

Webservice - Consulta de CNPJ via PEX

Bastante semelhante ao conteúdo **Webservice - Consulta CEP via PEX**, neste exemplo ao digitar o CNPJ e clicar no botão "**Consulta**", o processo trará todas as informações referentes à aquele CNPJ.

image-1640260776228.png

Formulário com evento ao sair no campo do tipo **Botão**.

image-1640260809718.png

PEX com webservice no botão de consulta.

image-1640260830927.png

```
const
    cs_cnpj          = ' 1' ;
    cs_nome          = ' 2' ;
    cs_uf            = ' 4' ;
    cs_bairro        = ' 5' ;
    cs_logradouro    = ' 6' ;
    cs_numero        = ' 7' ;
    cs_cep           = ' 8' ;
    cs_cidade        = ' 9' ;
    cs_complemento   = '10';
    cs_telefone      = '11';
var
    lsCorpoRequisicao,
    lscaminhoURL,
    Result,
    lsCNPJ,
    lsSQL           : String;
    loRESTClient    : TRESTClient;
    loRESTRequest    : TRESTRequest;
    loRESTResponse   : TRESTResponse;
    loAuthBasica     : THTTPBasicAuthenticator;
```

```

lojson          : tjsonobject;
loCDS           : Tlibcds;
begin
  //limpa campos
  aoFormularios.GetJSON(cs_nome).SetStr(' TEXT0' , '');
  aoFormularios.GetJSON(cs_uf).SetStr(' TEXT0' , '');
  aoFormularios.GetJSON(cs_bairro).SetStr(' TEXT0' , '');
  aoFormularios.GetJSON(cs_logradouro).SetStr(' TEXT0' , '');
  aoFormularios.GetJSON(cs_numero).SetStr(' TEXT0' , '');
  aoFormularios.GetJSON(cs_cep).SetStr(' TEXT0' , '');
  aoFormularios.GetJSON(cs_cidade).SetStr(' TEXT0' , '');
  aoFormularios.GetJSON(cs_complemento).SetStr(' TEXT0' , '');
  aoFormularios.GetJSON(cs_telefone).SetStr(' TEXT0' , '');

  //verifica se campo CNPJ está vazio
  if TSTR.EliminaAlfa(aoFormularios.GetJSON(CS_CNPJ).GetStr(' TEXT0' )) <> '' then
  begin
    //retira caracteres especiais do CNPJ
    lsCNPJ := aoFormularios.GetJSON(CS_CNPJ).GetStr(' TEXT0' );
    lsCNPJ := TSTR.StrTran(lsCNPJ, '-', '');
    lsCNPJ := TSTR.StrTran(lsCNPJ, '.', '');
    lsCNPJ := TSTR.StrTran(lsCNPJ, '/', '');
    //verifica se é um CNPJ válido
    if not TSTR.Testa_CNPJ(lsCNPJ) then
      begin
        aoMensagem.SetStr(' MENSAGEM' , 'Erro! CNPJ: ' +
aoFormularios.GetJSON(CS_CNPJ).GetStr(' TEXT0' ) + ' inválido. ');
        aoMensagem.SetInt(' FOCO' , 43);
        aoMensagem.SetStr(' TIPO' , ' ERRO' );
        aoMensagem.SetInt(' TIMEOUT' , 6000);
        aoFormularios.GetJSON(cs_cnpj).SetStr(' TEXT0' , '');
        Abort;
      end;
    end;
  end;

  //verifica se CNPJ está cadastrado no banco de dados

  loCDS := of_CriaCDSPorSQL(Format(' SELECT C.FANTASIA, F.NOME UNIDADE FROM CLIFOREMP C JOIN
CLIENTE CLI ON CLI.IDCLIFOREMP = C.IDCLIFOREMP JOIN FILIAL F ON F.IDFILIAL =
CLI.IDFILIALRELACAO ' +
                                ' WHERE C.TIPOCLIFOREMP = 0 AND C.CNPJ =

```

```

%0: s' , [ TSTR. Aspa( lsCNPJ) ]));

try
  if loCDS.of_TemDados then
    begin
      aoMensagem.SetStr(' MENSAGEM', ' CNPJ ' + aoFormularios.GetJSON( CS_CNPJ).GetStr(' TEXTO' )
+ ' já cadastrado para o cliente: ' + loCDS.GetStr([' FANTASIA' ]) + ' na Unidade: ' +
loCDS.GetStr([' UNIDADE' ]));

      aoMensagem.SetInt(' FOCO', 43);
      aoMensagem.SetStr(' TIPO', ' ERRO' );
      aoMensagem.SetInt(' TIMEOUT', 6000);
      aoFormularios.GetJSON( CS_CNPJ).SetStr(' TEXTO', '');
      Abort;
    end;
  finally
    loCDS.FREE;
  end;

//inicia integração com web service

Result      := '';
loRESTClient := nil;
loRESTRequest := nil;
loRESTResponse := nil;
lscaminhoURL  := ' http://www.receitaws.com.br/v1/cnpj/' + lsCNPJ;

if length(lsCNPJ) > 1 then
  begin

    //executa o requisição
    try
      loRESTClient := TRESTClient .Create(nil);
      loRESTRequest := TRESTRequest .Create(nil);
      loRESTResponse := TRESTResponse.Create(nil);

      loRESTClient.RaiseExceptionOn500 := False;
      loRESTClient.BaseURL             := lscaminhoURL;
      loRESTClient.ContentType         := '';

      loRESTRequest.Method             := rmGet;
      loRestRequest.Resource := '';
    
```

```

loRESTRequest.Client := loRESTClient;
loRESTRequest.Response := loRESTResponse;
try
    loRESTRequest.Execute;
except
end;

if TSTR.of_IsNotNullEma(loRESTResponse.JSONText) then
    Result := loRESTResponse.JSONText
else
    Result := loRESTResponse.Content;

//aoMensagem.SetStr(' MENSAGEM', lscaminhoURL);
lojson := tjson.op_StringToJSON(loRESTResponse.JSONText);

//preenche campos com as informações retornadas do web service

aoFormularios.GetJSON( cs_nome
).SetStr(' TEXT0', lojson.GetStr(' nome' ));
aoFormularios.GetJSON( cs_uf
).SetStr(' TEXT0', lojson.GetStr(' uf' ));
aoFormularios.GetJSON( cs_bairro
).SetStr(' TEXT0', lojson.GetStr(' bairro' ));
aoFormularios.GetJSON( cs_logradouro
).SetStr(' TEXT0', lojson.GetStr(' logradouro' ));
aoFormularios.GetJSON( cs_numero
).SetStr(' TEXT0', lojson.GetStr(' numero' ));
aoFormularios.GetJSON( cs_cep
).SetStr(' TEXT0', lojson.GetStr(' cep' ));
aoFormularios.GetJSON( cs_cidade
).SetStr(' TEXT0', lojson.GetStr(' municipio' ));
aoFormularios.GetJSON( cs_complemento
).SetStr(' TEXT0', lojson.GetStr(' complemento' ));
aoFormularios.GetJSON( cs_telefone
).SetStr(' TEXT0', lojson.GetStr(' telefone' ));

{lsSQL := 'select idcidade from cidade where upper(REMOVE_CHAR_ESP(cidade.descricao)) =
' + TSTR.Aspa(lojson.GetStr(' municipio' )) + ' and cidade.uf = ' +
TSTR.Aspa(lojson.GetStr(' uf' ));
//aoMensagem.SetStr(' MENSAGEM', lsSQL);
loCDS := of_CriaCDSporSQL(lsSQL);

```

```

try
    if loCDS.of_TemDados then
        begin
            aoFormularios.GetJSON( cs_uf
).SetStr(' TEXT0' , lojson.GetStr(' uf' ));
            aoFormularios.GetJSON( CS_CIDADE
).SetStr(' TEXT0' , loCDS.GetStr([' IDCIDADE' ]));
            end;
        finally
            loCDS.FREE;
        end;
    }

finally
    loAuthBasica .Free;
    loRESTClient .Free;
    loRESTRequest .Free;
    loRESTResponse.FREE;
end;
end
else
    begin
        aoMensagem.SetStr(' MENSAGEM' , 'Para poder efetuar a busca, você precisa informar um CNPJ
válido' );
        aoMensagem.SetInt(' FOCO' , 43);
        aoMensagem.SetStr(' TIPO' , ' INFO' );
        aoMensagem.SetInt(' TIMEOUT' , 6000);
        aoFormularios.GetJSON( cs_cnpj).SetStr(' TEXT0' , '' );
    end;
end;
end;

```

Revisão #2

Criado 23 December 2021 08:54:07 por Nicolly Andrielly

Atualizado 6 April 2022 09:07:47 por Nicolly Andrielly