فایل هفتم درس پایگاه داده ها

با سلام و آرزوی سلامتی برای تمامی دانشجویان، فایل مربوطه برای تدریس جلسه هفتم کلاس پایگاه داده ها در دوره آموزش مجازی تهیه شده است.

موفق باشید-اسبقی

ادامه بحث جبر رابطه ای

S(s#, s_name,s_city)(تهیه کننده)

P(p#, p_name,p_color,p_weight)(قطعه)

SP(s#,p#, number)(تولید)

s

S#	S_name	S_city
s1	Ali	tehran
s2	Mina	karaj
s3	Reza	tehran
s4	Hasan	shiraz

sp

S#	P#	number
s1	p1	700
s1	p2	90
s2	p2	100
s3	p4	200
s3	p2	50
s4	p5	100
s1	p4	70
s1	p5	20
s1	p3	30

p

P#	P_name	P_color	P_weight
p1	x	red	10g
p2	y	blue	30g
p3	z	yellow	5g
p4	k	red	15g
p5	g	blue	2g

سیستم بانک

- بعنوان مثالی دیگر یک سیستم بانک با جداول زیر را در نظر بگیرید:

Branch(b_name, b_city, assets)(شعبه بانک)

Customer(c#, c_name, c_city,c_street)(مشتری)

Account(acc#, c#, b_name, balance)(حساب بانکی)

Loan(L#, b_name, amount)(وام)

Borrower(c#, L#)(وام گیرنده)

مثال

Branch(b_name, b_city, assets)
Customer(c#, c_name, c_city,c_street)
Account(acc#, c#, b_name, balance)
Loan(L#, b_name, amount)
Borrower(c#, L#)

- اسامی مشتریان ساکن ‘تهران’؟

Π c-name(σc_city=‘tehran’ (Customer))

- شماره مشتریانی که در شعبه ‘مرکزی’ حساب دارند؟

Π c#(σb_name=‘markazi’ (Account))

- مشخصات کامل مشتریان ساکن خیابان ‘آزادی’؟

σc_street=‘Azadi’ (Customer)

مثال

Branch(b_name, b_city, assets)
Customer(c#, c_name, c_city,c_street)
Account(acc#, c#, b_name, balance)
Loan(L#, b_name, amount)
Borrower(c#, L#)

- تعداد حساب های بانکی شعبه ‘مرکزی’؟

Π count(acc#)(σb_name=‘markazi’(Account))

- اسامی مشتریانی که در شعبه ‘مرکزی’ حساب بانکی دارند؟

Π Costomer.c-name(σCustomer.c#=Account.c# $\cap$ Account.b_name = ‘markazi’ (Customer×Account))

- شماره وام های شعبه ‘مرکزی’ و میزان بیش از 10 میلیون؟

Π L#(σb_name = ‘markazi’ $\cap$ amount > 10,000,000(Loan))

مثال

Branch(b_name, b_city, assets)
Customer(c#, c_name, c_city,c_street)
Account(acc#, c#, b_name, balance)
Loan(L#, b_name, amount)
Borrower(c#, L#)

- شماره مشتریانی که وام دریافت نموده اند؟

$\Pi_{c\#}$ (Borrower))

- شماره مشتریانی که موجودی حساب بانکی آنها بیشتر از 1 میلیون است؟

$\Pi_{c\#}(\sigma_{balance > 1,000,000}$(Account))

- شماره مشتریانی که از شعبه ‘مرکزی’ وام دریافت نموده اند؟

$\Pi_{Borrower.c\#}(\sigma_{Loan.L\# = Borrower.L\# \ \wedge \ Loan.b_name = \text{‘markazi’}}$ (Loan×Borrower))

مثال

- شماره تهیه کنندگانی که ساکن ‘تهران’ هستند یا قطعه ‘p2’ را تهیه کرده اند.

sp

S#	P#	number
s1	p1	700
s1	p2	90
s2	p2	100
s3	p4	200
s3	p2	50
s4	p5	100
s1	p4	70
s1	p5	20
s1	p3	30

s

S#	S_name	S_city
s1	Ali	tehran
s2	Mina	karaj
s3	Reza	tehran
s4	Hasan	shiraz

(П S#(БP#=‘p2’ (SP)))U (П S#(БS_city=‘tehran’ (S))

S#
s1
s2
s3

U

S#
s1
s3



S#
s1
s2
s3

مثال

- شماره قطعات به رنگ ‘آبی’ که توسط تهیه کننده’s2’ تهیه شده اند؟

sp

S#	P#	number
s1	p1	700
s1	p2	90
s2	p2	100
s3	p4	200
s3	p2	50
s4	p5	100
s1	p4	70
s1	p5	20
s1	p3	30

p

P#	P_name	P_color	P_weight
p1	x	red	10g
p2	y	blue	30g
p3	z	yellow	5g
p4	k	red	15g
p5	g	blue	2g

$(\Pi_{P\#}(\sigma_{S\#='s2'}(SP))) \cap (\Pi_{P\#}(\sigma_{P_color='blue'}(P)))$

P#
P2

$\cap$

P#
p2
p5



P#
p2

مثال

- شماره تهیه کنندگانی ساکن ‘تهران’ که قطعه ‘p1’ را تهیه نکرده اند؟

sp

S#	P#	number
s1	p1	700
s1	p2	90
s2	p2	100
s3	p4	200
s3	p2	50
s4	p5	100
s1	p4	70
s1	p5	20
s1	p3	30

s

S#	S_name	S_city
s1	Ali	tehran
s2	Mina	karaj
s3	Reza	tehran
s4	Hasan	shiraz

(П S#(бS_city=‘tehran’ (S))) - (П S#(бP#=‘p1’ (SP)))

S#
s1
s3

-

S#
s1



S#
s3

عملگرهای جبر رابطه ای

- عملگر تقسیم(division)

عملگر تقسیم در پرس و جوهایی بکار می رود که عبارت «برای تمامی» « for all » دارند. عملگر تقسیم یک رابطه با m+n صفت خاصه را بر یک رابطه با n صفت تقسیم نموده و نتیجه رابطه ای با m صفت خواهد بود.

نماد عملگر: A÷R

A

a	b
7	x
3	x
3	y
7	y
4	x

÷

R

b
x
y

→

a
3
7

مثال

- اسامی تهیه کنندگان قطعه ’p3‘؟

s

S#	S_name	S_city
s1	Ali	tehran
s2	Mina	karaj
s3	Reza	tehran
s4	Hasan	shiraz

sp

S#	P#	number
s1	p1	700
s1	p2	90
s2	p2	100
s3	p4	200
s3	p2	50
s4	p5	100
s1	p4	70
s1	p5	20
s1	p3	30

Π S.S_name(σS.S#=SP.S# $\wedge$ SP.P#=‘p3’(S×SP))

Π S.S_name(σSP.P#=‘p3’(S⋈SP))

S_name
Ali

مثال

- شماره تهیه کنندگانی که تمامی قطعات را تولید کرده اند؟

sp

S#	P#	number
s1	p1	700
s1	p2	90
s2	p2	100
s3	p4	200
s3	p2	50
s4	p5	100
s1	p4	70
s1	p5	20
s1	p3	30

p

P#	P_name	P_color	P_weight
p1	x	red	10g
p2	y	blue	30g
p3	z	yellow	5g
p4	k	red	15g
p5	g	blue	2g

$\Pi_{SP.S\#,SP.P\#}(SP) \div \Pi_{P.P\#}(P)$

S#
s1

مثال

- اسامی تهیه کنندگان قطعات با رنگ ‘زرد’؟

sp

S#	P#	number
s1	p1	700
s1	p2	90
s2	p2	100
s3	p4	200
s3	p2	50
s4	p5	100
s1	p4	70
s1	p5	20
s1	p3	30

p

P#	P_name	P_color	P_weight
p1	x	red	10g
p2	y	blue	30g
p3	z	yellow	5g
p4	k	red	15g
p5	g	blue	2g

s

S#	S_name	S_city
s1	Ali	tehran
s2	Mina	karaj
s3	Reza	tehran
s4	Hasan	shiraz

Π S.Sname(σP.P#=SP.P# $\wedge$ P.P_color =‘yellow’ $\wedge$ SP.S# = S.S#((P×SP)×S))

S_name
Ali

مثال

Branch(b_name, b_city, assets)
Customer(c#, c_name, c_city,c_street)
Account(acc#, c#, b_name, balance)
Loan(L#, b_name, amount)
Borrower(c#, L#)

- شماره مشتریان دارای حساب بانکی که وام گرفته اند؟

(Π c# (Borrower))∩ (Π c# (Account))

- شماره مشتریانی که وام دریافت کرده اند یا حساب بانکی دارند؟

(Π c# (Borrower))U (Π c# (Account))

مثال

- شماره مشتریان ساکن ‘تهران’ که تاکنون وام نگرفته اند؟

Branch(b_name, b_city, assets)
Customer(c#, c_name, c_city,c_street)
Account(acc#, c#, b_name, balance)
Loan(L#, b_name, amount)
Borrower(c#, L#)

(Π c#(σc_city=‘tehran’ (Costomer))) - (Π c# (Borrower))

- شماره مشتریانی که در تمامی شعبه های شهر ‘تهران’ حساب بانکی دارند؟

Π c#,b_name(Account) ÷ Π b_name (σb_city=‘tehran’ (Branch))

مثال

- اسامی مشتریانی که از از شعبه ‘مرکزی’ وام دریافت نموده اند؟

Branch(b_name, b_city, assets)
Customer(c#, c_name, c_city,c_street)
Account(acc#, c#, b_name, balance)
Loan(L#, b_name, amount)
Borrower(c#, L#)

Π Customer.c_name(бLoan.L# = Borrower.L# ∩ Loan.b_name = ‘markazi’ ∩ customer.c# = Borrower.c#
(Loan×Borrower)×Customer)

- حذف تمامی حسابهای مشتری ‘c1’؟

Account ← Account - бc# =‘c1’(Account)

زبان پرس و جوی SQL

زبان پرس و جوی SQL(Structured Query Language)

مفاهیم اصلی:

-DDL (Data Definition Language)- زبان تعریف داده ها

- DML(Data Manipulation Language)-زبان پردازش و دستکاری داده ها

-View Definition- تعریف دید

-DCL(Data Control Language) - زبان مدیریت و کنترل داده ها

-Autorization-مجوز دهی

-Integrity- جامعیت پایگاه داده ها

–Transaction Control کنترل تراکنش ها

دستورات DDL

- دستورات DDL، مجموعه دستوراتی را در برمی گیرند که مربوط به ایجاد، حذف و تغییرات روی جداول پایگاه داده ها می باشد.
- ایجاد، حذف و تغییرات پایگاه
- دستور ایجاد یک جدول
- دستور حذف یک جدول
- دستور تغییرات روی ساختار یک جدول
  - اضافه کردن یک ستون
  - حذف یک ستون
  - تغییر میدان یک ستون

دستورات ایجاد پایگاه داده بانک

```
CREATE    DATABASE   BANK;

CREATE  TABLE   Branch(
b_name    char(10)  not null,
b_city       varchar(30),
Assets    integer,
Primary Key(b_name));

CREATE  TABLE   Customer(
C#    char(10)   not null,
c_name    varchar(20),
c_city       varchar(30),
c_street    varchar(30),
Primary Key(c#));
```

دستورات ایجاد پایگاه داده بانک

```
CREATE TABLE Account(
acc# char(10) not null,
c# char(10) not null,
b_name char(10) not null,
balance integer,
Primary Key(acc#),
Foreign Key(c#) References Customer(c#),
Foreign Key(b_name) References Branch(b_name));
```

دستور حذف جدول در پایگاه داده

- DROP TABLE table name

- با این دستور یک جدول بصورت کامل حذف می شود، بر همین اساس بعد از درج داده در جداول پایگاه داده، این دستور اجرا نمی گردد.
- چون با حذف یک جدول، کل داده های آن حذف شده و همچنین تمامی ارجاعات به این جدول نیز خدشه دار می گردد که این ممکن است باعث از کار افتادن کل پایگاه شود. بنابراین بعد از ورود داده، از این دستور استفاده نمی گردد.

دستورات تغییرات جداول در پایگاه داده

- ALTER TABLE …..
  - اضافه کردن یک ستون جدید در جدول

```
ALTER TABLE Customer ADD melli# char(10)
```

  - حذف یک ستون از جدول

```
ALTER TABLE Customer DROP melli#
```

  - تغییر حوزه میدان یک ستون

```
ALTER TABLE Customer MODIFY c_street char(40)
```

دستورات DML– کار با محتوای پایگاه داده

- دستورات DML دربرگیرنده کلیه عملیات مربوط به کار با محتوای پایگاه داده می باشد:
- خواندن (read) با دستور (select)
- حذف کردن(delete)
- اضافه کردن(insert)
- بروزرسانی و تغییر دادن(update)

دستور select

- در حالت کلی دستور select بصورت زیر تعریف می شود:

Select a1,a2,……an

from r1,r2,…..,rn

[where p]

[Order by a1,a2,……an]

a1,a2,....an: صفات خاصه

r1,r2,....,rn: اسامی جداول(, به معنای حاصلضرب کارتزین می باشد.)

P: شرط گزینش

Order by: دستور مرتب سازی نتایج پرس و جو

مثال

Branch(b_name, b_city, assets)
Customer(c#, c_name, c_city,c_street)
Account(acc#, c#, b_name, balance)
Loan(L#, b_name, amount)
Borrower(c#, L#)

- اسامی مشتریان ساکن ‘تهران’؟

Π c-name(σc_city=‘tehran’ (Customer))

```
select  c_name
from    Customer
where   c_city = ‘tehran’
```

- شماره مشتریانی که در شعبه ‘مرکزی’ حساب دارند؟

```
select  c#
from    Account
where   b_name = ‘markazi’
```

Π c#(σb_name=‘markazi’ (Account))

- مشخصات کامل مشتریان ساکن خیابان ‘آزادی’؟

```
select  *
from    Customer
where   c_street = ‘Azadi’
```

σc_street=‘Azadi’ (Customer)

مثال

Branch(b_name, b_city, assets)
Customer(c#, c_name, c_city,c_street)
Account(acc#, c#, b_name, balance)
Loan(L#, b_name, amount)
Borrower(c#, L#)

- مشخصات کامل حسابهای بانکی؟

```
select   *
from    Account
```

- شماره مشتریان وام گیرنده از شعبه ‘مرکزی’؟

```
select  Borrower.c#
from    Borrower, Loan
where   Borrower.L# = Loan.L#   and
Loan.b_name = ‘markazi’
```

- تعداد حسابهای شعبه ‘مرکزی’؟

```
select  count(acc#)
from    Account
where   b_name = ‘markazi’
```

مثال

- شماره حسابهایی که مقدار آنها 1000 یا 2000 یا 3000 یا 5000 است؟

```
select  acc#
from    Account
where   balance = 1000 or  balance = 2000  or  balance = 3000  or balance = 5000
```

در زبان پرس و جوی SQL، پرس و جوی بالا را می توان با دستور in بصورت زیر نوشت:

```
select  acc#
from    Account
where   balance  in(1000 , 2000  , 3000  , 5000)
```

عملکرد not in نیز برعکس in می باشد.

مثال

- مشخصات وام هایی که مقدار آنها بین 10 میلیون تا 20 میلیون است؟

```
select   *
from    Loan
where   amount>=10,000,000 and   amount <= 20,000,000
```

چنین پرس و جوهایی را نیز می توان با دستور between بصورت زیر نوشت:

```
select   *
from    Loan
where  amount   between  10,000,000   and   20,000,000
```

در مقابل، not between نیز عملکرد عکس خواهد داشت.

مثال

- اسامی مشتریانی که آدرس خیابان آنها نامشخص است؟

```
select   c_name
from    Customer
where   c_street   is null
```

- اسامی مشتریانی که آدرس آنها لزوما مشخص باشد؟

```
select   c_name
from    Customer
where   c_street   is  not null
```

مثال
دستور order by برای مرتب سازی نتایج پرس و جو

- شماره مشتریانی که در شعبه ‘مرکزی ’حساب های با موجودی بیش از 10 میلیون دارند؟مرتب شده بصورت صعودی

```
select  c#
from    Account
where   b_name = ‘markazi’ ∩ balance>10,000,000
Order by   balance
```

- شماره وام های تعریف شده در شعبه ‘مرکزی ’به ترتیب مقدار نزولی؟

```
select  L#
from    Loan
where   b_name = ‘markazi’
Order by  amount  desc
```

- شماره مشتریان حسابهای شعبه ‘مرکزی’ به ترتیب موجودی نزولی و درصورت برابری موجودی به ترتیب صعودی شماره حساب؟

```
select  c#
from    Account
where   b_name = ‘markazi’
Order by   balance  desc, acc#  asc
```

مثال
distinctبرای حذف نتایج تکراری

- تعداد حسابهای بانکی شعبه ‘مرکزی’ ؟

```
select   count(acc#)
from    Account
where   b_name = ‘markazi’
```

- تعداد مشتریان حسابهای بانکی شعبه ‘مرکزی’ ؟

```
select   count(  distinct  c#)
from    Account
where   b_name = ‘markazi’
```

- اسامی شعبه هایی که در آنها وام تعریف شده است؟

```
select    distinct   b_name
from    Loan
```

```
select    all   b_name
from    Loan
```