資料庫管理系統
第一次練習

匯入
<ex>以學生成績_new 為例，匯入
附加

建立資料表create table
CREATE TABLE Students(
sid char(4) NOT NULL
name varchar(12) NOT NULL
tel varchar(15)
birthday date
PRIMARY KEY (sid)
)

練習建立預設值

1.建立create default
CREATE default city_def as '台南市'

2.繫結

第二次練習

<ex>以northwind 為例，查詢員工的[員工編號],[姓名],[職稱],歲和年資
select [員工編號],[姓名],[職稱],歲=year(getdate())-year([出生日期]) ,year(getdate())-year([雇用日期]) 年資 from [dbo].[員工]

<ex>建立使用者自訂資料型別:address char(60)
sp_addtype address, 'char(60)'

<ex>以northwind 為例，再刪除參考主鍵或更新外來鍵時會導致違反參考原則，資料庫管理系統可能有三種處理方式

1.限制用法
alter table [產品資料]
add constraint fk_products_restrict
foreign key ([類別編號])
references [dbo].[產品類別]([類別編號])

2.連鎖法
alter table [產品資料]
add constraint fk_產品資料_cascade
foreign key ([類別編號])
references [dbo].[產品類別]([類別編號])
on delete cascade

3.空值化法
alter table [產品資料]
add constraint fk_產品資料_null

```
foreign key ([類別編號])
references [dbo].[產品類別]([類別編號])
on delete set null
```

查詢 產品編號 53(鹽水鴨)

```
select *
from [dbo].[訂貨明細],[dbo].[產品資料]
where [dbo].[訂貨明細].產品編號=[dbo].[產品資料].產品編號
and [dbo].[訂貨明細].[產品編號]=53
```

因其不再販售 而刪除鹽水鴨

```
delete [dbo].[產品資料]
where [產品編號]=53
```

因為在訂貨明細中設定的外來鍵是限制不允許刪除

第三次練習

學校School關聯式資料庫綱要

```
Students ( sid, name, birthday, tel )
Parents ( sid, name, relationship )
Std-Address ( sid, address )
Results ( sid, e_no, grade, date )
Exams ( e_no, title, type, c_no )
Courses ( c_no, title, credits )
Instructors ( eid, department, rank, SSN )
Employees ( SSN, name, city, street, tel )
Classes ( eid, sid, c_no, time, room, grade )
```

Create database

```
create database s3a632093
```

```
use s3a632093
```

Create table

本範例將示範已建立在s3a632093資料庫中的三個資料表(job、employee及publishers)的完整資料表定義，包含所有的條件約束定義

建立jobs

```
CREATE TABLE jobs
(
  job_id  smallint not null
   IDENTITY(1,1)
     PRIMARY KEY,
  job_desc       varchar(50)     NOT NULL
     DEFAULT 'New Position - title not formalized yet',
  min_lvl tinyint NOT NULL
     CHECK (min_lvl >= 10),
  max_lvl tinyint NOT NULL
       check (max_lvl >= 100)
)
```

匯入資料

輸入資料庫pubs中的jobs

```
set IDENTITY_INSERT jobs on
go
insert jobs (job_id, job_desc, min_lvl, max_lvl) select * from pubs..jobs
```

建立publishers

```
CREATE TABLE publishers
(
  pub_id  char(4) NOT NULL
      CONSTRAINT UPKCL_pubind PRIMARY KEY
      CHECK (pub_id IN ('1389', '0736', '0877', '1622', '1756')
        OR pub_id LIKE '99[0-9][0-9]'),
  pub_name      varchar(40)     NULL,
  city        varchar(20)     NULL,
  state     char(2) NULL,
  country     varchar(30)     NULL
        DEFAULT('USA')
)
```

匯入資料

```
 insert into publishers select  *from pubs.. publishers
```

建立employee

```
sp_addtype empid, 'char(9)','null'

CREATE TABLE employee
(
  emp_id  empid
    CONSTRAINT PK_emp_id PRIMARY KEY
    CONSTRAINT CK_emp_id CHECK (emp_id LIKE
      '[A-Z][A-Z][A-Z][1-9][0-9][0-9][0-9][0-9][FM]' or
      emp_id LIKE '[A-Z]-[A-Z][1-9][0-9][0-9][0-9][0-9][FM]'),
    /* Each employee ID consists of three characters that
    represent the employee's initials, followed by a five
    digit number ranging from 10000 through 99999 and then the
    employee's gender (M or F). A (hyphen) - is acceptable
    for the middle initial. */
  fname   varchar(20)     NOT NULL,
  minit   char(1) NULL,
  lname   varchar(30)     NOT NULL,
  job_id  smallint        NOT NULL
    DEFAULT 1
    /* Entry job_id for new hires. */
    REFERENCES jobs(job_id),
  job_lvl tinyint
    DEFAULT 10,
    /* Entry job_lvl for new hires. */
  pub_id  char(4) NOT NULL
```

```
    DEFAULT ('9952')
    REFERENCES publishers(pub_id),
    /* By default, the Parent Company Publisher is the company
    to whom each employee reports. */
  hire_date      datetime       NOT NULL
    DEFAULT (getdate())
    /* By default, the current system date is entered. */
)
```

匯入資料

```
insert into  employee select  *from pubs..  employee
```

在employee中練習輸入新進員工
手動鍵入

insert into輸入

```
insert into  employee([emp_id],[fname],[lname]) values ('AAA54321M','王','翔')
```

建立計算欄位資料表

```
CREATE TABLE mytable
  (
   [low] int,
   high int,
   myavg AS ([low] + high)/2
  )
```

ER_Diagram

建立計算欄位資料表

```
 CREATE TABLE 估價
 ( 編號 int  IDENTITY,
   單價 money,
   數量 int,
   總價 AS 單價 * 數量
 )
```

Alter table
欄位
改變前

移除主鍵

```
 ALTER TABLE [dbo].[publishers]
 drop [UPKCL_pubind]
```

增加主鍵

```
ALTER TABLE dbo.publishers
ADD PRIMARY KEY (pub_id)
```

jobs中job_id

修改前
```
 ALTER TABLE [dbo].[jobs]
 drop [PK__jobs__6E32B6A58D0E0F1C]
```

修改後
```
ALTER TABLE [dbo].[jobs]
ADD PRIMARY KEY (job_id)
```

-- 完整的資料表定義

--本範例將展示已建立在 pubs 資料庫中的三個資料表 (jobs、employee、及 publishers) 的完整資料表定義，包含所有的條件約束定義。

```
/* ************************ jobs table ************************ */
CREATE TABLE jobs
(
  job_id  smallint not null
   IDENTITY(1,1)
     PRIMARY KEY,
  job_desc       varchar(50)     NOT NULL
     DEFAULT 'New Position - title not formalized yet',
  min_lvl tinyint NOT NULL
     CHECK (min_lvl >= 10),
  max_lvl tinyint NOT NULL
      check (max_lvl > min_lvl)
)


create rule r2 as
/* ***************** publishers table ******************* */
CREATE TABLE publishers
(
  pub_id  char(4) NOT NULL
       CONSTRAINT UPKCL_pubind PRIMARY KEY
       CHECK (pub_id IN ('1389', '0736', '0877', '1622', '1756')
         OR pub_id LIKE '99[0-9][0-9]'),
  pub_name      varchar(40)     NULL,
  city        varchar(20)     NULL,
  state      char(2) NULL,
  country      varchar(30)     NULL
        DEFAULT('USA')
)

/* ************************ employee table ************************ */
CREATE TABLE employee
(
  emp_id  empid
     CONSTRAINT PK_emp_id PRIMARY KEY
     CONSTRAINT CK_emp_id CHECK (emp_id LIKE
       '[A-Z][A-Z][A-Z][1-9][0-9][0-9][0-9][0-9][FM]' or
       emp_id LIKE '[A-Z]-[A-Z][1-9][0-9][0-9][0-9][0-9][FM]'),
     /* Each employee ID consists of three characters that
     represent the employee's initials, followed by a five
     digit number ranging from 10000 through 99999 and then the
```

```
    employee's gender (M or F). A (hyphen) - is acceptable
    for the middle initial. */
  fname   varchar(20)    NOT NULL,
  minit   char(1) NULL,
  lname   varchar(30)    NOT NULL,
  job_id  smallint       NOT NULL
    DEFAULT 1
    /* Entry job_id for new hires. */
    REFERENCES jobs(job_id),
  job_lvl tinyint
    DEFAULT 10,
    /* Entry job_lvl for new hires. */
  pub_id  char(4) NOT NULL
    DEFAULT ('9952')
    REFERENCES publishers(pub_id),
    /* By default, the Parent Company Publisher is the company
    to whom each employee reports. */
  hire_date       datetime       NOT NULL
    DEFAULT (getdate())
    /* By default, the current system date is entered. */
)


-- 使用運算式定義計算資料行
-- 本範例將說明使用運算式 ((low + high)/2) 計算出 myavg 計算資料行。

CREATE TABLE mytable
  (
   [low] int,
   high int,
   myavg AS ([low] + high)/2
  )

-- 建立資料表
CREATE TABLE 估價
 ( 編號 int  IDENTITY,
   單價 numeric(5,1),
   數量 int,
   總價 AS 單價 * 數量
 )

-- 插入資料

INSERT 估價 VALUES (21.5, 8)
INSERT 估價 VALUES (12, 3)
INSERT 估價 VALUES (6, 8)
INSERT 估價 VALUES (4, 3)
SELECT * FROM 估價

--增加資料型態
sp_addtype type_of_price, 'smallmoney', 'null'

--增加一個聯絡人欄位
ALTER TABLE bookstores
```

```
  add  聯絡人 varchar(30)  NULL

-- 改變欄位資料型態
ALTER TABLE bookstores
  ALTER COLUMN 聯絡人
        varchar(25)
        NULL

-- 建立id,no為主鍵
ALTER TABLE orders
    WITH CHECK ADD
    CONSTRAINT PK_id_no
    PRIMARY KEY (id,no)

-- 改變欄位-刪除聯絡人欄位
ALTER TABLE bookstores
   DROP COLUMN 聯絡人

-- 取消外來鍵限制的檢查
ALTER TABLE orders
   NOCHECK CONSTRAINT FK_Orders_books
-- 啟動外來鍵限制的檢查
ALTER TABLE orders
   CHECK CONSTRAINT FK_Orders_books

ALTER TABLE orders
   CHECK CONSTRAINT FK_Orders_bookstores

-- 改變表格名稱
EXEC sp_rename 'pubs', 'publishers'

--改變表格欄位名稱
EXEC sp_rename 'publishers.address' , '通訊處', 'COLUMN'

--改變資料庫名稱
EXEC sp_rename 'tsao' , 'ttt', 'database'

-- 殺資料
select * from  估價
truncate TABLE 估價

-- 殺資料和結構
drop TABLE 估價
select * from 估價

 use learn
ALTER TABLE 選課資料表 ADD
/* Add a column referencing another column in the same table. */
CONSTRAINT fk_a 學號 REFERENCES 學生資料表(學號)

ALTER TABLE orders
  ADD   CONSTRAINT FK_a   FOREIGN KEY(no)      REFERENCES bookstores(no) ,
          CONSTRAINT FK_b   FOREIGN KEY(id)       REFERENCES books(id)
```

```
SET IDENTITY_INSERT jobs Off
SET IDENTITY_INSERT jobs ON

insert into jobs select * from pubs..jobs
insert into publishers select * from pubs..publishers
insert into employee select * from pubs..employee

select * into #temp from dbo.publishers
select * from #temp

Create Default My_default As 'Mr.'
Sp_bindefault my_default,'bookstores.name'
sp_unbindefault 'bookstores.name'
DROP DEFAULT my_default

DROP DEFAULT ee,kk

use bob

Create Rule My_rule
As
@id like '[A-Z][1-2][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9]'

create rule rr
as @qty between 300 and 500

Sp_bindrule my_rule,'employee.emp_id'


sp_addtype type_of_name, 'char(8)','null'

sp_droptype type_of_price


-- 改變資料表以加入新資料行
--本範例會新增允許 Null 值的資料行，且沒有透過 DEFAULT 定義所提供的值。各資料列在新資料行中具有 NULL。

CREATE TABLE doc_exa ( column_a INT)
GO
ALTER TABLE doc_exa ADD column_b VARCHAR(20) NULL
ALTER TABLE doc_exa ALTER column_b VARCHAR(10) NULL



-- 改變資料表以卸除資料行
--本範例會修改資料表以移除資料行。

CREATE TABLE doc_exb ( column_a INT, column_b VARCHAR(20) NULL)
GO
ALTER TABLE doc_exb DROP COLUMN column_b


CREATE TABLE dbo.mytable
( low int, high int, myavg AS (low + high)/2 ) ;
```

```
--設定外來鍵
CREATE TABLE ORDERS
(Order_ID integer,
Order_Date date,
Customer_SID integer,
Amount double,
Primary Key (Order_ID),
Foreign Key (Customer_SID) references CUSTOMER(SID));

--計算 保存期限
CREATE TABLE 估價
 ( 編號 int  IDENTITY,
   單價 numeric(5,1),
   數量 int,
  日期 datetime,
   總價 AS 單價 * 數量,
   保存期限  as year(getdate())-year(日期)
 )
```

What is a foreign key?
A foreign key means that values in one table must also appear in another table.

The referenced table is called the parent table while the table with the foreign key is called the child table. The foreign key in the child table will generally reference a primary key in the parent table.

A foreign key with a cascade delete means that if a record in the parent table is deleted, then the corresponding records in the child table with automatically be deleted. This is called a cascade delete.

A foreign key with a cascade delete can be defined in either a CREATE TABLE statement or an ALTER TABLE statement.

```
--Using a CREATE TABLE statement
--The syntax for creating a foreign key using a CREATE TABLE statement is:

CREATE TABLE table_name
(column1 datatype null/not null,
column2 datatype null/not null,
...
CONSTRAINT fk_column
  FOREIGN KEY (column1, column2, ... column_n)
  REFERENCES parent_table (column1, column2, ... column_n)
  ON DELETE CASCADE
);
```

For example:

```
CREATE TABLE supplier
( supplier_id numeric(10) not null,
 supplier_name varchar2(50) not null,
```

```
 contact_name varchar2(50),
 CONSTRAINT supplier_pk PRIMARY KEY (supplier_id)
);



CREATE TABLE products
( product_id numeric(10) not null,
 supplier_id numeric(10) not null,
 CONSTRAINT fk_supplier
   FOREIGN KEY (supplier_id)
  REFERENCES supplier(supplier_id)
  ON DELETE CASCADE
);
```

In this example, we've created a primary key on the supplier table called supplier_pk. It consists of only one field - the supplier_id field. Then we've created a foreign key called fk_supplier on the products table that references the supplier table based on the supplier_id field.

Because of the cascade delete, when a record in the supplier table is deleted, all records in the products table will also be deleted that have the same supplier_id value.

We could also create a foreign key (with a cascade delete) with more than one field as in the example below:

```
--拒絕刪除(Restrict)
CREATE TABLE supplier
( supplier_id numeric(10) not null,
 supplier_name char(50) not null,
 contact_name char(50),
 CONSTRAINT supplier_pk PRIMARY KEY (supplier_id)
);

CREATE TABLE products
( product_id numeric(10) not null,
 supplier_id numeric(10) not null,
 supplier_name char(50) not null,
 CONSTRAINT fk_supplier_comp
   FOREIGN KEY (supplier_id)
  REFERENCES supplier(supplier_id)
);

--連鎖性刪除(Cascade)
```

In this example, our foreign key called fk_foreign_comp references the supplier table based on two fields - the supplier_id and supplier_name fields.

The cascade delete on the foreign key called fk_foreign_comp causes all corresponding records in the products table to be cascade deleted when a record in the supplier table is deleted, based on supplier_id and supplier_name.

```
--Using an ALTER TABLE statement
--The syntax for creating a foreign key in an ALTER TABLE statement is:

ALTER TABLE table_name
add CONSTRAINT constraint_name
  FOREIGN KEY (column1, column2, ... column_n)
  REFERENCES parent_table (column1, column2, ... column_n)
  ON DELETE CASCADE;

--For example:

ALTER TABLE products
add CONSTRAINT fk_supplier
  FOREIGN KEY (supplier_id)
  REFERENCES supplier(supplier_id)
  ON DELETE CASCADE;

--In this example, we've created a foreign key (with a cascade delete) called fk_supplier that
references the supplier table based on the supplier_id field.




--設成空值
ALTER TABLE products
add CONSTRAINT fk_supplier
  FOREIGN KEY (supplier_id)
  REFERENCES supplier(supplier_id)
on delete set null;

We could also create a foreign key (with a cascade delete) with more than one field as in the
example below:

ALTER TABLE products
add CONSTRAINT fk_supplier
  FOREIGN KEY (supplier_id, supplier_name)
  REFERENCES supplier(supplier_id, supplier_name)
  ON DELETE CASCADE;

use pubs

--暫存資料表
select * into #temp from dbo.publishers
select * from #temp

--將關聯員工的屬性，員工編號、姓名、職稱和地址，在輸出時，更改為編號、員工姓名、職務和通
訊地址，在關聯代數的表示式
select emp_id as 員工編號,fname as 名 ,姓=lname,到職日=hire_date from dbo.employee



sp_addtype 'address', 'char (35)', 'null'

Create Rule r1 As  @odate <= getdate()

Sp_bindrule r1,'訂單.訂單日期'
```