

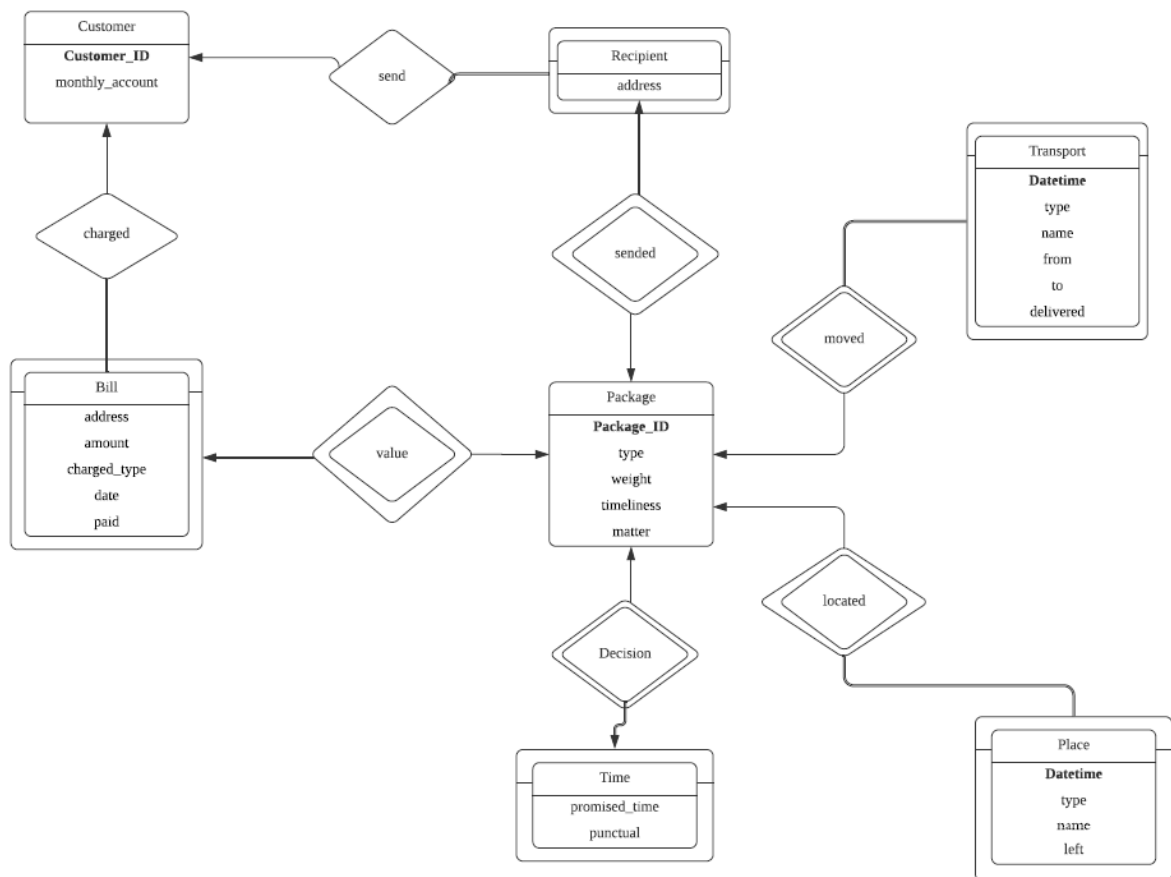
Project1

과목: 데이터베이스

학번: 20150614

이름: 어대영

1. E-R Diagram



2. 공통 설명

핵심 set은 Customer와 Package이다. 다른 set으로는 Bill, Recipient, Time, Place, Transport가 있다.

<Customer>

Customer_ID(PK) : Customer의 고유 번호로 식별하는 기준이 된다.

monthly_account : 매달 일정 금액을 납부하고 서비스를 이용하는 고객이 금액을 납부할 계좌이다. 개인마다 하나의 계좌가 각각 부여된다.

<Package>

Package_ID(PK) : 배송할 상품의 고유 번호로 식별하는 기준이 된다.

type : 소포의 유형으로 flat envelope, small box, large box 등을 값으로 갖는다.

weight : 소포의 무게이다.

timeliness : 배달 시간을 나타내는 속성으로 overnight, second day, longer 등을 값으로 갖는다.

matter : 배달할 수 없는 소포인지 나타내는 속성으로 hazardous, international, none을 값으로 갖는다. none인 경우 두 경우에 해당하지 않는, 바로 배달할 수 있는 소포를 뜻한다.

<Bill>

Package_ID(FK) : 소포마다 1개의 bill을 갖는다

Customer_ID : bill을 지불할 고객 번호이다.

address : bill을 보낼 주소이다.

amount : 부과된 금액이다.

charged_type : 고객의 지불 방식으로 monthly, credit, prepaid 등을 값으로 갖는다.

date: bill이 청구된 날짜이다.

paid: 지불여부를 확인하는 속성으로 True, False를 값으로 갖는다.

<Recipient>

Package_ID(FK) : 소포마다 1명의 수령인을 갖는다

Customer_ID : 수령하는 사람의 고객 번호로 고객 간의 배달을 가정한다.

address : 소포가 배달될 주소이다.

<Time>

Package_ID(FK) : 소포마다 1개의 시간정보를 갖는다.

promised_time : 소포의 배달 예정 시각으로 제 시간에 배달이 됐는지 체크하는 기준이 된다.

punctual : 소포가 제시간에 배달 되었는지를 나타내는 속성으로 True와 False, none을 값으로 갖는다. none인 경우 아직 배달되지 않은 것을 뜻한다.

<Transport >

Package_ID(FK) : 소포마다 n개의 운송수단 이용 정보를 갖는다.

Datetime(PK) : 소포가 운송수단에 실려진 시간을 값으로 갖는다. 같이 이동하는 소포들의 경우 실린 시간이 몇 분 차이가 나더라도 기준 시간을 정한 뒤 같은 시간을 입력한다. 예를 들어서 2020년 4월 23일 트럭에 100개의 소포를 실을 때 첫번째 소포를 1시에 싣고 마지막 소포를 2시에 싣었을 때 기준 시각을 1시30분으로 하면 100개의 소포에 대한 Datetime은 202004231330이 된다.

type : truck, airplane과 같이 운송수단 종류를 값으로 갖는다.

name : 1721같이 운송수단의 번호를 값으로 갖는다.

from : 운송수단이 출발한 곳을 값으로 갖는다.

to : 운송수단이 향하는 곳을 값으로 갖는다.

delivered : to로의 배달이 끝났는지 여부를 나타내는 속성으로 True, False를 값으로 갖는다.

<Place >

Package_ID(FK) : 소포마다 n개의 장소 정보를 갖는다.

Datetime(PK) : 소포가 해당 장소에 도착한 시간으로 운송수단에 실린 소포들 모두 동일한 시간 정보를 갖는다.

type : 장소의 종류로 warehouse, post office 등의 값을 갖는다.

name : 장소의 구체적인 정보로 예를 들어서 1번 warehouse의 경우 1을 값으로 갖는다.

left : 소포가 해당 장소를 떠났는지를 나타내는 속성으로 True와 False를 값으로 갖는다.

3. E-R Diagram

주된 set인 Customer와 Package의 Primary key인 ID를 FK이자 PK로 갖는 set들은 weak entity set 이 된다. 두 set을 제외한 다른 set들은 모두 package에 대한 정보를 갖고 있다. 따라서 Package_ID를 FK로 갖게 되고 이는 entity를 구분하는 기준이 되기 때문에 두 set을 제외한 다른 모든 set은 weak entity set이 된다.. Bill, Recipient, Time의 경우 소포당 1개의 정보를 갖기 때문에 Package_ID만을 PK로 갖고 Package와 one to one의 관계를 갖는다. Transport와 Place의 경우 소포당 여러 개의 entity를 가질 수 있기 때문에 Package와 one to many의 관계를 갖는다. 마지막으로 Customer는 여러 번의 배송 주문을 통해 여러 개의 Bill과 다수의 Recipient를 갖기 때문에 Bill과 Recipient의 경우 Customer와 one to many의 관계를 갖는다.

3. Relational Scheme diagram:

Customer와 Package는 다른 set과 non identifying relationship을 갖는다. 두 set을 제외한 나머지 set들은 식별에 Package_ID를 필요로 하기 때문에 identifying relationship을 갖는다. set간의 관계 (one to one, one to many 등)는 E R Diagram의 관계와 같다. Customer는 Recipient에게 소포를 보내므로 send의 relationship을 갖고 요금이 청구되기 때문에 Bill과 charged의 관계를 갖는다.

Package는 Recipient에게 보내지므로 sented의 관계를 갖고 각 소포마다 가격이 다르게 책정되므로 value의 관계를 갖는다. 또 소포는 어떤 Place에 위치하므로 located의 관계를 갖고 Transport에 의해 운반되므로 moved의 관계를 갖는다. 마지막으로 배송시간이 결정되므로 Time과 Decision의 관계를 갖는다.

4. Queries

<Assume truck 1721 is destroyed in a crash. (1)Find all customers who had a package on the truck at the time of the crash. (2)Find all recipients who had a package on that truck at the time of the crash. (3)Find the last successful delivery by that truck prior to the crash.>

1) with crashed as (select Package_ID, Datetime

from Transport_History

where type='truck' and name='1721' and delivered='False')

select B.Customer_ID

from Bill B

where B.Package_ID in (select Package_ID

from Transport_History as T

where T.Datetime=(select max(C.Datetime)

from crashed C))

2) with crashed as (select Package_ID, Datetime

from Transport_History

where type='truck' and name='1721' and delivered='False')

select R.Customer_ID

from Recipient R

where R.Package_ID in (select Package_ID

from Transport_History as T

where T.Datetime=(select max(C.Datetime)

```

                                from crushed C))

3) with crushed as (select Package_ID, Datetime
                        from Transport_History
                        where type='truck' and name='1721' and delivered='True')

select B.Customer_ID
from Bill B
where B.Package_ID in (select Package_ID
                        from Transport_History as T
                        where T.Datetime=(select max(C.Datetime)
                                           from crushed C))

```

<Find the customer who has shipped the most packages in the past year.>

*past year = 2019

```

with delivery as (select Customer_ID, count(*) num
                  from Bill
                  where date between 20190000 and 20200000
                  group by Customer_ID)

select D1.Customer_ID
from delivery D1
where D1.num=(select max(D2.num)
              from delivery D2)

```

설명: Bill에서 date는 연월일의 값을 갖는다. 20190000~20200000은 2019년도에 해당하는 Bill을 선택할 것이고 이를 Customer_ID로 묶고 개수를 센다. 이를 활용해서 개수 값이 가장 큰 Customer의 ID를 출력한다.

< Find the customer who has spent the most money on shipping in the past year>

with delivery as (select Customer_ID, sum(amount) pay

from Bill

where date between 20190000 and 20200000

group by Customer_ID)

select D1.Customer_ID

from delivery D1

where D1.pay=(select max(D2.pay)

from delivery D2)

<Find those packages that were not delivered within the promised time.>

select Package_ID

from Time

where punctual='False'

< Generate the bill for each customer for the past month. Consider creating several types of bills.

(1) A simple bill: customer, address, and amount owed. (2) A bill listing charges by type of service.

(3) An itemize billing listing each individual shipment and the charges for it.>

*past month=03,

(1) with last_month_bill as (select Customer_ID, address, amount, date

from Bill

where data between 20200300 and 20200400

order by date)

select Customer_ID, address, amount

from last_month_bill B1

order by Customer_ID

(2) with last_month_bill as (select Package_ID, Customer_ID, address, amount, date
from Bill
where data between 20200300 and 20200400
order by date)

select B.Customer_ID, P.type, B.address, B.amount
from last_month_bill B, Package P
where B.Package_ID=P.Package_ID
order by B.Customer_ID, P.type

(3) with last_month_bill as (select Customer_ID, address, amount, date, charged_type
from Bill
where data between 20200300 and 20200400
order by date)

select Customer_ID, address, amount, charged_type
from last_month_bill
group by Customer_ID, charged_type