

```

////////////////////////////////////
//
// APPSYS.PRG
//
// Copyright:
//   Alaska Software, (c) 1997-2001. All rights reserved.
//
// Contents:
//   AppSys() - Creates default application window
//
// Remarks:
//   This file is part of the XppRt0.lib.
//
// Syntax:
//   The function AppSys() is called automatically during
//   the programm startup.
//
////////////////////////////////////

#include "xbp.ch"

*****
* Function AppSys() to create default output devices
*****

PROCEDURE AppSys()

#define DEF_ROWS    25
#define DEF_COLS    80
#define DEF_FONTHEIGHT 16
#define DEF_FONTWIDTH 8

LOCAL oCrt, nAppType := AppType()
LOCAL aSizeDesktop, aPos

DO CASE

    // PM Mode: create an XbpCrt instance
    CASE nAppType == APPTYPE_PM

        // Compute window position (center window
        // on the Desktop)
        aSizeDesktop := AppDesktop():currentSize()
        aPos          := { (aSizeDesktop[1]-(DEF_COLS * DEF_FONTWIDTH)) /2, ;
                          (aSizeDesktop[2]-(DEF_ROWS * DEF_FONTHEIGHT)) /2 }

        // Create XbpCRT object
        oCrt := XbpCrt():New ( NIL, NIL, aPos, DEF_ROWS, DEF_COLS )
        oCrt:FontWidth := DEF_FONTWIDTH
        oCrt:FontHeight := DEF_FONTHEIGHT
        oCrt:title     := AppName()
        oCrt:icon      := 1
        oCrt:SysMenu   := .T.

    #ifdef __OS2__
        oCrt:FontName := "System VIO"
    #endif
    #ifdef __WIN32__

```

```
oCrt:FontName := "Alaska Crt"
#endif
oCrt:Create()

// Init Presentation Space
oCrt:PresSpace()
oCrt:setModalState( XBP_DISP_APPMODAL )

// XbpCrt gets active window and output device
SetAppWindow ( oCrt )

// VIO or NOVIO Mode: create a RootCrt instance
CASE nAppType == APPTYPE_VIO .OR. nAppType == APPTYPE_NOVIO

// Create RootCrt object
//
// The IVar :CreateBuffer determines the behaviour of the RootCrt
// when it is created in a shell window.
//
// :CreateBuffer == .T. : The RootCrt will create a new
//                        screen buffer with the same size
//                        as the console window. This causes
//                        the effect that outputs with
//                        OutStd() or printf() wont be
//                        displayed.
// :CreateBuffer == .F. : The RootCrt will use the screen
//                        buffer of the window where it runs
//                        in.
oCrt := RootCrt():New()
oCrt:CreateBuffer := .T.
oCrt:Create()

// RootCrt gets active window and output device
SetAppWindow ( oCrt )

ENDCASE

RETURN

// EOF
```