

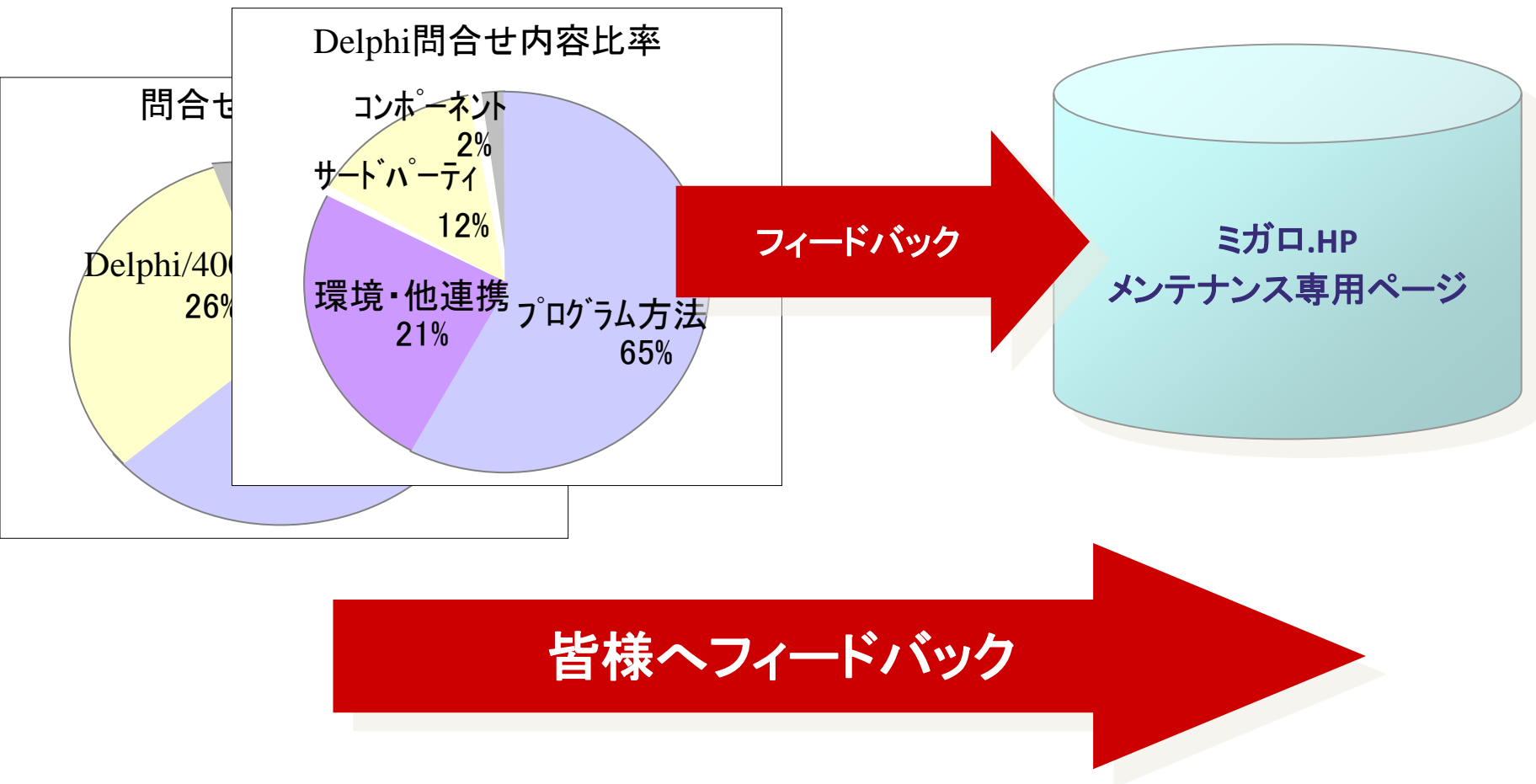
【セッションNo. 3】

知って得する！現役ヘルプデスクが答える
Delphiテクニカルエッセンス 10.0

株式会社ミガロ.
RAD事業部 技術支援課
吉原 泰介

■ 第1回資料

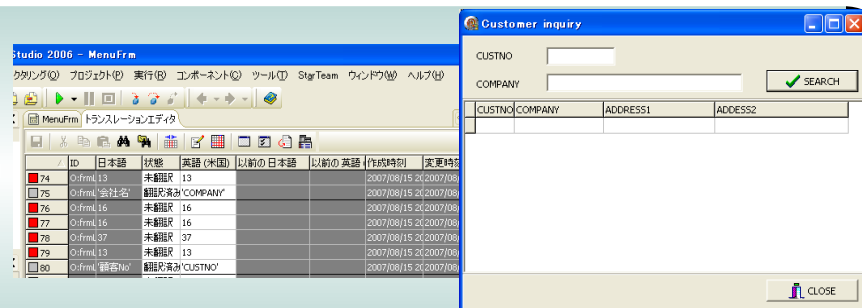
✉ サポートには月100件弱、年間1000件以上のお問合せ



■ これまでのセッションを振り返ってみよう！

第1回

- 【Q1】Excel出力パフォーマンスを改善したい！
- 【Q2】DBGridでこんなことがしたい！
- 【Q3】クライアント端末の情報を取得したい！
- 【Q4】多言語環境に対応するには？



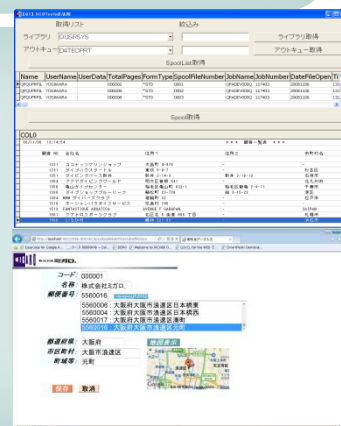
第2回

- 【Q1】DBGridで簡単にソートを行いたい！
- 【Q2】VB-ReportVer3.0で効率よく出力したい！
- 【Q3】TreeViewでメニューを制御したい！
- 【Q4】一覧明細にローカル画像を表示したい！
- 【Q5】StringGridで文字列を縦表示したい！



第3回

- 【Q1】オブジェクトのリストを取得するには？
- 【Q2】SQLでメンバを扱うには？
- 【Q3】デバッグモードを判断するには？
- 【Q4】他プログラムからExeを起動するには？
- 【Q5】OS/400 V6R1上でDelphi/400は使えますか？
- 【QA】Cookieを利用した制御
- 【QB】IWTextの右寄せ表示
- 【QC】開発モードの変更方法
- 【QD】IIS上での動作の違い



■ これまでのセッションを振り返ってみよう！

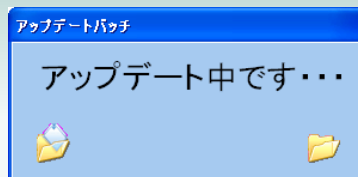
第4回

- 【Q1】RUNQRYをDelphi/400から利用しよう！
- 【Q2】ファンクションキー/Tabキー入力を制御しよう！
- 【Q3】ClientDataSetでグループ制御しよう！
- 【Q4】CSVファイルをTableで読み込もう！
- 【Q5】ExcelファイルをADOTableで読み込もう！

REGION	STORE	ADDR	TELNO
北海道	〇×店	札幌市〇×区1-1	xxxx-xxxx-xxxx
北海道	××店	札幌市××区2-2	xxxx-xxxx-xxxx
北海道	△△店	札幌市△△区3-3	xxxx-xxxx-xxxx
北海道	〇〇店	札幌市〇〇区2-1	xxxx-xxxx-xxxx
北海道	×〇店	札幌市×〇区1-4	xxxx-xxxx-xxxx
北海道	△×店	札幌市△×区3-2	xxxx-xxxx-xxxx
北海道	〇△店	札幌市〇△区1-1	xxxx-xxxx-xxxx
北海道	〇×店	旭川市〇×区1-2	xxxx-xxxx-xxxx
北海道	××店	旭川市××区4-2	xxxx-xxxx-xxxx
北海道	〇×店	小樽市〇×区3-3	xxxx-xxxx-xxxx
北海道	〇〇店	北見市〇〇区4-4	xxxx-xxxx-xxxx
青森県	〇×店	青森市〇×区1-1	xxxx-xxxx-xxxx
青森県	△△店	八戸市△△区2-2	xxxx-xxxx-xxxx
青森県	〇×店	八戸市〇×区5-5	xxxx-xxxx-xxxx
岩手県	××店	盛岡市××区2-3	xxxx-xxxx-xxxx
宮城県	〇×店	仙台市〇×区3-3	xxxx-xxxx-xxxx
宮城県	×△店	仙台市×△区4-2	xxxx-xxxx-xxxx
宮城県	〇〇店	古川市〇〇区1-3	xxxx-xxxx-xxxx
秋田県	△×店	秋田市△×区3-3	xxxx-xxxx-xxxx

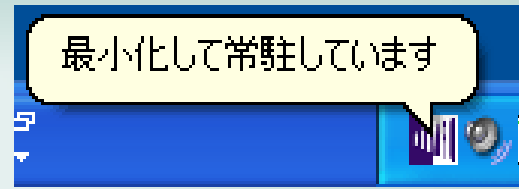
第5回

- 【Q1】サイレントインストールを行うには？
- 【Q2】Flash動画を扱うには？
- 【Q3】AnsiStringとUnicodeStringの違いは？
- 【Q4】複数アプリ間でパラメータを受け渡すには？



第6回

- 【Q1】IBM i を使ったファイルのアップロード/ダウンロード
- 【Q2】トレイアイコンを利用したアプリケーションの常駐
- 【Q3】リモートデバッガで運用PCのデバッグ調査
- 【Q4】Delphi/400 V2010 新機能紹介- ジェスチャアプリケーション作成-



■ これまでのセッションを振り返ってみよう！

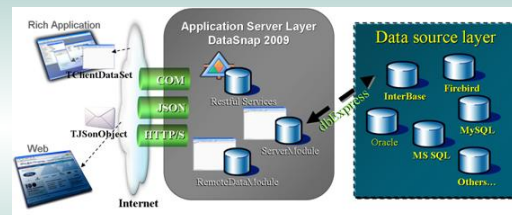
第7回

- 【Q1】OpenOfficeOrgを利用するには？
- 【Q2】メール送信機能を作成するには？



第8回

- 【Q1】ExcelファイルをPDFに変換するには？
- 【Q2】EXEのサイズを小さくするには？
- 【Q3】C/Sアプリで3階層接続を行うには？



ドキュメントライブラリ

Projects

名前	更新日時	サイズ
Project1.exe	2011/06/13 20:27	21 KB

第9回

- 【Q1】PageControl応用テクニック
- 【Q2】DLLモジュールの開発手法

2315	ダイビングショップマーメイド
2354	マリンセンター楽柿浦
2975	ダイブリゾートバビヤ
2984	サンセットダイビングサービス
3041	マリンショップアーク



第10回

- 【Q1】画面をドッキングするには？
- 【Q2】データを暗号化するには？

全40トピック
413Pの技術情報公開

【アジェンダ】



お客様より年間1,000件以上お問合せ頂いている
テクニカルサポートからの技術フィードバック！

12月	11月	10月	9月	8月	7月	6月	5月	4月	3月	2月	1月	12月	11月	10月	9月	8月	7月	6月	5月	4月	3月	2月	1月
...

Q1. 画面をドッキングするには？

ドッキングの基本

応用テクニック

Q2. データを暗号化するには？

```
305A06092B0601040182375803A04D304B060A2B060104018  
237580301A03D303B02030200010202660202020080040855  
9FC1759553FF0504101B8177242BB3F8AE8B1F368747368C02  
04102520185443074795D13D9DF792A42F72
```

簡単な暗号化

CAPICOMを利用した高度な暗号化

■ Q1. 画面をドッキングするには？



【ご質問】

複数画面を結合したり、分離する操作を実現したいのですが、どのようなコーディング必要ですか？



【回答】

Delphi/400にはドッキングと呼ばれる機能が用意されています。

ドッキング機能を利用すると、画面の結合や分離をプロパティの設定で制御することができますので、コーディングはほとんど必要ありません。

■ Q1. 画面をドッキングするには？

● ドッキング機能例

複数画面を1画面にドッキング(結合)/分離できる

The screenshot displays a financial software interface with several docked windows. The main window shows a budget summary table for May. Other docked windows include a detailed budget table, a summary table, and a bar chart. Red arrows indicate the docking process.

科目	予定	実績	達成率
総売上高(=B+E)++++ A	21,500,000	20,671,500	96.15%
機械売上高 B			
受取口銭 C			
機械収益(=C).... D			
仕切型売上高 E	21,500,000	20,671,500	96.15%
仕切型売上原価 F	17,200,000	17,108,100	99.47%
仕切型収益(=E-F) G	4,300,000	3,590,000	83.49%
付替口銭..... H			
総収益(=D+G+H)++++ K	4,300,000	3,590,000	83.49%
経費合計(=M+O)++++ L	2,238,000	2,803,000	125.25%
直接費小計..... M	1,390,000	1,968,000	141.58%
通信費	10,000	38,400	384.00%
交際・会議費			
旅費・交通費	50,000	82,500	165.00%
運賃	5,000	-14,000	-280.00%
荷造費		137,000	
合計金額	62,500		

科目	予定	実績(確定済)	達成率
総売上高(=B+E)++++ A	18,500,000	19,162,500	103.58%
機械売上 受取口銭			
機械収益(=C).... D			
仕切型売上高 E	18,500,000	19,162,500	103.58%
仕切型売上原価 F	15,900,000	15,402,700	96.87%
仕切型収益(=E-F) G	2,600,000	3,866,400	148.71%
付替口銭..... H			
総収益(=D+G+H)++++ K	2,600,000	3,866,400	148.71%
経費合計(=M+O)++++ L	1,933,000	3,300,000	170.72%
直接費小計..... M	819,000	2,240,000	273.50%
通信費	30,000	40,300	134.33%
交際・会議費			
旅費・交通費	60,000	153,800	256.33%
運賃	50,000	223,000	446.00%

科目	予定	実績(確定済)	達成率
総売上高(=B+E)++++ A	302,000,000	320,273,500	106.05%
機械売上高 B			
受取口銭 C			
機械収益(=C).... D			
仕切型売上高 E	302,000,000	304,339,200	100.77%
仕切型売上原価 F	254,600,000	257,269,900	101.05%
仕切型収益(=E-F) G	47,400,000	48,862,700	103.09%
付替口銭..... H			
総収益(=D+G+H)++++ K	47,400,000	48,862,700	103.01%
経費合計(=M+O)++++ L	27,192,000	33,649,000	123.75%
直接費小計..... M	17,181,000	24,124,000	140.41%
通信費		350,000	419.40%
交際・会議費		695,000	838.80%
旅費・交通費		336,000	474.02%
運賃		670,000	810.00%

摘要	金額	照会No.	照会先
WAN専用線料金	6,000	HIVZ980KI2	001450
8601基本料金(11/03)	700	HDN5801	001450
8601コレクトコール料	700	HDN5801	001450
8601電報料(11/03)	1,200	HDN5801	001450
2828基本料金(11/03)	5,000	HDN10485	001450
2828通話料(11/03)	1,800	HDN10485	001450
8602基本料金(11/03)	4,700	HDN64913	001450
8602通話料(11/03)	1,100	HDN64913	001450
2898基本料金(11/03)	5,000	HDN99730	001450
2898通話料(11/03)	200	HDN99730	001450
8603FAX使用料(11/03)	5,300	HDN5880	001450
(11/03)通話料配賦	2,000	HDN2031	001450
(11/03)通話料配賦	4,700	HDN2032	001450
(11/03)通話料配賦	400	HDN2033	001450
(11/03)通話料配賦	1,500	HDN7303	001450

月	予定	実績
4月	18,500,000	19,162,500
5月	21,500,000	20,671,500
6月	21,500,000	20,671,500
7月	21,500,000	20,671,500
8月	21,500,000	20,671,500
9月	21,500,000	20,671,500
10月	21,500,000	20,671,500
11月	21,500,000	20,671,500
12月	21,500,000	20,671,500
1月	21,500,000	20,671,500
2月	21,500,000	20,671,500
3月	21,500,000	20,671,500

■ Q1. 画面をドッキングするには？

- ドッキングの仕組み（構成）

ドッキング先（親）

【Q1】画面をドッキングするには？

✓ 受注情報 ✓ 取引先情報 ✓ 担当者情報 ✓ 在庫情報

閉じる(C)

ドッキング元（子）

受注情報

受注番号 MG00264
受注日 2012/06/12
受注品番 SU02-GF
取引先 0082
伝票区分 E2
取引区分 直販

倉庫区分 01
納入先 0082
備考 6/21までに納品

閉じる(C)

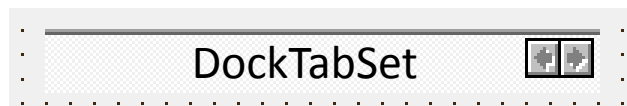
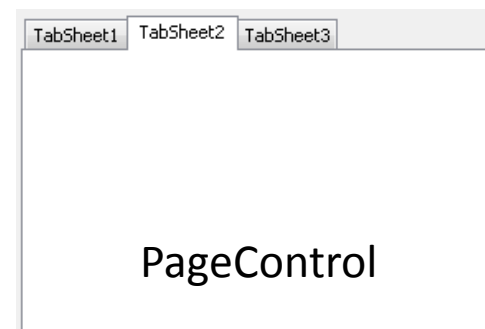
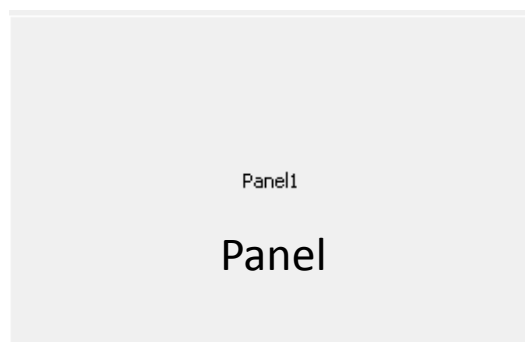
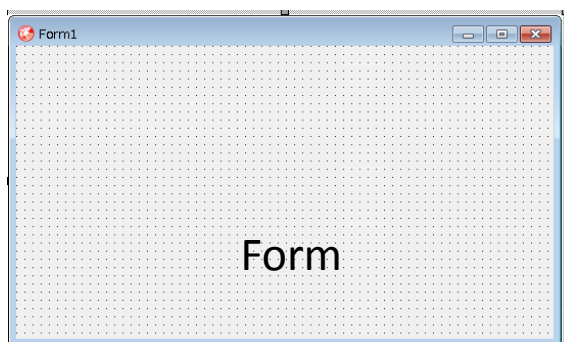
結合/分離が自由

■ Q1. 画面をドッキングするには？

- ドッキングの仕組み（利用できるコンポーネント）

ドッキング先（親）となれるコンポーネント

ドッキング先にはコンテナコンポーネント（土台）を使うことができます。

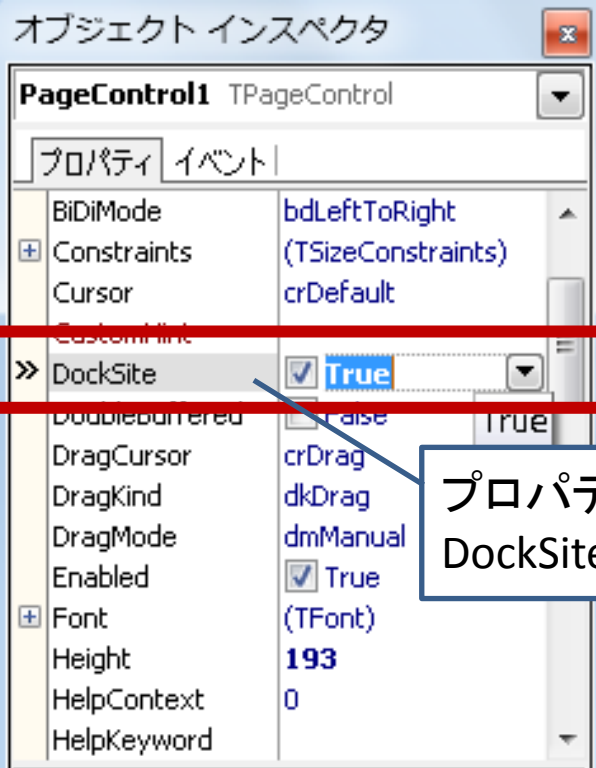


・・・などの
コンテナコンポーネント

■ Q1. 画面をドッキングするには？

- ドッキングの仕組み(プロパティの制御)

ドッキング先(親)の設定



オブジェクト インспекタ

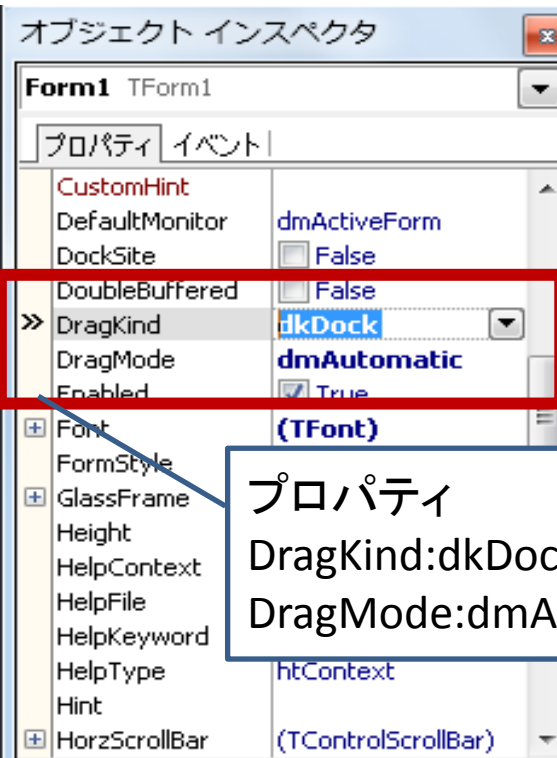
PageControl1 TPageControl

プロパティ イベント

BiDiMode	bdLeftToRight
Constraints	(TSizeConstraints)
Cursor	crDefault
CustomHint	
>> DockSite	<input checked="" type="checkbox"/> True
DoubleBuffered	<input type="checkbox"/> False
DragCursor	crDrag
DragKind	dkDrag
DragMode	dmManual
Enabled	<input checked="" type="checkbox"/> True
Font	(TFont)
Height	193
HelpContext	0
HelpKeyword	

プロパティ
DockSite:True

ドッキング元(子)の設定



オブジェクト インспекタ

Form1 TForm1

プロパティ イベント

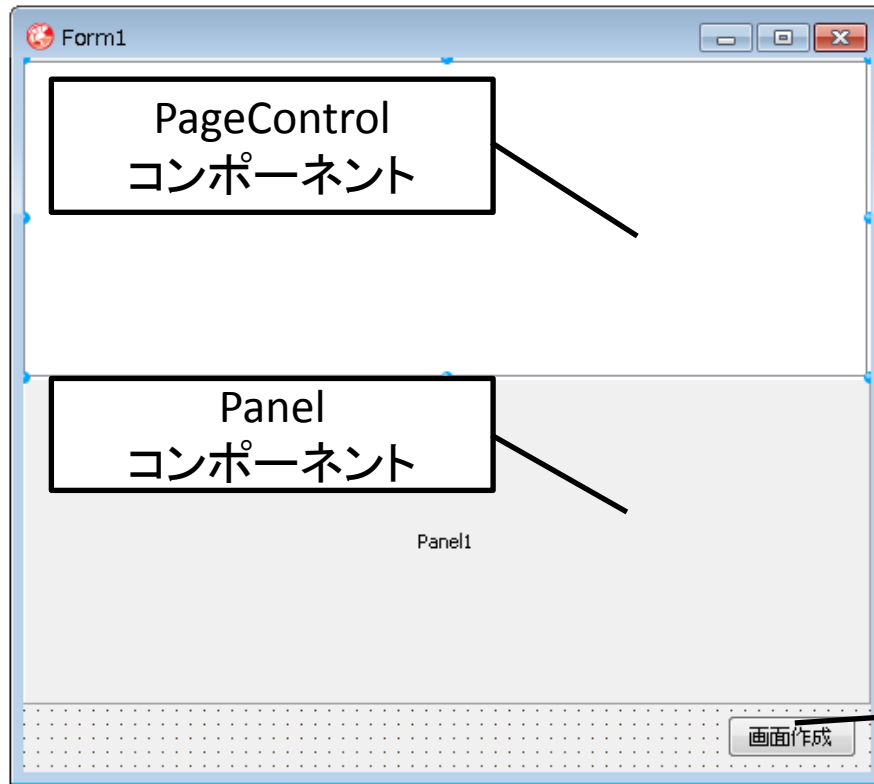
CustomHint	
DefaultMonitor	dmActiveForm
DockSite	<input type="checkbox"/> False
DoubleBuffered	<input type="checkbox"/> False
>> DragKind	dkDock
DragMode	dmAutomatic
Enabled	<input checked="" type="checkbox"/> True
Font	(TFont)
FormStyle	
GlassFrame	
Height	
HelpContext	
HelpFile	
HelpKeyword	
HelpType	htContext
Hint	
HorzScrollBar	(TControlScrollBar)

プロパティ
DragKind:dkDock
DragMode:dmAutomatic

■ Q1. 画面をドッキングするには？

- ドッキングの基本 作成例①(プロパティ)

ドッキング先(親)



Buttonコンポーネント:
イベント処理は後述

■ Q1. 画面をドッキングするには？

- ドッキングの基本 作成例② (プロパティ)

ドッキング元(子)

A screenshot of a Windows form titled 'Form2' containing a calendar component. The calendar shows a grid of days from 1 to 31, with the date 19 highlighted. The form has standard Windows window controls (minimize, maximize, close) in the top right corner.

Calendarコンポーネント
(画面構成例)

A screenshot of the 'Object Inspector' window in a development environment. The 'Form1 TForm1' is selected, and the 'プロパティ' (Properties) tab is active. The 'DragKind' property is set to 'dkDock' and 'DragMode' is set to 'dmAutomatic'. A red box highlights these two properties. A yellow callout box points to these properties with the text: 'プロパティ(Form) DragKind:dkDock DragMode:dmAutomatic'.

■ Q1. 画面をドッキングするには？

- ドッキングの基本 作成例③(イベント)

ドッキング先(親)

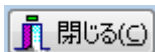
画面作成ボタンの処理(ソース)

ドッキング元(子)の画面ユニットをusesに追加

```
procedure TForm1.Button1Click(Sender: TObject);  
begin  
  Form2 := TForm2.Create(Application); //画面(子)を作成  
  Form2.Show; //画面(子)を表示  
end;
```

ドッキング元(子)

ポイント



閉じるボタンをつける場合、
BitBtnのbkcloseで閉じるボタンを作成すると
ドッキング時にドッキング先も閉じてしまうので注意！

■ Q1. 画面をドッキングするには？

- サンプルの実行

Form1

日	月	火	水	木	金	土
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31		

画面作成

②ドッキング/解除

Form2

日	月	火	水	木	金	土
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31		

①画面作成

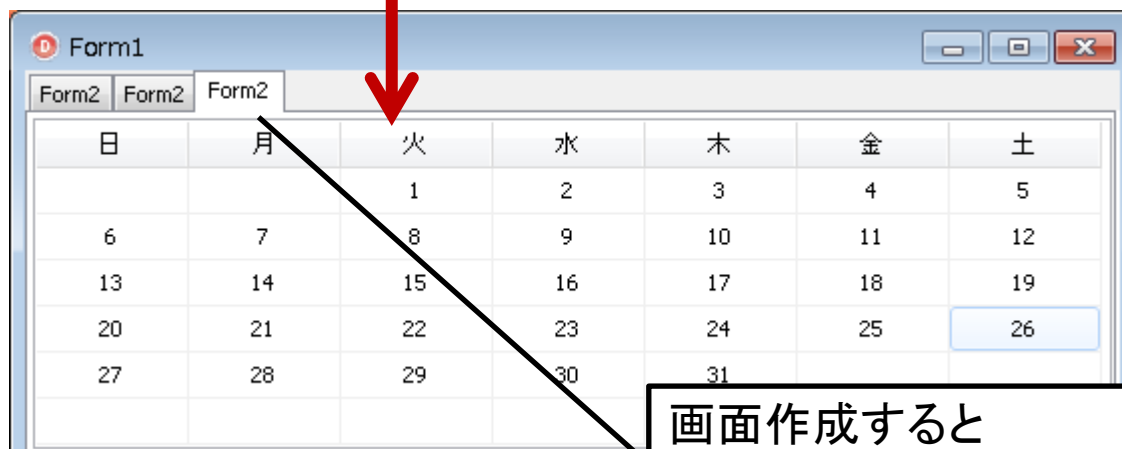
■ Q1. 画面をドッキングするには？

- 応用テクニック① プログラムでドッキングさせる

ドッキング先(親)

画面作成ボタンの処理(ソース)

```
procedure TForm1.Button1Click(Sender: TObject);  
begin  
  Form2 := TForm2.Create(Application); //画面 (子) を作成  
  Form2.ManualDock(PageControl1); //画面 (子) をPagecontrolにドッキングさせる  
  Form2.Show; //画面 (子) を表示  
end;
```



画面作成すると
直接ドッキングできる。

■ Q1. 画面をドッキングするには？

- 応用テクニック② プログラムでドッキング解除する

ドッキング元(子)

ダブルクリックの処理(ソース)

ドッキング先(親)の画面ユニットをusesに追加

```
procedure TForm2.Calendar1DbClick(Sender: TObject);
begin
  if Floating then //フロート(ドッキング解除状態)であればドッキング
  begin
    ManualDock(Form1.PageControl1);
  end
  else //ドッキング状態であればドッキング解除
  begin
    ManualFloat(Rect(0, 0, Width, Height));
  end;
end;
```

カレンダーをダブルクリックするとドッキングを切替える。

ポイント

ドッキングの状態はFloatingというプロパティで判断できる。

Form1 window showing a calendar grid. The calendar has columns for days of the week (日, 月, 火, 水, 木, 金, 土) and rows for dates. The date 26 is highlighted. Three instances of Form2 are docked on top of the calendar.

Form2 window showing a calendar grid. The calendar has columns for days of the week (日, 月, 火, 水, 木, 金, 土) and rows for dates. The date 26 is highlighted.

ダブルクリックで切替

■ Q1. 画面をドッキングするには？

- 応用テクニック③ ドッキング先を制御する

ドッキング先(親)

ドッキング制限の処理(ソース)

GetSiteInfoイベント

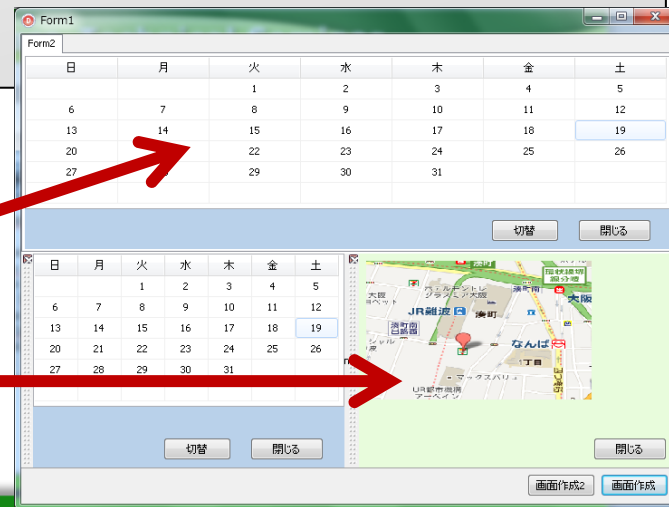
```
procedure TForm1.PageControl1GetSiteInfo(Sender: TObject; DockClient: TControl;  
  var InfluenceRect: TRect; MousePos: TPoint; var CanDock: Boolean);  
begin  
  if DockClient is TForm3 then //ドッキングする画面がXXXXであれば  
  begin  
    CanDock := False; //ドッキングを無効にする  
  end;  
end;  
end;
```



ドッキング
できない



ドッキングできる

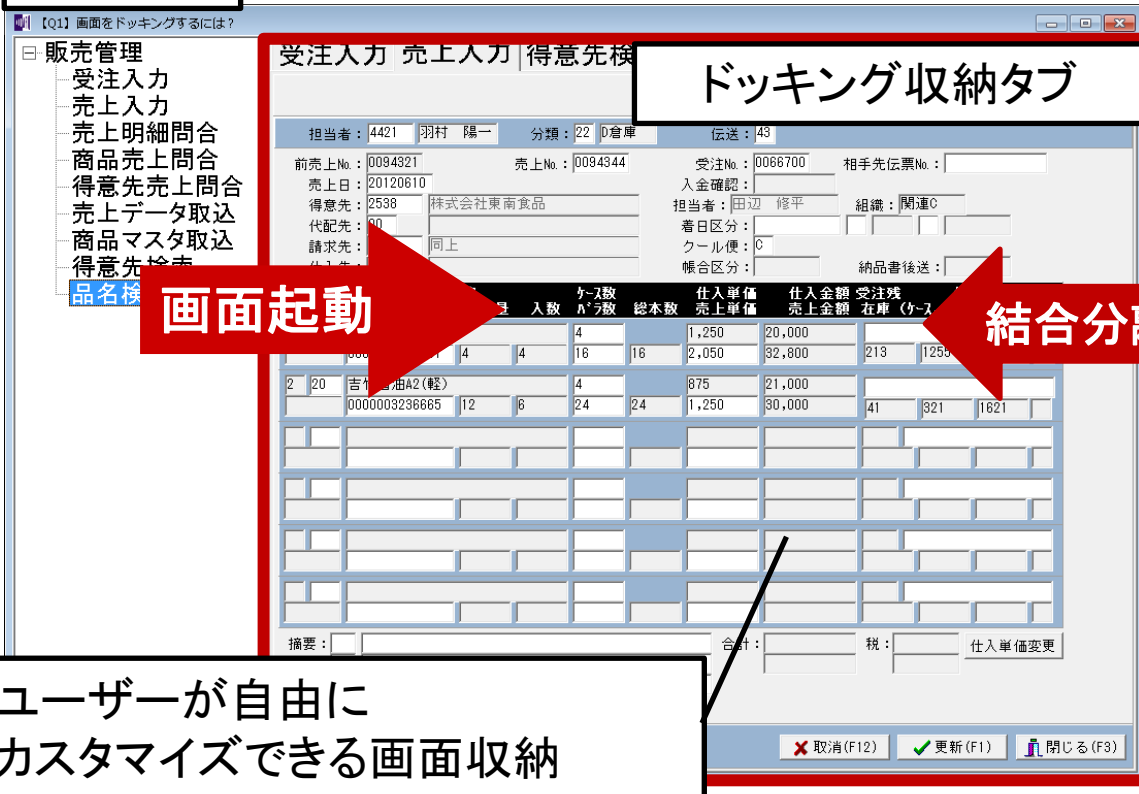


■ Q1. 画面をドッキングするには？

● ドッキングを活用した開発例



メニュー

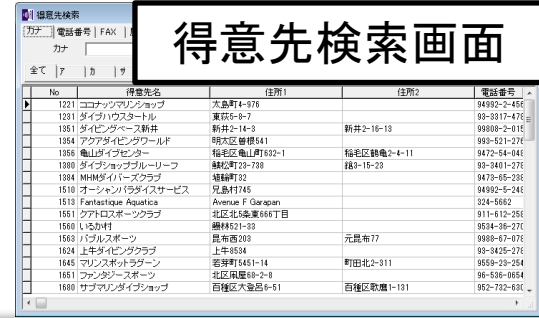


画面起動

結合分離可能

ユーザーが自由に
カスタマイズできる画面収納

※ドッキングさせる画面は、別々に開発できるのでプログラムも煩雑になりません。



■ Q2. データを暗号化するには？



【ご質問】

画面で入力したデータを暗号化して保存したいのですが、どのようにデータを加工すればいいのでしょうか？



【回答】

暗号化を行う場合、元になるデータに対して鍵(キー)を使って、プログラムで加工したり復元します。プログラムロジックで加工することもできますし、既成の暗号APIやソフトを利用して加工することもできます。

■ Q2. データを暗号化するには？

- 暗号化とは



■ 暗号化

インターネットなどのネットワークを通じて文書や画像などのデジタルデータをやり取りする際に、通信途中で第三者に盗み見られたり改ざんされたりされないよう、決まった規則に従ってデータを変換すること。

IT用語辞典抜粋

つまり..

データの内容を他人には分からなくするための方法です。

■ Q2. データを暗号化するには？

● 暗号化の用途

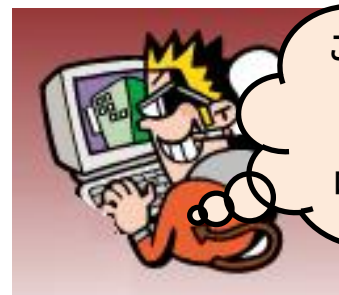
例えば、ユーザー管理のファイルにパスワードを保存する場合、そのまま保存するよりも暗号化して保存したほうが機密性を高められます。

もしユーザー管理ファイルを参照されてしまうと・・・

UID	ユーザー名	パスワード
▶ 2112	村上浩一	12345
2113	上野雄大	PASSWORD
2114	佐藤信司	201206
2115	名倉美穂	NAGURA
2116	佐竹蓮	PASSWORD

UID	ユーザー名	パスワード
▶ 2112	村上浩一	ZWJ_P
2113	上野雄大	;\$*8269!
2114	佐藤信司	YUHYUO
2115	名倉美穂	%\$>>78
2116	佐竹蓮	;\$*8269!

暗号化なし



暗号化



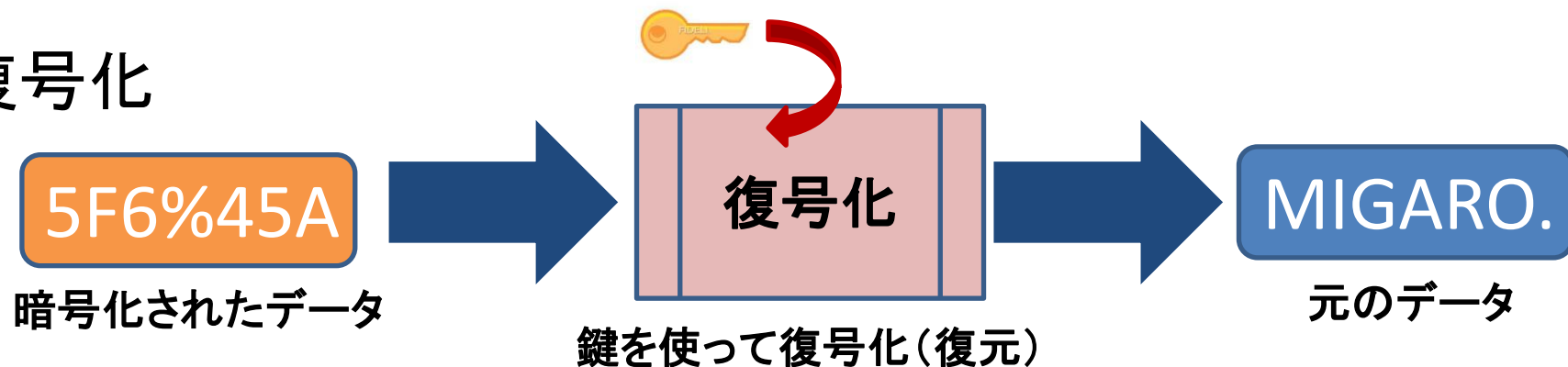
■ Q2. データを暗号化するには？

- 暗号化の手法

暗号化



復号化



※暗号の方式には暗号化と復号化に同じ鍵を使う共通鍵方式だけでなく、復号では別の鍵を使う公開鍵方式もあります。

■ Q2. データを暗号化するには？

- 簡単な暗号化例

暗号化/復号化	
データ	ミガロ.
鍵	key

暗号化/復号化	
データ	ゴドゥEhs
鍵	key

※ここでのサンプルはボタンを押して暗号化/復号化していますが、実際にはデータ登録時に暗号化、データ抽出時に復号化するような使い方ができます。

■ Q2. データを暗号化するには？

- 簡単な暗号化 作成例①(関数)

暗号化/復号化関数(ソース)

```
private
  function Encryption(data, key: String):String;

/* パラメータでデータを鍵を受け取り暗号/復号化を行う */
function TForm1.Encryption(data, key: String): String;
var
  i : Integer; //データループ用
  j : Integer; //鍵ループ用
begin
  j := 1; //鍵の初期値設定
  for i:= 1 to Length(data) do //入力データの桁数分、暗号化/復号を行う
  begin
    data[i] := Char(Ord(data[i]) xor Ord(key[j])); //データ1文字を対応する鍵1文字で加工
    if j = Length(key) then j:= 1 else Inc(j); //鍵の桁数が足りない場合は、最初から再利用
  end;
  Result := data; //暗号化/復号化されたデータを返却
end;
```

対象データと鍵を受け取って
暗号/復号データを返却

■ Q2. データを暗号化するには？

- 簡単な暗号化 作成例②(イベント)

暗号化/復号化ボタンの処理(ソース)

```
procedure TForm1.Button1Click(Sender: TObject);  
begin  
    //入力データと入力鍵を暗号関数で変換する。  
    memData.Text := Encryption(memData.Text, edtKey.Text);  
end;
```

暗号化関数に
データと鍵を渡すだけ

暗号化/復号化

データ ミガロ.

鍵 key

■ Q2. データを暗号化するには？

- 関数化すれば一括で変換も容易
memDataの代わりにデータセットをループで処理する

暗号化/復号化

データ

鍵 key

暗号化/復号化

UID	ユーザー名	パスワード
2112	村上浩一	12345
2113	上野雄大	PASSWORD
2114	佐藤信司	SATO06
2115	名倉美穂	NAGURA
2116	佐竹蓮	19820921
2117	石橋亮介	201206

暗号化/復号化

データ

鍵 key

暗号化/復号化

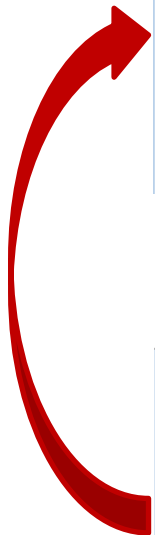
UID	ユーザー名	パスワード
2112	村上浩一	ZWJ_P
2113	上野雄大	;\$*8269!
2114	佐藤信司	8\$-\$UO
2115	名倉美穂	%\$>>78
2116	佐竹蓮	Z¥AYU@YT
2117	石橋亮介	YUHYUO

プログラムロジックで簡易な暗号化を実現！


■ Q2. データを暗号化するには？

- 高度な暗号化例

CAPICOM暗号化		CAPICOM復号化	
データ	ミガロ.		
鍵	key		



CAPICOM暗号化		CAPICOM復号化	
データ	305A06092B0601040182375803A04D304B060A2B060104018 237580301A03D303B02030200010202660202020080040857 D2B23369CCE6350410EABEA61574538E13887278FC108B0D3 4041000EE1FDB0C4B778CB0603C1E9B30C858		
鍵	key		



■ Q2. データを暗号化するには？

- 高度な暗号化を行う -Crypto APIを使った暗号化-

■ Crypto API

CryptoAPIとは、Microsoftが提供しているアプリケーションが暗号化およびデジタル証明書サービスを利用できる標準フレームワークのことである。



IT用語辞典抜粋

CryptoAPIは・・・

「CAPICOM」というCOMの形で用意されているので、

Delphi/400に取り込んで簡単に利用することが出来ます。

■ Q2. データを暗号化するには？

- Delphi/400にCAPICOMを取り込む手順
COMコントロール取り込み手順



■ Q2. データを暗号化するには？

- Delphi/400にCAPICOMを取り込む手順

③リストから[CAPICOM]を選択

説明	バージョン	ファイル名
C:\Windows\System32\wwan...	1.0	C:\W...
CAPICOM v2.1 Type Library	2.1	C:\P...
CertCli 1.0 Type Library	1.0	C:\W...
CertEnc 1.0 Type Library	1.0	C:\W...
CertEnroll 1.0 Type Library	1.0	C:\W...
ChartFX 2.0 OLE Custom Control	2.0	C:\W...
cic 1.0 Type Library	1.0	C:\W...
CloneHelper 1.0 Type Library	1.0	C:\W...
Cmi20	2.0	C:\W...

④取り込み先の指定
パレットページ名 : 任意のパレット名
ユニットディレクトリ名 : 取込み先のフォルダパス

クラス名(C): Settings=TSettings,EKU=TEKU,EKUs=TEKUs ...
パレットページ名(P): COM
ユニットディレクトリ(U): C:\デモプログラム\第10回\Session3\デモサン ...
検索パス(S): \$(BDSLIB)\$(Platform)release;\$(BDSUSERDIR)¥
 コンポーネントラッパーを生成する(S)

CAPICOMがPCにインストールされていない場合はMicrosoftからダウンロードして導入できます。

※再配布モジュールも同様です。

■ Q2. データを暗号化するには？

- Delphi/400にCAPICOMを取り込む手順

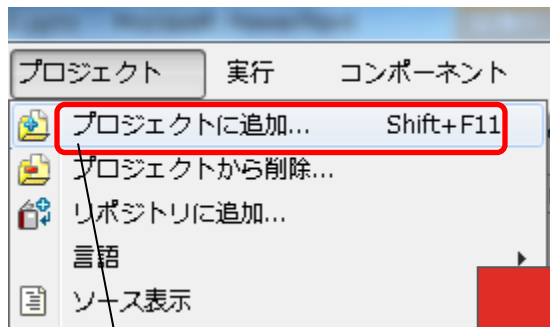
The image shows a Delphi IDE window with the 'コンポーネントのインポート' (Import Components) dialog box open. The dialog has a 'ユニットの作成' (Unit Creation) section with four radio buttons. The first option, 'ユニットの作成(U)', is selected and highlighted with a red box. A red arrow points from this box to the 'CAPICOM_TLB' unit source code in the main editor. The code shows the unit name and a comment in Japanese explaining that the unit is generated from a type library. A second red box highlights the text '⑥指定されたフォルダにユニットソースが自動生成' (Unit source is automatically generated in the specified folder), which is positioned over the code.

⑤ユニットの作成を選択

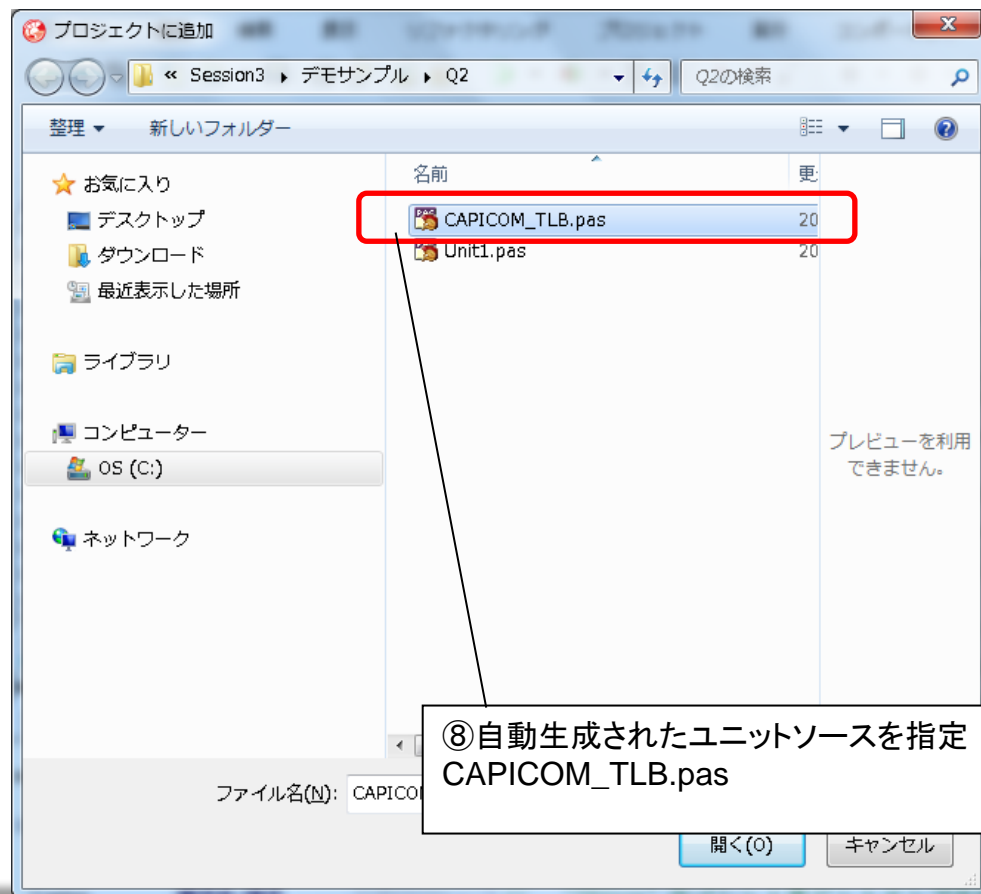
⑥指定されたフォルダにユニットソースが自動生成

■ Q2. データを暗号化するには？

- Delphi/400にCAPICOMを取り込む手順
自動生成されたユニットを組み込む



⑦使用したいプロジェクトで
生成ファイルを追加



⑧自動生成されたユニットソースを指定
CAPICOM_TLB.pas

■ Q2. データを暗号化するには？

- Delphi/400にCAPICOMを取り込む手順

⑨ユニットを使うで
CAPICOM_TBL
を選択

```
private
  { Private 宣言 }
  function Encryption(data,Key: String):String;
public
  { Public 宣言 }
end;

var
  Form1: TForm1;

implementation
uses CAPICOM_TLB;
{$R *.dfm}
```

⑩プログラムのuses節に
CAPICOM_TLB
が追加される

■ Q2. データを暗号化するには？

- 自動生成されたCAPICOMユニットの使い方

TEncryptedDataクラス

contentプロパティ

暗号化 / 複合する内容

SetSecretメソッド

鍵を設定するメソッド

Encryptメソッド

暗号化を実行するメソッド

Decryptメソッド

復号化を実行するメソッド

TUtilitiesクラス

BinaryToHexメソッド

バイナリをHEXに変換するメソッド

HexToBinaryメソッド

HEXをバイナリに変換するメソッド

■ Q2. データを暗号化するには？

- CAPICOMを利用した高度な暗号化 作成例①(関数)

暗号化関数(ソース)

```
private
  function CAPICOMEncryption(data, key: String):String;

/* パラメータでデータを鍵を受け取り暗号化を行う */
function TForm1.CAPICOMEncryption(data, key: String): String;
var
  Encpt : TEncryptedData; //暗号復号クラス
  Util : TUtilities;      //ユーティリティクラス
begin
  Encpt := TEncryptedData.Create(Self); //クラス生成
  Util := TUtilities.Create(Self);     //クラス生成
  try
    Encpt.SetSecret(key , CAPICOM_SECRET_PASSWORD); //鍵のセット
    Encpt.Content := data;                          //データのセット
    //暗号して結果返却 (バイナリデータなのでHEX変換)
    Result := Util.BinaryToHex(Encpt.Encrypt(CAPICOM_ENCODE_BINARY));
  finally
    Encpt.Free; //クラス破棄
    Util.Free;  //クラス破棄
  end;
end;
```

CAPICOMのクラスを作成

HEXデータとして暗号化

■ Q2. データを暗号化するには？

- CAPICOMを利用した高度な暗号化 作成例②(関数)

復号化関数(ソース)

```
private
function CAPICOMDecryption(data, key: String):String;

/* パラメータで暗号データを鍵を受け取り復号化を行う */
function TForm1.CAPICOMDecryption(data, key: String): String;
var
  Encpt : TEncryptedData; //暗号復号クラス
  Util  : TUtilities;     //ユーティリティクラス
  Buff  : WideString;    //復号処理用
begin
  Encpt := TEncryptedData.Create(Self); //クラス生成
  Util  := TUtilities.Create(Self);    //クラス生成
  try
    Encpt.SetSecret(key , CAPICOM_SECRET_PASSWORD); //鍵のセット
    Buff := Util.HexToBinary(data);                //暗号データのセット
    Encpt.Decrypt(Buff);                           //復号化
    Result := Encpt.Content;                       //結果返却
  finally
    Encpt.Free; //クラス破棄
    Util.Free;  //クラス破棄
  end;
end;
```

CAPICOMのクラスを作成

HEXデータをバイナリ化して
から復号化

■ Q2. データを暗号化するには？

- CAPICOMを利用した高度な暗号化 作成例③(イベント)

CAPICOM暗号化ボタンの処理(ソース)

```
procedure TForm1.btnCAPEncClick(Sender: TObject);  
begin  
    //入力データと入力鍵をCAPICOM暗号関数で変換する。  
    memData.Text := CAPICOMEncryption(memData.Text, edtKey.Text);  
end;
```

CAPICOM暗号化

CAPICOM復号化

データ ミガロ.

鍵 key

■ Q2. データを暗号化するには？

- CAPICOMを利用した高度な暗号化 作成例④(イベント)

CAPICOM復号化ボタンの処理(ソース)

```
procedure TForm1.btnCAPDecClick(Sender: TObject);  
begin  
    //入力データと入力鍵をCAPICOM復号関数で変換する。  
    memData.Text := CAPICOMDecryption(memData.Text, edtKey.Text);  
end;
```

CAPICOM暗号化 CAPICOM復号化

データ 305A06092B0601040182375803A04D304B060A2B060104018
237580301A03D303B02030200010202660202020080040857
D2B23369CCE6350410EABEA61574538E13887278FC108B0D3
4041000EE1FDB0C4B778CB0603C1E9B30C858

鍵 key

■ Q2. データを暗号化するには？

- 関数化すれば一括で変換も容易
データセットをループで処理する

UID	ユーザー名	パスワード
2112	村上浩一	12345
2113	上野雄大	PASSWORD
2114	佐藤信司	SATO06
2115	名倉美穂	NAGURA
2116	佐竹蓮	19820921
2117	石橋亮介	201206
2118	加瀬太	ADMIN

項目のサイズには注意が必要

UID	ユーザー名	パスワード
2112	村上浩一	305A06092B0601040182375803A04D3
2113	上野雄大	306206092B0601040182375803A0553
2114	佐藤信司	305A06092B0601040182375803A04D3
2115	名倉美穂	305A06092B0601040182375803A04D3
2116	佐竹蓮	306206092B0601040182375803A0553
2117	石橋亮介	305A06092B0601040182375803A04D3
2118	加瀬太	305A06092B0601040182375803A04D3

CAPICOMを利用するだけで高度な暗号化を簡単に実現！



パスワード

顧客情報

個人情報

Delphi/400で暗号化対応！



開発でお困りの点は
お気軽にミガロ.サポートへ
ご相談ください！



皆様のご質問をお待ちしております！