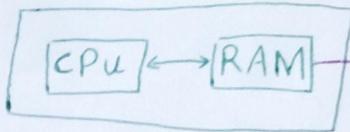


Digit e
رقم
۰, ۱

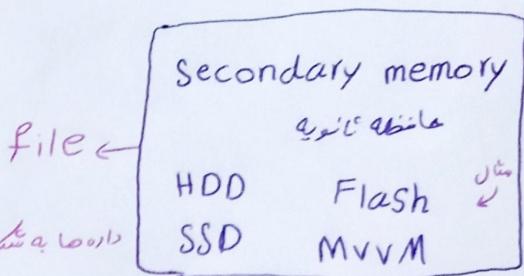
Computer

$$0 + 1 = 1$$



Arby
Linked-List
Tree
Graph
Stack

↑
Data structure
ساخته های داده ها

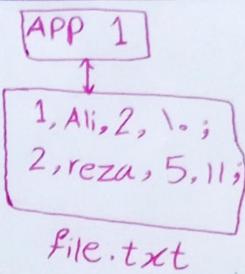


داده ها به سطل فایل ذخیره می شوند

① Text file

فایل

استفاده از شابل صاف کرده آن فایل را می توان
از نمایش بخوبی.



داده ها در ساختار صاف به نام فایل ذخیره می شوند
اگر بخواهیم داده ها در فایل ذخیره کنیم چند طریق برای داشته باشیم:

مثال ① : در دست اول مشخصه های صفتی برای شفتمین نام کمتر
روییت دارم مشخصه های صفتی برای شفتمین نام کمتر

مشخصه های با علاوه کاراکتر (۰) از قاعده جدا می شوند

مشخصه های صفتی با انفرادی برای این اسم کارن (۰) از قاعده جدا می شوند.
حالا ما اگر یک ایجاد کنیم داشته باشیم باید این دست را بین خود و از این دست را تا سخواند. میں کارهای این ایجاد کریش انجام
می دهد. اولین کار را بخواهی که از نام من بعد دست را درج من کنم، بعد دست را حذف کنم، دست را آپدیت کنم،
دست را استخراج کنم، همانگونه حجم فایل زیارت شود، این عملیات به سختی انجام می شوند.

برای رفع این مشکل باید این فایل را بر روی سرور آورده و بعد همه کارهای این فایل به سرور متصل می شوند و در تابیس پیوست
در گذشته مشکل این را بخواهیم داشت و پس از آن داده های این فایل را باید نیز پیوست. اما زمانه توییس نیز هم زیاد برآید و نوبت

باید باید باشد که همه سطح و اساس کار باید کاهش داده را باید بردازد

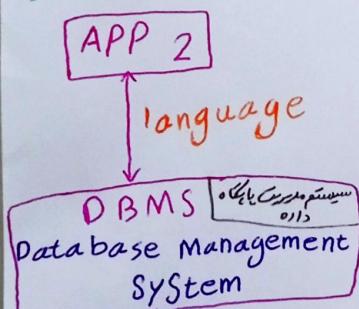
حالا گفتند اگر من خواهیم ایجاد کریش بسازید، دست آن را درون یک مخزن بخواهیم که آن مخزن دست را باشد که این نیز

می باشد اس. حالا سفارشات آن بآن تابیل صاف های ریگر چیزیست؟

هم: آدمین بین برنامه و در تابیس پیوستی یا باید من نظره باشد

هم: DBMS تاردارن

ظبطیغه DBMS این بزرگه برنامه توییس نیاز به راستی سطحی و ساخت فایل را نداشتند باشد.



logic and structure



Relational Model

دلایل بسطه ای

هم: دلایل بسطه ای براساس روابط بین مجموعه ها، مجموعه های خالی و جبر لایه ای است.
نته: دلایل بسطه ای در عبارت جداگانه است.

(DBMS) نام داده بانی (RDBMS) نام داده بانی DBMS ها

oracle @ MySQL @ MS SQL server @ MS Access @ RDBMS های دنی سمعان
SQLite @ Sybase @ DB2 @

نته: با این دست ادبیاتی دلایل بسطه ای در عبارت جداگانه است.

SQL (Structured Query Language)

برنامه های

field	Table	column
Row		Column
ID	Name	C
		A
		B
		...
		...
		...

نته: قسم سهون آمی دی (ID Column) است. بازگشت فیلد آمی دی (ID field)

مدل دستور SQL:

- ساخت جدول یا ریتاپس جدول CREATE
- تغییر ساختار جدول مثل (اضافه کردن ستون ALTER
- حذف جدول یا ریتاپس DROP
- حذف صورتی داره های جدول - رونو TRUNCATE
- حذف جدول

- افزودن رکور (جدید) INSERT
- دریافت رکوردهای جدید UPDATE
- حذف رکوردهای از جدول DELETE

- انتخاب و نمایش داده ها از یک یا چند جدول SELECT

- اعطای رسترسی برای GRANT
- لغو رسترسی برای REVOKE

- ذخیره سفیری داشتی COMMIT
- بازگرداندن سفیری ROLLBACK
- رتصوری خطا
- ایجاد نقطه بازگردانی SAVEPOINT
- تنظیم محدوده کار SET TRANSACTION

P

PK ↓

PID	Pname	Pcolor
1	CPU	Silver
2	Flash	Black
3	GPU	Black
4	Tablet	White

S

PK ↓

SID	Sname	Scity
1	AMD	CA
2	ATI	CA
3	Sony	To
4	LG	Se

SP

FK ↓ **FK** ↓

PID	SID	Qty
1	1	10
1	2	20
2	1	50
3	2	100

create Database Mydb;

use Mydb;

create Table PC

PID int Primary key Identity(1,1),

Pname nchar(10) not null,

Pcolor nchar(10) Default ('white')

);

create Table SC

SID int Primary key identity(0),

Sname nvarchar(30),

Scity nchar(10) check('CA', 'Se', 'To', 'Bnd', 'Minab')

);

create Table SPC

PID int ,

SID int ,

Qty int Default(0),

Constraint fk_P Foreign Key(PID)
References PC(PID),

constraint fk_S Foreign Key(SID)
References SC(SID),

این رستورانی مایک بایگانه دارد
بانام db1 رایم سازر.

Create database db1;

use db1;
create table test(
id int,
name nchar(10))

insert into test(id, name) values(50, 'ali');
Select * from test;

خروجی	
id	name
50	ali

create table test(
id int Primary key identity(1,1),
name nchar(10));
drop table test;

/*DML*/

insert into test(id, name) values(10, 'Reza');
insert into test(name) values('Ali');
Select * from test;

خروجی		
	id	name
1	1	Reza
2	2	Reza
3	3	Ali

- ① این رستور جدول test را به طور کامل حذف می کند (هم ساخت و نعم را داشت)
② این رستور سعی می کند همه مقدار id و name را در جدول وارد کند. اما یکی از ستون identity id تعریف شده، نباید به آن مقدار رسانی بدم و گزئه خطایم دهد.
③ در اینجا فقط ستون name مقدار رصی می شود. ستون id به صورت خودکار (با identity مقدار رصی) می شود و این روش درست است.

- ④ نمایش های اطلاعات جدول test
کاربرد primary key (کلید اصلی): ① نمی گذرد مقدار تکراری وارد شد ② نمی گذرد مقدار [null] داشته باشد.
کاربرد drop table: ① صعبی دارهها و ساختار جدول یا کی می شود ② برگشت برگر نیست (گراینه کابی پائمه باشی)

① این خط ام کوید از ریتا میس به نام db1 استفاده کن.
یعنی از این پس بعد همه رستورات خواه این ریتا میس اجرا می شود.

- ② این خانواده داران یک جدول جدید به نام tset است با درستون:
ستون id از نوع عدد صحیح (int) که یعنی رشته ای با طول ثابت ۱۰ کاراکتر
ستون name از نوع nchar(10) که یعنی رشته ای با طول ثابت ۱۰ کاراکتر
@ این خط یک ریتف (سطر) جدید وارد جدول tset می کند
مقدار id برابر با ۵۰ و مقدار name برابر با 'ali' است
③ این رستور تمام اطلاعات موجود در جدول tset را نمایش می دهد
علمانت * یعنی صهی ستون ها (در اینجا id و name)

```

Create table test(
    id int Primary Key identity (1,1),
    name nchar (10)
);

```

```
drop table test;
```

/*DML*/

```
insert into test (id, name) values (10, 'Reza');
```

```
insert into test (name) values ('Ali');
```

```
Select * from test;
```

```
update test set name = 'Hadi' where id=2;
```

	id	name
1	1	Reza
2	2	Hadi
3	3	Ali

١٠ توضیح خط آخر

update test : مخفی توان جدول test را برایش کنی

set name = 'Hadi' : مقدار سعی name براي id هر چهار عدد.

Where id=2 : فقط برای رفع کردن id باید این تغییرات را اعمال کرد.

```
Create table test(
```

```
id int Primary Key identity (1,1),
```

```
name nchar (10)
```

```
);
```

```
drop table test;
```

/*DML*/

```
insert into test (id, name) values (10, 'Reza');
```

```
insert into test (name) values ('Ali');
```

```
Select * from test;
```

```
update test set name = 'Hadi' where id=2;
```

```
delete from test where id=1;
```

خروجی

	id	name
1	2	Hadi
2	3	Ali

١١ توضیح خط آخر

آن مقداری از ستون id را برای با این شکر

پس سطر اول پاک کنم شود.

Select * from P;

Select * from S;

Select * from SP;

	sid	sname	scity
1	1	AMD	CA
2	2	ATI	LA
3	3	LMartn	CA
4	4	Sony	TO

	Pid	Pname	Pcolor
1	1	CPU	Silver
2	2	GPU	Black
3	3	RAM	Black
4	4	Tabil	white

: خروجی

	Pid	sid	qty
1	1	1	10
2	1	2	20
3	2	2	50
4	3	3	1...
5	1	2	2...

Select * from P;

Select * from S;

Select * from SP;

Select * from P left outer join SP on P.Pid = SP.Pid;

	Pid	Pname	Pcolor	Pid	sid	qty
1	1	CPU	Silver	1	1	10
2	1	CPU	Silver	1	2	20
3	1	CPU	Silver	1	2	200
4	2	GPU	Black	2	2	50
5	3	RAM	Black	3	3	100
6	4	Tabel	white	NULL	NULL	NULL

④ توضیح خط آخر

: تمام ستون ها هم از جدول P و هم از جدول SP نهایت راره شود.

: یعنی جدول اصلی که از آن سرچ می کنیم جدول P است.

: left outer join SP

یعنی من خواصیم که left outer join بین جدول P و جدول SP انجام بریم.

left join : یعنی همه ردیف جدول P حتی اگر صحیح (راده ای) مطابق در SP نراشته باشد، نهایت راره شوند.
اگر داره متناظری در جدول SP هیچ نشود، مقدار ستون های SP بایر NULL خواهد بود.

: این شرط اتصال (join condition) on P.Pid = SP.Pid است.

: یعنی وقتی مقدار ستون Pid در جدول P با مقدار Pid در جدول SP برابر باشد، دور دیگر به هم وصل می شوند.

Select * from P;

Select * from S;

Select * from SP;

Select * from P left outer join SP on pid = SP.Pid;

Select P.Pid, P.Pname, P.Pcolor, SP.QTY

from P left outer join SP on P.Pid = SP.Pid;

Select P.pid, P.pname, P.pcolor, SUM(SP.QTY) as 'QTY'

from P left outer join SP on P.Pid = SP.Pid

group by P.Pid, P.Pname, P.Pcolor;

insert into SP(Pid, sid, qty) values (3, 2, 20); ←

	Pid	Pname	Pcolor	QTY
1	1	CPU	Silver	230
2	2	GPU	Black	50
3	3	RAM	Black	120
4	4	Tablet	White	NULL

insert into SP(): يعنی من خواصیم که مسأله جدول (جوره) SP اضافه کنیم. ⑩

(Pid, Sid, Qty): اینهاستون های مخصوص که خواصیم بار آنها مقدار وار کنیم.

values (3, 2, 20): یعنی محصول با شناسه ۳ تا ۲۰ تعداد تسلیم نماییم که در باعث نمایم ۲ ب تعداد ۲۰

Select * from P where pid=1 not in (select pid from SP);

شکوه: اگر بجا خط شماره ۱ که با ← نشان داده شده است از کد بالا استفاده کنیم، خروجی برین مثل است:

	Pid	Pname	Pcolor
1	1	CPU	Silver

نمایش اطلاعات مخصوصی با $Pid=1$ شکوه در صورت که این شناسه در جدول SP وجود را شناخته باشد. ⑪

Select * from P: انتخاب همه ستون های جدول P

Where Pid=1: فقط رینگ کار با ۱ شناسه را در نظر میگیرد.

not in (Select pid from SP): شرط گذاشته شده که pid باشد در جدول SP وجود نداشته باشد.

دستورات SQL Join

① LEFT OUTER JOIN

② INNER JOIN

③ Right OUTER JOIN

① همه رکوردهای جدول سمت چپ را برمی‌گرداند، حتی اگر در جدول سمت راست مقابله‌ی جیزی نباشد.
اگر داره‌ای در سمت راست نباشد، مقدارش NULL می‌شود.
مثال: نمایش همه مشتری‌ها حتی اگر خرید نکرده باشند.

② فقط رکوردهایی را می‌آورد که در هر دو جدول داره‌ی مشترک داشته باشند (براساس شرط join).
اگر در یکی از جدول‌ها تطابق نباشد، آن را برمی‌نشاند و داره نمی‌شود.
مثال: فقط نمایش مشتری‌هایی که خرید انجام ندارند.

③ بر عکس مورداً ول. همه رکوردهای جدول سمت راست را برمی‌ورده، حتی اگر در سمت چپ جیزی نباشد.
اگر داره‌ای در سمت چپ نباشد، مقدارش NULL می‌شود.
مثال: نمایش همه سفارش‌هایی که اگر هنوز مشتری برایش عرض نشده باشد.

تمرین: هر سه وجوه بتوانید که فهرست تمامی محصولاتی که خرید نمایش نشده را نمایش دهد.

Select P.Pid, P.Pname, P.Pcolor, SUM(SP.qty) as 'QTY'
from P left outer join SP on P.Pid = SP.Pid
group by P.Pid, P.Pname, P.Pcolor;

تمرین: کوئی بنویسید که فهرست محصولاتی که خرید نشده را نمایش دهد

Select P.Pid, P.Pname, P.Pcolor
from P left outer join SP on P.Pid = SP.Pid
where SP.qty is NULL;