```
unit HUDirNameEntry;

{$mode objfpc}{$H+}

//========================================================================================
//
// Unit : HUDirNameEntry.pas
//
// Description :
//
// Called By :
//
// Calls :  HUConstants
//          HUMessageBoxes : HUConfirmMsgYN
//          HUValidations : ValidDirectoryCharacter
//
// Ver. : 1.0.0
//
// Date : 20 Jan 2019
//
//========================================================================================

interface

uses
  Buttons, Classes, SysUtils, FileUtil, Forms, Controls, Graphics, Dialogs, StdCtrls,
  // Application units
  AppSettings,
  // HULibrary units
  HUConstants, HUMessageBoxes, HUValidations;

type

  { TdlgHUDirNameEntry }

  TdlgHUDirNameEntry = class(TForm)
    bbtCancel: TBitBtn;
    bbtOK: TBitBtn;
    edtDirName: TEdit;
    procedure bbtCancelClick(Sender: TObject);
    procedure bbtOKClick(Sender: TObject);
    procedure edtDirNameKeyPress(Sender: TObject; var Key: char);
    procedure edtDirNameKeyUp(Sender: TObject; var Key: Word; Shift: TShiftState );
    procedure FormCreate(Sender: TObject);
    procedure FormShow(Sender: TObject);
  private
    vstrOriginalDirName : string;
    fDirName : string;
    fDirPath : string;
    function GetDirName : string;
    procedure SetDirName(Dir : string);
    function GetDirPath : string;
    procedure SetDirPath(DirPath : string);
  public
    property pDirName : string read GetDirName write SetDirName;
    property pDirPath : string read GetDirPath write SetDirPath;
  end;// TdlgHUDirNameEntry

var
  dlgHUDirNameEntry: TdlgHUDirNameEntry;
```

```

implementation

{$R *.lfm}

//========================================================================================
//          PRIVATE CONSTANTS
//========================================================================================
const
  cintMinDirNameCharacters = 1;
  cintMaxDirNameCharacters = 20;

//========================================================================================
//          PUBLIC CONSTANTS
//========================================================================================

//========================================================================================
//          PRIVATE VARIABLES
//========================================================================================

//========================================================================================
//          PUBLIC VARIABLES
//========================================================================================

//========================================================================================
//          PRIVATE ROUTINES
//========================================================================================

//========================================================================================
//          PUBLIC ROUTINES
//========================================================================================

//========================================================================================
//          PROPERTY ROUTINES
//========================================================================================
function TdlgHUDirNameEntry.GetDirName: string;
begin
   Result := fDirName;
end;// function TdlgHUDirNameEntry.GetDirName

//----------------------------------------------------------------------------------------
procedure TdlgHUDirNameEntry.SetDirName(Dir: string);
begin
    fDirName := Dir;
end;// procedure TdlgHUDirNameEntry.SetDirName

//========================================================================================
function TdlgHUDirNameEntry.GetDirPath: string;
begin
   Result := fDirPath;
end;// function TdlgHUDirNameEntry.GetDirPath

//----------------------------------------------------------------------------------------
procedure TdlgHUDirNameEntry.SetDirPath(DirPath: string);
begin
    fDirPath := DirPath;
end;// procedure TdlgHUDirNameEntry.SetDirPath

//========================================================================================
//          MENU ROUTINES
```

```
//=========================================================================================

//=========================================================================================
//          COMMAND BUTTON ROUTINES
//=========================================================================================

//=========================================================================================
//          CONTROL ROUTINES
//=========================================================================================
procedure TdlgHUDirNameEntry.bbtCancelClick(Sender: TObject);
begin
  if HUConfirmMsgYN('Confirm', 'Do you really want to Cancel ?') = mrNo then
  begin
    edtDirName.SetFocus;
    ModalResult := mrNone;
    Exit;
  end;

  ModalResult := mrCancel;

end;// procedure TdlgHUDirNameEntry.bbtCancelClick

//-----------------------------------------------------------------------------------------
procedure TdlgHUDirNameEntry.bbtOKClick(Sender: TObject);
var
  vstrDirFullPath : string;
begin

  if edtDirName.Text = '' then
  begin
    showmessage('You Must Enter a Valid Directory Name');
    edtDirName.SetFocus;
    ModalResult := mrNone;
    Exit;
  end;// if edtDirName.Text = ''

  pDirName := edtDirName.Text;

  // Confirm Dir DOES NOT already exist
  pDirPath := pDirPath + '\' + pDirName;

  If DirectoryExists(pDirPath) then
  begin
    showmessage('Directory ' + pDirPath + ' Already Exists!');
    edtDirName.Text := '';
    edtDirName.SetFocus;
    ModalResult := mrNone;
    Exit;
  end;// If DirectoryExists(pDirPath)

end;// procedure TdlgHUDirNameEntry.bbtOKClick

//=========================================================================================
procedure TdlgHUDirNameEntry.edtDirNameKeyPress(Sender: TObject; var Key: char);
begin
  Key := HUValidations.ValidDirectoryCharacter(Key);
end;// procedure TdlgHUDirNameEntry.edtDirNameKeyPress

//-----------------------------------------------------------------------------------------
procedure TdlgHUDirNameEntry.edtDirNameKeyUp(Sender: TObject; var Key: Word;
  Shift: TShiftState);
begin
  if Length(edtDirName.Text) < cintMinDirNameCharacters then
    edtDirName.Color := bclrError
  else
    edtDirName.Color := bclrOk;
end;// procedure TdlgHUDirNameEntry.edtDirNameKeyUp

//========================================================================================
//          FILE ROUTINES
//========================================================================================

//========================================================================================
//          FORM ROUTINES
//========================================================================================
procedure TdlgHUDirNameEntry.FormCreate(Sender: TObject);
begin
  edtDirName.MaxLength := cintMaxDirNameCharacters;
end;// procedure TdlgHUDirNameEntry.FormCreate

//========================================================================================
procedure TdlgHUDirNameEntry.FormShow(Sender: TObject);
begin

  edtDirName.Text := PDirName;

  if Length(edtDirName.Text) < cintMinDirNameCharacters then
    edtDirName.Color := bclrError
  else
    edtDirName.Color := bclrOk;

  edtDirName.SetFocus;

end;// procedure TdlgHUDirNameEntry.FormShow

//========================================================================================
end.// unit HUDirNameEntry

```