RELATIONAL MODEL

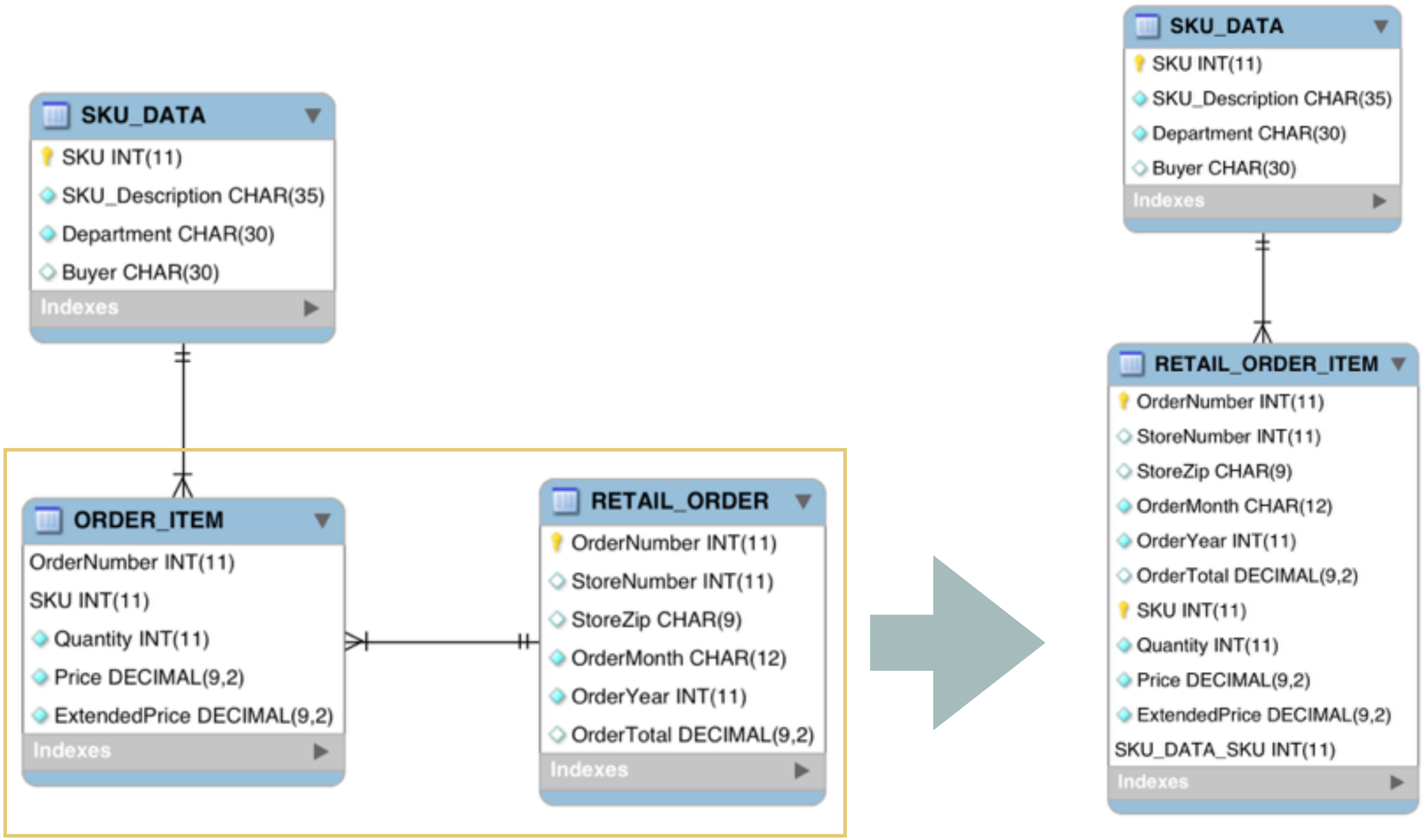
revised by 김태연

OBJECTIVES

- To understand basic relational terminology
- To understand the characteristics of relations
- To be able to identify functional dependencies, determinants, and dependent attributes
- To identify primary, candidate, and composite keys
- To be able to identify possible insertion, deletion, and update anomalies in a relation

HOW MANY TABLES?

- Should we store these two tables as they are, or should we combine them into one table in our new database?



THE RELATIONAL MODEL

- Introduced in 1970
- Created by E.F. Codd
- He was an IBM engineer
- The model used mathematics known as "relational algebra"
- Now the standard model for commercial DBMS products.

Codd. E.F., "A Relational Model of Data for Large Shared Data Banks", CACM, June 1970

RELATION

- In relational database theory, a relation, as originally defined by E. F. Codd, is a set of tuples (d_1, d_2, ..., d_n), where each element d_j is a member of D_j, a data domain.

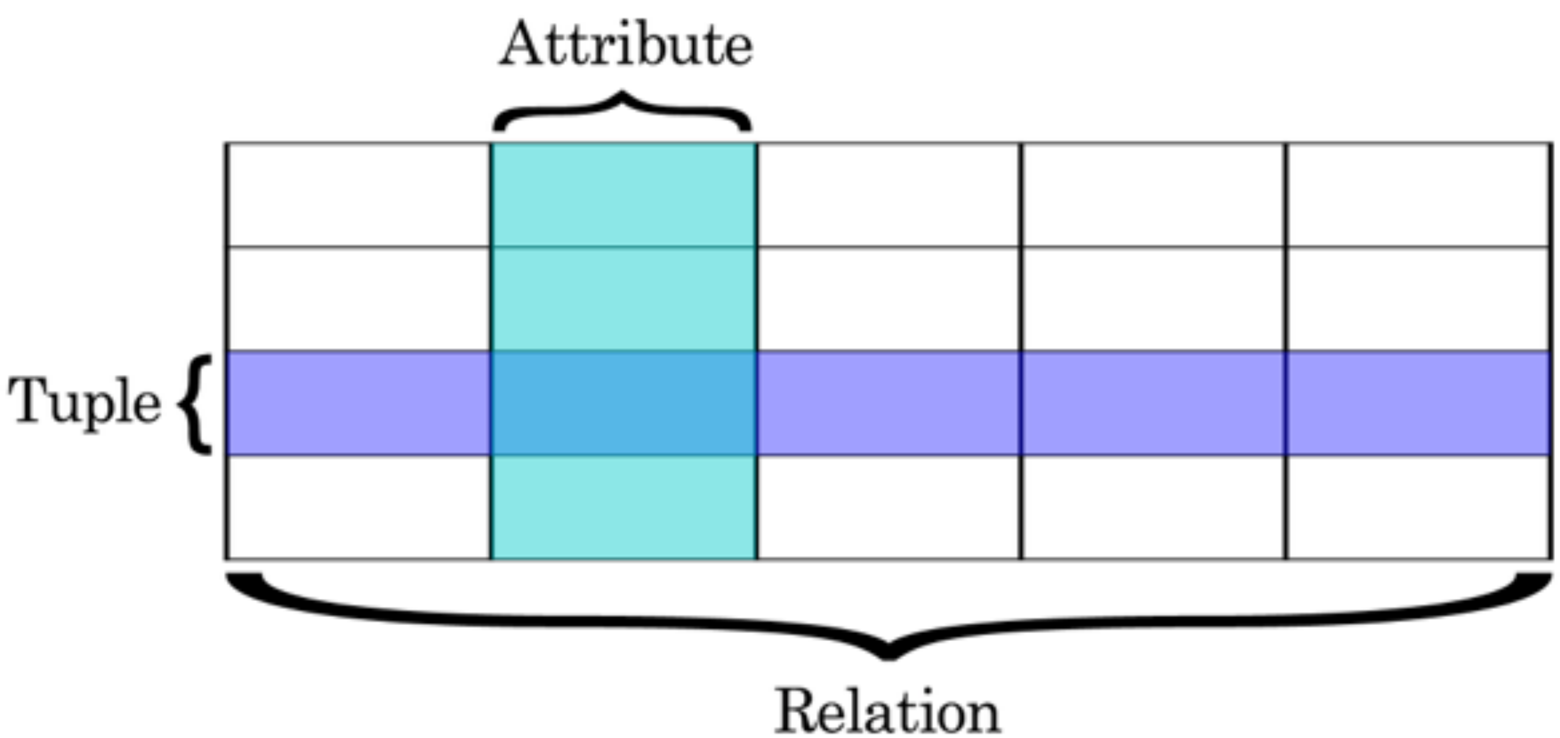


Table	Column	Row
Relation	Attribute	Tuple

Relation (database), Wikipedia

RELATION EXAMPLES

- Three named attributes: 'ID' from the domain of integers, and 'Name' and 'Address' from the domain of strings.
- A predicate for this relation might be "Employee number ID is known as Name and lives at Address”.
  - Employee 102 is known only by that name, Yonezawa Akinori, and does not live anywhere else but in Naha, Okinawa

ID (Integer)	Name (String)	Address (String)
102	Yonezawa Akinori	Naha, Okinawa
202	Murata Makoto	Sendai, Miyagi
104	Sakamura Ken	Kumamoto, Kumamoto
152	Matsumoto Yukihiro	Okinawa, Okinawa

Relation (database), Wikipedia

CHARACTERISTICS OF RELATIONS

- Relational DBMS products store data about entities in relations, which are a special type of table.
- A relation is a two-dimensional table that has the following characteristics:
  - Rows contain data about an entity.
  - Columns contain data about attributes of the entity.
  - All entries in a column are of the same kind.
  - Each column has a unique name.
  - Cells of the table hold a single value.
  - The order of the columns is unimportant.
  - The order of the rows is unimportant.
  - No two rows may be identical.

An entity is some identifiable thing that users want to track such as Order, Customer, etc.

삼라만상 (森羅萬象):[명사] 우주에 있는 온갖 사물과 현상.

EMPLOYEE RELATION

"Columns contain data about attributes of the entity."

"All entries in a column are of the same kind."

"Each column has a unique name."

"Rows contain data about an entity."

"No two rows may be identical."

EmployeeNumber	FirstName	LastName	Department	Email	Phone
100	Jerry	Johnson	Accounting	JJ@somewhere.com	834-1101
200	Mary	Abernathy	Finance	MA@somewhere.com	834-2101
300	Liz	Smathers	Finance	LS@somewhere.com	834-2102
400	Tom	Caruthers	Accounting	TC@somewhere.com	834-1102
500	Tom	Jackson	Production	TJ@somewhere.com	834-4101
600	Eleanore	Caldera	Legal	EC@somewhere.com	834-3101
700	Richard	Bandalone	Legal	RB@somewhere.com	834-3102

"The order of the rows and the columns is unimportant."

"Cells of the table hold a single value."

TABLES THAT ARE NOT RELATIONS

Multiple Entries per Cell

EmployeeNumber	FirstName	LastName	Department	Email	Phone
100	Jerry	Johnson	Accounting	JJ@somewhere.com	834-1101
200	Mary	Abernathy	Finance	MA@somewhere.com	834-2101
300	Liz	Smathers	Finance	LS@somewhere.com	834-2102
400	Tom	Caruthers	Accounting	TC@somewhere.com	834-1102, 834-1191, 834-1192
500	Tom	Jackson	Production	TJ@somewhere.com	834-4101
600	Eleanore	Caldera	Legal	EC@somewhere.com	834-3101
700	Richard	Bandalone	Legal	RB@somewhere.com	834-3102, 834-3191

Order of Rows Matters

EmployeeNumber	FirstName	LastName	Department	Email	Phone
100	Jerry	Johnson	Accounting	JJ@somewhere.com	834-1101
200	Mary	Abernathy	Finance	MA@somewhere.com	834-2101
300	Liz	Smathers	Finance	LS@somewhere.com	834-2102
400	Tom	Caruthers	Accounting	TC@somewhere.com	834-1102
				Fax:	834-9911
				Home:	723-8795
500	Tom	Jackson	Production	TJ@somewhere.com	834-4101
600	Eleanore	Caldera	Legal	EC@somewhere.com	834-3101
				Fax:	834-9912
				Home:	723-7654
700	Richard	Bandalone	Legal	RB@somewhere.com	834-3102

FUNCTIONAL DEPENDENCY

- Functional dependency is a constraint between two sets of attributes in a relation from a database.
- The attribute on the left side of the functional dependency is called the determinant.
- Composite determinant is a determinant of a functional dependency that consists of more than one attribute.
- Functional dependencies may be based on equations
- Function dependencies are not equations!

```
StudentID —> StudentName
StudentID —> (DormName, DormRoom, Fee)
ExtendedPrice = Quantity X UnitPrice
(Quantity, UnitPrice) —> ExtendedPrice
```

FUNCTIONAL DEPENDENCIES IN THE SKU_DATA TABLE

- SKU —> (SKU_Description, Department, Buyer)
- Anything else?

SKU	SKU_Description	Department	Buyer
100100	Std. Scuba Tank, Yellow	Water Sports	Pete Hansen
100200	Std. Scuba Tank, Magenta	Water Sports	Pete Hansen
101100	Dive Mask, Small Clear	Water Sports	Nancy Meyers
101200	Dive Mask, Med Clear	Water Sports	Nancy Meyers
201000	Half-dome Tent	Camping	Cindy Lo
202000	Half-dome Tent Vestibule	Camping	Cindy Lo
301000	Light Fly Climbing Harness	Climbing	Jerry Martin
302000	Locking Carabiner, Oval	Climbing	Jerry Martin

PROPERTIES OF FUNCTIONAL DEPENDENCIES

- Given that X, Y, and Z are sets of attributes in a relation R, one can derive several properties of functional dependencies:
  - Reflexivity: If Y is a subset of X, then $X \rightarrow Y$
  - Augmentation: If $X \rightarrow Y$, then $XZ \rightarrow YZ$
  - Transitivity: If $X \rightarrow Y$ and $Y \rightarrow Z$, then $X \rightarrow Z$
- From these rules, we can derive these secondary rules:
  - Union: If $X \rightarrow Y$ and $X \rightarrow Z$, then $X \rightarrow YZ$
  - Decomposition: If $X \rightarrow YZ$, then $X \rightarrow Y$ and $X \rightarrow Z$
  - Pseudotransitivity: If $X \rightarrow Y$ and $WY \rightarrow Z$, then $WX \rightarrow Z$
  - Composition: If $X \rightarrow Y$ and $Z \rightarrow W$, then $XZ \rightarrow YW$

Functional dependency, Wikipedia

CLOSURE OF FUNCTIONAL DEPENDENCY

- The closure is essentially the full set of values that can be determined from a set of known values for a given relationship using its functional dependencies.
- Imagine the following list of FD's. We are going to calculate a closure for A from this relationship.
  - 1. $A \rightarrow B$
  - 2. $B \rightarrow C$
  - 3. $AB \rightarrow D$
- The closure would be as follows:
  - a) $A \rightarrow A$ (by Armstrong's reflexivity)
  - b) $A \rightarrow AB$ (by 1. and (a))
  - c) $A \rightarrow ABD$ (by (b), 3, and Armstrong's transitivity)
  - d) $A \rightarrow ABCD$ (by (c), and 2)

A is a good candidate key!

Functional dependency, Wikipedia

WHAT MAKES DETERMINANT VALUES UNIQUE?

- A determinant is unique in a relation if and only if, it determines every other column in the relation.
- You cannot find the determinants of all functional dependencies simply by looking for unique values in one column:
  - Data set limitations
  - Must be logically a determinant
- Best strategies are to think about the nature of the business activity.

KEY

- A key is a combination of one or more columns that is used to identify rows in a relation.
- A candidate key is a key that determines all of the other columns in a relation.
- A primary key is a candidate key selected as the primary means of identifying rows in a relation.
  - There is only one primary key per relation.
  - The primary key may be a composite key.
  - The ideal primary key is short, numeric, and never changes.

SURROGATE KEY

- A surrogate key is an artificial column added to a relation to serve as a primary key.
  - DBMS supplied
  - Short, numeric, and never changes—an ideal primary key
  - Has artificial values that are meaningless to users
  - Normally hidden in forms and reports

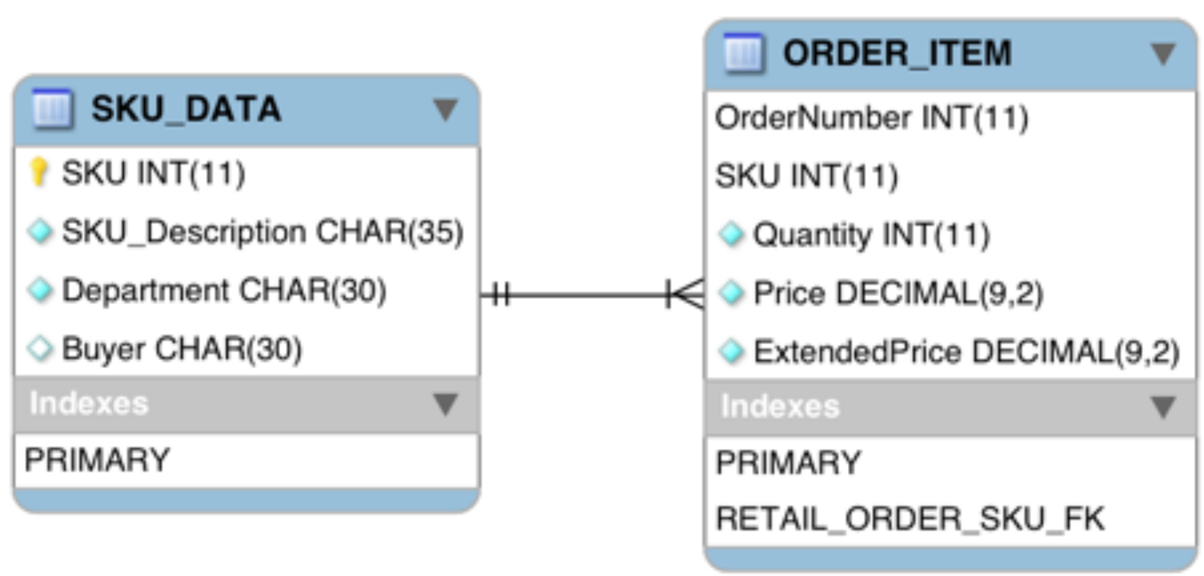
```
RENTAL_PROPERTY without surrogate key:
     RENTAL_PROPERTY (Street, City,
  State/Province, Zip/PostalCode, Country, Rental_Rate)

RENTAL_PROPERTY with surrogate key:
        RENTAL_PROPERTY (PropertyID, Street, City,
    State/Province, Zip/PostalCode, Country, Rental_Rate)
```

FOREIGN KEY

- A foreign key is the primary key of one relation that is placed in another relation to form a link between the relations.
  - A foreign key can be a single column or a composite key.
  - The term refers to the fact that key values are foreign to the relation in which they appear as foreign key values.

```
SKU_DATA    (SKU, SKU_Description, Department, Buyer)
ORDER_ITEM  (OrderNumber, SKU, Quantity, Price,
      ExtendedPrice)
```



INTEGRITY CONSTRAINTS

- Entity Integrity
  - A Primary key attributes can’t be null.
- Referential Integrity
  - Limits the values of the foreign key to those already existing as primary key values in the corresponding relation.
- Domain Integrity
  - Valid set of values for an attribute
  - Data type, length or size, null value allowed, the value unique or not
- Foreign Key Integrity
  - cascade update related fields and delete related rows.

MODIFICATION ANOMALIES

```
ItemNumber -> (EquipmentType, AcquisitionCost)
RepairNumber -> (RepairDate, RepairCost)
```

- Deletion anomaly
  - Delete the data for repair number 2100
- Insertion anomaly
  - Enter the first repair for a piece of equipment
- Update anomaly
  - Alter a value of AcquisitionCost 3500 to 5000 in the last row

	ItemNumber	EquipmentType	AcquisitionCost	RepairNumber	RepairDate	RepairCost
1	100	Drill Press	3500.00	2000	2013-05-05	375.00
2	200	Lathe	4750.00	2100	2013-05-07	255.00
3	100	Drill Press	3500.00	2200	2013-06-19	178.00
4	300	Mill	27300.00	2300	2013-06-19	1875.00
5	100	Drill Press	3500.00	2400	2013-07-05	0.00
6	100	Drill Press	3500.00	2500	2013-08-17	275.00

NORMAILIZATION

> "A technique for producing a set of relations with desirable properties, given the data requirements of an enterprise."
>
> \- Thomas Connolly -Database Systems

- 1NF—a table that qualifies as a relation is in 1NF.
- 2NF—a relation is in 2NF if all of its nonkey attributes are dependent on all of the primary keys.
- 3NF—a relation is in 3NF if it is in 2NF and has no determinants except the primary key.
- Boyce-Codd Normal Form (BCNF)—a relation is in BCNF if every determinant is a candidate key.

1NF – SKU_DATA

- Checking against the definition of 1NF, this relation is in 1NF.

```
SKU_DATA (SKU, SKU_Description, Department, Buyer)
```

SKU_DATA

	SKU	SKU_Description	Department	Buyer
1	100100	Std. Scuba Tank, Yellow	Water Sports	Pete Hansen
2	100200	Std. Scuba Tank, Magenta	Water Sports	Pete Hansen
3	101100	Dive Mask, Small Clear	Water Sports	Nancy Meyers
4	101200	Dive Mask, Med Clear	Water Sports	Nancy Meyers
5	201000	Half-dome Tent	Camping	Cindy Lo
6	202000	Half-dome Tent Vestibule	Camping	Cindy Lo
7	301000	Light Fly Climbing Harness	Climbing	Jerry Martin
8	302000	Locking Carabiner, Oval	Climbing	Jerry Martin

2NF – SKU_DATA

- SKU and SKU_Description are candidate keys.
- Since SKU is a single column primary key, all non-key attributes are determined by SKU, and the relation is in 2NF.

```
SKU_DATA (SKU, SKU_Description, Department, Buyer)


SKU —> (SKU_Description, Department, Buyer)

SKU_Description —> (SKU, Department, Buyer)

Buyer —> Department
```

3NF – SKU_DATA

- The term non-key attribute means an attribute that is neither (1) a candidate key itself, nor (2) part of a composite candidate key.
- Therefore, the only non key attribute is Buyer, and it is a determinant. So, The relation is not in 3NF.

```
SKU_DATA_2 (SKU, SKU_Description, Buyer)
BUYER (Buyer, Department)

Where SKU_DATA_2.Buyer must exist in BUYER.Buyer

SKU —> (SKU_Description, Department, Buyer)
SKU_Description —> (SKU, Department, Buyer)
Buyer —> Department
```

BCNF – SKU_DATA

- A relation is in BCNF if and only if it is in 3NF and every determinant is a candidate-key.
- In SKU_DATA_2, both determinants are determinant keys, so SKU_DATA_2 is in BCNF.
- In BUYER, the determinant is a determinant key, so BUYER is in BCNF.

SKU	SKU_Description	Buyer
100100	Std. Scuba Tank, Yellow	Pete Hansen
100200	Std. Scuba Tank, Magenta	Pete Hansen
101100	Dive Mask, Small Clear	Nancy Meyers
101200	Dive Mask, Med Clear	Nancy Meyers
201000	Half-dome Tent	Cindy Lo
202000	Half-dome Tent Vestibule	Cindy Lo
301000	Light Fly Climbing Harness	Jerry Martin
302000	Locking Carabiner, Oval	Jerry Martin

Buyer	Department
Cindy Lo	Camping
Jerry Martin	Climbing
Nancy Meyers	Water Sports
Pete Hansen	Water Sports