INF10025 Data Management and Analytics

Task 2 – Pass and Credit

Overview

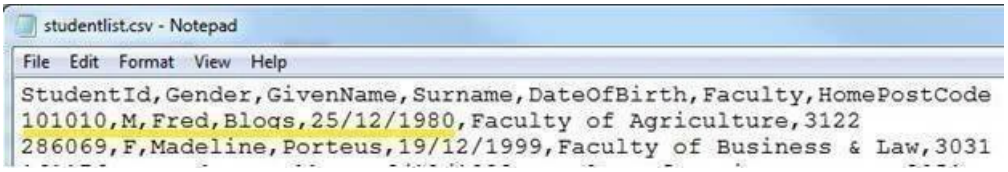
- This week you will continue to compile tasks for your learning portfolio. As per the first task: complete tasks, document them (usually by making screenshots) and submit online.
- In this task we focus on parameters, data exchange, data analytics, and data visualisation.
- Read through the tasks once first before getting started.
- To create your submission, download the files T02P.DOCX and T02C.DOCX from Canvas. Paste the required screen captures from the tasks below into these files.
- When complete, use the File / Export menu option to generate the files T02P.PDF and T02C.PDF.
- **Submit both files into Canvas via Assignment**
- **Check Canvas → Assignments for Due Date. Tasks submitted after the due date without a prior written extension will not be accepted.**

Pass Tasks

Completion Criteria: For the Pass Task to be marked "Complete" eleven (11) sub-tasks must be marked "Correct".

Pass 2a

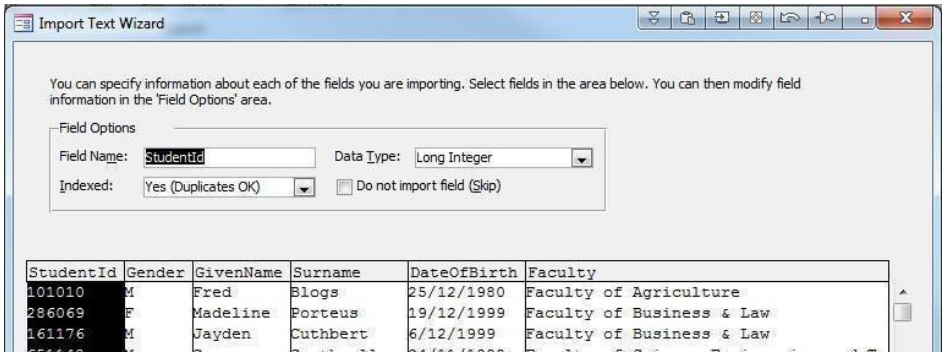
- Download the **studentlist.csv** data file from Canvas
- Look at the file name in File Manager.
- DO NOT double click the file (as it may cause Excel to open).
- Instead Right Click the file name and edit the file using Notepad or Notepad ++ (or similar editor).
  Note: MS Word is **not** a good choice as it may add hidden formatting characters.
- **Add a new row** of student details below the headings.
  Use **your student name** and **student ID** (drop the X if your student id contains an 'X').
  Use **your** current home address **postcode**.
  This **screen capture** depicts an example of a student in this unit named **Fred Blogs.**



- **Save your CVS file.**

- **Import** the file into MS Access
- ***When importing files that have dates make sure the Date format in your Access (or your computer) has the same format of the Data you are importing***Check Lecture 3 slide No.34

- Take a screen shot during the **Field Options process** – so that **your details appear in Row 1.** (See Fred's details in the screen shot example)



- Make sure the table name is **Student9999**, where 9999 are the last 4 digits of your student ID.
- Finally take a **screen capture** of the student table **datasheet** with **your name displayed** in the list.
- **Paste the 3 screen captures in the appropriate position in the document named T02P.DOCX**

Pass 2b

- Type answers to these questions:
  - Why does double clicking typically cause a csv file (*such as student.csv) to open in Excel rather than Notepad?
  - What steps can you take to ensure that when you double click a .csv file it will cause Notepad.Exe to open that file (instead of Excel)?
- **Write your answers in the appropriate position in the document named T02P.DOCX. You may use screenshots to illustrate your answer.**

Pass 2c

- Add a new **field** to the **student** table.

  Fieldname: **FeesPaid**. Type: **Currency**.
- **Save** the changes.

- Open DataSheet view of the Student table.
- Sort the data into Descending Student ID sequence (The largest student number at the top of the list)
- **Screen Capture** the top 10 rows.

- You are about to execute Queries which will modify data in your student table.
- At this point, you must **make a copy of the Student Table.**
- **Right Click** on the **student table** and select **COPY**.
- Then **Right Click** anywhere **beneath** the table names and select **Paste**.
- Ensure that the new table is called STUCOPY9999 (where 9999 are replaced with the last 4 digits of your id).
- If you need to recreate the student table (because your update queries perform inappropriately, then copy and paste the backup copy of your Student table).
- **Screen Capture** the list of tables that includes both the Student and StuCopy Tables.
- **Paste the screen captures in the appropriate position in the document named T02P.DOCX**

Pass 2d

- **Close** the **Student** table if it is open.
- Create an **Update Query** named **T2PD_Update_9999** (replace **9999** with the last 4 digits of your student ID)
- The query must set the **FeesPaid** to **5000** for all **Male** students in the student table.
- **Screen Capture** the Query Grid Design.
- Run the Query. **Screen Capture** the same 10 rows as in the previous task.
- **Paste both screen captures in the appropriate position in the document named W02P.DOCX**

Pass 2e

- **Close** the student table if it is open.
- Create a **Delete** Query named **T2PE_Delete_9999** (replace **9999** with the last 4 digits of your student ID)
- The query must **delete** students that **have a surname beginning with 'M'**
- **Screen Capture** the Query Grid Design.
- Run the Query. **Screen Capture** the top 10 rows in Descending Student ID sequence.
- **Paste the screen captures in the appropriate position in the document named T02P.DOCX**

Pass 2f

- **Close** the student table if it is open.
- Create a **Select** Query named **T2PF_Param_9999** (replace **9999** with the last 4 digits of your student ID)
- The query must display the StudentId, Given Name, Surname, Gender, Home PostCode
- The query must have a **parameter**.
- The parameter must ask the user a **postcode** value.
- Run the Query a few times. At least one of these must use the same PostCode that you assigned to 'your' row in the table.
  Create 3 screen captures, each shows the first 5 rows of **a different postcode** (only use postcodes that exist within the table, one of which is your postcode).
- **Paste the screen captures in the appropriate position in the document named T02P.DOCX**

You should be able to reach this point at the end of Week 03

Pass 2g

You are now going to use **MS Power BI** to complete the remainder of this Pass task. Use the same **moviedatabaseDAD.accdb** that you used in the Pass tasks.

Make sure all tables have a 9999 extension (e.g. movie9999) where 9999 is the last 4 digits of your student id.

- **Close MS Access.**
- Now, open **Power BI Desktop**
- Choose the **Get Data** option.
- Choose the Microsoft Access file named **moviedatabaseDAD.accdb from Task 1 or 2 folder** (now with your tables renamed with the last 4 digits of your student ID)
- **Import** data from all tables
- Click on the **Relationships Window** icon

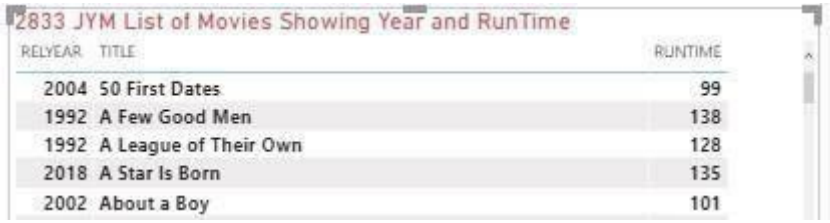
- Double Click on each line of the relationship **and go the Cross Filter section** - make sure the **BOTH** box is ticked, not Single
- Check that **relationships** between the five tables are correct.
- Take a **screen capture** of the **Relationships**
- **Paste the screen captures in the appropriate position in the document named T02P.DOCX**

For the following tasks you need to include screenshots of Values/Filters area where your visualisation is defined

Pass 2h

- Create a **Table** visualisation.
- Add the fields TITLE, RELYEAR, RUNTIME and RATINGCODE.
- Click onto the each of the fields and make sure that Don't Summarise is ticked
- Go to the Filters Section for either Relyear or Runtime and set a filter of your own choosing. For eg you could choose Films released where Release Year is 2005 or greater or Go to Runtime and set a filter of your choosing on the Run time. Make sure it is your own work
- **Sort** the **Relyear** column into **ascending** sequence
- Add a **heading** to the visualisation that includes your **student ID** and the **title** of the visualisation.
- Try a few of the formatting functions – colour, change the size of the font of the heading and anything else you like but make sure it is clear when doing a screenshot.

  It should look similar to the screenshot below (note, your column list will be different from what's depicted on the screenshot).



- **Screen capture** the first **10 rows** displayed (click on the Visualisation so it appears clearer)
- **Screen capture** the Values/Filters section (Right Hand side under the Visualisations)
- **Paste both screen captures in the appropriate position in the document named T02P.DOCX**

Pass 2i

Create a **Donut Chart** visualisation**.**

- The **donut chart** must display the **count of Movies by RatingCode**
- Add a **heading to** the visualisation that includes your **student ID** and the **title** of the visualisation
- Format Title Font size and colour to your liking.
- The **data labels** must show **both** the **Rating Code** and the **Count of movies**.
- **Screen capture** the visualisation.
- **Screen capture** the Values section (Right Hand side under Visualisations)
- **Paste both screen capture in the appropriate position in the document named T02P.DOCX**

Pass 2j

Create a **Matrix** visualisation**.**

- It must display the **number of actors in each movie by gender.**
- If you have totals appearing beneath each row, they can be **removed** via **Format Settings / General / Total Row**

- Add a **heading** to the visualisation that includes your **student ID** and the **title** of the visualisation.
- Your result should look similar to the screenshot below

Student Id 1234567 Gender Casting by Title

TITLE	F	M	Total
50 First Dates	1	4	5
A Few Good Men		6	6
A League of Their Own	3	3	6
About a Boy	3	1	4
Adventureland	2	3	5

- **Screen capture** the Values/Filters section (Right Hand side under Visualisations)
- **Screen capture** the first **10 rows** of the **visualisation**.
- **Paste the screen capture in the appropriate position in the document named T02P.DOCX**

Troubleshooting:

In the above tasks, sometimes **the same values** are repeated for every row. (You may not have done the cross filter part correctly in 2G)

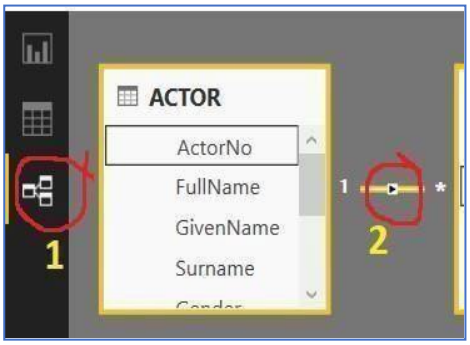
E.g. 123 females and 349 males for every movie title

(Note: the values may differ from semester to semester as we change tasks but the troubleshooting approach is the same)

A League of Their Own	123	349	472
About a Boy	123	349	472

These values are obviously **incorrect**. The first movie did NOT have 472 actors! Nor did the second movie.

- Follow these instructions to solve the problem.



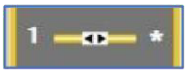
Steps 1 & 2

Click on the **Relationship Design** screen icon and observe the arrow on the **relationship line** between the tables.

The problem occurs when the line has a **single direction** arrowhead.

You need to change from single headed to **double headed**.

From this 

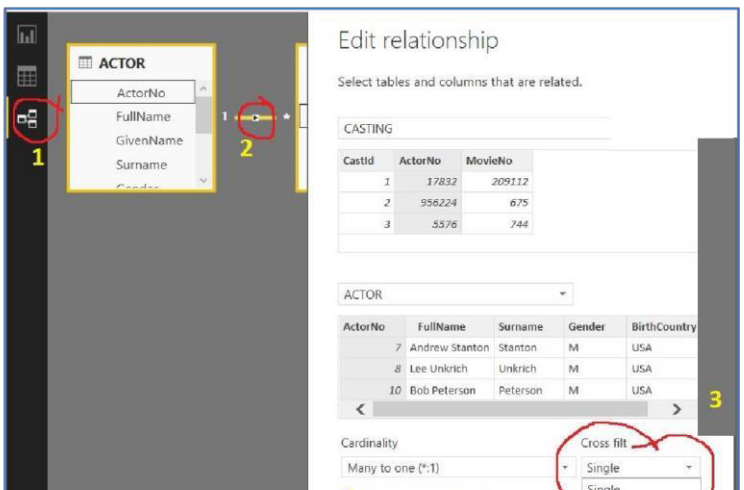
to this 

Step 3.

This is done by **clicking** on the single headed arrow.

An Edit screen appears.

Change the Cross Filter arrow direction to **Both**.



Pass 2k

- Create a **Map** visualisation (and **not** a Filled Map visualisation)
- It must display the **count of actors** by **birth country**.
- Add a **heading** to the visualisation that includes your **student ID and the title** of the visualisation.
- **Screen capture** the visualisation.
- **Screen capture** the Values / Filters section (Right Hand side under Visualisations)
- **Paste the screen captures in the appropriate position in the document named T02P.DOCX**

Pass 2L

- Create **another type of Visualisation (Not A Map)** that uses the same data as above. It must display the count of actors by birth country.
- Adjust one of the **Filters** so that **only countries** with at **least 4 actors or more** are displayed.
- Add a **heading** to the visualisation that includes your **student ID and the title** of the visualisation.
- Increase fonts sizes and colour to make Visualisation clearer to read **Screen capture** the visualisation.
  Screen capture the filter section and the section under the visualisation (right hand side)
- **Paste the screen captures in the appropriate position in the document named T02P.DOCX**

Pass 2m

- Go back to **MS Access** and **create a new table**.
- The table could be about **ANYTHING** you want. It could be a table of your favourite vegetables or it could be a list of Houses that you have lived in, or details about your favourite songs, sports players, electronics.
- Table name must end with **the last 4 digits of your ID**.
- Add fields to the table. The table must have at least 5 columns. The table must use at least **2 Data Types** (Text, Number, Date…). **One of the columns should contain data that can be summed or counted** within a query that will be created in the future).
- Specify a **Primary Key**.
- **Screen capture** the Table Design.
- Now add at least 10 records to this table.
- **Screen capture** the Data View of this table.
- Now **create a Query** that has a **parameter.**
- Use the parameter in the Criteria section so that only some of the records will be displayed.
- **Screen capture** the Query Design grid
- **Screen capture** the running of the query with the specified Parameter
- **Screen capture** Data View once the query has been run/executed.
- **Paste the screen captures in the appropriate position in the document named T02P.DOCX**

Credit Tasks

Completion Criteria: For the Credit Task to be marked "Complete" Eight (8) sub-tasks must be marked "Correct".

Credit 2a

- Download the file named **custdetails.xlsx** from Canvas
- This file contains two worksheets that contain data about: **Customers & Customer Types.**
  Note: The Customer data contains two dates (date of birth and last purchase date).
- **Import** this data into Access **tables** and create appropriate **relationships**.
- The table names must be: **CUST9999** and **CUST_TYPE9999** (where 9999 represents the last 4 digits of your student id).
- **Screen Capture** the first few rows of the CUST9999 datasheet view
- **Screen Capture all** the rows of the CUST_TYPE9999 datasheet view
- **Paste the screen captures in the appropriate position in the document named T02C.DOCX**

Credit 2b

- Create a Query named **T2CB_9999** (where 9999 is the last 4 digits of your student id).
- The query is based on the **CUST9999** table.
- The query must display the customer id, given name, family name, Date of Birth, **The Age of the Customer** (based on today's date). This will involve a calculation that uses **the Now() and DateDiff functions**.
- **Screen capture** the Query Grid Design.
- **Screen capture** the **first 10 rows** in the datasheet once the query has been executed.
- **Paste the screen captures in the appropriate position in the document named T02C.DOCX**
- You can use these references (or any others that you find) to assist with the **DateDiff()** function and the **Now()** function:

  http://www.techonthenet.com/access/functions/date/datediff.php

  http://www.techonthenet.com/access/functions/date/now.php

  http://msaccesstips.blogspot.com.au/2011/08/calculating-date-differenceusing.html
-
-

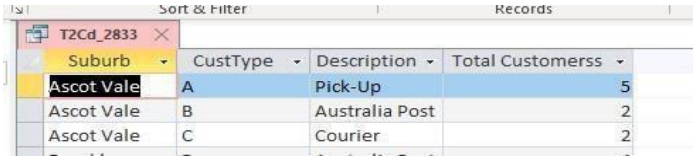
Credit 2c

- Create a **Parameter Query** named **T2CC_9999**
  This query must use the existing **query** named **T2CB_9999** as its source.
- This Query must have a **parameter** that asks the user to enter a **number(Age)**
- Only records where the Customer's Age is **equal to or less than** the **parameter** value must be displayed
- **Screen Capture** the Query Grid Design.
- **Screen Capture** the Query Design with the Parameter entry showing **Age you entered** before you Run the query (You will have to work out a number so that some results show up)
- **Screen capture all of the rows(if more than 20 just do 20)** in the datasheet once the query has been executed.
- **Paste the screen captures in the appropriate position in the document named T02C.DOCX**

Credit 2d

- Create a Query named **T2CD_9999** based on the **CUST9999** and **CUST_TYPE9999** tables (where 9999 should be replaced with the last 4 digits of your student ID)
- This query will require use of **Totals**.
- The query must **count** the number of customers for each **combination** of **Suburb**, **Customer Type** and **Customer Description**.
- The results are to be **sorted** on Ascending Suburb/ Ascending Type / Ascending Description / Descending Total.

- The results should look similar to this example (although different order / sort of rows):



Suburb	CustType	Description	Total Customerss
Ascot Vale	A	Pick-Up	5
Ascot Vale	B	Australia Post	2
Ascot Vale	C	Courier	2

- **Screen Capture** the Query Grid Design.
- **Screen Capture** the first 12 rows of the results.
- **Paste the screen captures in the appropriate position in the document named T02C.DOCX**

Credit 2e

- **Export** the results of the **T2CD_9999** query to a **text** file named **T2CD_9999.TXT**.
- The file must have **headings** in row 1, use a **semicolon (;) delimiter** and **single quotes** around text.
- **Include Field Names in first Row**
- **Do NOT tick the box** that says "Export Data with Formatting"
- Screen capture the export step showing **selection of a delimiter**.
- The output must look similar to this example but different delimiter and quotes around the text:

```
"Suburb","CustType","Description","Total Customerss"
"Ascot Vale","A","Pick-Up",5
"Ascot Vale","B","Australia Post",2
"Ascot Vale","C","Courier",2
"Brooklyn","B","Australia Post",4
```

- **Open** the file in **Notepad** or **Notepad++** (or similar text editor).

  Screen Capture the first 10 rows of data.

  Paste the screen captures in the appropriate position in the document named T02C.DOCX

Credit 2f

- You are now going to return to **MS Power BI** to complete the remainder of this credit task. Use the same **moviedatabaseDAD.accdb** that you used in the Pass tasks.
- Each actor in the Actor table owns a number of dogs and / or a number of cats.
- You are going to create a calculation called TotalPets for each actor.
- Go to the **Data window** in Power BI
- Add a **new column** that has the name **TotalPets**.
- Add a **formula** that adds the NoOfDogs to NoOfCats.
- Now create **visualisation** that uses TotalPets
  E.g. Average pets per country or Total Pets by Gender or something else
  (You may **use any** appropriate visualisation tool that shows some meaningful data).

- Add a **heading** to the visualisation that includes your **student ID** and the **title** of the visualisation.
- **Screen Capture** the visualisation.
- **Screen Capture the formula that you used to calculate TotalPets.**
- **Paste the screen captures in the appropriate position in the document named T02C.DOCX**

Credit 2g

- Go to the **Data** Window.
- Create a **new column** in the Actor table (Click on the bottom portion of the New Measure icon to create a new column). Do **not** create a new measure (confusing huh?)
- The new column is to be called **DayNo_9999**
- Use the **Weekday()** function to calculate the day number in the week based on the actor's date of birth, i.e. birthday on a Sunday will return a value of 1, birthday on a Monday will return 2…

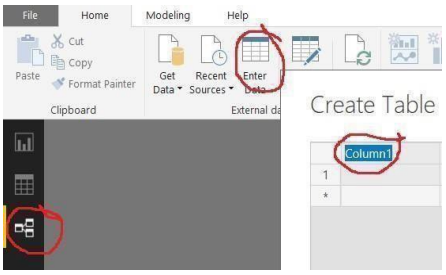
ActorNo	FullName	GivenName	Surname	Gender	Birthdate	IMDBid	BirthCountry	DayNo_4567
1231	Julianne Moore	Julianne	Moore	F	3/12/1960 12:00:00 AM	nm0000194	USA	7
1233	Philip Seymour Hoffn	Philip	Seymour Hoff	M	23/07/1967 12:00:00 AM	nm0000450	USA	1
1245	Scarlett Johansson	Scarlett	Johansson	F	22/11/1984 12:00:00 AM	nm0424060	USA	5

e.g.

- **Screen Capture** the formula that you used.
- Create a **donut chart** visualisation with appropriate data labels and title.
- **Screen Capture** the visualisation.
- **Paste the screen captures in the appropriate position in the document named T02C.DOCX**

Credit 2h

- Go to the **Data** Window.
- Create a **new table.** This can be done by clicking the **Enter Data** icon.
- A Create Table window appears. Click on Column1 heading and replace the text with **DayNo_9999**. The 2nd column should be named **DayName**.
- The contents of the table must be as shown in the diagram.



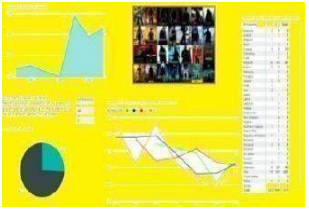
- At the bottom of the Create table window, add a table name: **WeekDays9999.**
- Go to the **Relationships** Window.
- Create a **Relationship** between Actor4567 and WeekDays9999 based on the column named **DayNo_9999.**
- **Screen Capture** the Relationship between the tables.
- **Paste the screen captures in the appropriate position in the document named T02C.DOCX**

DayNo_4567	DayName
1	Sunday
2	Monday
3	Tuesday
4	Wednesday
5	Thursday
6	Friday
7	Saturday

Credit 2i

- Create a **pie chart** visualisation with appropriate **data labels** and **title**.
- The chart must show the **Count of actors** by **Birth DayName** (the legend must show the actual Name of the Day, not Day Number)
- **Screen Capture** the visualisation.
- **Paste the screen captures in the appropriate position in the document named T02C.DOCX**

Credit 2j



- Finally make a report using the table that you created in **Pass Task 2m** above.
- The report must have 4 different types of **visualisations** and at least **one slicer.**
- Add a **text box** that contains your **id** and **name plus a report description**.
- Each visualisation must have a text box or heading. (E.g. Bar chart showing the total of my friends by age group)
- Add an **image**(s) to the report that has something to do with the content of your report. (e.g. If you created a table about trains, then you would include a picture of a locomotive).
- Modify the **background** colour and/or image of the report.
- **Screen Capture** the report (similar to the example shown).
- **Paste the screen captures in the appropriate position in the document named T02C.DOCX**

Support Materials

Microsoft Access

Importing and Exporting Data

Chapter 6 – Importing and Exporting Data

Chapter 10 – Action Queries & Crosstab Queries

Microsoft Power BI

- **Getting Started**

https://powerbi.microsoft.com/enus/documentation/powerbidesktop-gettingstarted/

https://www.youtube.com/playlist?list=PL1N57mwBHtN2q1WbU5O29rrn_A0lkVv9p

- **Potential Issues and Fixes**

'**32/64bit**' error – for fix, please refer to the lecture slides

'**microsoft.ace.oledb.12.0**' error – fix is here:
https://www.microsoft.com/enus/download/details.aspx?id=13255