
Posted by on Mon, 27 Feb 2023 13:35:10 GMT
[View Forum Message](#) <> [Reply to Message](#)

```
unit uEgg;
```

```
interface
```

```
uses  
  uChicken;
```

```
type
```

```
  TEgg = class  
  private  
    FChicken: TChicken;  
  published  
    property Chicken: TChicken read FChicken write FChicken;  
  end;
```

```
implementation
```

```
  ...  
end.
```

```
unit uChicken;
```

```
interface
```

```
uses  
  uEgg;
```

```
type
```

```
  TChicken = class  
  private  
    FEgg: TEgg;  
  published  
    property Egg: TEgg read FEgg write FEgg;  
  end;
```

```
implementation
```

```
  ...  
end.
```

unit incubator;

interface

type

TAbstractChicken = class;

TAbstractEgg = class

```
public
  property Chicken: TAbstractChicken read GetChicken write SetChicken;
  function GetChicken:TAbstractChicken; virtual; abstract;
  procedure SetChicken( Chicken:TAbstractChicken ); virtual; abstract;
end;
```

TAbstractChicken = class

```
public
  property Egg: TAbstractEgg read GetEgg write SetEgg;
  function GetEgg:TAbstractEgg; virtual; abstract;
  procedure SetEgg( Egg:TAbstractEgg ); virtual; abstract;
end;
```

unit uEgg;

interface

uses

incubator

;

type

```
TEgg = class( TAbstractEgg )
  private
    fChicken : TAbstractChicken;
  public
    function GetChichen:TAbstractChicken; override;
    procedure SetChicken( Chicken:TAbstractChicken ); override;
  published
    property Chicken;
end;
```

implementation

```
uses
  uChicken
;
```

```
TEgg.GetChicken:TAbstractChicken;
begin
  Result := fChicken;
end;
```

```
TEgg.SetChicken( aChicken:TAbstractChicken );
begin
  fChicken := aChicken;
end;
```

unit uChicken;

interface

```
uses
  incubator
;
```

type

```
TChicken = class( TAbstractChicken )
  private
    fEgg : TAbstractEgg;
    procedure DoSomethingWithEgg;
  public
    function GetEgg:TAbstractEgg; override;
    procedure SetEgg( aEgg:TAbstractEgg ); override;
  published
    property Egg;
end;
```

implementation

```
uses
  uEgg
;
```

```
TChicken.GetEgg:TAbstractEgg;
begin
  Result := fEgg;
end;
```

```
TChicken.SetEgg( aEgg:TAbstractEgg );
begin
  fEgg := aEgg;
end;
```

```
procedure TChicken.DoSomethingWitnEgg;
begin
  DoSomething( TEgg( fEgg ) );
end;
```

unit coop;

interface

```
uses
  uChicken
  , uEgg
;
```

```
function CreateChickenWithEgg:TChicken;
```

implementation

```
function CreateChickenWithEgg:TChicken;
var
  IChicken : TChicken;
  IEgg     : TEgg;
begin
  IChicken := TChicken.Create;
  IEgg     := TEgg.Create;
  IChicken.Egg := IEgg;
  IEgg.Chicken := IChicken;
  Result      := IChicken;
end;
```

Posted by _____ on Tue, 28 Feb 2023 11:36:42 GMT

[View Forum Message](#) <> [Reply to Message](#)
