**Assignment II**

1. **Which data movement utility supports the CURSOR input file?**

First of all, we have to understand the working of CURSOR file. The use of this file is declared against an SQL query.

The load utility will process the entire result of the query associated with the specified cursor whether or not the cursor has been used to fetch rows.

1. **What other privileges are needed to load a tale if a person already has LOAD authority?**
2. **What will this command do?**

**db2look –d department –a –e –m –x –f –o db2look.sql**

The ***db2look*** command generates the DDL (Data Definition Language) statements by object type. ***-d*** Alias name of the production database that is to be queried. ***department*** can be the name of a Db2. In other words, the ***department*** is the name of the database. ***-a*** as it generates DDL statements for objects that were created by any user, including inoperative objects. ***-e*** as it extracts all DDL statements for the database objects, as our database object is alias. ***-m*** generates the UPDATE statements that are required to replicate the statistics on tables, statistical views, columns, and indexes. ***-x*** as it generates authorization DDL statements such as GRANT statements. ***-f*** this extracts the configuration parameters and registry variables that affect the query optimizer. ***-o*** Writes the output to the ***db2look.sql*** file. If you do not specify an extension, the .sql extension is used. If you do not specify this parameter, output is written to standard output.

1. **What is the difference between LOAD and IMPORT utility?**

In large scale organizations, it is often necessary to move data from Db2 Production subsystems to Test subsystems. Of the various technologies that are available to achieve this business objective, circumstances determine which is most appropriate.

There are various data movement options available in Db2 like load/unload utility, import utility, export utility, Ingest utility, db2move command etc.

|  |  |  |
| --- | --- | --- |
| Method | **LOAD utility** | **IMPORT utility** |
| Purpose | To efficiently move large quantities of data into newly created tables, or into tables that already contain data. | To insert data from an external file into a table, hierarchy, view, or nickname. |
| Best practice usage | This utility is best suited to situations where performance is your primary concern. This utility can be used as an alternative to the import utility. It is faster than the import utility because it writes formatted pages directly into the database rather than using SQL INSERTS. In addition, the load utility allows you the option to not log the data or use the COPY option to save a copy of the loaded data. Load operations can fully exploit resources, such as CPUs and memory on SMP and MPP environments. | The import utility can be a good alternative to the load utility in the following situations:   * where the target table is a view * the target table has constraints and you don't want the target table to be put in the Set Integrity Pending state * the target table has triggers and you want them fired. |

1. **How you know when to use IMPORT Vs LOAD utility?**

When we need to view a target table, and the target table has constraints, we don’t want the target table to be put in the Set Integrity Pending state. If the above situations are occurred then we only face towards the IMPORT utility. IMPORT utility is mainly use to insert data from an external file into a table, hierarchy, view, or nickname.

On the other hand. i.e. LOAD utility, where we have some concern in the performance of primary, if you want to writes formatted pages directly into the database so you may use LOAD utility. LOAD utility also allows you the option to not log the data or use the COPY option to save a copy of the loaded data. Load operations can fully exploit resources, such as CPUs and memory on SMP and MPP environments.

1. **Name and Explain the phases of LOAD process.**

A complete load process consists of four phases:

• Load phase

• Build phase

• Delete phase

• Index copy phase

Load Phase:

In the load phase, the LOAD utility scans the input file for any invalid data rows that do not comply with the table definition.

Build Phase:

During the build phase, indexes are produced based on the index keys collected during the load phase. The index keys are sorted during the load phase, and index statistics are collected (if the statistics use profile option was specified).

Delete Phase:

In the load phase, the utility only rejects rows that do not comply with the column definitions. Rows that violated any unique constraint are deleted.

Index copy Phase:

During the index copy phase, index data is copied from a system temporary table space to the original table space.