

**19F CST8276\_010 Advanced Database Topics**

**Vi Thi Phuong Pham**

**Exercise 03**

# Evidence of Learning

1. Variable

Create variables with const and let

*//regex expression for (,)*

const regex = /,(?=(?:[^"]\*"[^"]\*")\*[^"]\**$*)/;

*//print out my name*

console.log("Name: Vi Thi Phuong Pham - 040886894");

*//path to the csv file*

let path = "canadianCheeseDirectory.csv";

let max = 5;

let counter = 0;

From lines 11 to 19 in exercise 3.

1. Loops

I am using a simple For loop and an enhanced For loop in this example

*//loop through totalrows to get data with selected columns*

*//put it into rows. So now rows is perfect data*

*for* (let i = 0; i <= max; i++) {

      let row = [];

*for* (const col of columnIndex) {

        row.push(totalRows[i][col]);

      }

      rows.push(row);

    }

From lines 67 to 73 in exercise 3.

1. File input-output

In this exercise, I use fs, readline modules along with createReadStream and createInterface method to read the file line by line.

*//import fs, readline*

const fs = require("fs");

const readline = require("readline");

let readFile = fs.createReadStream(path);

let rl = readline.createInterface({ input: readFile });

From lines 7, 8 and 54, 55

1. Simple Data structure

I use multiple arrays in the exercise like array totalRows (line 44), rows (line 47), header (line 49), data (line 52)

*for* (let i = 0; i <= max; i++) {

      let row = [];

*for* (const col of columnIndex) {

        row.push(totalRows[i][col]);

      }

      rows.push(row);

    }

*//put the first array of rows into header*

    header = rows[0];

*//put the leftover data into data array*

    data = rows.slice(1);

This code above indicate how I use arrays to handle data. From line 67 to 78

1. Methods

Throughout the exercise, I have used different built in methods including (map, for each, push, split, slice…)

*//put the leftover data into data array*

    data = rows.slice(1);

*//convert rows into objects*

    const cheeseRecordList = data.map((rowData, rowIndex) => {

*//create object obj*

      const cheeseRecord = {};

*//for each header, put data into object*

      header.forEach((headerName, colIndex) => {

        cheeseRecord[headerName] = rowData[colIndex];

      });

*//print out the object*

*// console.log(cheeseRecord);*

*return* cheeseRecord;

    });

    console.log(cheeseRecordList);

  }

I use map for array data in line 81, slice for data in line 78, forEach for array header in line 85.

# Research