

Q & A (vol.4)

분리문자 없이 날짜 데이터 출력

날짜를 출력할 때 "1998/01/01"이나 "1998.01.01"처럼 분리문자가 꼭 생기는 것 같습니다. Informix에서 날짜 데이터를 출력할 때 분리문자 없이 "19980101"이나 "980101"의 형식으로 출력할 수 있습니까? 있다면 그 방법은 무엇인가요? .

A Informix에서 DATE 형태의 데이터는 정수로 저장됩니다. 이 정수 데이터를 DBDATE라는 환경변수의 내용에 따라 단지 출력형태만을 결정하는 것입니다. 특별히 DBDATE 변수에 값을 지정하지 않았을 경우에는 "MDY4/"의 형식, 즉 "01/15/1998"의 형식이 기본입니다. 변경을 원한다면 DBDATE의 환경변수를 setting하여 바꿀 수 있습니다.

Korn shell 의 경우

```
$) export DBDATE=y4md.
```

C shell 의 경우

```
$) setenv DBDATE y4md.
```

Bourne shell 의 경우(Korn shell 도 가능)

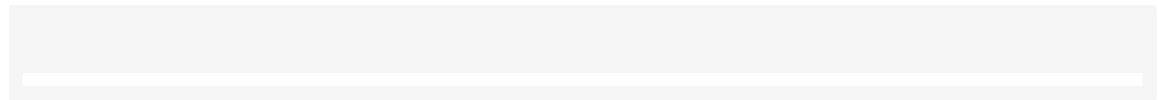
```
$) DBDATE=y4md.
```

```
$) export DBDATE
```

위의 형식으로 setting할 수 있으며, 대소문자는 구별하지 않습니다.위와 같이 DBDATE 환경 변수를 지정하면 "1998.01.15"의 형식으로 출력됩니다.

이와 마찬가지로 분리문자를 출력하지 않는 것도 역시 DBDATE 환경 변수를 조정함으로써 가능합니다.

이 경우, DBDATE를 "Y4MD0"으로 설정하십시오.



위와 같이 setting하면 각각 "19980115"와 "980115"의 형식으로 출력될 것입니다. 여기서 '0(zero)'은 분리문자로서 아무 것도 출력하지 않겠다는 의미입니다.

mirroring없이 사용하던 OnLine 시스템에서 mirroring을 필요로 하는 dbspace 추가

I처음에 mirroring 없이 사용하던 OnLine 시스템에서 mirroring을 필요로 하는 dbspace를 추가하려 합니다. Q2 여느 dbspace처럼 추가하려 했으나 mirror chunk는 추가할 수가 없었습니다.OnLine 초기화시에 mirroring 옵션을 주지 않은 경우에는 mirror chunk를 추가할 수 없습니까? 할 수 있다면 어떤 단계로 가능한가요?

```

MIRROR          1          # Mirroring flag (Yes = 1, No = 0)

MIRRORPATH&     # Path for device containing mirrored root

MIRROROFFSET    0          # Offset into mirrored device (Kbytes)

```

위의 내용이 \$InformixDIR/etc/\$ONCONFIG file의 MIRROR에 관한 부분입니다.

dbspace를 추가시키는 명령은 다음과 같습니다.

```

$) onspaces -c -d custdbs -p /dev/rdsk/d0s0t0 -o 500 -s

500000 -m /dev/rdsk/d1s0t0 1000

```

위 명령은 /dev/rdsk/d0s0t0에 500KBytes의 offset을 주고 500,000KBytes size의 primary chunk와 /dev/rdsk/d1s0t0에 1000KBytes 의 offset을 준 mirror chunk를 가진 custdbs라는 dbspace를 추가시켜 줍니다. 기본적으로 mirror chunk는 primary chunk와 같은 크기로 생성됩니다.

dbspace의 추가는 onmonitor를 사용하여서도 가능합니다.

onmonitor의 Dbspaces 메뉴의 Create 옵션에서 추가시킬 수 있습니다. 다음 화면을 참고하십시오.

```

Press ESC to build a new DBspace.Press Interrupt to cancel the option and return to the Dbspaces
menu. Press F2 or CTRL-F for field level help. CREATE DBSPACE DbSpace Name [custdbs ]
Mirror [Y] Temp [N] PRIMARY CHUNK INFORMATION: Full Pathname [/dev/rdsk/d0s0t0 ] Offset
[ 500] Kbytes Size [ 500000] Kbytes MIRROR CHUNK INFORMATION: Full Pathname
[/dev/rdsk/d1s0t0 ] Offset [ 500] Kbytes

```

sql 구문을 prompt line에서 실행하는방법 또는 sql 문장을
file로 저장했다가 prompt line에서 실행할 수 있는 방법

SQL 구문을 dbaccess나 isql을 띄우지 않고 prompt line에서 실행하는 방법이 있습니까?

Q₃

AK Alaska

HI Hawaii

CA California

OR Oregon

WA Washington

ID Idaho

NV Nevada

MT Montana

WY Wyoming

IN Indiana

MI Michigan

OH Ohio

PA Pennsylvania

DE Delaware

NJ New Jersey

NY New York

CT Connecticut

VT Vermont

RI Rhode Island

MA Massachusetts

NH New Hampshire

ME Maine

PR Puerto Rico

KR 한국

24 row(s) retrieved.

```
> select customer_num, fname, lname from customer where
```

```
customer_num >120;
```

```
customer_num fname      lname
```

```
121 Jason      Wallack
```

```
122 Cathy      O'Brian
```

```
123 Marvin      Hanlon
```

```
124 Chris      Putnum
```

```
125 James      Henry
```

```
126 Eileen     Neelie
```

```
127 Kim        Satifer
```

```
128 Frank      Lessor
```

```
8 row(s) retrieved.
```



```
> insert into state values ("BK","Bangkok");
```

```
1 row(s) inserted.
```

```
> update customer set (fname,lname)=("Boyoung", "Rhee")
```

```
where customer_num=101;
```

```
1 row(s) updated.
```

```
> delete from state where code="AB";
```

```
1 row(s) deleted.
```

```
> close database;
```

```
Database closed.
```

또한 file로 저장된 sql 문을 실행할 때는 다음과 같이 사용할 수 있습니다.

이 예는 /tmp/Informix/a.sql이 stores7 데이터베이스를 사용하는 질의어일 경우입니다.

```
k400:/post01/byrhee> dbaccess stores7 /tmp/Informix/a.sql
```

Sysmaster 데이터베이스를 이용한 DB size 계산

Informix 7.x를 사용하는 user입니다. OnLine을 처음 초기화할 때 sysmaster라는 DB가 생성되고 이 DB에는 OnLine이 관리하는 각종 정보가 들어간다고 알고 있습니다. 이 DB를 이용하여 특정 DB의

Q4 size를 구할 수 있습니까?

```
database sysmaster;
```

```
Database selected.
```

```
> select sum(size) from sysextents where dbsname="bluesky";
```

```
(sum)
```

```
5245
```

```
1 row(s) retrieved.
```

```
>
```

또한 다음의 shell script를 작성해 놓으면 편리하게 사용할 수 있을 것입니다.

```
#!/bin/sh PATH=/usr/ucb:$PATH export PATH #clear echo "Database Name : " read DB if test $DB  
then echo echo Executing $DB DB size .... else continue fi dbaccess - - > /dev/null 2>&1 << !>  
/tmp/tmpcheck database sysmaster; set isolation to dirty read; select "SIZE : " || sum(size)||  
PAGES" from sysextents where dbsname = "$DB";! echo cat /tmp/tmpcheck | grep SIZE echo Wrm  
/tmp/tmpcheck 2> &- 이 script를 실행하면 다음과 같습니다. k400:/post01/byrhee/sh> dbsize7  
Database Name : bluesky Executing bluesky DB size .... SIZE : 5245 PAGES  
k400:/post01/byrhee/sh>
```

SQL에서 Jullian Date를 select할 수 있습니까?

Q5

로 나타내는 것입니다.

```
CREATE PROCEDURE julian_date (d DATE)
```

```
    RETURNING INTEGER;
```

```
    RETURN (YEAR(d)*1000) + (d -MDY(1,1,YEAR(d))) + 1;
```

```
END PROCEDURE;
```

```
SELECT order_date,julian_date(order_date) FROM j_test;
```

```
order_date (expression)
```

```
1996.01.15      1996015
```

```
1996.03.25      1996085
```

```
1996.07.29      1996211
```

1996.08.05 1996218

1996.11.10 1996315

1996.12.31 1996366

1997.02.28 1997059

1997.06.25 1997176

1997.12.31 1997365

두 번째에서는

1998 년 1 월 15 일 = 35809

1997 년 12 월 16 일 = 35779

로 표현해보고자 합니다.

```
CREATE PROCEDURE julian(ddd DATE)
```

```
    RETURNING INTEGER;>
```

```
    DEFINE j_date INTEGER;
```

```
    LET j_date=ddd;
```

```
    RETURN j_date;
```

```
END PROCEDURE;
```

```
SELECT order_date,julian(order_date) FROM j_test;
```

order_date (expression)

1996.01.15	35078
------------	-------

1996.03.25	35148
------------	-------

1996.07.29	35274
1996.08.05	35281
1996.11.10	35378
1996.12.31	35429
1997.02.28	35488
1997.06.25	35605
1997.12.31	35794

extent size를 계산하는 shell script

새로운 테이블을 생성할 때 미리 테이블의 크기를 예상하여 그 크기를 명시하게 되면 데이터를 효과적으로 저장할 수 있다고 들었습니다. 그 방법은 무엇이며, 테이블의 예상 크기를 어떻게 결정할 수 있습

Q6 니까?


```
CREATE TABLE cust
( cust_num serial not null,
    name char(10),
    phone char(14),
    addr1 char(30),
    addr2 char(30),
    zipcode char(7),
    memo char(50) )
IN dbspace1
EXTENT SIZE 500 NEXT SIZE 200
LOCK MODE ROW;
```

위 문장에서 주어진 각 extent 의 size 는 KBytes 단위입니다.

이처럼 테이블의 크기를 미리 예상할 수 있다면 extent size를 명시함으로써 테이블에 입력될 데이터가 저장될 수 있는 알맞은 공간을 미리 확보함으로써 보다 효과적으로 데이터 관리를 할 수 있을 것입니다. 이러한 extent size는 해당 테이블의 한 행의 크기와 입력될 행의 수와 또한 index 크기에 따라 달라질 수 있습니다.

