

DB2 backup and recovery

Task. 1. Backup offline. Recovery.

1. Start command line processor
2. Start DB2 instance (db2start)
3. Read from the configuration file of the database TESTDB parameters defining the transaction logs:
db2 get db cfg for testdb
number of primary log files – LOGPRIMARY
number of secondary log files – LOGSECOND
log file size (in 4kB) – LOGFILSIZ
the parameter LOGARCHMETH1 defines the method of archiving of logs: circular logging (default, LOGARCHMETH1=OFF and LOGARCHMETH2=OFF) or archival logging (LOGARCHMETH1 not set to OFF).
4. Connect to TESTDB db as STUDENT
5. List tables in db TESTDB
6. Create in the db TESTDB table employee with fields:
ID int not null primary key,
NAME varchar(15),
SALARY decimal(10,2)
(if the table already exists, drop it first)
7. Add two rows to this table
8. On disc C, create the folder db2backup (in this folder we will store the backups)
9. Make the full offline backup of the db TESTDB to the folder db2backup
backup db testdb to c:/db2backup (put the correct path to the directory)
save the information of the backup time (as **t1**), you will need this information for the recovery; to read the time of backups, you can also use
list history backup all for db testdb
10. Check, if the connection to TESTDB is still active?
(During the full offline backup there can be no other active connections to the db. You can force all other users and application with: force applications all).
11. Connect to TESTDB
12. Insert to table employee two more rows
13. Drop the db TESTDB
14. Restore TESTDB using its backup:
restore db testdb from c:/db2backup taken at **t1**
15. Connect to TESTDB and read table employee. What data were recovered?

Task. 2. Backup online and rollforward recovery.

In order to use online backups and the mechanism of rollforward recovery, the database must enable the archival logging method of archiving of transaction logs

1. In the node catalog of DB2 create a folder logarch (for storing the transaction log files)
2. Change the configuration parameters of the TESTDB database in such a way that the archive logging is enabled (put the correct path to the logarch folder)
update db cfg for testdb using LOGARCHMETH1 disk: c:/db2/logarch
3. Connect to db TESTDB
(After changing the method of archiving, the db is in the BACKUP PENDING state. In order to connect to the db, make first the full backup. Make the full backup and save the time of it as **t2**.)
4. Add to table employee two rows

5. Disconnect from TESTDB and drop it
6. Restore db TESTDB from the backup made in point 3
 restore db testdb from c:/db2backup taken at **t2**
 (after restoring, the db is in the ROLL FORWARD PENDING state. In order to connect to the db, we must first make the rollforward recovery (to recover data from the transaction logs, in this way we can recover data that were changed between the moment of the backup and the moment of the db crash)
 Recover all data from transaction logs:
 rollforward db testdb to end of logs and complete
 (if we do not want to recover data from logs, we change the state of db to normal with rollforward db testdb stop)
7. Connect to TESTDB and read table employee. What data were recovered?
8. Read the system time and save it as **t3**. Delete from table employee the last added row.
9. Restore db from the backup, like in point 6, but rollforward all the changes, except the last delete operation:
 rollforward db testdb to **t3** using local time
 (the time **t3** must be given in the format yyyy-mm-dd-hh.mm.ss)
 Stop the rollforward.
10. Connect to TESTDB and read table employee. What data were recovered?

Task. 3. Incremental backups

1. Change the configuration parameter TRACKMOD, which defines if the incremental backups are allowed
 update db cfg for testdb using TRACKMOD ON
2. Connect to TESTDB
3. In order to make the incremental backup, full backup must be done first. Make the full backup (online) of TESTDB, save the time of the backup as **t4**.
 backup db testdb online to c:/db2backup
4. Connect to TESTDB and change the salaries of all employees to 3000
5. Make the incremental backup (this backup store only the changes since last full backup)
 backup db testdb incremental to c:/db2backup
 Save the time **t5** of the backup. Compare the sizes of files with full and incremental backups.
6. To restore db from incremental backup, we need to use the incremental and full backups in the right order. During the recovery, it is recommended to use the option AUTOMATIC (let DB2 to decide on the sequence of backups to use)
 restore db testdb incremental automatic from c:/db2backup taken at **t5**
 We can list the sequence of backups to use in the restore with
 db2ckrst -d testdb -t t5
7. Connect to TESTDB and change the salaries of all employees to 6000
8. Do points 5 and 6 but make the delta backup instead