

# LAB REPORT

## CSE – 4508

**Name** : Abdullah  
**ID** : 200041126  
**Lab Group** : 1B

**Submitted on** : 17<sup>th</sup> August, 2023

**Submitted to** : MD. SHIHAB SHAHRIAR SIR

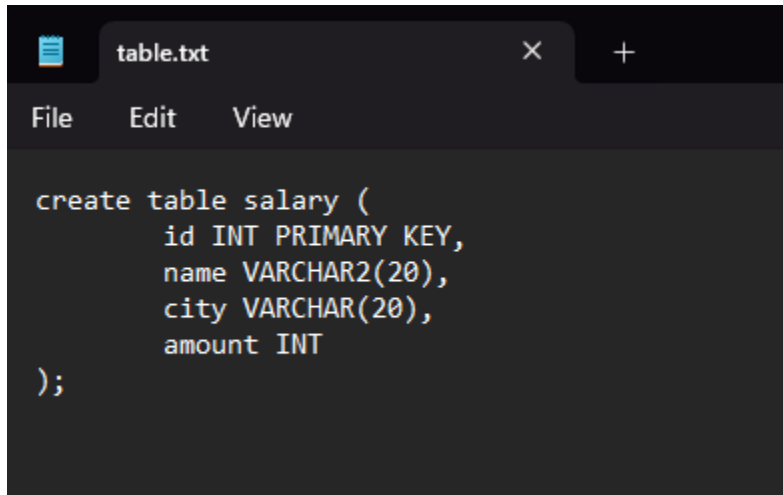
During today's laboratory session, we acquired proficiency in employing the SQL Loader utility to transfer data from a text file into an Oracle table. This utility seamlessly accommodates text files in formats such as .txt and .csv. Our focus for this session was on loading data from a .csv file.

The subsequent steps elucidate the process of directly loading data from a CSV file into the database.

#### **Procedure to input data:**

- ❖ First, when we look at the CSV file, we notice that the data is separated by commas. Each line represents one thing and is finished with a space.  
Inside each line, there are four separate pieces of information separated by commas. These pieces of information are the ID, Name, Address, and Salary of the employees. Because of this, the table we create in the database needs four columns. These columns should be for Integer, varchar2, varchar2, and number data types, in that order.

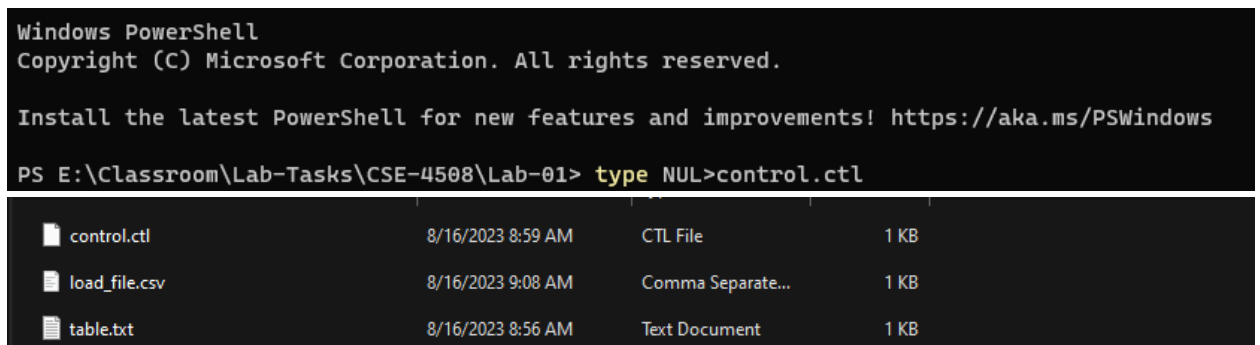
- ❖ Next on the basis of gathered information the table should be created in the database after the user logs in into the database.



```
File Edit View

create table salary (
    id INT PRIMARY KEY,
    name VARCHAR2(20),
    city VARCHAR(20),
    amount INT
);
```

- ❖ Create a control file (.ctl) in any directory, preferably in the same location as the CSV file. Name the control file appropriately and open it using a text editor like Notepad.
- ❖ We are opening it using PowerShell command.



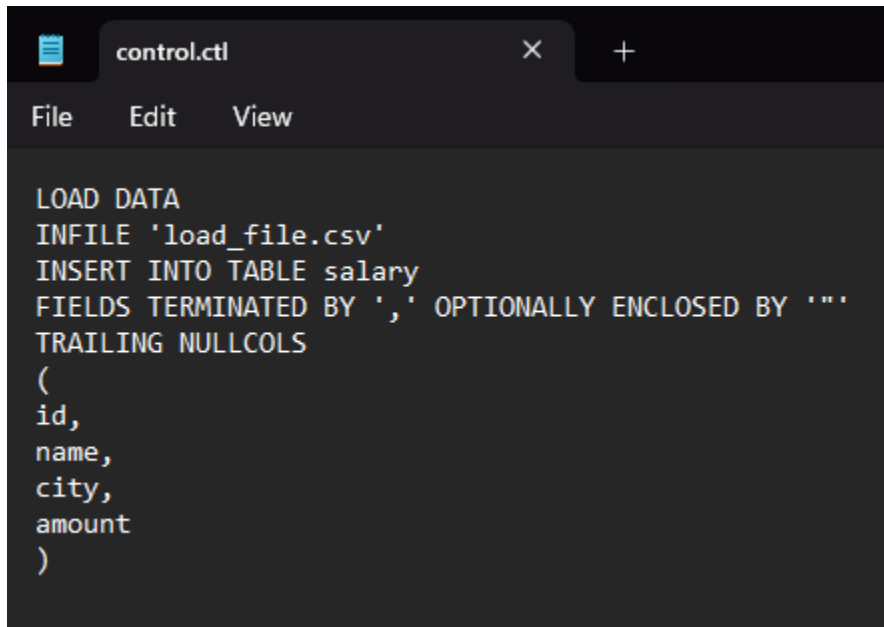
```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS E:\Classroom\Lab-Tasks\CSE-4508\Lab-01> type NUL>control.ctl
```

|               |                   |                   |      |
|---------------|-------------------|-------------------|------|
| control.ctl   | 8/16/2023 8:59 AM | CTL File          | 1 KB |
| load_file.csv | 8/16/2023 9:08 AM | Comma Separate... | 1 KB |
| table.txt     | 8/16/2023 8:56 AM | Text Document     | 1 KB |

- ❖ After the blank file is opened, the following lines of code should be written in the .ctl file and saved.

A screenshot of a text editor window with a dark theme. The title bar shows a tab labeled 'control.ctl' with a close button (X) and a new file button (+). The menu bar includes 'File', 'Edit', and 'View'. The code is written in a light-colored font on a dark background. It is a SQL control file script for loading data from a CSV file into a table named 'salary'.

```
LOAD DATA
INFILE 'load_file.csv'
INSERT INTO TABLE salary
FIELDS TERMINATED BY ',' OPTIONALLY ENCLOSED BY '"'
TRAILING NULLCOLS
(
  id,
  name,
  city,
  amount
)
```

- ❖ Afterward, we enter the SQLLoader prompt, connecting it to the relevant control file and log file as shown below. At this stage, the user's credentials and the control file are supplied. Additionally, a custom log file is needed, allowing the option to select a preferred name. This process concludes with the database being updated using the provided values, as demonstrated in the accompanying visual

reference.





```
Windows PowerShell
PS E:\Classroom\Lab-Tasks\CSE-4508\Lab-01> sqldr userid=user41126/iutcse control=control.ctl log=track.log

SQL*Loader: Release 11.2.0.2.0 - Production on Thu Aug 17 21:24:01 2023

Copyright (c) 1982, 2009, Oracle and/or its affiliates. All rights reserved.

Commit point reached - logical record count 34
Commit point reached - logical record count 35
```

❖ This is the current condition of the file directory

|   |                   |                   |      |
|---|-------------------|-------------------|------|
|  control.ctl   | 8/16/2023 8:59 AM | CTL File          | 1 KB |
|  load_file.csv | 8/16/2023 9:08 AM | Comma Separate... | 1 KB |
|  table.txt     | 8/16/2023 8:56 AM | Text Document     | 1 KB |
|  track.log     | 8/16/2023 9:26 AM | Text Document     | 2 KB |

## ❖ We can check the track.log file now

```
track.log
File Edit View

SQL*Loader: Release 11.2.0.2.0 - Production on Wed Aug 16 09:08:48 2023

Copyright (c) 1982, 2009, Oracle and/or its affiliates. All rights reserved.

Control File:   control.ctl
Data File:      load_file.csv
Bad File:       load_file.bad
Discard File:   none specified

(Allow all discards)

Number to load: ALL
Number to skip: 0
Errors allowed: 50
Bind array:     64 rows, maximum of 256000 bytes
Continuation:   none specified
Path used:      Conventional

Table SALARY, loaded from every logical record.
Insert option in effect for this table: INSERT
TRAILING NULLCOLS option in effect

  Column Name          Position  Len  Term Encl Datatype
-----
ID                     FIRST     *    ,  O("") CHARACTER
NAME                   NEXT     *    ,  O("") CHARACTER
CITY                   NEXT     *    ,  O("") CHARACTER
AMOUNT                 NEXT     *    ,  O("") CHARACTER

Table SALARY:
  35 Rows successfully loaded.
  0 Rows not loaded due to data errors.
  0 Rows not loaded because all WHEN clauses were failed.
  0 Rows not loaded because all fields were null.

Space allocated for bind array:          66048 bytes(64 rows)
Read  buffer bytes: 1048576

Total logical records skipped:           0
Total logical records read:              35
Total logical records rejected:           0
Total logical records discarded:          0

Run began on Wed Aug 16 09:08:48 2023
Run ended on Wed Aug 16 09:08:48 2023

Elapsed time was:      00:00:00.05
CPU time was:          00:00:00.00
```

❖ Now, if we enter the database and select the table, the contents of .csv file will be displayed there.

| ID | NAME | CITY     | AMOUNT |
|----|------|----------|--------|
| 1  | ABC  | Dhaka    | 6000   |
| 2  | BCD  | Rajshahi | 2000   |
| 3  | CBA  | Dhaka    | 6000   |
| 4  | BCE  | Rajshahi | 7000   |
| 5  | EBC  | Rangpur  | 3400   |
| 6  | BEC  | Rajshahi | 4300   |
| 7  | CAB  | Sylhe    | 6800   |
| 8  | DEC  | Rajshahi | 1100   |
| 9  | DCE  | Sylhet   | 7300   |
| 10 | DBC  | Dhaka    | 2700   |
| 11 | DCB  | Rajshahi | 8400   |
| ID | NAME | CITY     | AMOUNT |
| 12 | DCC  | Sylhet   | 3800   |
| 13 | EDC  | Dhaka    | 7200   |
| 14 | GAB  | Rangpur  | 2800   |
| 15 | BAG  | Rangpur  | 7300   |
| 16 | ABG  | Rajshahi | 9100   |
| 17 | GAC  | Rangpur  | 4800   |
| 18 | CAG  | Dhaka    | 8300   |
| 19 | CGA  | Rajshahi | 1900   |
| 20 | ACG  | Sylhet   | 5700   |
| 21 | GDA  | Sylhet   | 8600   |
| 22 | DAG  | Dhaka    | 5800   |
| ID | NAME | CITY     | AMOUNT |
| 23 | AGD  | Rajshahi | 4700   |
| 24 | ADG  | Rangpur  | 2800   |
| 25 | HAC  | Rangpur  | 1900   |
| 26 | HRT  | Rajshahi | 5200   |
| 27 | ABH  | Sylhet   | 8300   |
| 28 | HBA  | Rangpur  | 7300   |
| 29 | ACH  | Sylhet   | 9200   |
| 30 | HDC  | Dhaka    | 3600   |
| 31 | HUS  | Rajshahi | 8700   |
| 32 | IUT  | Dhaka    | 12000  |
| 33 | RUT  | Rangpur  | 9800   |
| ID | NAME | CITY     | AMOUNT |
| 34 | UIU  | Tangail  | 2200   |
| 35 | HJK  | Mirpur   | 9500   |

35 rows selected.

This depicts the method by which data is brought into the database from a .csv file, and sqlldr automatically handles any errors using suitable control file directives.