Sub Reports

This section introduces Sub-Reports and provides an example using simple reports to demonstrate the Sub-Report functionality. Following the example step-by-step also provides extra benefit by guiding you to:

1. Use one data driver for two different layout programs.
2. Use commands to control the display of data on the left side and a graph to the right of a report.
3. Flex report headers based on whether they are called as a sub-report or stand-alone.
4. Use commands to control the flow of the program.

    1. This section assumes that you have already completed the sections Creating a Table View Layout and Converting Existing Programs to Layout Programs Using a Driver Program.
    2. Sub-Reports are layout programs that can be executed as stand-alone programs or called from another layout program. Using Sub-Reports enables you to re-use and combine layout programs, and display information from multiple layout programs into one report executed by one parent program.
    3. The flexibility and usability of layout programs is increased by the ability to call and display them from within another layout program. You might have multiple layout programs that contain information that you want to combine in to one report. Marking the layout programs as Sub-Reports enables them to be called and displayed in other layout programs.
       Another common use of Sub-Reports is to execute a driver program to gather data and store it in a record structure. The information is then available for other sub-reports or sections in the layout program.
       The following steps are covered to demonstrate the use of Sub-Reports:
5. Create a data driver program that stores general information about persons in a record structure. We will use the data in the record structure for two separate layout programs that will be marked as Sub-Reports. The naming convention used for the data driver program is **1_*your_initials*_PRSN_Data_Driver**.
6. Create a layout program that graphs the distribution of birthdates by month. The naming convention to be used for the first layout program is **1_*your_initials*_Graph_BD**.
7. Create a table view layout program to display information about persons on a grid. The naming convention to be used for the second layout program is **1_*your_initials*_Person_View**.
8. Create a parent program that calls the graph and table view programs to display the information in one report. The naming convention to be used for the parent layout program is **1_*your_initials*_Parent**.

Step 1: Creating the Driver Program

This driver program is used to retrieve general person information, such as names, birthdates and gender and stores them in to a record structure.

1. Using DVDev, from the File menu select **New** > **Program**.
2. In the Program Name box, enter **1_*your_initials*_PRSN_Data_Driver** and click **OK**.
3. Copy the following commands into your file. Place the code after the Create command and before the key words *End Go*.

4. At the end of the file, enter **1_*your_initialsPRSN_Data_Driver go*** *after the key words _End Go*.
5. Save and compile the source code file.
6. Press CTRL+L to access the listing and verify there is data loaded in the record structure.

Step 2: Creating Layout Program for Sub-Report 1

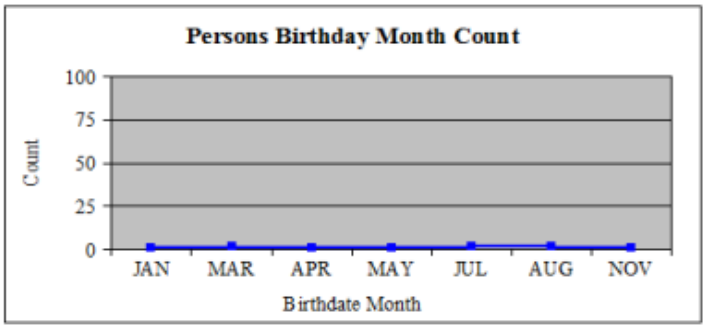
We will create a layout program to calculate the number of birthdays that fall within a month for a specific gender, and display that information in a list and also in a graph. The final output is a report formatted like the following example:

Birthdate Distribution by Month

Page: 1 of 2

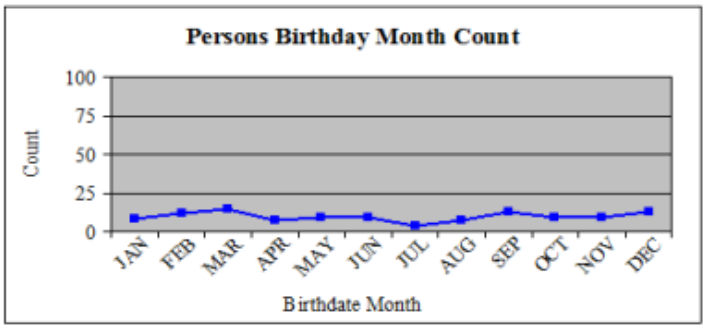
N/A

JAN	1.00
MAR	2.00
APR	1.00
MAY	1.00
JUL	2.00
AUG	2.00
NOV	1.00



Female

JAN	8.00
FEB	12.00
MAR	15.00
APR	7.00
MAY	9.00
JUN	9.00
JUL	4.00
AUG	7.00
SEP	13.00
OCT	9.00
NOV	9.00
DEC	13.00



1. From the File menu, select New.
2. From the File Type list, select Layout Program.
3. In the Program Name box, enter **1_your_initials_BD_Graph** and click OK.
4. On the Report Properties dialog box, click Finish. A layout with a single section named DetailSection is created.
5. From the Tools menu, select Prompt Builder.
6. Click Add to add a second prompt to ask for a gender from the user.
7. On the General tab, set the options as follows:

   Prompt Display:Select a specific gender or Any(*) for all types
   Prompt Name:GEN
   Control Type:Code Set
   Prompt Type:Expression
8. On the Code Set tab, set the options as follows:
   1. Enter **57** for the Code Set
   2. Check the Include Any(*) option
   3. Click the Define Any(*) box and change the Primary key value to **0.0** and click **OK**.
   4. Click the Set Default option and highlight the Any(*) option and save the prompt form.

   Set the Layout Driver that will be used to populate the record structure.
9. From the Tools menu, select **Set Layout Driver** and enter **1_your_initials_PRSN_Data_Driver** in the Name box.

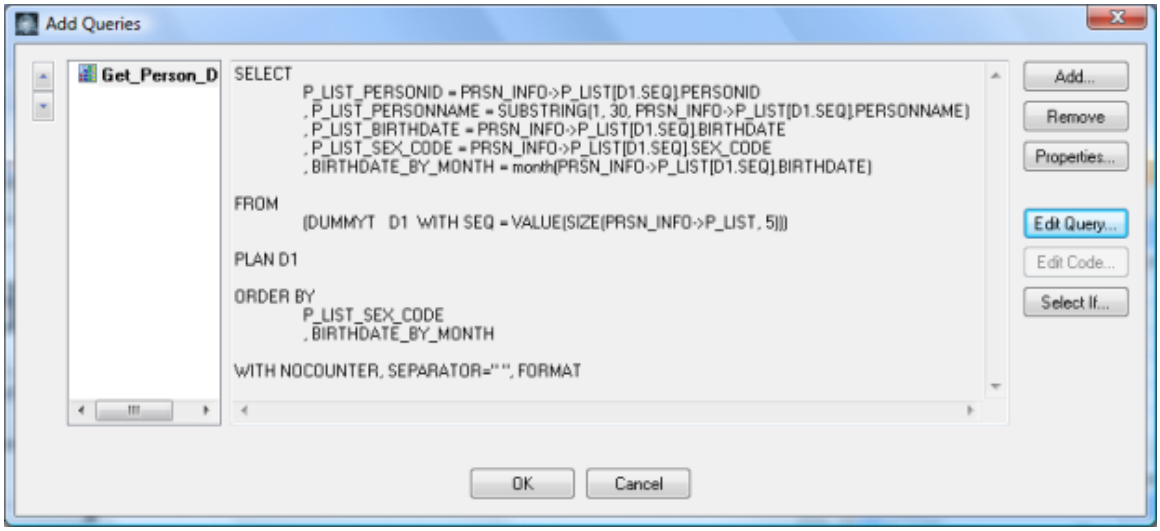
   If your 1_*your_initials*_PRSN_Data_Driver program is a cclgroup1 object, append **:group1** to the program name when you enter it in the Name box. Use CCLPROT to determine if your program is a cclgroup1 or cclgroup0 (DBA) object.
10. In the Parameters box, enter **$Outdev, $GEN** and click **OK** to close the Layout Driver dialog box.

    When this program is run, it executes the driver program which loads data into a record structure. For this layout program to format the data from the record structure, the record structure needs to be recognized by the layout program and in the driver program. In this example, the record structure is defined in the data driver program with the scoping option of PersistScript. The PersistScript scoping option allows a child process to declare a public variable and make it known to the parent process. Using this option allows the record structure to be known to the layout program.
    We need to access the values in the record structure list. A simple way to access the values in the record structure list is to use a query to assign the values in the list to select expressions.
11. From the Tools menu, select **Query Builder** and then click **Add**.
12. In the Query Name box, enter **Get_Person_Data,** select the Associate Layout option and click **OK**.
13. In the **Tables** tab, expand the Record Structures tree and select the User Defined category. The PRSN_INFO record structure is displayed in the Tables column.
14. Double-click the PRSN_INFO record structure to move it to the Selected Tables column.

15. Click the **Fields** tab, expand the P_list tree, and double-click each of the items to add them to the Selected Fields list. Double-clicking a record structure list item adds the DUMMYT table to the **Tables** tab and creates expressions in the Selected Fields column using *dummyt_alias*.seq to reference each position of the record structure list. Creating an expression for each item in a record structure makes it easier to reference the items in your layout.

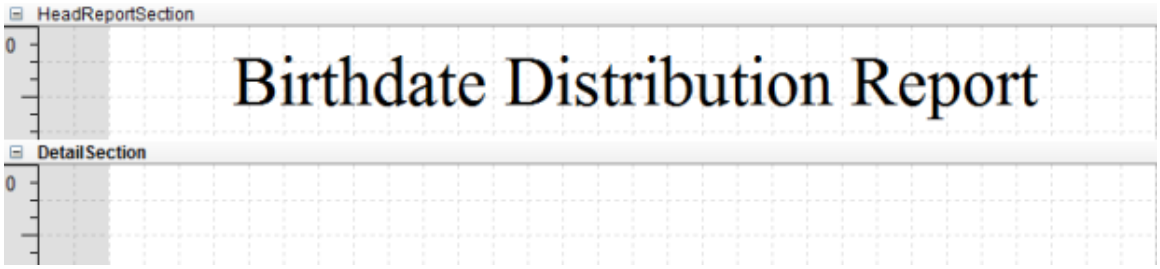
    Now we will create a new expression to return an integer value representing the month of the birth date for a person by using the MONTH( ) function. This expression will be used as a sub sort so that our data is logically grouped by gender and then sequenced by their birth month.
16. Click **Add Expr** and create a new expression. Copy and paste the following command:

17. Click **OK** to close the Add Expr dialog box and click the **Sort** tab.
18. Select P_LIST_SEX_CODE for the primary sort. Then sub-sort by BIRTHDATE_BY_MONTH.
19. Click **Close**. The Add Queries dialog box opens and displays the query similar to the following example:



20. Click **OK** to close the Add Queries dialog box.
21. Check the Head Report Reportwriter Section checkbox and click **Yes** to add the HeadreportSection layout section.
22. Use the Label tool to add a report title, **Birthdate Distribution by Month,** to the HeadReportSection.
23. Use the format toolbar to set the following:
    1. The font to Times
    2. The font size to 36
    3. Center the text
24. Use the vertical resize to drag the DetailSection bar close to the title text.

    Your layout should resemble the following example:



25. Click the Head Page Reportwriter Section checkbox and click **Yes** to the message to add the HeadpageSection layout section.
26. Use the Text tool and add a field to top left side of the section to display the page number and the number of pages.
27. Double-click on the text field and enter **RPT_PageOfPage**.
28. Select the Properties for the item and enter **Page_Of_Page** for the Name property and select Yes for the Font Bold property.
29. Use the vertical resize to drag the DetailSection up to the bottom of the field you just added.
30. From the Layout Workspace check the Head P_List_Sex_Code Reportwriter Section checkbox and click **Yes** to add the HeadP_List_Sex_Code layout section.
31. Select the Properties for the section and change the Name property to **GenderSection**.
32. Use the Line tool to draw a line horizontally across the very top of the section and change the Pen Size property to .025.
33. Use the Text tool to add a field that is under the line to the top left hand side of the section to the display the gender.
34. Double-click the text field and copy and paste the following command:

35. Select the Properties for the item and change the Name property to **Gender**, the Font Bold to Yes and the Font Color to Blue.
36. Use the vertical resize to drag the DetailSection up to the bottom of the field you just added.

    We need to add logic to the program to determine if there is enough room on the page to display the information in the GenderSection which consists of the line, the gender and the 12 months of the year, all displayed vertically down the page. Here is an example of the GenderSection:

If there is not enough room to display the line, gender and 12 months, logic must be added to the program to create a page break and to display the section on the next page. The commands we will use to determine if there is enough space on the page needs to happen before the GenderSection executes.

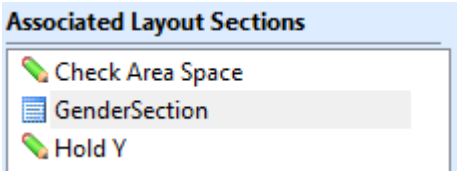
37. From the Layout Workspace dialog, add a Code Segment above the GenderSection. Copy and paste the following commands:

38. Click on the Properties for the Code Segment and change the Name property to **Check Area Space**.

    We need to add commands after the GenderSection has rendered to hold off where the Y value is, so that it can be used later on to properly place the graph to right of the text.

39. Add a code segment below the GenderSection. Copy and paste the following commands:

40. Click the Properties for the Code Segment and change the Name property to **HOLD Y**. Your Associated Layout Sections for the Head P_List_Sex_Code Reportwriter section should resemble the following example:

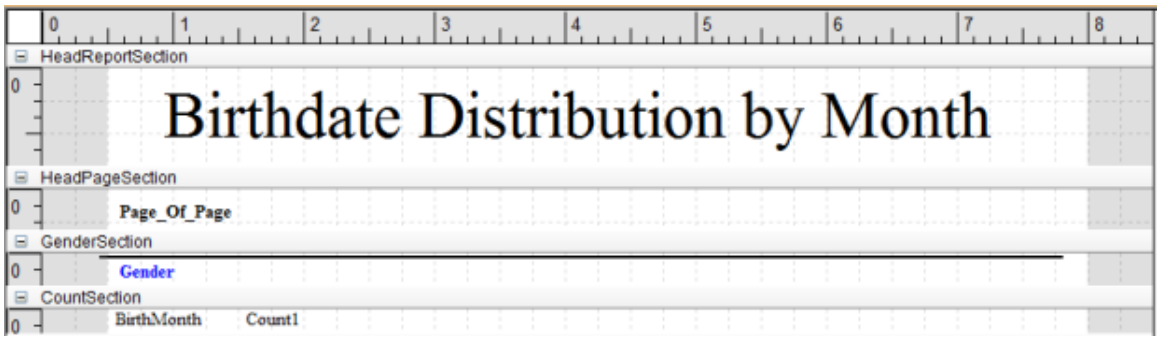


    In this layout program, we do not need the DetailSection to display any items, so it can be deleted.

41. Click in the DetailSection and from the Edit menu select **Remove** > **Section**. Click **Yes** to remove the section.
42. Select the Foot Birthdate_By_Month option and click **Yes** to add the FootBirthdate_By_MonthSection.
43. Change the Name property to **CountSection**.
44. Use the Text tool to add a field to the top left side of the section to display the birth month.
45. Double-click the text field and copy and paste the following command to be used as the source:

46. Select the Properties for the item and change the Name property to **BirthMonth**.
47. From the LayoutWorkspace dialog, add a new Code Segment above the CountSection and add the following command:

48. Use the Text tool to add a field to the right of the BirthMonth to display the number of birthdates calculated for each month.
49. Double-click on the text field and enter **MyCount**.
50. Select the Properties for the item and change the Name property to **Count1**.
51. Use vertical re-size to drag the bottom of the section to the bottom of the fields you just added. Your layout should look similar to the following example:



52. Press CTRL+F5 to execute your layout program. Click **Yes** when prompted to save the layout. The prompt form opens. Select Any(*) for the second parameter and click **Execute**. Your output should look similar to the following format:

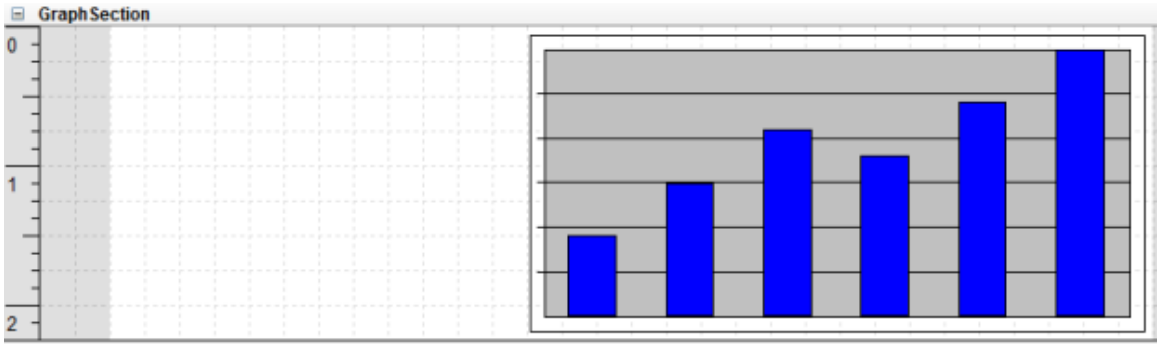
Birthdate Distribution by Month

Page: 1 of 2

N/A	
JAN	1.00
MAR	2.00
APR	1.00
MAY	1.00
JUL	2.00
AUG	2.00
NOV	1.00
Female	
JAN	8.00
FEB	12.00
MAR	15.00
APR	7.00
MAY	9.00
JUN	9.00
JUL	4.00
AUG	7.00
SEP	13.00
OCT	9.00
NOV	9.00
DEC	13.00

To finish this report, add a graph to show the number of birthdays in each month per gender that displays to the right of the months listed vertically.

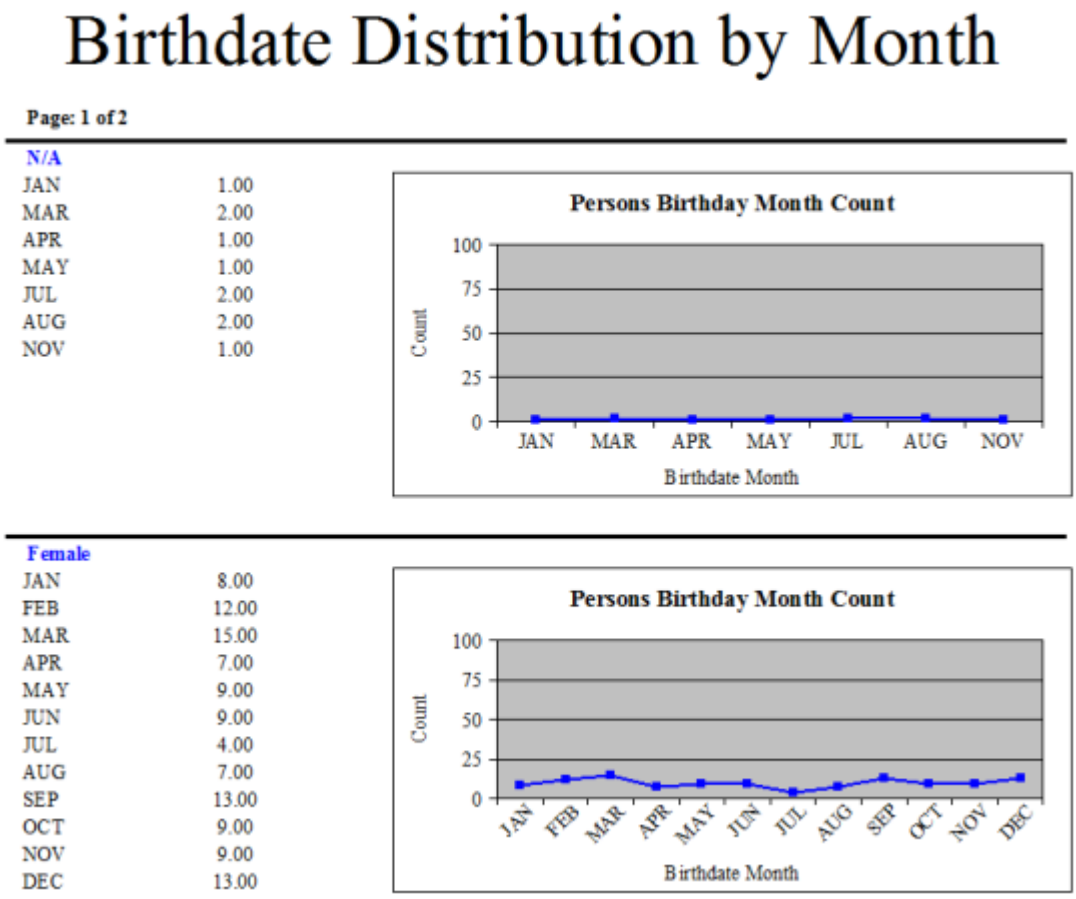
53. From the Layout Workspace, select the Foot P_List_Sex_Code option and click **Yes** to the message to add a layout section.
54. Select the Properties for the section and change the Name property to **GraphSection** and the Height property to **2.25**.
55. Using the Graph tool, create a graph that covers all of the right side of the Graph Section. Your GraphSection should look like the following example:



56. Double-click in the Graph and change the Graph Type to Line.
57. Click the **Data Source** tab and from the Select Query list, select Get_Person_Data.
58. Select BIRTHDAY_BY_MONTH for the Axis Field.
59. Enter **FORMAT(P_LIST_BIRTHDATE,"MMM;;Q")** as the Axis Label.
60. Click the **Series** tab and enter **Count** as the Series Name.
61. Enter **count(P_LIST_BIRTHDATE)** as the Data Values.
62. Click the Data Range tab and enter **0** for the Minimum, **100** for the Maximum and **25** for the Index.
63. Click the **Titles** tab. Enter **Persons Birthday Month Count** as the Chart Title*.* Enter **Birthdate Month** as the (X) Axis Title. Enter **Count** as the (Y) Axis Title.
64. Click **OK** to exit the graph properties dialog box.
65. From the Workspace dialog, click on the Foot P_LIST_SEX_CODE reportwriter section (click the text, not the box) and add a Code Segment above the GraphSection. Copy and paste the following command:

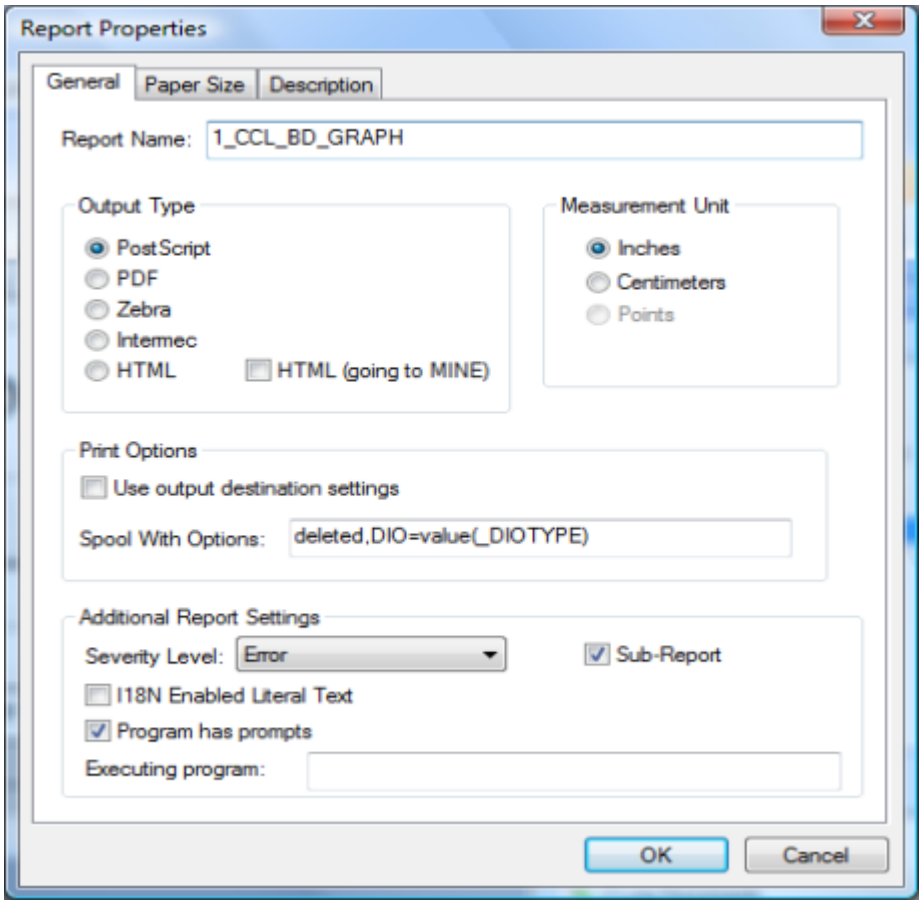
66. Add a Code Segment below the GraphSection to reset the Y position. Copy and paste the following commands:

67. Press CTRL+F5 to execute your layout program. Click **Yes** when prompted to save the layout. The prompt form opens. Select Any(*) for the second parameter and click **Execute**. Your output should look like the following example:



This report runs as a stand-alone program. However, we also want to call it from another layout program as a Sub-Report. It needs to be made a Sub-Report by changing its Report Properties.

68. From the Edit menu select **Report Properties**. Select the Sub-Report option located in the Additional Report Writer Settings.



The sub-report option places code in the program that is used to detect if the program is being called or used as a sub-report from another layout program.

69. Save the file.

You have completed one of the Sub-Reports and will now create one more.

Step 3: Creating the Table View Program – Sub-Report 2

This next layout program calls the same data driver as 1_*your_initialsBD* Graph and displays the detailed information about the persons stored in the record structure using a table view. The final output will be a report formatted like the following example:

	Person ID	Name	Birth Date	Gender
1	593694	Smith, Trina	01/01/01 00:00:00	Female
2	594976	Tremmel, Sue	07/29/23 00:00:00	Female
3	593758	Lee, Khim	03/10/28 00:00:00	Female
4	595361	Baker, Ginger	10/03/28 00:00:00	Female
5	593747	Limshue, Katia	02/17/30 00:00:00	Female
6	595125	Lou, Marta	07/08/30 00:00:00	Female
7	594102	Pill, Jillene	11/10/33 00:00:00	Female
8	593927	Boban, Leanne	06/18/35 00:00:00	Female
9	595232	Imara, Lauris	07/09/35 00:00:00	Female
10	593924	Dock, Alison	10/17/35 00:00:00	Female
11	594274	Teach, Kathleen	09/15/36 00:00:00	Female
12	594937	Linn, Kerri	08/21/37 00:00:00	Female
13	595112	Friends, Colleen	05/05/38 00:00:00	Female

1. In DVDev, from the File menu select New.
2. From the File Type list, select Layout Program.
3. In the Program Name box, enter **1_your_initials_Person_View** and click OK.
4. From the Report Layout options, select the Table View option. Enter **5** for the number of column options.
5. From the Tools menu, select **Prompt Builder**.

Copy the prompts from the 1_*your_initialsBD_Graph layout program to 1your_initials*_Person_View.

6. From the Tools menu, select **Transfer Objects** and enter **1_*your_initials*_BD_Graph**.
7. Click **Browse** to find the program.
8. From Task menu, select **Copy Object**. Enter **1_*your_initials*_Person_View** and click **OK**.

   The same data driver program used to populate the graph in 1_*your_initials*_BD_Graph is used to populate this table view.

9. From the Tools menu, select **Set Layout Driver**. The Layout Driver dialog box is opened.
10. In the Name box, enter **1_*your_initials*_PRSN_Data_Driver**.

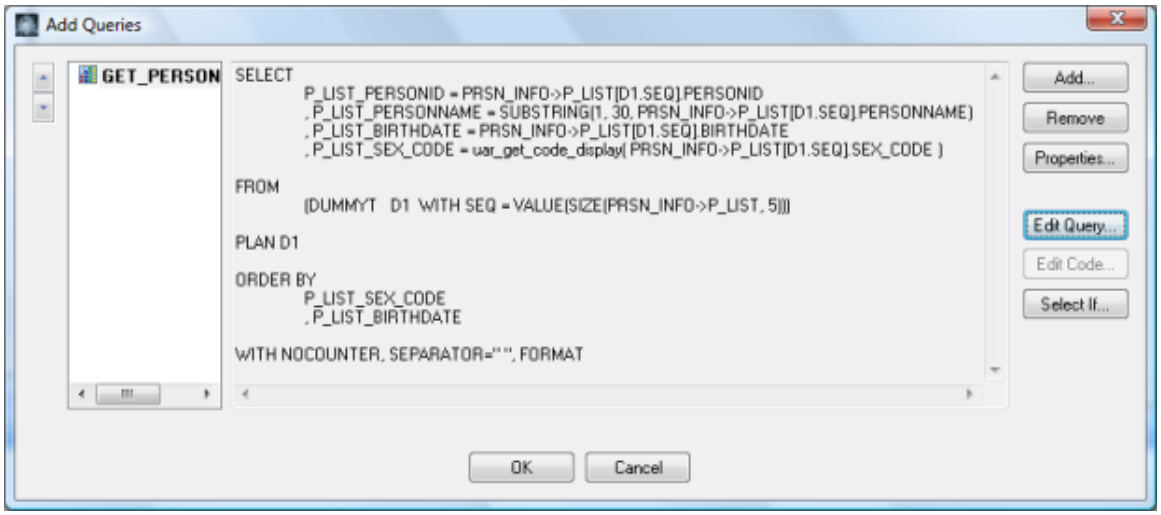
    If your 1_*your_initials*_PRSN_Data_Driver program, is a cclgroup1 object, append **:group1** to the program name when you enter it in the Name box. Use CCLPROT to determine if your program is a cclgroup1 or cclgroup0 (DBA) object.

11. In the Parameters box, enter **$Outdev, $GEN** and click **OK** to close the Layout Driver dialog box.

    We need to access each item in the record structure list, which can be done by using a query to assign the values in the list to select expressions.
12. From the Tools menu, select **Query Builder** and click **Add**.
13. In the Query Name box, enter **Get_Person_Data**, select the Associate Layout option and click **OK**.
14. On the **Tables** tab, expand the Record Structures tree and select the User Defined category. The PRSN_INFO record structure is displayed in the Tables column.
15. Double-click the PRSN_INFO record structure to move it to the Selected Tables column.
16. Click the **Fields** tab, expand the P_list tree, and double-click each of the items to add them to the Selected Fields list.
17. In the Selected Fields list, edit the expression created for the PRSN_INFO->P_LIST[D1.SEQ].SEX_CODE by adding **UAR_GET_CODE_DISPLAY( )** to the item:

    Your expression should look similar to the following example:

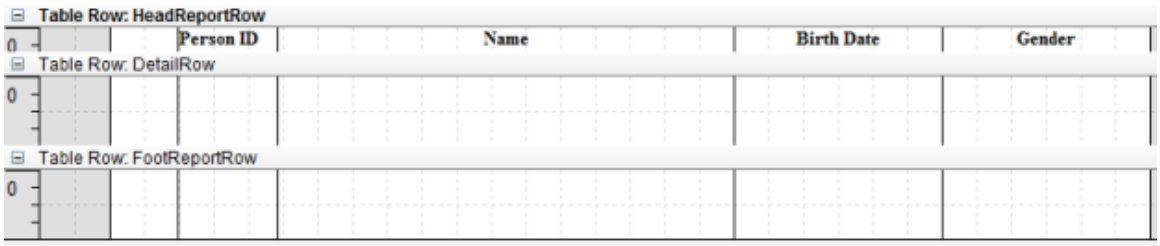
18. From the **SORT** tab, select P_LIST_SEX_CODE and then P_LIST_BIRTHDATE.
19. Click **Close**. Your query should look similar to the following example:



20. Click **OK** to close the Add Queries dialog box.

    The HeadReportRow is used to display column headers. The DetailRow is used to display the actual ID and name of each person returned by the query. The FootReportRow is used to show a total count of how many persons are listed. Currently, these rows are divided into cells of equal size. The first cell is used to display a sequence number and the second cell is used to display a person's ID. These cells should be made smaller to accommodate the data.

21. With no cell selected, move the pointer over the right vertical edge of the first cell. When the pointer changes to the horizontal resize ↔ ,click and drag the vertical edge to the left to make all of the cells in the first column about a half inch in width.
22. Click in the second cell of the HeadReportRow. Click again and enter **"Person ID"** as the text. Remember that quotes must be placed around any text item.
23. Modify the Font Bold property to Yes.
24. Deselect the cell and change the vertical edge of all of the cells in the second column to three quarters inch in width.
25. Click the third cell of the HeadReportRow. Wait a moment, click again, and enter **"Name"** as the text.
26. Modify the Font Bold to Yes and the Align Horizontal property to Center.
27. Click the fourth cell of the HeadReportRow. Wait a moment, click again, and enter **"Birth Date"** as the text.
28. Modify the Font Bold to Yes and the Align Horizontal property to Center.
29. Click the fifth cell of the HeadReportRow. Wait a moment, click again, and enter **"Gender"** as the text.
30. Use the vertical resize to drag the bottom of the HeadReportRow up to just below the items you added above. Your table should look similar to the following example:



The first cell of the DetailRow will be used to display the row number. Initialize a variable in the HeadReportRow that will be incremented and used to display the row number.

31. From the Layout Workspace click the Head Report reportwriter section (the text not the checkbox) and add a Code Segment below the HeadReportRow and enter the following command:

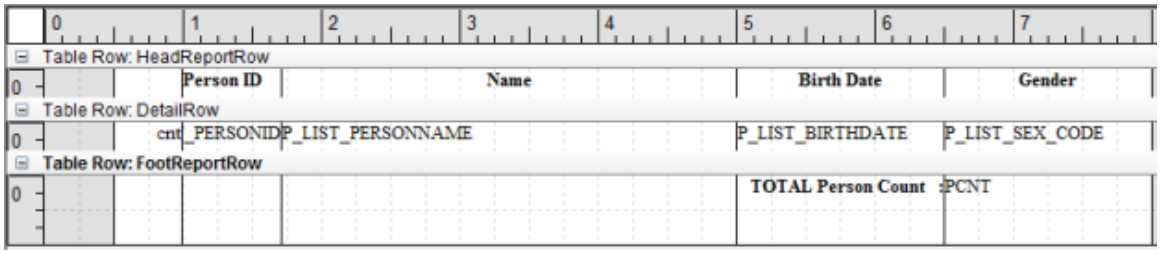
32. Click the Detail Section (the text, not the checkbox) and add a Code Segment above the DetailRow and enter the following command:

33. Click in the first cell of the DetailRow. Click again and enter **CNT**.
34. From the Drop Items dialog box, drag the following items to the DetailRow:
    1. P_LIST_PERSONID to the second cell
    2. P_LIST_PERSONNAME to the third cell
    3. P_LIST_BIRTHDATE to the fourth cell
    4. P_LIST_SEXCODE to the fifth cell

35. Access the Properties for P_LIST_PERSONID and wrap the CNVTINT() function around the item so that it looks similar to the following example: **cnvtint(P_LIST_PERSONID)**. Using the function converts the data to an integer data type and keeps trailing zeros from displaying with the person ID.
36. Change the Align Horizontal property to Right.
37. Access the Properties for P_LIST_BIRTHDATE and wrap the FORMAT() function around the value so that it looks similar to the following example:

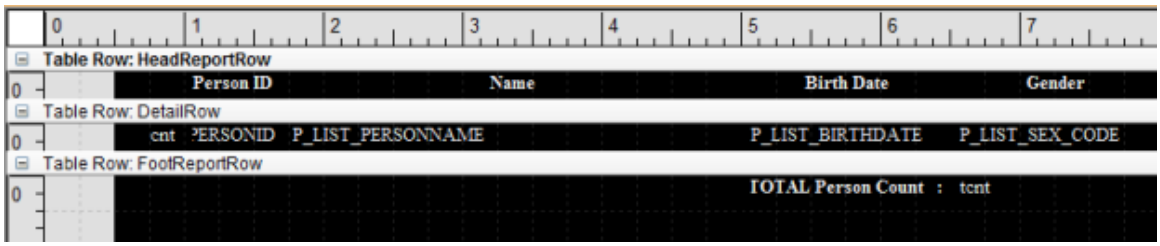
Use vertical resize to drag the bottom of the DetailRow up to just below the items you have just added.

38. Click the fourth cell of the FootReportRow. Click again and enter **"TOTAL Person Count"** as the text.
39. Modify the Font Bold property to Yes and Align Horizontal property to Right.
40. Click the fifth cell of the FootReportRow. Click again and enter **CNT.**
41. Change the Name property to **PCNT**. Your table should look similar to the following example:



For display purposes, add extra spaces to each of the cells so that the data is not displayed directly next to a vertical edge.

42. Click on the box directly to the left of the Horizontal ruler. The entire table is highlighted.



If your ruler is not visible, select it from the View menu.

43. From the Properties, change Padding Left and Padding Right to Yes.
44. Change the Padding Size to **0.050**.
45. Press CTRL+F5 to execute your layout program. Click **Yes** when prompted to save the layout. The prompt form opens. Select Any(*) for the second parameter and click **Execute**. Your report should look similar to the following example:

	Person ID	Name	Birth Date	Gender
1	593694	Smith, Trina	01/01/01 00:00:00	Female
2	594976	Tremmel, Sue	07/29/23 00:00:00	Female
3	593758	Lee, Khim	03/10/28 00:00:00	Female
4	595361	Baker, Ginger	10/03/28 00:00:00	Female
5	593747	Limshue, Katia	02/17/30 00:00:00	Female
6	595125	Lou, Marta	07/08/30 00:00:00	Female
7	594102	Pill, Jillene	11/10/33 00:00:00	Female
8	593927	Boban, Leanne	06/18/35 00:00:00	Female
9	595232	Imara, Lauris	07/09/35 00:00:00	Female
10	593924	Dock, Alison	10/17/35 00:00:00	Female
11	594274	Teach, Kathleen	09/15/36 00:00:00	Female
12	594937	Linn, Kerri	08/21/37 00:00:00	Female
13	595112	Friends, Colleen	05/05/38 00:00:00	Female

46. Look at the last page of your report and verify that the total number of people are being calculated and displayed.

200	8773482	Revoir, Gene	12/12/23 00:00:00	Male
			TOTAL Person Count	200

47. Close the output display window.

    This report runs as a stand-alone program, however, we also want to call it from another layout program as a Sub-Report. It needs to be made a Sub-Report by changing its Report Properties.
48. From the Edit menu, select **Report Properties**. Select the Sub-Report option located in the Additional Report Writer Settings.
49. Save the file.

    We have created two layout programs that can be executed as stand-alone programs. These programs are independent of each other and allow you to retrieve information from the specific program. However, we have also marked these programs as Sub-Reports so that they can be called from another layout program to combine the information into one report. To demonstrate how to combine the reports, we will create a parent program to call each of the programs and display their information on one report.

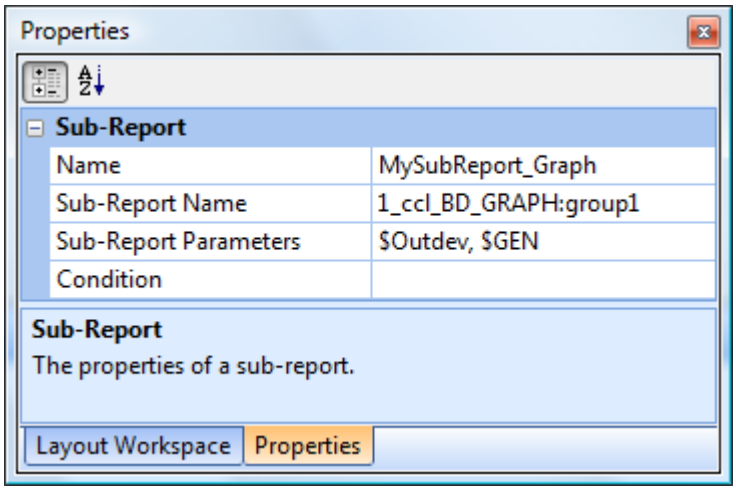
Step 4: Creating the Parent Layout Program

1. From the File menu, select **New**.
2. From the File Type List, select **Layout Program**.
3. In the Program Name box, enter **1_*your_initials*_Parent**, click **OK** and then click **Finish**.

   Copy the prompts from the 1_*your_initialsPerson_View layout program to 1your_initials*_Parent.
4. Select Transfer objects from the Tools menu and enter **1_*your_initials*_Person_View**.
5. Click **Browse** to find the program.
6. From Task menu, select **Copy Object**. Enter **1_*your_initials*_Parent** for the destination object name and click **OK**.

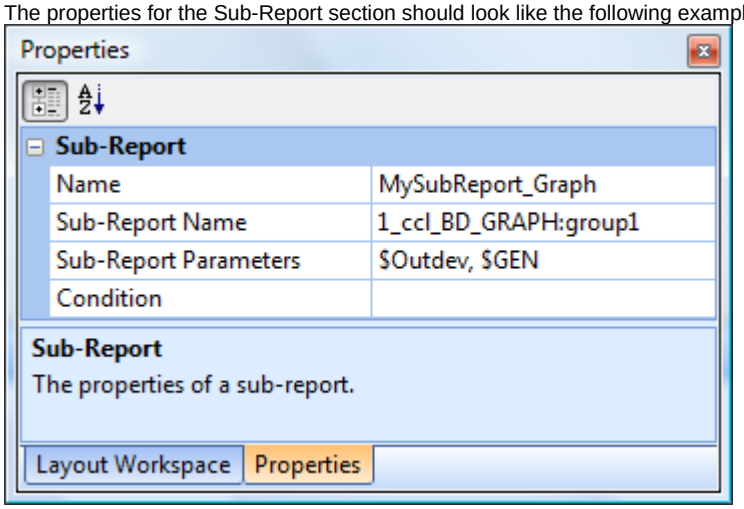
   In this layout program, the DetailSection is not tied to a query and is only executed one time. We can use this section to create the title for our report.
7. In the DetailSection, use the Label tool to add a report title, **Person Data - Birthdate Distribution Report**.
8. Use the format toolbar to set the following example:
    1. The font to Times
    2. The font size to 24
    3. Center the text
9. From the Edit menu, select **Insert** > **Sub-Report**. A new sub-report section named Sub-Report: MySubReport(0) is created.
10. Modify the Name property to **MySubReport_View**.
11. Modify the Sub-Report Name property to the program containing the table view, 1_*your_initials*_Person_View. Append **:group1** to the object name if necessary.
12. Enter **$Outdev, $GEN** for the Sub-Report Parameters property.

    The properties for the Sub-Report section should look like the following example:



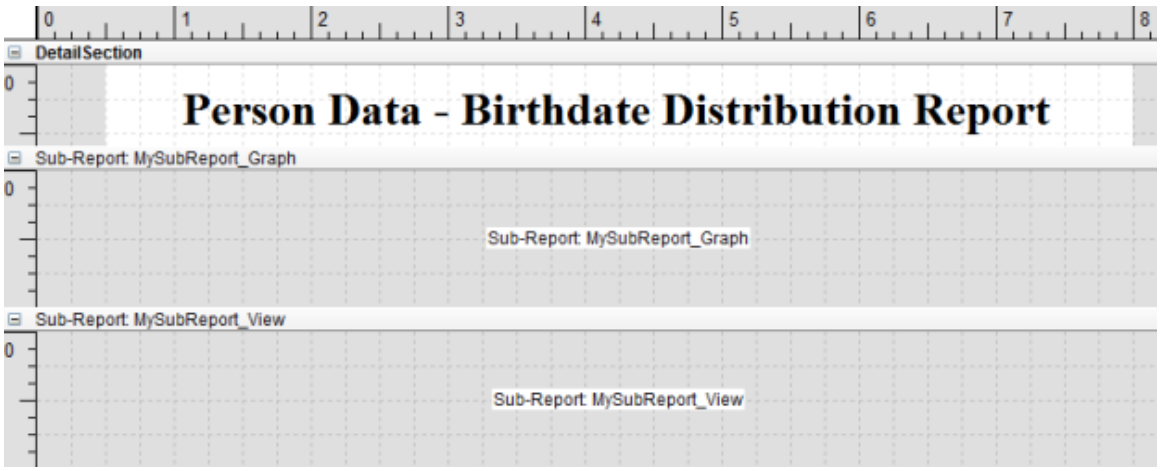
13. From the Edit menu, select **Insert** > **Sub-Report**. A new sub-report section named Sub-Report: MySubReport(1) is created.
14. Modify the Name property to **MySubReport_Graph**.
15. Modify the Sub-Report Name property to the program containing the graph, 1_*your_initials*_PRSN_Graph. Append :**group1** to the object name if necessary.
16. Enter **$Outdev $GEN** for the Sub-Report Parameters property.

    The properties for the Sub-Report section should look like the following example:



17. Validate that the sections are displayed in the layout in the following order and rearrange them if needed:
    1. DetailSection
    2. MySubreport_Graph
    3. MySubreport_View

    Your layout should look similar to the following example:



18. Press CTRL+F5 to execute your layout program. Click **Yes** when prompted to save the layout. The prompt form opens.
19. Select Female for the second parameter and click **Execute**. The report should have two titles, and then below them display the information from the two sub-reports. Your report should look similar to the following format:

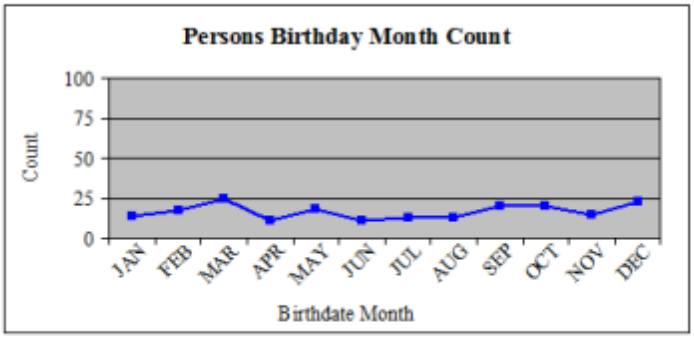
Person Data - Birthdate Distribution Report

Birthdate Distribution by Month

Page: 1 of 5

Female

JAN	14.00
FEB	17.00
MAR	25.00
APR	11.00
MAY	18.00
JUN	11.00
JUL	13.00
AUG	13.00
SEP	20.00
OCT	20.00
NOV	15.00
DEC	23.00



	Person ID	Name	Birth Date	Gender
1	593694	Smith, Trina	01/01/01 00:00:00	Female
2	594976	Tremmel, Sue	07/29/23 00:00:00	Female
3	593758	Lee, Khim	03/10/28 00:00:00	Female
4	595361	Baker, Ginger	10/03/28 00:00:00	Female
5	593747	Limshue, Katia	02/17/30 00:00:00	Female
6	595125	Lou, Marta	07/08/30 00:00:00	Female
7	594102	Pill, Jillene	11/10/33 00:00:00	Female
8	593927	Boban, Leanne	06/18/35 00:00:00	Female
9	595232	Imara, Lauris	07/09/35 00:00:00	Female
10	593924	Dock, Alison	10/17/35 00:00:00	Female
11	594274	Teach, Kathleen	09/15/36 00:00:00	Female

We need to control the logic so that when the 1*_your_initials*BD_Graph *is executed as a stand-alone program it displays the title _Birthdate Distribution Report*, however, when it executes as a Sub-Report from 1*_your_initials*Parent, *the title should display as _Person Data - Birthdate Distribution Report*. The conditional logic in 1*_your_initials*{_}BD_Graph controls the display of the title by checking for a variable populated when the program is called as a Sub-Report.

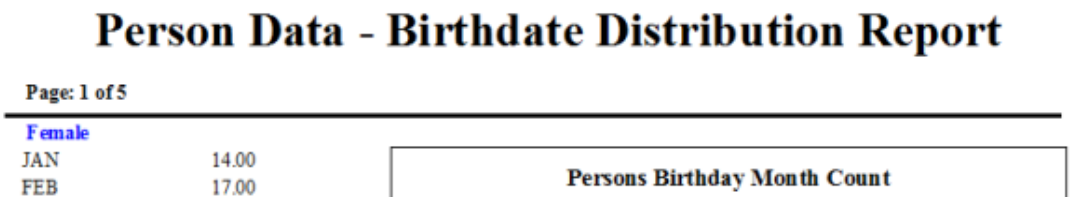
20. Open the layout program 1*_your_initials*_BD_Graph.
21. Access the Properties for the HeadreportSection and click the **ellipses** button on the Condition property.
22. Enter the following command and then click **OK**:

If the program is executed as a stand-alone program, the *bSubReport variable is 0 and the title from 1your_initials*_BD_Graph is rendered. If it is not 0, then the report is being called from a parent process and will not render the title.

23. Press CTRL+S to save and include the change.

Execute 1*_your_intials*_Parent to verify only one title is displayed.

24. Most likely, you now have multiple files open. Click on the tab for 1*_your_intials*_Parent and press CTRL+F5 to execute the program.
25. Select Female for the second parameter. Now your report should only have the title from the parent program displaying, similar to the following example:



26. Close the output window.

Execute your program for all genders.

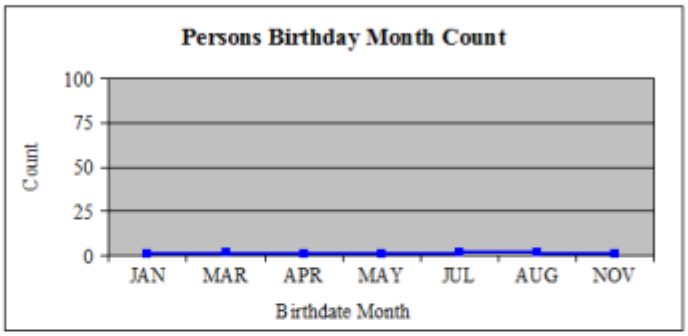
27. Press CTRL + F5 and select Any(*) for the second parameter. The report should have one title and display the graphs for each gender and the table view of the person information. The first page of the report will display the graphs, similar to the following example:

Person Data - Birthdate Distribution Report

Page: 1 of 6

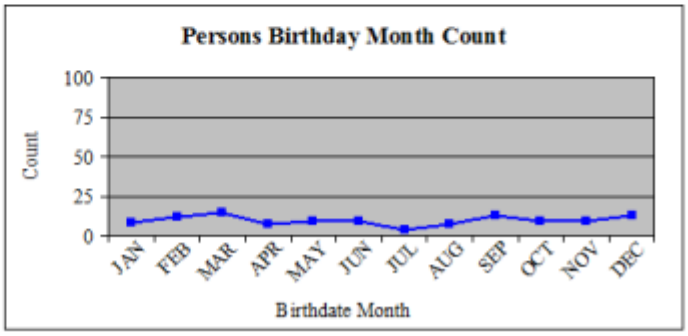
N/A

JAN	1.00
MAR	2.00
APR	1.00
MAY	1.00
JUL	2.00
AUG	2.00
NOV	1.00



Female

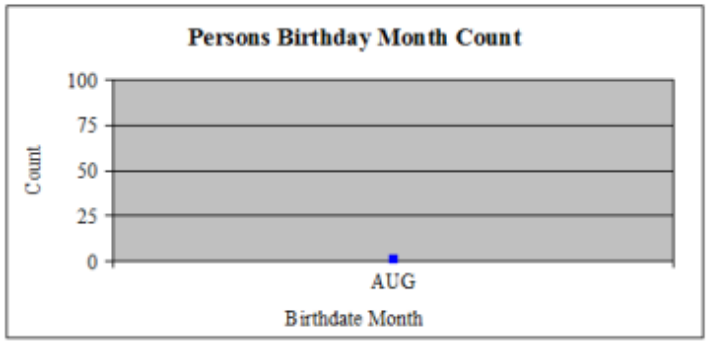
JAN	8.00
FEB	12.00
MAR	15.00
APR	7.00
MAY	9.00
JUN	9.00
JUL	4.00
AUG	7.00
SEP	13.00
OCT	9.00
NOV	9.00
DEC	13.00



There are a series of graphs for each gender, and once they are done, the next page displays the person information, similar to the following example:

Unknown

AUG	1.00

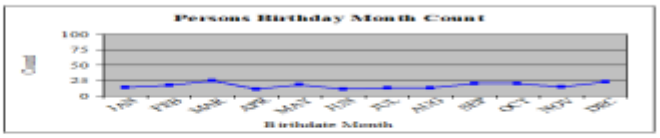


Person Data - Birthdate Distribution Report

Page: 1 of 5

Female

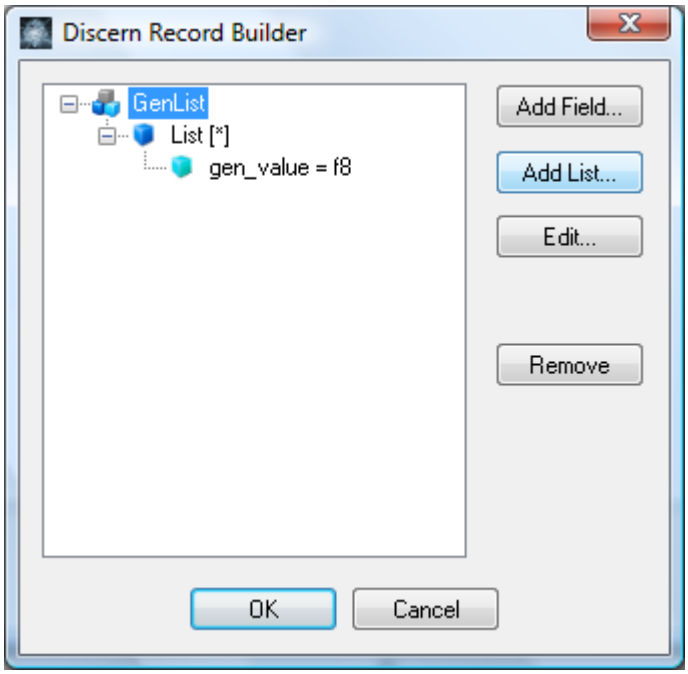
JAN	14.00
FEB	17.00
MAR	28.00
APR	11.00
MAY	18.00
JUN	11.00
JUL	13.00
AUG	13.00
SEP	20.00
OCT	20.00
NOV	15.00
DEC	23.00



	Person ID	Name	Birth Date	Gender
1	303604	Smith, Trina	01/01/01 00:00:00	Female
2	594976	Trammel, Sue	07/29/23 00:00:00	Female
3	503758	Lee, Khim	03/10/28 00:00:00	Female
4	595361	Baker, Ginger	10/03/28 00:00:00	Female
5	593747	Linsine, Katia	02/17/30 00:00:00	Female
6	303123	Lau, Maria	07/08/30 00:00:00	Female
7	594102	Pil, Allene	11/10/33 00:00:00	Female
8	503037	Reban, Leann	08/18/34 00:00:00	Female
9	595232	Imara, Laura	07/09/35 00:00:00	Female
10	593024	Deck, Alison	10/17/35 00:00:00	Female
11	594274	Teach, Kathleen	09/13/36 00:00:00	Female

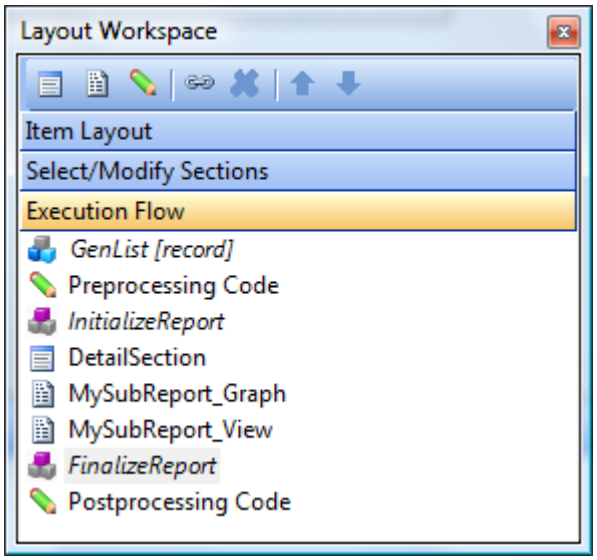
When Any(*) is chosen at the prompt, the series of graphs are displayed for each gender, and at the end displays the complete list of the person information used to create the graphs. We could re-arrange the display of the report so that for each gender the graph displays, and immediately following the gender specific person information that makes up the graph. We need to detect when the user select Any(*) at prompt and control the flow of the program to display the graph and person information for one gender, then create a page break and display the graph and person information for each subsequent gender. One way to accomplish this is to create looping logic to execute the Sub-Reports for each gender.
In the parent program we can build a record structure containing the code_value of the genders or gender that the user enters at the gender prompt. The sub-reports can execute for each of the code_values stored in the record structure using *For/Endfor* looping logic. Each code_value can be passed individually to the sub-reports.

28. With the 1_*your_intials*_Parent open, from the Tools menu, select Record Builder and click **New Record**.
29. Enter **GenList** for the Record Name and click **OK**.
30. Click **Add List** and enter **List** for the Name and *** for the Occurs. Click *OK**.
31. Select the newly-added List. Click **Add Field** and enter **gen_value** for the Name, select **f8** for the Type, and click **OK**. Your record structure should look like the following example:



32. Click **OK** to close the Discern Record Builder dialog box, and click **OK** again to close the Add Records dialog box.

    This record structure should be loaded before the Sub-Reports are executed. We can use the Preprocessing Code section, which provides the ability to place commands that are needed for processing in the program.
33. From the LayoutWorkspace, select Execution Flow. Your Execution Flow should resemble the following example:



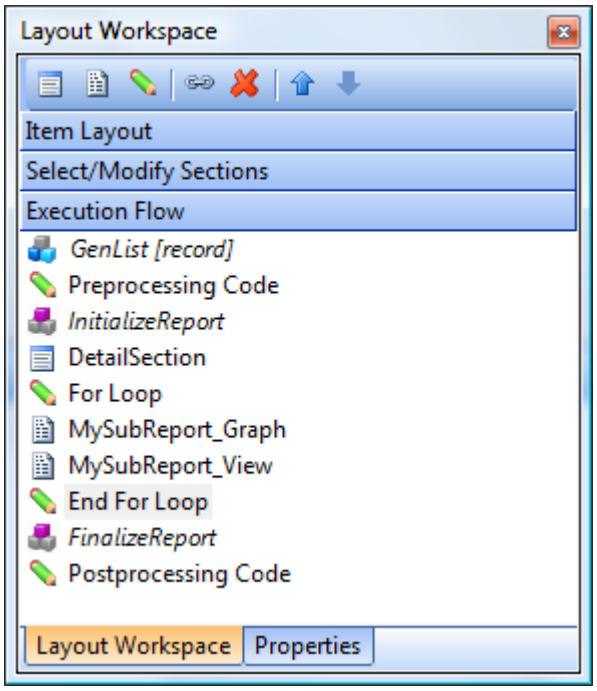
34. Double-click the Preprocessing Code, then copy and paste the following query which is used to load the record structure with one or multiple code_values:

35. Click **OK** to close the Layout Source[Preprocessing Code].

    The *For* loop needs to be wrapped around the both of the Sub-Reports.
36. From the Execution Flow, click **New Code Segment**.
37. Use the Up arrow to move the Code Segment above MySubReport_Graph.
38. Double-click the code segment, then copy and paste the following commands:

39. Click **OK** and access the Properties for the Code Segment.

40. Enter **For Loop** for the Name property.
41. From the Execution Flow, click **New Code Segment**.
42. Verify that the new code segment is below MySubReport_View. Move the segment if necessary.
43. Double-click on the code segment and enter **ENDFOR**.
44. Click **OK** and access the Properties for the Code Segment.
45. Enter **End For Loop** for the Name property. Your Execution Flow section should resemble the following example:



One final step is to modify the parameter passed to each of the Sub-Reports so that one code_value at a time is passed to the programs. The record item GenList->list[_MyList].gen_value contains the code_value.

46. Access the Properties for MySubReport_Graph and modify the Sub-Report Parameters to the following example:

    $Outdev, GenList->list[_MyList].gen_value
47. Access the Properties for MySubReport_View and modify the Sub-Report Parameters to the following example:

    $Outdev, GenList->list[_MyList].gen_value
48. Press CTRL+F5 and click **Yes** when prompted to save the layout.
49. Select Any(*) for the second parameter and click **Execute**. The report should display the graph and person information for one gender. At the end of the information for that gender, there should be a page break. Then the next page should show the graph and person information for the next gender.

Once you complete these steps, continue on to the next part of Use Discern Layout Builder, Manually Calling Layout Subroutines From Report Programs