Ontario One Call Challenge

The challenge is essentially a data manipulation problem that is very typical in database systems. There are two data files attached to this email (ticketInfo.xlsx and MasterStationCode.xlsx) that are in .xlsx format. You can convert the data files to whatever format you feel comfortable with. I suggest MySQL or PostgreSQL for creating the stored procedure.

The ticketInfo.xlsx file describes how the part numbers in the MasterStationCode.xlsx file are interrelated. Please use the TicketInfo file to construct a tree view showing all the station code assigned to the master code in the TicketInfo file. There should be 10 levels in your tree view if the code is working properly except “OOCTEST15” master station code that has 12 level (station code) assigned to it. Parent Node: master_code and child Node: station_code.

Once you have done the tree view, please add a datagrid view that shows the component parts of the part that is currently selected in the tree view. If the station code has no components, the datagrid view should be blank. One datagrid will have a sum of all the station code selected and separate datagrid for each station code selected. Please include the following columns in the datagrid view:

1. Member_code, count of ticket number belongs to the ticketInfo file but the rest of the data must come from the MasterStationCode file. Stored procedure calculation: day_to_close, Time_to_respond, and Compliance. **Call stored procedure by passing the parameter.**

2. The form should look something like the screenshot below when you are done. Note that there are two buttons: Populate Data in Tree. This will populate the Tree View and disable once the Tree View is fully populated.

3. Exit from Application.

Instructions:

1. Read the excel file provided and save it into database

2. Call the stored procedure into the web application with passing selected parameters.

3. Create a stored procedure:
    1. Calculate day_to_close field using SQL query:
        a. IF renegotiated_date IS NULL THEN processed_date – closed_date

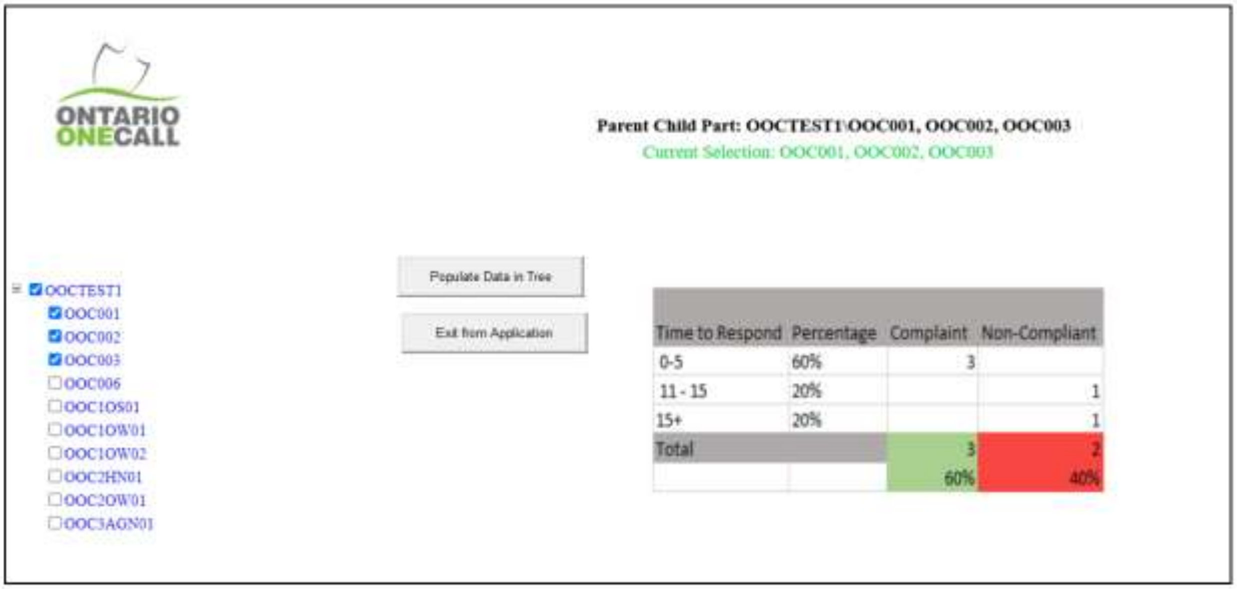
        b. IF renegotiated_date IS NOT NULL THEN renegotiated_date – closed_date

        c. IF renegotiated_date and closed_date IS NULL THEN processed_date - current_date

2. Calculate Time_to_respond using SQL query:
    a. IF day_to_close is below 5, then Time_to_respond = “0-5”
    b. IF day_to_close is below 11, then Time_to_respond = “5-10”
    c. IF day_to_close is below 16, then Time_to_respond = “11-15”
    d. IF day_to_close is above 16, then Time_to_respond = “15+”
3. Calculate day_to_close is below 5 then it is “Compliance” else “Non-Compliance”

3. Final Submission: Attach MySQL or PostgreSQL Script(Single Run script)

4. After click event on “Populate Data in Tree”, it will pass the parameter by selecting the checkbox and perform the above calculation and return the data in the below temporary table. Parameters will be passed by stored procedure. **Calculation will be performed over the database and the return table, not into the web application**.



Parent Child Part: OOCTEST1\OOC001, OOC002, OOC003

Current Selection: OOC001, OOC002, OOC003

- ☑ OOCTEST1
    - ☑ OOC001
    - ☑ OOC002
    - ☑ OOC003
    - □ OOC006
    - □ OOC1OS01
    - □ OOC1OW01
    - □ OOC1OW02
    - □ OOC2HN01
    - □ OOC2OW01
    - □ OOC3AGN01

Time to Respond	Percentage	Complaint	Non-Compliant
0-5	60%	3	
11 - 15	20%		1
15+	20%		1
Total		3	2
		60%	40%