How to make a single instance of your application

```
unit USingleInst;

interface

uses
  Windows,SysUtils,Classes;

 type
   TSingleInst=class(TComponent)
   public
    constructor Create(AOwner:TComponent);override;
    destructor Destroy();override;
   end;

procedure Register;

implementation

var
  PrvInst :TSingleInst =nil;
  MutHandle :THandle;

procedure Register;
begin
  RegisterComponents('Exemples', [TSingleInst]);
end;

procedure IniInstance();
var
  Erreur: Integer;
  Mut :string;
begin
  Mut := StringReplace(ParamStr(0),'\','',[rfReplaceAll]);
  SetLastError(NO_ERROR);

  MutHandle :=CreateMutex(nil, False,PChar(Mut));
  Erreur := GetLastError;

  if (Erreur = ERROR_ALREADY_EXISTS) or (Erreur = ERROR_ACCESS_DENIED) then
  begin
    MutHandle := 0;
    Halt;
  end;
end;

{ TSingleInst }

constructor TSingleInst.Create(AOwner: TComponent);
begin
  if PrvInst = nil then
    PrvInst := inherited Create(AOwner)
  else
    raise Exception.Create('Only one instance of TSingleInst can be used');
end;

destructor TSingleInst.Destroy;
begin
  if PrvInst = Self then
     PrvInst := nil;
  inherited;
end;

initialization
  IniInstance();

finalization
  if MutHandle  0 then
    CloseHandle(MutHandle);
```

```
end.
```

Author:	Montor
Contributor:	Montor
Added:	2011-01-02
Last updated:	2011-01-04