

Using FoxPro to Send or Retrieve data, using WebConnect WWHTTP library (any HTTP client library will work - such as curl)

[Freshbooks API documentation](#)

[Freshbooks](#) provides simple and inexpensive on-line timekeeping, billing, and support for the small and one-person consultant. Payments can be received online through Paypal, Google and major gateways, and invoices can be sent via email or snail-mail. Automated and recurring invoicing is supported.

Retrieving your Client List:

```
SET PROCEDURE TO classes\wwHTTP ADDITIVE
```

```
* The following is pulled almost verbatim from the API docs page
* EXCEPT <per_page> - which defaults to 25! Unless you have fewer than 25 items to be
  returned,
* increase this or call recursively for additional pages! Max items per page = 100
TEXT TO strReq NOSHOW PRETEXT 15
  <?xml version="1.0" encoding="utf-8"?>
  <request method="client.list">
  <!-- The page number to show (Optional) -->
  <page>1</page>
  <!-- Number of results per page, default 25 (Optional) -->
  <per_page>100</per_page>
  <!-- One of 'active', 'archived', 'deleted' -->
  <folder>active</folder>
  </request>
ENDTEXT
```

```
oHTTP = CREATEOBJECT("wwHttp")
oHTTP.cUserAgent = [CondoConduit Invoicing]
oHTTP.nHTTPPostMode = 4    && Ideal for posting XML
oHTTP.AddPostKey(strReq)   && This is where we add the request above to post to
FreshBooks
lcFBUserName = [xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx]    && Use your own Freshbooks
Authentication Token - it changes when you change your FB password
lcFBPassword = lcFBUserName    && Some http libraries require a password be used - it
can be anything
  * Freshbooks API has a single entry point for all requests -
https://yourcompanyid.freshbooks.com/api/2.1/xml-in
lcResult = oHTTP.HTTPGet("https://ideatellc.freshbooks.com/api/2.1/xml-
in",lcFBUserName,lcFBPassword)
IF [status="fail"] $ lcResult    && Check for Error Condition in returned XML
  * Bad result
  STRTOFILE(strReq,"FBRequest.xml",0)
  STRTOFILE(lcResult,"FBResult.xml",0)
  lcReturnValue = [Error: ] + STREXTRACT(lcResult,[<error>],[</error>])
ELSE
  * CAUTION HERE: FoxPro's XMLtoCURSOR does not like the <response></response> Node,
so strip it out of the response
  lcResult = STRTRAN(lcResult,[<response xmlns="http://www.freshbooks.com/api/"
status="ok">],[ ])
  lcResult = STRTRAN(lcResult,[</response>],[ ])
  XMLTOCURSOR(lcResult,'Clients')
  * You now have the results in a cursor - do what you will from this point forward.
  * I'm creating a JSON response that can be called using AJAX from a web page
  loSerializer = CREATEOBJECT("wwJsonSerializer")
```

```

    lcReturnValue = loSerializer.Serialize("cursor:TQuery")
    loSerializer.Destroy()
ENDIF
oHTTP.Destroy()

```

Retrieving Invoices:

Each Client is assigned an integer ID, which must be used when performing client specific actions. For example to grab all invoices for a single Client, use the following (documented [here](#)) :

```

<?xml version="1.0" encoding="utf-8"?>
<request method="invoice.list">
  <!-- Filter by client (Optional) -->
  <client_id>3</client_id>
  <!-- Filter by recurring id (Optional) -->
  <recurring_id>10</recurring_id>
  <!-- Filter by status (Optional) -->
  <status>draft</status>
  <!-- Return invoices dated after this arg (Optional) -->
  <date_from>2007-01-01</date_from>
  <!-- Return invoices dated before this arg (Optional) -->
  <date_to>2007-04-01</date_to>
  <!-- Return invoices modified after this arg (Optional) -->
  <updated_from>2007-01-01 00:00:00</updated_from>
  <!-- Return invoices modified before this arg (Optional) -->
  <updated_to>2007-01-02 00:00:00</updated_to>
  <!-- Page number to return, default is 1 (Optional) -->
  <page>1</page>
  <!-- Number of results per page, default is 25 (Optional) -->
  <per_page>10</per_page>
  <!-- One of 'active', 'archived', 'deleted' (Optional)-->
  <folder>active</folder>
</request>

```

Returning all open Invoices, or all Invoices for a single Client:

```

*****
*** Function: FreshBilling
*** Assume:
*** Created: 01/11/2011
*** Revised:
*** Copyright: (c) 2011, Ideate, LLC
*****
FUNCTION FreshBilling()

```

```

#IF .F.
  LOCAL Request as wwRequest, Response as wwResponse
#ENDIF

```

```

lcPropCode = ""
lcPropCode = ALLTRIM(Request.QueryString("PropCode"))
llGetClients = .F.

```

```
THIS.TableOpen('Props','CondoUsers')
```

```
loUserInfo=THIS.CheckLogin()
```

```

IF INDEXSEEK(lcPropCode,.T., "Props", "PropCode") AND Props.FBClientCode > 0
  lcFBClientCode = ALLTRIM(STR(Props.FBClientCode))

```

```

strReq = [ <?xml version="1.0" encoding="utf-8"?>] + CRLF + ;
[<request method="invoice.list">] + CRLF + ;
[ <per_page>100</per_page>] + CRLF + ;
[ <client_id>] + lcFBClientCode + [</client_id>] + CRLF + ;
[</request>] + CRLF
ELSE
TEXT TO strReq NOSHOW PRETEXT 15
  <?xml version="1.0" encoding="utf-8"?>
  <request method="invoice.list">
    <!-- Number of results per page, default is 25 (Optional) -->
    <per_page>100</per_page>
    <!-- Filter by status (Optional) -->
    <status>unpaid</status>
  </request>
ENDTEXT
ENDIF

*** Connect to the server
oHTTP = CREATEOBJECT("wwHttp")
oHTTP.cUserAgent = [CondoConduit Invoicing]
oHTTP.nHTTPPostMode = 4
oHTTP.AddPostKey(strReq)
lcFBUserName = [xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx]
lcFBPassword = lcFBUserName
lcResult = oHTTP.HTTPGet("https://ideatellc.freshbooks.com/api/2.1/xml-in",lcFBUserName,lcFBPassword)
* _Ciphertext = lcResult
IF [status="fail"] $ lcResult
  * Bad result
  STRTOFILE(strReq,"FBRequest.xml",0)
  STRTOFILE(lcResult,"FBResult.xml",0)
  lcReturnValue = [Error: ] + STREXTRACT(lcResult,[<error>],[</error>])
ELSE
  lcResult = STRTRAN(lcResult,[<response xmlns="http://www.freshbooks.com/api/"
status="ok">],[,[])
  lcResult = STRTRAN(lcResult,[</response>],[,[])
  lcResult = STRTRAN(lcResult,[>0<],[>0.00<]) && force decimal values rather than
FALSE
  XLTOCURSOR(lcResult,'Invoices')
  SELECT Invoices.Number, Organization, First_Name, Last_Name, Status, ;
    Amount, Amount_Outstanding, DTOC(Invoices.Date) AS InvDate, Props.PropCode,
Invoices.URL as LinkURL ;
  FROM Invoices ;
  INNER JOIN Props ON Invoices.Client_ID = Props.FBClientCode ;
  ORDER BY InvDate DESC ;
  INTO CURSOR TQuery
  loSerializer = CREATEOBJECT("wwJsonSerializer")
  lcReturnValue = loSerializer.Serialize("cursor:TQuery")
  loSerializer.Destroy()
ENDIF
oHTTP.Destroy()
USE IN SELECT('TQuery')
USE IN SELECT('Invoices')
Response.Write(lcReturnValue)

ENDFUNC && FreshBilling

```

Other online services used:

[Outright](#) for simple accounting, links to Freshbooks AND Bank and Credit Card accounts for easy-as-pie maintenance.

[ProvideSupport](#) for economical monitoring of web site activity from any platform (Win, Mac, Linux and Web) - includes proactive chatting

[Paessler](#) Server Monitoring - free for up to 100 nodes

[Producteev](#) - simple task manager that syncs with Google Tasks (has iPhone client) and the fabulous [Astrid](#) task manager for Android

[UserVoice](#) - Track and vote on suggestions from your users