

Mérési jegyzőkönyv – Adatbázisok Laboratórium

I. mérés: Oracle

Név:	Szabó Bence Farkas
Neptun kód:	RF57V5
Feladat kódja:	30-VASUT
Mérésvezető neve:	Csapó Tamás
Mérés időpontja:	2018-02-19 12:15
Mérés helyszíne:	HSZK N
Megoldott feladatok:	1,2,3,4,5,6
Elérhető pontszám (plusz pontok nélkül):	8,5p

Mérési feladatok megoldása

1. feladat: Jelszó megváltoztatása

A jelszót a connectios fül alatt az adatbázis kapcsolaton jobb egérrel kattintva a „reset passord” menüben lehet megváltoztatni.

Az új jelszó: 'bence1995'

2. feladat: Jogosultságok

Közvetlenül, mint felhasználó nem kapunk rendszer szintű jogokat. Az alábbi jogokat a HALLGATO_ROLE-on keresztül kapjuk meg: Objektum jogosultságok:

General Roles Sys Privs Object Privs Consumer Group P			
▼ Actions...			
	Object Privilege	Sc...	Object
1	SELECT	OKTATAS	CIM
2	SELECT	OKTATAS	EMBER
3	SELECT	OKTATAS	IGAZOLVANYOK
4	SELECT	OKTATAS	MNB_DEVIZA
5	SELECT	OKTATAS	SZEMELY
6	SELECT	OKTATAS	SZEMELYEK
7	SELECT	OKTATAS	SZEMELYEK2
8	SELECT	OKTATAS	VISITS

Rendszer jogosultságok:

General Roles Sys Privs Object Privs Consumer Group Privs		
▼ Actions...		
	System Privilege	Admin Option
1	ALTER SESSION	NO
2	CREATE DATABASE LINK	NO
3	CREATE SYNONYM	NO
4	CREATE VIEW	NO
5	SELECT ANY DICTIONARY	NO

További jogosultságokat az alábbi szerepeken keresztül kapunk:



General Roles Sys Privs Object Privs Consumer Group		
▼ Actions...		
	Role	Admin Option
1	CONNECT	NO
2	RESOURCE	NO
3	SELECT_CATALOG_ROLE	NO

Magyarázat

A jogosultságokat a DBA fül Security - Users menüpontjában tekinthetjük meg az adott felhasználóra kattintva. Itt a roles fülön látható, hogy a felhasználó milyen szerepet kapott, amelyen keresztül a jogokat kapja. Ezeket a jogokat a Security – Roles menüpontban találjuk az adott szerepre (HALLGATO_ROLE) kattintva (itt a system privileges és object privileges alatt, illetve láthatjuk, hogy még milyen szerepeket kapunk a még az adott szerepen keresztül).

3. feladat: fizikai struktúra

Details Datafiles Free Space Objects Usage Chart SQL

 Actions...

FILE_NAME	FILE_ID	TOTAL_BYTES	USED_BYTES	FREE_BYTES	BLOCKS	AUTOEXTENSIBLE	MAXBYTES	MAXBLOCKS	STATUS	FRAG_IDX
1 /oracle/oradata/SZGLAB/okt01.dbf	5	500	42,2	457,81	64000	NO	(null)	0	AVAILABLE	27,55
2 /oracle/oradata/SZGLAB/okt02.dbf	7	100	34	66	12800	YES	(null)	4194176	AVAILABLE	41,97

Magyarázat

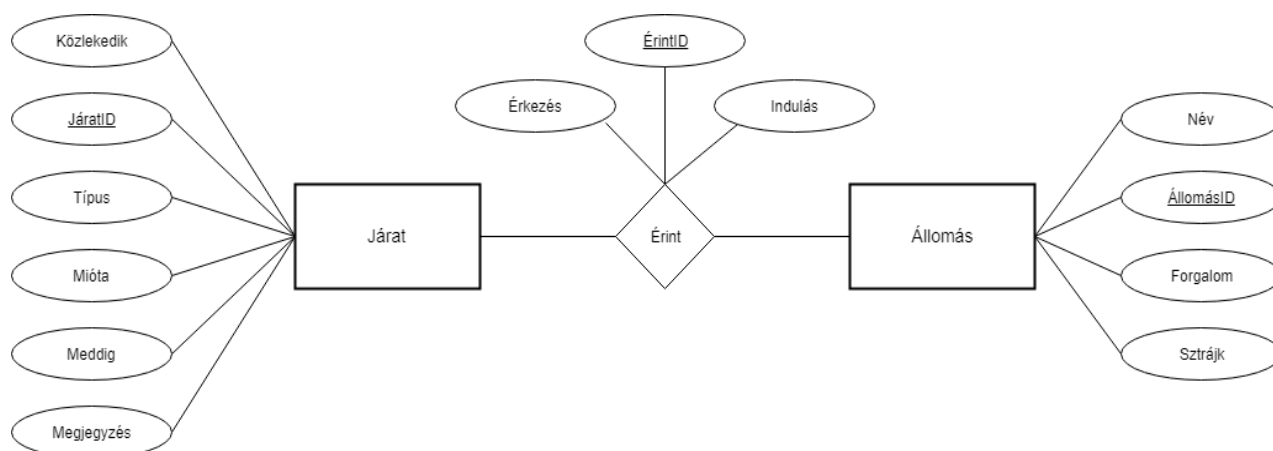
Az OKT táblahelyhez a fent látható adatfájlok tartoznak, melyeknek mérete (lefoglalt) a TOTAL_BYTES, ebből a USED_BYTES ami ténylegesen foglalt. A kihasználtság a USED_BYTES/ TOTAL_BYTES.

Név	Méret	Kihasználtság
okt01.dbf	500MB	8,4%
okt02.dbf	100MB	34%

*megfigyelés: Az SQL Developer a fájlok méretét Byte-nak írja, de ez a valóságban MegaByte (MB).

4. feladat: Modellezés

ER-diagramm:



Magyarázat

A feladathoz a fent látható ER diagramot terveztem meg. Ezen látható egy Járat és egy Állomás entitás, melyek között N:N kapcsolat áll fenn. Ez reprezentálja, hogy egy járat érint egy állomást. Értelem szerűen egy járat több állomáson is megállhat, illetve egy állomáson több járat is áthaladhat.

Az ALLOMAS tábla létrehozása

PK	Name	Data Type	Size	Not Null	Default	Comment
	ALLOMAS_ID	NUMBER		<input checked="" type="checkbox"/>		
	ALLOMAS_AZON	VARCHAR2	4	<input checked="" type="checkbox"/>		
	NEV	VARCHAR2	40	<input checked="" type="checkbox"/>		
	FORGALOM	NUMBER	4	<input checked="" type="checkbox"/>		
	SZTRAJK	NUMBER	4	<input checked="" type="checkbox"/>		

A tábla létrehozásnak SQL kódja:

Létrehozzuk az ALLOMAS táblát, ahol az ALLOMAS_ID egyedi szám és egyben a tábla kulcs, az ALLOMAS_AZON egy 4 karakter hosszú karakterlánc, a NEV egy 40 karakter hosszú karakterlánc, a FORGALOM és SZTRAJK pedig 4 jegyű számok. A táblában egyik attribútum sem lehet Null.

```
CREATE TABLE ALLOMAS
(
    ALLOMAS_ID NUMBER(5, 0) NOT NULL
, ALLOMAS_AZON VARCHAR2(4 CHAR) NOT NULL
, NEV VARCHAR2(40 CHAR) NOT NULL
, FORGALOM NUMBER(4, 0) NOT NULL
, SZTRAJK NUMBER(4, 0) NOT NULL
)
```

Beállítjuk az ALLOMAS_ID-t elsődleges kulcsként.

```
, CONSTRAINT ALLOMAS_PK PRIMARY KEY
(
    ALLOMAS_ID
)
USING INDEX
(
    CREATE UNIQUE INDEX ALLOMAS_PK ON ALLOMAS (ALLOMAS_ID ASC)
    LOGGING
    TABLESPACE OKT
    PCTFREE 10
    INITRANS 2
    STORAGE
    (
        BUFFER_POOL DEFAULT
    )
    NOPARALLEL
)
ENABLE
)
LOGGING
TABLESPACE OKT
PCTFREE 10
INITRANS 1
STORAGE
(
    BUFFER_POOL DEFAULT
)
NOCOMPRESS
NO INMEMORY
NOPARALLEL;
```

Az ALLOMAS_AZON beállítása egyedinek, illetve az alfanumerikus karakterek ellenőrzése REGEXP_LIKE kifejezéssel.

```
CREATE UNIQUE INDEX ALLOMAS_AZON ON ALLOMAS (ALLOMAS_AZON ASC)
LOGGING
TABLESPACE OKT
PCTFREE 10
INITRANS 2
STORAGE
(
    BUFFER_POOL DEFAULT
)
NOPARALLEL;

ALTER TABLE ALLOMAS
ADD CONSTRAINT ALLOMAS_AZON_CHK CHECK
(REGEXP_LIKE (ALLOMAS_AZON, '[0-9a-zA-Z]{4}'))
ENABLE;
```


Ellenőrizzük, hogy a forgalom 0 és 9000 közé esik e.

```
ALTER TABLE ALLOMAS
ADD CONSTRAINT ALLOMAS_FORGALOM_CHK CHECK
(FORGALOM >= 0 AND FORGALOM <= 9000)
ENABLE;
```

Ellenőrizzük, hogy a sztrájk esetén fogadható emberek száma nagyobb e mint a tényleges fogadóképesség 20%-a.

```
ALTER TABLE ALLOMAS
ADD CONSTRAINT ALLOMAS_SZTRAJK_CHK CHECK
(SZTRAJK >= (FORGALOM*0.2))
ENABLE;
```

A JARAT tábla létrehozása:

Columns: 🔍 name + - 📄						
PK	Name	Data Type	Size	Not Null	Default	Comment
	JARAT_ID	NUMBER		<input checked="" type="checkbox"/>		
	JARAT_AZON	NUMBER	4	<input checked="" type="checkbox"/>		
	TIPUS	VARCHAR2	2	<input checked="" type="checkbox"/>		
	MIOTA	DATE		<input type="checkbox"/>		
	MEDDIG	DATE		<input type="checkbox"/>		
	MEGJEGYZES	VARCHAR2	40	<input type="checkbox"/>		
	KOZLEKEDIK	VARCHAR2	7	<input checked="" type="checkbox"/>		

A tábla létrehozásnak SQL kódja:

Létrehozzuk az JARAT táblát, ahol az JARAT_ID egyedi szám és egyben a tábla elsődleges kulcsa, az JARAT_AZON egy 4 számjegyű egyedi szám, a TIPUS egy 2 karakter hosszú karakterlánc, a MIOTA és MEDDIG dátum típusú mezők, a MEGJEGYZES egy 40 karakter hosszú karakterlánc, a KOZLEKEDIK pedig egy 7 karakter hosszú karakterlánc. A táblában a MEGJEGYZES, a MIOTA és a MEDDIG attribútumok értéke lehet Null (üresen hagyható).

```
CREATE TABLE JARAT
(
    JARAT_ID NUMBER(5, 0) NOT NULL
, JARAT_AZON NUMBER(4, 0) NOT NULL
, TIPUS VARCHAR2(2 CHAR) NOT NULL
, MIOTA DATE
, MEDDIG DATE
, MEGJEGYZES VARCHAR2(40 CHAR)
, KOZLEKEDIK VARCHAR2(7 CHAR) NOT NULL
)
```

Beállítjuk az JARMU_ID-t elsődleges kulcsként.

```
, CONSTRAINT JARAT_PK PRIMARY KEY
(
    JARAT_ID
)
USING INDEX
(
    CREATE UNIQUE INDEX JARAT_PK ON JARAT (JARAT_ID ASC)
    LOGGING
    TABLESPACE OKT
    PCTFREE 10
    INITRANS 2
    STORAGE
    (
        BUFFER_POOL DEFAULT
    )
    NOPARALLEL
)
ENABLE
)
LOGGING
TABLESPACE OKT
PCTFREE 10
INITRANS 1
STORAGE
(
    BUFFER_POOL DEFAULT
)
NOCOMPRESS
NO INMEMORY
NOPARALLEL;
```

A JARAT_AZON beállítása egyedinek.

```
CREATE UNIQUE INDEX JARAT_AZON ON JARAT (JARAT_AZON ASC)
LOGGING
TABLESPACE OKT
PCTFREE 10
INITRANS 2
STORAGE
(
    BUFFER_POOL DEFAULT
)
NOPARALLEL;
```

Ellenőrizzük, hogy a MIOTA dátum kisebb legyen mint a MEDDIG dátum.

```
ALTER TABLE JARAT
ADD CONSTRAINT JARAT_DATE_CHK CHECK
(MIOTA < MEDDIG)
ENABLE;
```


Ellenőrizzük, hogy a KOZLEKEDIK (a járat mely napokon közlekedik), csak 7 karakter hosszú, 0-1 karakterekből álljon.

```
ALTER TABLE JARAT
ADD CONSTRAINT JARAT_KOZL_CHK CHECK
(REGEXP_LIKE(KOZLEKEDIK, '[0-1]{7}'))
ENABLE;
```

Ellenőrizzük, hogy a TIPUS csak az 'SZ' azaz személy, 'GY' azaz gyors, 'IC' és 'EC' értékeket vehesse fel.

```
ALTER TABLE JARAT
ADD CONSTRAINT JARAT_TIPUS_CHK CHECK
(TIPUS = 'SZ' OR TIPUS = 'GY' OR TIPUS = 'IC' OR TIPUS = 'EC')
ENABLE;
```

Az ERINT tábla létrehozása:

PK	Name	Data Type	Size	Not Null	Default	Comment
	ERINT_ID	NUMBER		<input checked="" type="checkbox"/>		
	ALLOMAS_ID	NUMBER		<input checked="" type="checkbox"/>		
	JARAT_ID	NUMBER		<input checked="" type="checkbox"/>		
	INDUL	DATE		<input type="checkbox"/>		
	ERKEZIK	DATE		<input type="checkbox"/>		

A tábla létrehozásnak SQL kódja:

Létrehozzuk az ERINT táblát, ahol az ERINT_ID egyedi szám és egyben a tábla elsődleges kulcsa, az ALLOMAS_ID és JARAT_ID is egyedi szám és idegen kulcs, az INDUL és ERKEZIK mező pedig dátum típusú.

```
CREATE TABLE ERINT
(
    ERINT_ID NUMBER(5, 0) NOT NULL
,   ALLOMAS_ID NUMBER(5, 0) NOT NULL
,   JARAT_ID NUMBER(5, 0) NOT NULL
,   INDUL DATE
,   ERKEZIK DATE
)
```


Beállítjuk az ERINT_ID-t elsődleges kulcsként.

```
, CONSTRAINT ERINT_PK PRIMARY KEY
(
    ERINT_ID
)
USING INDEX
(
    CREATE UNIQUE INDEX ERINT_PK ON ERINT (ERINT_ID ASC)
    LOGGING
    TABLESPACE OKT
    PCTFREE 10
    INITRANS 2
    STORAGE
    (
        BUFFER_POOL DEFAULT
    )
    NOPARALLEL
)
ENABLE
)
LOGGING
TABLESPACE OKT
PCTFREE 10
INITRANS 1
STORAGE
(
    BUFFER_POOL DEFAULT
)
NOCOMPRESS
NO INMEMORY
NOPARALLEL;
```

Beállítjuk az JARMU_ID-t és ALLOMAS_ID-t idegen kulcsnak.

```
ALTER TABLE ERINT
ADD CONSTRAINT ERINT_FK1 FOREIGN KEY
(
    ALLOMAS_ID
)
REFERENCES ALLOMAS
(
    ALLOMAS_ID
)
ENABLE;

ALTER TABLE ERINT
ADD CONSTRAINT ERINT_FK2 FOREIGN KEY
(
    JARAT_ID
)
REFERENCES JARAT
(
    JARAT_ID
)
ENABLE;
```

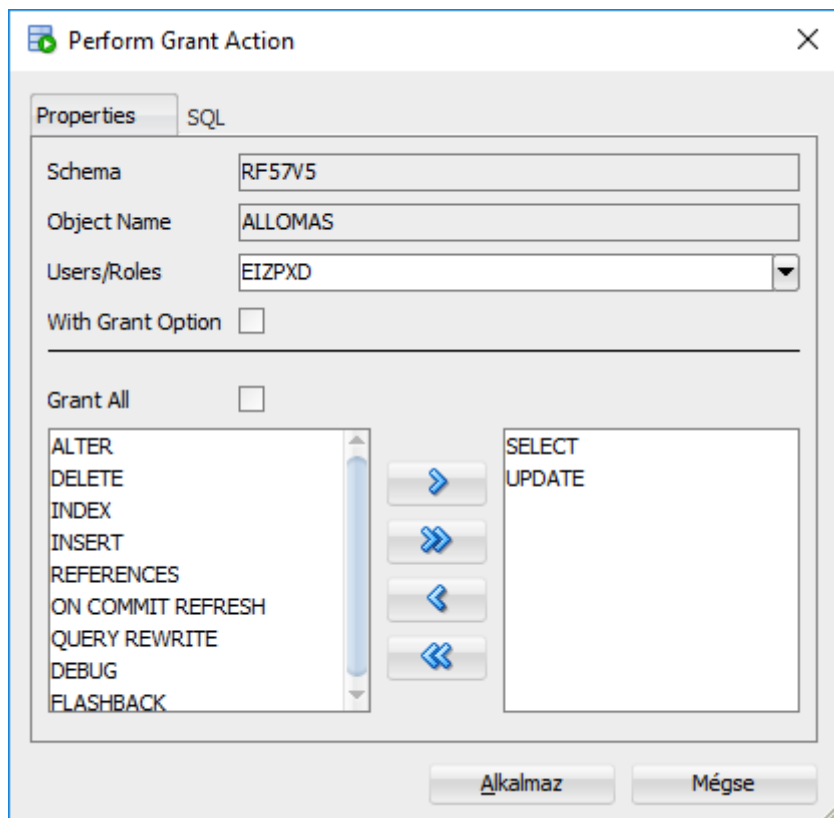
Ellenőrizzük, hogy a ERKEZIK idő hamarabb legyen mint az INDUL idő, vagy az ERKEZIK idő NULL. (Láthatjuk, hogy ha az indulás NULL akkor kisebb lesz mint az érkezés ideje, tehát az állomás végállomás lesz. Ha az érkezés ideje NULL, az állomás induló állomás lesz.)

```
ALTER TABLE ERINT
ADD CONSTRAINT ERINT_DATES_CHK CHECK
(INDUL < ERKEZIK OR ERKEZIK IS NULL)
ENABLE;
```

5. feladat: Jogok

A jogok beállítása az adott táblán jobb egérrel kattintva a privileges, azon belül pedig a grant menüpont alatt érhető el.

Az alábbi képen az ALLOMAS táblára adunk jogokat EIZPXD felhasználónak (csak SELECT és UPDATE jogokat kap).



A kiadott jogokat a tábla Grants nézetében tekinthetjük meg:

	PRIVILEGE	GRANTEE	GRANTABLE	GRANTOR	OBJECT_NAME
1	SELECT	EIZPXD	YES	RF57V5	ALLOMAS
2	UPDATE	EIZPXD	YES	RF57V5	ALLOMAS

A fenti jogok kiadása az alábbi SQL kódot futtatja:

```
grant UPDATE, SELECT on "RF57V5"."ALLOMAS" to "EIZPXD";
```

Azokat a jogokat, amiket más felhasználóktól kaptunk a DBA fül security, users pontjában, saját felhasználónkat kikeresve tekinthetjük meg. Itt az object privileges nézetben láthatók a táblákra kapott jogok:

	Object Privilege	Schema	Object	Grant Option
1	SELECT	EIZPXD	JARAT	YES
2	UPDATE	EIZPXD	JARAT	YES
3	SELECT	EIZPXD	MENETREND	YES
4	UPDATE	EIZPXD	MENETREND	YES
5	SELECT	EIZPXD	REPULOGE	YES
6	UPDATE	EIZPXD	REPULOGE	YES

A táblázatból láthatjuk, hogy EIZPXD felhasználó JARAT, MENETREND és REPULOGE táblákra kaptunk SELECT és UPDATE jogokat.

Script futtatása:

```
column grantor format a8;
column grantee format a8;
column table_name format a20;
column privilege format a20;
select grantor
      , grantee
      , table_name
      , privilege
      , initcap(grantable) grant_opt
from all_tab_privs
where grantor = user
      or grantee = user
order by grantor, grantee, table_name, privilege
;
```

A fenti scriptet futtatva az alábbi kimenet jelenik meg:

GRANTOR	GRANTEE	TABLE_NAME	PRIVILEGE	GRA
EIZPXD	RF57V5	JARAT	SELECT	Yes
EIZPXD	RF57V5	JARAT	UPDATE	Yes
EIZPXD	RF57V5	MENETREND	SELECT	Yes
EIZPXD	RF57V5	MENETREND	UPDATE	Yes
EIZPXD	RF57V5	REPULOGEP	SELECT	Yes
EIZPXD	RF57V5	REPULOGEP	UPDATE	Yes
RF57V5	EIZPXD	ALLOMAS	SELECT	No
RF57V5	EIZPXD	ALLOMAS	UPDATE	No
RF57V5	EIZPXD	ERINT	SELECT	No
RF57V5	EIZPXD	ERINT	UPDATE	No
RF57V5	EIZPXD	JARAT	SELECT	No
GRANTOR	GRANTEE	TABLE_NAME	PRIVILEGE	GRA
RF57V5	EIZPXD	JARAT	UPDATE	No
RF57V5	PUBLIC	RF57V5	INHERIT PRIVILEGES	No
13 rows selected				

Ebben láthatjuk a kiadott és kapott jogainkat:

- SELECT és UPDATE jogokat adtunk EIZPXD felhasználónak az ALLOMAS, JARAT és ERINT tábláinkra
- SELECT és UPDATE jogokat kaptunk EIZPXD felhasználótól a JARAT, MENETREND és REPULOGEP táblákra

6. feladat: Zárak

A zárok aktiválásához egy táblán tranzakciót kell kezdeményezünk. A zár aktív marad, amíg a tranzakció tart. A változtatások ideje alatt a módosított adatok nem vesznek el és más felhasználók sem férhetnek hozzá a zárolt táblákhoz. Amikor kiadjuk a commit parancsot (a tranzakció végeztével), a program frissíti a táblák adatait majd feloldja a zárokat, így a változtatások a többi felhasználó számára is láthatóvá válnak.

Két féle zár van erre a célra:

- TM zárok, melyek a táblákat zárolják
- TX zárok, melyek a tranzakcióknak fenntartott zárok. Ezekre azért van szükség, hogy a tábla egy sorához egyszerre csak egy tranzakció férhessen hozzá

Beszúrás az ALLOMAS táblába:

❖ USERNAME	❖ SESSION_ID	❖ TYPE	❖ TYPE_NAME	❖ TYPE_DESCRIPTION	❖ OBJECT_NAME	❖ PARTITION_NAME
1 RF57V5	606 TM	DML	Synchronizes accesses to an object	RF57V5.ALLOMAS	(null)	
2 RF57V5	606 TM	DML	Synchronizes accesses to an object	RF57V5.ERINT	(null)	
3 RF57V5	606 TX	Transaction	Lock held by a transaction to allow other transactions to wait for it	(null)	(null)	
4 RF57V5	606 AE	Edition Lock	Prevent Dropping an edition in use	(null)	(null)	

Az alábbi SQL parancsok kiadása alatt képen látható zárok aktiválódnak. A két TM lock zárolva tartja az ALLOMAS és ERINT táblákat, mert az ERINT tábla használja az ALLOMAS táblát (az ERINT tartalmazza az ALLOMAS kulcsát idegen kulcsként).

```
INSERT INTO ALLOMAS (ALLOMAS_ID,ALLOMAS_AZON,NEV,FORGALOM,SZTRAJK) VALUES
(00001,'BP01','Nyugati pu.','7500,5000);
INSERT INTO ALLOMAS (ALLOMAS_ID,ALLOMAS_AZON,NEV,FORGALOM,SZTRAJK) VALUES
(00002,'BP02','Keleti pu.','8000,4000);
INSERT INTO ALLOMAS (ALLOMAS_ID,ALLOMAS_AZON,NEV,FORGALOM,SZTRAJK) VALUES
(00003,'GYOR','Győr','5500,2000);
INSERT INTO ALLOMAS (ALLOMAS_ID,ALLOMAS_AZON,NEV,FORGALOM,SZTRAJK) VALUES
(00004,'SZLK','Szolnok','6500,5000);
```

Beszúrás a JARAT táblába:

	USERNAME	SESSION_ID	TYPE	TYPE_NAME	TYPE_DESCRIPTION	OBJECT_NAME	PARTITION_NAME
1	RF57V5	606	TM	DML	Synchronizes accesses to an object	RF57V5.JARAT	(null)
2	RF57V5	606	TM	DML	Synchronizes accesses to an object	RF57V5.ERINT	(null)
3	RF57V5	606	TX	Transaction	Lock held by a transaction to allow other transactions to wait for it	(null)	(null)
4	RF57V5	606	AE	Edition Lock	Prevent Dropping an edition in use	(null)	(null)

Az alábbi SQL parancsok kiadása alatt képen látható zárok aktiválódnak. A két TM lock zárolva tartja a JARAT és ERINT táblákat, mert az ERINT tábla használja a JARAT táblát (az ERINT tartalmazza a JARAT kulcsát idegen kulcsként).

```
INSERT INTO JARAT (JARAT_ID,JARAT_AZON,TIPUS,MIOTA,MEDDIG,MEGJEGYZES,KOZLEKEDIK)
VALUES (00001,1112,'IC',date'2014-10-16',null,null,'1111111');
INSERT INTO JARAT (JARAT_ID,JARAT_AZON,TIPUS,MIOTA,MEDDIG,MEGJEGYZES,KOZLEKEDIK)
VALUES (00002,2112,'EC',date'2009-05-02',date'2012-12-31','Megszűnt','0000011');
INSERT INTO JARAT (JARAT_ID,JARAT_AZON,TIPUS,MIOTA,MEDDIG,MEGJEGYZES,KOZLEKEDIK)
VALUES (00003,2562,'GY',null,null,null,'1111111');
INSERT INTO JARAT (JARAT_ID,JARAT_AZON,TIPUS,MIOTA,MEDDIG,MEGJEGYZES,KOZLEKEDIK)
VALUES (00004,1236,'SZ',null,null,null,'0111010');
```

Beszúrás az ERINT táblába:

USERNAME	SESSION_ID	TYPE	TYPE_NAME	TYPE_DESCRIPTION	OBJECT_NAME	PARTITION_NAME
1 RF57V5	606 AE	Edition Lock	Prevent Dropping an edition in use		(null)	(null)
2 RF57V5	606 TM	DML	Synchronizes accesses to an object		RF57V5.ERINT	(null)
3 RF57V5	606 TX	Transaction	Lock held by a transaction to allow other transactions to wait for it		(null)	(null)
4 RF57V5	606 TM	DML	Synchronizes accesses to an object		RF57V5.JARAT	(null)
5 RF57V5	606 TM	DML	Synchronizes accesses to an object		RF57V5.ALLOMAS	(null)

Az alábbi SQL parancsok kiadása alatt képen látható zárok aktiválódnak. Mivel az ERINT a JARAT és az ALLOMAS táblákat kapcsolja össze (és a ERINT tartalmazza mindkettő kulcsát idegen kulcsként), lezárja a JARAT és az ALLOMAS táblákat is.

```
INSERT INTO ERINT(ERINT_ID,ALLOMAS_ID,JARAT_ID,INDUL,ERKEZIK) VALUES
(00001,00001,00001,to_date('2017-10-15 11:30:00','YYYY-MM-DD HH24:MI:SS'),null);
INSERT INTO ERINT(ERINT_ID,ALLOMAS_ID,JARAT_ID,INDUL,ERKEZIK) VALUES
(00002,00002,00002,null,to_date('2018-02-25 14:40:00','YYYY-MM-DD HH24:MI:SS'));
INSERT INTO ERINT(ERINT_ID,ALLOMAS_ID,JARAT_ID,INDUL,ERKEZIK) VALUES
(00003,00004,00004,to_date('2018-02-22 14:25:00','YYYY-MM-DD
HH24:MI:SS'),to_date('2018-02-22 14:27:30','YYYY-MM-DD HH24:MI:SS'));
INSERT INTO ERINT(ERINT_ID,ALLOMAS_ID,JARAT_ID,INDUL,ERKEZIK) VALUES
(00004,00003,00003,to_date('2018-03-01 17:45:00','YYYY-MM-DD
HH24:MI:SS'),to_date('2018-03-01 17:50:00','YYYY-MM-DD HH24:MI:SS'));
```