5.1. Define the following terms as they apply to the relational model of data: *domain, attribute, n-tuple, relation schema, relation state, degree of a relation, relational database schema,* and *relational database state.*

① domain : range of values of attribute

② attribute : components of relation

③ relation schema : description of relation

$R(A_1, A_2, \cdots, A_n)$

④ relation state : Relation state at particular time

$r_i = r(R) = \{t_1, t_2, \cdots, t_n\}$

⑤ degree of relation

: # of attributes in a relation schema

⑥ relational database schema

$S = \{R_1, R_2, \cdots R_n\}$

description of relations in DB

⑦ relational database state

$DB = \{r_1, r_2, \cdots, r_n\}$

↑ relation state

5.2. Why are tuples in a relation not ordered?

relation = set of tuples

Mathematically, tuples are not ordered.

5.3. Why are duplicate tuples not allowed in a relation?

Because primary key for each relation distinguishes every tuples in that relation.

5.4. What is the difference between a key and a superkey?

super key is all the possible combinations of attributes which can distinguish tuples

key (candidate key) is minimum attribute combination

5.5. Why do we designate one of the candidate keys of a relation to be the primary key?

To easily access each tuple

single distinct key (primary key) is easier to deal with

5.7. Discuss the various reasons that lead to the occurrence of NULL values in relations.

- unknown
- not available
- inapplicable

) case 의 NULL

5.8. Discuss the entity integrity and referential integrity constraints. Why is each considered important?

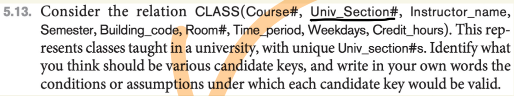
Entity integrity : primary key ≠ NULL

referential integrity : primary key – foreign key

⇓

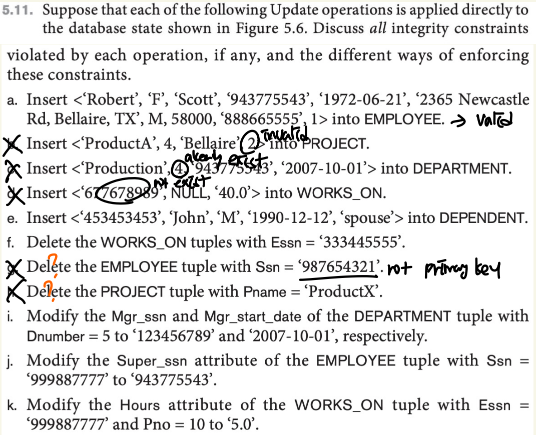
왜 중요?!

referencing tuple 이 exist 함을 guarantee 하기 위해서



5.13. Consider the relation CLASS(Course#, Univ_Section#, Instructor_name, Semester, Building_code, Room#, Time_period, Weekdays, Credit_hours). This represents classes taught in a university, with unique Univ_section#s. Identify what you think should be various candidate keys, and write in your own words the conditions or assumptions under which each candidate key would be valid.

{univ_section#, course#, semester}



5.11. Suppose that each of the following Update operations is applied directly to the database state shown in Figure 5.6. Discuss *all* integrity constraints violated by each operation, if any, and the different ways of enforcing these constraints.

a. Insert <'Robert', 'F', 'Scott', '943775543', '1972-06-21', '2365 Newcastle Rd, Bellaire, TX', M, 58000, '888665555', 1> into EMPLOYEE. → valid

✗ b. Insert <'ProductA', 4, 'Bellaire', 2> into PROJECT. invalid

✗ c. Insert <'Production', 4, '943775543', '2007-10-01'> into DEPARTMENT. already exist

✗ d. Insert <'677678989', NULL, '40.0'> into WORKS_ON. not exist

e. Insert <'453453453', 'John', 'M', '1990-12-12', 'spouse'> into DEPENDENT.

f. Delete the WORKS_ON tuples with Essn = '333445555'.

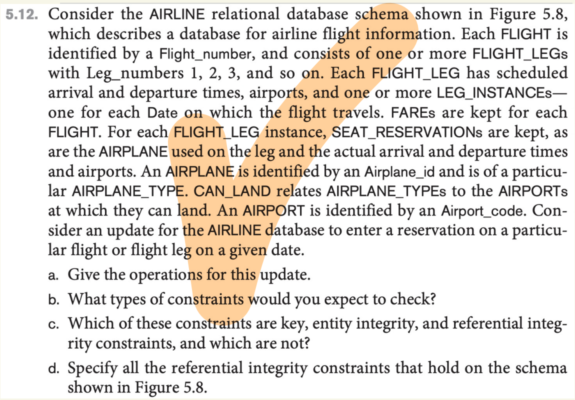
✗ g. Delete the EMPLOYEE tuple with Ssn = '987654321'. not primary key ?

✗ h. Delete the PROJECT tuple with Pname = 'ProductX'. ?

i. Modify the Mgr_ssn and Mgr_start_date of the DEPARTMENT tuple with Dnumber = 5 to '123456789' and '2007-10-01', respectively.

j. Modify the Super_ssn attribute of the EMPLOYEE tuple with Ssn = '999887777' to '943775543'.

k. Modify the Hours attribute of the WORKS_ON tuple with Essn = '999887777' and Pno = 10 to '5.0'.



5.12. Consider the AIRLINE relational database schema shown in Figure 5.8, which describes a database for airline flight information. Each FLIGHT is identified by a Flight_number, and consists of one or more FLIGHT_LEGs with Leg_numbers 1, 2, 3, and so on. Each FLIGHT_LEG has scheduled arrival and departure times, airports, and one or more LEG_INSTANCEs—one for each Date on which the flight travels. FAREs are kept for each FLIGHT. For each FLIGHT_LEG instance, SEAT_RESERVATIONs are kept, as are the AIRPLANE used on the leg and the actual arrival and departure times and airports. An AIRPLANE is identified by an Airplane_id and is of a particular AIRPLANE_TYPE. CAN_LAND relates AIRPLANE_TYPEs to the AIRPORTs at which they can land. An AIRPORT is identified by an Airport_code. Consider an update for the AIRLINE database to enter a reservation on a particular flight or flight leg on a given date.

a. Give the operations for this update.

b. What types of constraints would you expect to check?

c. Which of these constraints are key, entity integrity, and referential integrity constraints, and which are not?

d. Specify all the referential integrity constraints that hold on the schema shown in Figure 5.8.

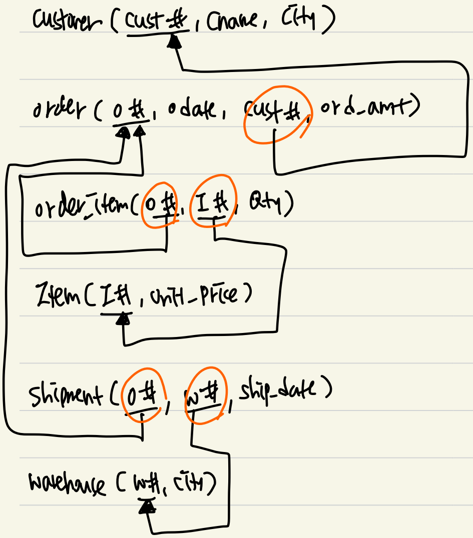
update fight or filght leg on given date

https://silo.tips/download/chapter-5-the-relational-data-model-and-relational-database-constraints

5.14. Consider the following six relations for an order-processing database application in a company:

CUSTOMER(Cust#, Cname, City)
ORDER(Order#, Odate, Cust#, Ord_amt)
ORDER_ITEM(Order#, Item#, Qty)
ITEM(Item#, Unit_price)
SHIPMENT(Order#, Warehouse#, Ship_date)
WAREHOUSE(Warehouse#, City)

Here, Ord_amt refers to total dollar amount of an order; Odate is the date the order was placed; and Ship_date is the date an order (or part of an order) is shipped from the warehouse. Assume that an order can be shipped from several warehouses. Specify the foreign keys for this schema, stating any assumptions you make. What other constraints can you think of for this database?



customer (cust #, Cname, City)

order (o #, odate, cust #, ord_amt)

order_item (o #, I #, Qty)

Item (I #, unit_price)

shipment (o #, w #, ship_date)

warehouse (w #, city)

5.15. Consider the following relations for a database that keeps track of business trips of salespersons in a sales office:

SALESPERSON(Ssn, Name, Start_year, Dept_no)
TRIP(Ssn, From_city, To_city, Departure_date, Return_date, Trip_id)
EXPENSE(Trip_id, Account#, Amount)

A trip can be charged to one or more accounts. Specify the foreign keys for this schema, stating any assumptions you make.

5.19. Consider a STUDENT relation in a UNIVERSITY database with the following attributes (Name, Ssn, Local_phone, Address, Cell_phone, Age, Gpa). Note that the cell phone may be from a different city and state (or province) from the local phone. A possible tuple of the relation is shown below:

Name	Ssn	Local_phone	Address	Cell_phone	Age	Gpa
George Shaw William Edwards	123-45-6789	555-1234	123 Main St., Anytown, CA 94539	555-4321	19	3.75

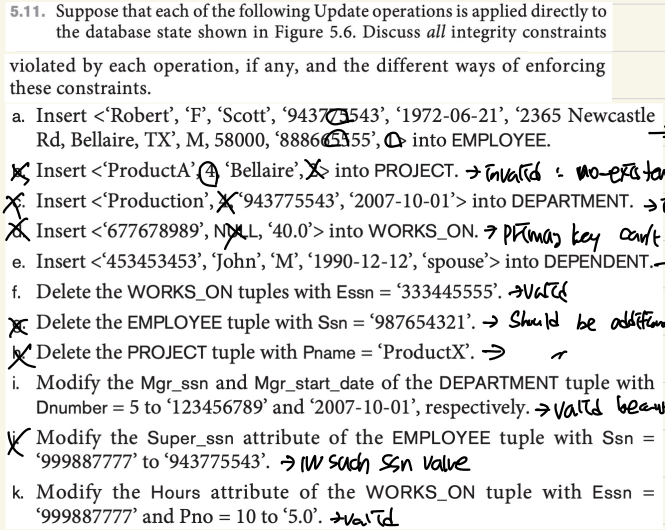
a. Identify the critical missing information from the Local_phone and Cell_phone attributes. (*Hint*: How do you call someone who lives in a different state or province?)

- Country number should be included / state number

b. Would you store this additional information in the Local_phone and Cell_phone attributes or add new attributes to the schema for STUDENT?

- change domain of two attributes

Re-Exercise



5.11. Suppose that each of the following Update operations is applied directly to the database state shown in Figure 5.6. Discuss *all* integrity constraints violated by each operation, if any, and the different ways of enforcing these constraints.

a. Insert <'Robert', 'F', 'Scott', '943775543', '1972-06-21', '2365 Newcastle Rd, Bellaire, TX', M, 58000, '888665555', 1> into EMPLOYEE. → valid

b. Insert <'ProductA', 4, 'Bellaire', 2> into PROJECT. → invalid : no-existance of foreign key

c. Insert <'Production', 4, '943775543', '2007-10-01'> into DEPARTMENT. → invalid : already existed primary key

d. Insert <'677678989', NULL, '40.0'> into WORKS_ON. → primary key can't be null

e. Insert <'453453453', 'John', 'M', '1990-12-12', 'spouse'> into DEPENDENT. → valid

f. Delete the WORKS_ON tuples with Essn = '333445555'. → valid

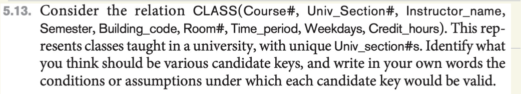
g. Delete the EMPLOYEE tuple with Ssn = '987654321'. → should be additional operation

h. Delete the PROJECT tuple with Pname = 'ProductX'. →

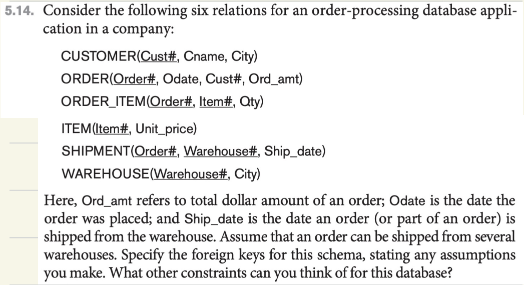
i. Modify the Mgr_ssn and Mgr_start_date of the DEPARTMENT tuple with Dnumber = 5 to '123456789' and '2007-10-01', respectively. → valid because they're ordinary attributes

j. Modify the Super_ssn attribute of the EMPLOYEE tuple with Ssn = '999887777' to '943775543'. → no such Ssn value

k. Modify the Hours attribute of the WORKS_ON tuple with Essn = '999887777' and Pno = 10 to '5.0'. → valid



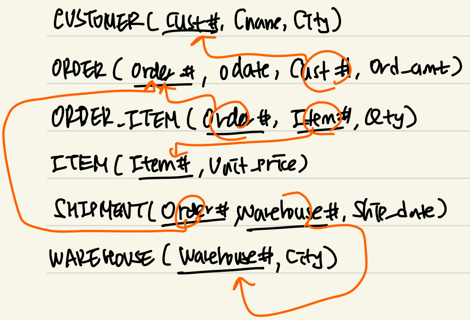
5.13. Consider the relation CLASS(Course#, Univ_Section#, Instructor_name, Semester, Building_code, Room#, Time_period, Weekdays, Credit_hours). This represents classes taught in a university, with unique Univ_section#s. Identify what you think should be various candidate keys, and write in your own words the conditions or assumptions under which each candidate key would be valid.



5.14. Consider the following six relations for an order-processing database application in a company:

CUSTOMER(Cust#, Cname, City)
ORDER(Order#, Odate, Cust#, Ord_amt)
ORDER_ITEM(Order#, Item#, Qty)
ITEM(Item#, Unit_price)
SHIPMENT(Order#, Warehouse#, Ship_date)
WAREHOUSE(Warehouse#, City)

Here, Ord_amt refers to total dollar amount of an order; Odate is the date the order was placed; and Ship_date is the date an order (or part of an order) is shipped from the warehouse. Assume that an order can be shipped from several warehouses. Specify the foreign keys for this schema, stating any assumptions you make. What other constraints can you think of for this database?


CUSTOMER (Cust#, Cname, City)
ORDER (Order#, odate, Cust#, Ord_amt)
ORDER_ITEM (Order#, Item#, Qty)
ITEM (Item#, Unit_price)
SHIPMENT(Order# Warehouse#, Ship_date)
WAREHOUSE (Warehouse#, City)