

sebastianfernandes.com

[Creating a progress bar (Progress CHUI)]

While progress bars are common in any Windows environment, it is not difficult to construct a "simulated widget" in a Progress character-based application. The code is simple, but the result is an appealing and effective visual indicator.

In the sample code that complements this tutorial, there is little other than the code that renders the progress bar. As a result, the display zips along in under a second! However, your application does much more and including such an indicator could add some noticeable benefits.

1. Define the variables and frame which will form the basis of our display. Note the character variable array `ctrlProgressbar`.

```
DEF VAR ctrlProgressbar AS CHAR FORM "x" EXTENT 20 INIT " " NO-UNDO.
DEF VAR currPercent    AS INT                                NO-UNDO.
DEF VAR currRecord     AS INT                                NO-UNDO.
DEF VAR currCell       AS INT                                NO-UNDO.
DEF VAR displayedCell  AS INT                                NO-UNDO.

DEF FRAME fProgbar
  "Processing .... " AT 15
  currPercent FORM ">>9%" NO-LABEL
  SKIP(1)
  " [" ctrlProgressbar[1 FOR 20] NO-LABEL "]"
  WITH SIZE 47 BY 5 ROW 8 CENTERED OVERLAY.
```

2. Define and open the query. Note that the query is opened with the `PRESELECT` option – more on this in a minute.

```
DEFINE QUERY qOrder FOR
  customer FIELDS (customer.cust-num customer.name customer.phone),
  order FIELDS (order.order-num order.order-date),
  order-line FIELDS (order-line.line-num order-line.price order-line.qty),
  item FIELDS (item.item-num item.item-name item.cat-desc).
OPEN QUERY qOrder PRESELECT EACH customer,
  EACH order OF customer,
  EACH order-line OF order,
  EACH item OF order-line NO-LOCK.
```

3. Now, the result list can be processed.

```
GET FIRST qOrder.  
DO WHILE AVAILABLE(customer):  
    ...  
    GET NEXT qOrder.  
END.
```

4. With every iteration, we track our place in the recordset by incrementing the variable currRecord.

```
ASSIGN  
    currRecord = currRecord + 1  
    currPercent = (currRecord / NUM-RESULTS("qOrder") * 100)  
    currCell = INT(TRUNC(currPercent / 5,0)).
```

Because we specified PRESELECT in step 2 – which built the entire recordset immediately – we can take advantage of the NUM-RESULTS function to return the total number of rows. With these pieces of information, we can easily calculate our percentage progress.

The last thing we need to determine is how much of our progress bar should be coloured. The progress bar will be constructed from the character variable array ctrlProgressbar that we set up in step 1. Since we defined this as an array of twenty elements, we calculate variable currCell accordingly.

5. Having determined how much of our progress bar should be coloured, we adjust the elements of character variable array ctrlProgressbar using the COLOR statement with the MESSAGES colour phrase.

```
IF (displayedCell < currCell)  
THEN DO:  
    displayedCell = currCell.  
    COLOR DISPLAY MESSAGES  
    ctrlProgressbar[displayedCell] WITH FRAME fProgbar.  
END.
```

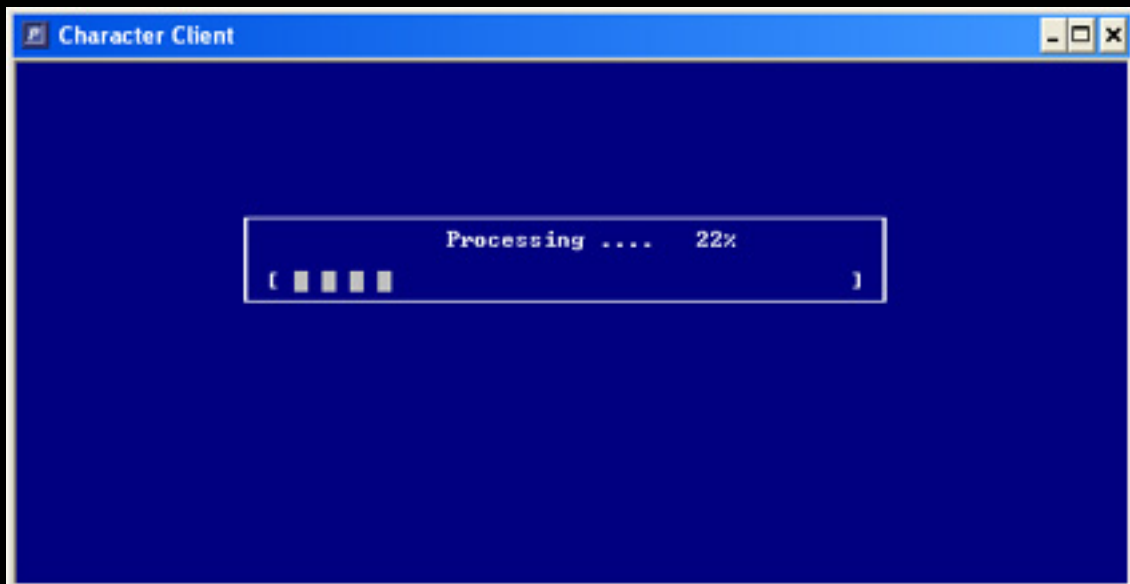
N.B. This procedure assumes there will be at least twenty records in our result set. If there are less than this number, some of the cells in our progress bar may be skipped. If you expect to process recordsets of less than twenty rows, you can either reduce the size of the array variable as required or modify the block above to colour all cells up to and including our target cell as follows:

```
IF (displayedCell < currCell) OR (currCell = 20)
THEN DO displayedCell = 1 TO currCell:
    COLOR DISPLAY MESSAGES
    ctrlProgressbar[displayedCell] WITH FRAME fProgbar.
END.
```

6. Finally, we display the current state of frame fProgbar.

```
DISP currPercent ctrlProgressbar[1 FOR 20] WITH FRAME fProgbar.
```

7. ... And we're done!



The completed procedure is as follows:

```

/* progbar.p */

DEF VAR ctrlProgressbar AS CHAR FORM "x" EXTENT 20 INIT " " NO-UNDO.
DEF VAR currPercent     AS INT                                NO-UNDO.
DEF VAR currRecord      AS INT                                NO-UNDO.
DEF VAR currCell        AS INT                                NO-UNDO.
DEF VAR displayedCell   AS INT                                NO-UNDO.

DEF FRAME fProgbar
  "Processing .... " AT 15
  currPercent FORM ">>9%" NO-LABEL
  SKIP(1)
  " [" ctrlProgressbar[1 FOR 20] NO-LABEL "]"
WITH SIZE 47 BY 5 ROW 8 CENTERED OVERLAY.

DEFINE QUERY qOrder FOR
  customer FIELDS (customer.cust-num customer.name customer.phone),
  order FIELDS (order.order-num order.order-date),
  order-line FIELDS (order-line.line-num order-line.price order-line.qty),
  item FIELDS (item.item-num item.item-name item.cat-desc).
OPEN QUERY qOrder PRESELECT EACH customer,
  EACH order OF customer,
  EACH order-line OF order,
  EACH item OF order-line NO-LOCK.

GET FIRST qOrder.
DO WHILE AVAILABLE(customer):

  ASSIGN
    currRecord = currRecord + 1
    currPercent = (currRecord / NUM-RESULTS("qOrder") * 100)
    currCell = INT(TRUNC(currPercent / 5,0)).

  IF (displayedCell < currCell)
  THEN DO:
    displayedCell = currCell.
    COLOR DISPLAY MESSAGES
    ctrlProgressbar[displayedCell] WITH FRAME fProgbar.
  END.

  DISP currPercent ctrlProgressbar[1 FOR 20] WITH FRAME fProgbar.

  /* <insert whichever processes need to be executed here> */

  GET NEXT qOrder.

END.

```