

c23ef898	----	5	Informix	-	0	0	0	0	0
	P--B								
c23efcd8	Y--	56	Informix	ttyp3	c25f3130	0	1	229	0
	P----								
c23f0118	Y--	55	Informix	ttyp7	c2498090	0	1	4	0
	P----								
c23f0558	----	8	Informix	-	0	0	0	0	0
	P--D								
c23f0998	Y--	89	Informix	ttypb	c251a680	0	1	1	0
	P----								
c23f1218	Y--	70	ychi	ttyq2	c24c4cd0	0	1	2	0
	P----								

8 active, 128 total, 17 maximum concurrent

K400:/post01/case/byrhee/test> onstat -g sql

Informix OnLine Version 7.24.UC7 -- On-Line -- Up 4

days 07:23:02 -- 17888 Kbytes

Sess	SQL	Current	Iso	Lock	SQL	ISAM	F.E.
Id	Stmt type	Database	Lvl	Mode	ERR	ERR	Vers
89	SELECT	bluesky	CR	Not Wait	0	0	7.20
70	-	bluesky	CR	Not Wait	0	0	7.20
56	SELECT	bluesky	CR	Not Wait	0	0	6.01
55	SELECT	bluesky	CR	Not Wait	0	0	7.20

K400:/post01/case/byrhee/test> onstat -g sql 55

Informix OnLine Version 7.24.UC7 -- On-Line -- Up 4

days 07:24:12 -- 17888 Kbytes

Sess	SQL	Current	Iso	Lock	SQL	ISAM	F.E.
Id	Stmt type	Database	Lvl	Mode	ERR	ERR	Vers
55	SELECT	bluesky	CR	Not Wait	0	0	7.20

Current statement name : i00043ff0_05bb9243

Current SQL statement :

```
select * from maint where mnt_invoice_num= 9900429
```

```
order by mnt_invoice_num
```

Last parsed SQL statement :

```
select * from maint where mnt_invoice_num= 9900429
```

```
order by mnt_invoice_num
```

위와 같은 사용자 정보는 sysmaster 데이터베이스를 이용해서도 확인할 수 있습니다.

다음의 질의어로 간단히 원하는 결과를 얻을 수도 있습니다.

```
database sysmaster;

select sysdatabases.name database,
       sysessions.username,
       sysessions.hostname,
       syslocks.owner sid

from syslocks, sysdatabases , outer sysessions

where syslocks.tabname = "sysdatabases"

and syslocks.rowidk = sysdatabases.rowid

and syslocks.owner = sysessions.sid

order by 1;
```

위 질문의 실행 결과는 다음과 같습니다.

database	username	hostname	sid
bluesky	Informix	kor-k400	56
bluesky	Informix	kor-k400	55
bluesky	Informix	kor-k400	89
bluesky	ychi	kor-k400	70
sysmaster	Informix	kor-k400	91

논리로그 파일 추가하기

발생할 수 있다고 들었는데, 그렇게 되지 않도록 하기 위해서 자주 backup을 하는 편이지만, 논리로그의 갯수나 크기를 변경하여 논리로그가 full 되는 시간을 늦추고 싶습니다. 어떻게 변경할 수 있을까요?

A

데이터베이스에 변경 작업이 많으면 많을수록 로깅을 한 데이터베이스는 논리로그에 기록을 많이 남기게 되어 있습니다. 따라서 그만큼 빨리 논리로그가 full이 될 수 있습니다.

이런 경우 논리로그를 충분한 크기로 유지해야 하는데, 처음 초기화할 때 지정된 논리로그 크기와 갯수를 \$InformixDIR/etc/\$ONCONFIG 파일을 수정한다고 해서 변경되지는 않습니다. 논리로그는 onparams 명령이나 onmonitor 에서 변경할 수 있으며 반드시 대기모드(quiet mode)에서만 가능합니다. 즉, 실행모드(online mode)에서는 논리로그를 변경할 수 없으므로 user session 이 있는 경우에는 변경할 수 없습니다.

초기의 논리로그는 rootdbs 에 생성됩니다. 추가하거나 변경하는 논리로그는 rootdbs 또는 다른 dbspace 에도 가능합니다.

```
K400:/bigfs/byrhee/env> onstat -l
```

```
Informix Universal Server Version 9.14.UC4 -- On-Line --
```

```
Up 00:17:24 -- 8816 Kbytes
```

```
Physical Logging
```

Buffer	bufused	bufsize	numpages	numwrits	pages/io
--------	---------	---------	----------	----------	----------

Buffer	bufused	bufsize	numrecs	numpages	numwrits	recs/pages	pages/io
L-2	0	8	1	1	1	1.0	1.0
	Subsystem	numrecs	Log Space used				
	OLDRSAM	1	32				

address	number	flags	uniqid	begin	size	used	%used
c9058e74	1	U-B-----	4	100427	500	296	59.20
c9058e90	2	U----C-L	5	10061b	500	326	65.20
c9058eac	3	U-B-----	3	10080f	500	500	100.00

onstat -l 명령은 물리로그와 논리로그의 정보를 보여줍니다. 아래쪽의 “Logical Logging” 부분을 살펴보십시오. 이 시스템에서는 1MB(500 Page)짜리 3 개의 논리로그를 사용하고 있습니다. 현재는 실행모드(On-Line 모드)이므로 논리로그를 추가할 수 없습니다. 대기모드로 변경하고 논리로그를 추가해보도록 하겠습니다.

```
K400:/bigfs/byrhee/env> onmode -s
This will perform a GRACEFUL SHUTDOWN -
Do you wish to continue (y/n)? Y
K400:/bigfs/byrhee/env> onstat -
Informix Universal Server Version 9.14.UC4 -- Quiescent --

Up 00:20:58 -- 8816 Kbytes
```

대기모드(Quiescent 모드)임을 확인하십시오. 이제 onparams 명령으로 논리로그를 추가하겠습니다.

```
K400:/bigfs/byrhee/env> onparams -a -d rootdbs -s 2000  
Logical log successfully added.  
K400:/bigfs/byrhee/env> onparams -a -d dbpace1 -s 2000  
Logical log successfully added.
```

처음 명령으로는 그 크기가 2000KB (2KB page 시스템에서는 1000 page)인 논리로그 파일을 rootdbs 에 추가했고, 두 번째 명령으로는 dbpace1 에 추가했습니다. onstat -l 로 추가된 논리로그 파일을 확인할 수 있습니다.

```
K400:/bigfs/byrhee/env> onstat -l  
Informix Universal Server Version 9.14.UC4 -- Quiescent  
  
-- Up 00:21:42 -- 8816 Kbytes
```

Physical Logging

Buffer	bufused	bufsize	numpages	numwrits	pages/io
--------	---------	---------	----------	----------	----------

address	number	flags	uniqid	begin	size	used	%used
c9058e74	1	U-B-----	4	100427	500	296	59.20
c9058e90	2	U---C-L	5	10061b	500	335	67.00
c9058eac	3	U-B-----	3	10080f	500	500	100.00
c9058ec8	4	A-----	0	1013d5	1000	0	0.00
c9058ee4	5	A-----	0	1015c9	1000	0	0.00

새로운 논리로그 파일이 4 번, 5 번으로 추가되었고 그 flag 가 "A"임을 확인하십시오.

onparams 명령으로 논리로그는 추가했으나, "A"라는 flag 를 가지고 있는 파일은 사용되지 않습니다. 여기서 ontape 명령으로 backup 을 한 번 해야만 새로 추가한 논리로그를 사용할 수 있습니다. ontape 명령으로 backup 하기 전 \$InformixDIR/etc/\$ONCONFIG 의 TAPEDEV 값을 /dev/null 로 바꾸십시오. 이는 실제 backup 을 위한 작업이 아니라 단지 새로 추가한 논리로그의 flag 를 "A"에서 "F"로 변경하기 위한 것이기 때문입니다.

```
K400:/bigfs/byrhee/env> ontape -s
Archive to tape device '/dev/null' is complete.
Program over.
K400:/bigfs/byrhee/env> onstat -l
Informix Universal Server Version 9.14.UC4 -- Quiescent --

Up 00:22:23 -- 8816 Kbytes

Physical Logging
```

Buffer	bufused	bufsize	numpages	numwrits	pages/io
--------	---------	---------	----------	----------	----------

P-1	4	8	10	3	3.33		
phybegin	physize	phypos	phyused	%used			
10003f	1000	385	4	0.40			
Logical Logging							
Buffer	bufused	bufsize	numrecs	numpages	numwrits	recs/pages	pages/io
L-1	0	8	47	12	12	3.9	1.0
	Subsystem	numrecs	Log Space used				
	OLDRSAM	47	2016				
address	number	flags	uniqid	begin	size	used	%used
c9058e74	1	U-B-----	4	100427	500	296	59.20
c9058e90	2	U---C-L	5	10061b	500	337	67.40
c9058eac	3	U-B-----	3	10080f	500	500	100.00
c9058ec8	4	F-----	0	1013d5	1000	0	0.00
c9058ee4	5	F-----	0	1015c9	1000	0	0.00

새로 추가한 논리로그의 flag 가 "F"임을 확인하십시오. 이제 이 논리로그 파일도 사용가능합니다.

이렇게 작업을 끝내신 후에는 다시 실행모드로 바꾸시어 사용자들이 데이터베이스에 접속할 수 있도록 하시면 됩니다.

```
K400:/bigfs/byrhee/env> onmode -m
```

```
K400:/bigfs/byrhee/env> onstat -
```

-- Up 00:23:45 -- 8816 Kbytes

ESQL/C ; Multi Connection

ESQL/C 프로그램을 작성하려 하는데, 두개의 DB Server에 CONNECT해야 할 필요가 있습니다. multi connection이 가능한가요?

Q₃

```
$char sname[15 + 1];

printf("Sample ESQL Program running.\n");

$connect to 'stores7@byrhee_731t' as 'con_1';

if (SQLCODE != 0) {

    printf("byrhee_731t DB connect error(%d)\n",

        sqlca.sqlcode);

    exit(1);

}

printf(" byrhee_731t DB connect success(%d)\n",

    sqlca.sqlcode);

$connect to 'stores7@maserv_tcp' as 'con_2';
```

```
if (SQLCODE != 0) {  
  
    printf("maserv_tcp DB connect error(%d)\n",  
  
        sqlca.sqlcode);  
  
    exit(1);  
  
}
```

```
printf("maserv_tcp DB connect success(%d)\n",  
  
    sqlca.sqlcode);
```

```
$declare sel_cursor cursor for
```

```
select code, sname into $code, $sname  
  
from state;
```

```
if (SQLCODE != 0) {
```

```
    printf("declare cursor error(%d)\n", sqlca.sqlcode);
```

```
        exit(1);

    }

    printf("declare cursor success(%d)\n", sqlca.sqlcode);

$open sel_cursor;

    if (SQLCODE != 0) {

        printf("cursor open error(%d)\n", sqlca.sqlcode);

        exit(1);

    }

    printf("cursor open success(%d)\n", sqlca.sqlcode);

    for (;;)

    {
```

```
$set connection 'con_2';

if (SQLCODE != 0) {

    printf("set connection maserv_tcp error (%d)\n",

        sqlca.sqlcode);

    exit(1);

}

printf("set connection maserv_tcp success (%d) :
SELECT\n",

    sqlca.sqlcode);

$fetch sel_cursor;

if (SQLCODE == SQLNOTFOUND)

    break;
```

```
$set connection 'con_1';

if (SQLCODE != 0) {

    printf("set connection byrhee_731t error (%d)\n",

        sqlca.sqlcode);

    exit(1);

}

printf("set connection byrhee_731t success (%d) :

    INSERT \n", sqlca.sqlcode);

$insert into state values ($code, $sname);

if (SQLCODE != 0) {

    printf("insert error(%d)\n", sqlca.sqlcode);

    exit(1);
```

```
    }

    printf("insert success(%d)\n", sqlca.sqlcode);

    i=i+1;

    } /* end for-loop */

    printf("%d 행이 insert 되었습니다.\n",i);

    $close sel_cursor;

    if (SQLCODE != 0) {

        printf("close sel_cursor error(%d)\n",
sqlca.sqlcode);

        exit(1);

    }

    printf("sel_cursor closed(%d)\n", sqlca.sqlcode);
```



```
$free sel_cursor;

if (SQLCODE != 0) {

    printf("free sel_cursor error(%d)\n", sqlca.sqlcode);

    exit(1);

}

printf("sel_cursor freed(%d)\n", sqlca.sqlcode);

$disconnect 'con_1';

if (SQLCODE != 0) {

    printf("disconnect byrhee_731t error(%d)\n",

        sqlca.sqlcode);

    exit(1);

}
```

```
    printf(" byrhee_731t disconnected (%d)\n",
sqlca.sqlcode);

$disconnect 'con_2';

if (SQLCODE != 0) {

    printf("disconnect maserv_tcp error(%d)\n",

        sqlca.sqlcode);

    exit(1);

}

printf(" maserv_tcp disconnected (%d)\n",
sqlca.sqlcode);

printf("\nSample Program over.\n");
```

```
} /* end of main() */
```

한 가지 주의할 점은 두 서버간의 환경변수 중 한 쪽 서버쪽에만 필요한 환경변수가 있다면 그 환경변수를 setting 해주고 unset 하는 작업이 꼭 필요하다는 점입니다. 다음과 같이 환경변수를 설정하는 내용으로 function 을 만들어 이 function 을 필요한 시점에 호출하여 실행하도록 하는 것이 좋습니다. 아래의 내용처럼 특히 ONCONFIG, LOCALE 변수와 InformixSQLHOSTS 변수를 주의하십시오.

```
to_env1()  
  
{  
  
    putenv("InformixSERVER=byrhee_731t");  
  
    putenv("InformixDIR=/CS1/IDS731UC");  
  
    putenv("ONCONFIG=onconfig.byrhee");  
  
    putenv("InformixSQLHOSTS=/CS1/byrhee/sqlhosts");  
  
    putenv("DB_LOCALE=en_us.8859-1");  
  
    putenv("SERVER_LOCALE=en_us.8859-1");  
  
    putenv("CLIENT_LOCALE=en_us.8859-1");  
  
}
```

```

to_env2()

{

    putenv("InformixSERVER=maserv_tcp");

    putenv("InformixDIR=/post01/Product/ON724UC1");

    putenv("ONCONFIG=onconfig.csdb");

    putenv("InformixSQLHOSTS=

/post01/Product/ON724UC1/etc/sqlhosts");

    putenv("DB_LOCALE=ko_kr.ksc");

    putenv("SERVER_LOCALE=ko_kr.ksc");

    putenv("CLIENT_LOCALE=ko_kr.ksc");

}

```

CHAR 자료형으로 저장된 칼럼의 계산

CHAR(5)로 정의된 칼럼이 있습니다. 이 칼럼의 다섯 자리가 각각 의미가 있어서 substring으로 사용되어야 하기 때문에 CHAR로 저장했습니다. 그런데 이 칼럼이

때때로 계산을 필요로 합니다. CHAR 형이기 때문에 계산을 할 수 없는데, 계산할 수

Q4 있는 방법이 있습니까?

A

내장 프로시저(Stored Procedure)를 사용하여 처리할 수 있습니다.

다음의 예를 참고하십시오.

```
create table test(no char(3),num char(5));
```

위와 같이 데이터가 저장되어 있을 때 아래의 질의는 다음과 같이 오류 367 번을 만납니다. 계산할 수 없는 칼럼을 계산에 사용했기 때문입니다.

```
select no, sum(num), avg(num) from test
```

```
group by no;
```

```
367: Sums and averages cannot be computed for
```

```
character columns.
```

위와 같은 질의를 다음의 내장 프로시저를 이용하여 처리하면 간단히 원하는 결과를 얻을 수 있습니다.

```
create procedure chartoint(val char(12))
```

```
insert into test values("333","40000");

insert into test values("333","");
```

위와 같이 데이터가 저장되어 있을 때 아래의 질의는 다음과 같이 오류 367 번을 만납니다. 계산할 수 없는 칼럼을 계산에 사용했기 때문입니다.

```
select no, sum(num), avg(num) from test

group by no;

367: Sums and averages cannot be computed for

character columns.
```

위와 같은 질의를 다음의 내장 프로시저를 이용하여 처리하면 간단히 원하는 결과를 얻을 수 있습니다.

```
create procedure chartoint(val char(12))

returning int;

if (val is null) or (val = "") then
```

```
        return 0;  
  
    else  
  
        return val;  
  
    end if;  
  
end procedure;
```

```
select no, sum(chartoint(num)),avg(chartoint(num))  
  
from test
```

```
222    60000 15000.0000000000
```

```
333    100000 20000.0000000000
```

CHAR 칼럼에 New Line을 입력하는 방법: 버전 9.1x

9.1 제품을 사용하는 User입니다. CHAR 칼럼에 New Line을 입력하려 합니다. Return Key를 받아들이는 방법이 있습니까?

Q₅


```
"Success is failure turned inside out.
```

```
The silver tint of the clouds of doubt,
```

```
so stick to the fight when you're hardest hit,
```

```
It's when things seem worst that you mustn't quit.");
```

```
282: Found a quote for which there is no matching quote.
```

위와 같이 `ifx_allow_newline()` 프로시저를 사용하지 않았을 때에는 282 번 오류가 발생합니다. 그러나 `ifx_allow_newline()` 프로시저를 실행한 후에 입력하게 되면 Return Key 를 받아들입니다.

```
execute procedure ifx_allow_newline('t');
```

```
insert into test values (
```

```
1,
```

```
so stick to the fight when you're hardest hit,
```

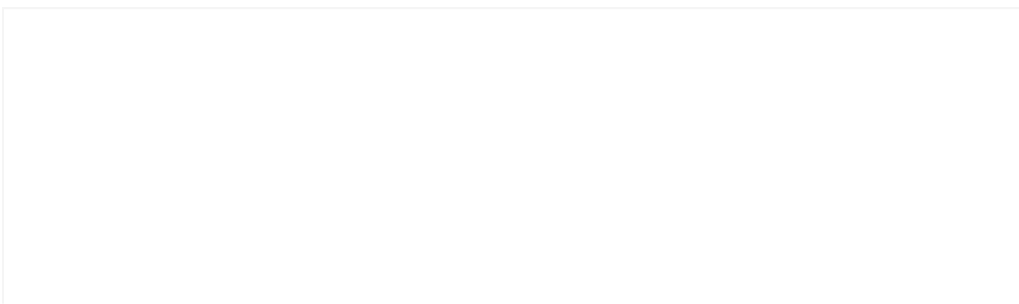
```
It's when things seem worst that you mustn't quit.");
```

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

DATE TIME 칼럼에서 특정 시간대만 찾아오기

DATE TIME으로 정의된 칼럼이 있습니다. 이 칼럼에서 특정 시간대 또는 특정 날짜에 해당하는 자료만 검색하는 방법을 알고 싶습니다.

Q6



```
where extend(d_time, hour to hour)="11";
```

```
no d_date    d_time
```

```
4 1999.11.01 1999-11-01 11:34:22
```

```
select * from d_time where
```

```
extend(d_time, year to day)="1999-11-01";
```

```
no d_date    d_time
```

```
4 1999.11.01 1999-11-01 11:34:22
```