkey and since the count of the `contains_` relation can be determined by the `setID` and `modelNumber`, `contains_` is in BCNF.

`describes` The `describes` relation is in BCNF. The `describes` relation includes two foreign keys that together also form the primary key. These two keys are `modelNumber` and `sku`. The `modelNumber` specifically determines a single `sku`. This means that `sku` is dependent on the `modelNumber` and that `modelNumber` is the superkey. Since the only non-trivial functional dependency in the `describes` relation involves the superkey determining the single other attribute, `describes` is in BCNF.

`canOrderFrom` The `canOrderFrom` relation is in BCNF. The `canOrderFrom` relation has an attribute called `leadTime`. This `leadTime` is dependent on the distribution center (identified by `centerID`) and the supplier (identified by `supplierID`). This `centerID` and `supplierID` are both part of the primary key for `canOrderFrom` and together make up its superkey. For this reason, `canOrderFrom` is in BCNF.

`stocks` The `stocks` relation is in BCNF. The `stocks` relation has a primary key of `sku`. Since `centerID` is dependent on `sku` and the `stocks` relation borrows both `sku` and `centerID` from other relations, `sku` is the superkey. For this reason, `stocks` is in BCNF.

`Chair` The `Chair` relation is in BCNF. The `Chair` relation is a subset of the `Item` relation. Each chair has one of each of its attributes that is strictly related to its `sku`. This makes the `sku` the superkey for the `Chair` relation.

As such, since each functional dependency is dependent on sku, Chair is in BCNF.

Table_ The Table_ relation is in BCNF. The Table_ relation is a subset of the Item relation. Each table has one of each of its attributes that is strictly related to its sku. This makes the sku the superkey for the Table_ relation. As such, since each functional dependency is dependent on sku, Table_ is in BCNF.

Desk The Desk relation is in BCNF. The Desk relation is a subset of the Item relation. Each desk has one of each of its attributes that is strictly related to its sku. This makes the sku the superkey for the Desk relation. As such, since each functional dependency is dependent on sku, Desk is in BCNF.

Stool The Stool relation is in BCNF. The Stool relation is a subset of the Item relation. Each stool has one of each of its attributes that is strictly related to its sku. This makes the sku the superkey for the Stool relation. As such, since each functional dependency is dependent on sku, Stool is in BCNF.

Cabinet The Cabinet relation is in BCNF. The Cabinet relation is a subset of the Item relation. Each cabinet has one of each of its attributes that is strictly related to its sku. This makes the sku the superkey for the Cabinet relation. As such, since each functional dependency is dependent on sku, Cabinet is in BCNF.

Bedframe The Bedframe relation is in BCNF. The Bedframe relation is a subset of the Item relation. Each bedframe has one of each of its attributes that is strictly related to its sku. This makes the sku the superkey for the Bedframe relation. As such, since each functional dependency is dependent on sku, Bedframe is in BCNF.

features_Feature This relation is in BCNF. The features_Feature relation has a key that it contains called description. This description, however long it may be, will be distinct meaning that each feature has a specific description fitting to that specific feature. This makes {modelNumber, description} the superkey for features_Feature. Also, since count_ is dependent specifically on the description of that feature for that model, it is dependent on the superkey; this means that features_Feature is in BCNF.

Group Work

Alexander: Modified the SQL to 1NF; looked over the conversions to BCNF.

Timothy: Modified the tables to 3NF or BCNF; created the descriptions of the tables.

Schuyler: Played a little catch-up and wrote up these descriptions.