

# **IQX Web Referencing Setup Guide**

**V1.01**

**Gareth Johnstone and Ivan Baldwin**



## Table of Contents

What you need to run it.....	3
IQX Database Update.....	3
SQL for the database table.....	3
Woodpecker File Update. ....	6
Job for WoodPecker.....	6
System Administrator Setup .....	12
IQXNet Setup.....	12
Switchable Views Setup .....	13
Setting up Branding.....	13
Reference Questionnaire Setup.....	15
Job Runner Scheduling.....	16
Web Reference – Sending out the reference requests.....	17
Reference Request Job - Send Web Reference Emails.xml .....	17
The Summary – Send Web Reference Summary.xml. ....	22

## What you need to run it

- A version of IQX with Branding 2.3.5.7 onwards.
- IQXHub (latest) and associated IQXWeb functionality
- nodeJS

IQX will set up IQX Hub and IQXWeb along with associated database and wpk changes. This will be a chargeable item.

## IQX Database Update

**This information is provided for information purposes only. Changes to the database should only be carried out by experienced personnel. Before any changes ensure a backup of the database has been made.**

Table ReferenceRequest should already be in the database. However, if the table is incomplete it should be deleted and the following SQL run to recreate the table.

ReferenceRequest (pears)

PKKey	Name	ID	Obj. ID	Data Type	Size	Scale	Comp.	Null	Uniq.	Value	Comment
1	<input checked="" type="checkbox"/> ReferenceRequestID	1	11752	char	20		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
2	<input type="checkbox"/> PersonID	2	11753	char	20		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
3	<input type="checkbox"/> DepartmentID	3	11754	char	2		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
4	<input type="checkbox"/> RefereeAddressName	4	11755	char	60		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
5	<input type="checkbox"/> RefereeSalutation	5	11756	char	60		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
6	<input type="checkbox"/> RefereePosition	6	11757	char	40		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
7	<input type="checkbox"/> CompanyName	7	11758	char	60		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
8	<input type="checkbox"/> Keyname	8	11759	char	60		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
9	<input type="checkbox"/> Addr1	9	11760	char	40		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
10	<input type="checkbox"/> Addr2	10	11761	char	40		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
11	<input type="checkbox"/> Addr3	11	11762	char	40		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
12	<input type="checkbox"/> Town	12	11763	char	30		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
13	<input type="checkbox"/> County	13	11764	char	30		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
14	<input type="checkbox"/> Country	14	11765	char	30		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
15	<input type="checkbox"/> PostCode	15	11766	char	20		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
16	<input type="checkbox"/> HomePhone	16	11767	char	100		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
17	<input type="checkbox"/> WorkPhone	17	11768	char	100		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
18	<input type="checkbox"/> MobilePhone	18	11769	char	100		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
19	<input type="checkbox"/> Fax	19	11770	char	100		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
20	<input type="checkbox"/> Email	20	11771	char	250		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
21	<input type="checkbox"/> Position	21	11772	char	100		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
22	<input type="checkbox"/> PositionNotes	22	11773	long varchar			<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
23	<input type="checkbox"/> URLRequestCode	23	11774	char	250		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
24	<input type="checkbox"/> Processed	24	11775	smallint			<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	0	
25	<input type="checkbox"/> StaffID	25	11776	char	20		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
26	<input type="checkbox"/> WhenCreated	26	11777	timestamp			<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	current timesta...	
27	<input type="checkbox"/> WhenCompleted	27	11778	timestamp			<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
28	<input type="checkbox"/> WhenProcessed	28	11779	timestamp			<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
29	<input type="checkbox"/> WhoProcessed	29	11780	char	20		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
30	<input type="checkbox"/> WhenSent	30	11781	timestamp			<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
31	<input type="checkbox"/> WhenReturned	31	11782	timestamp			<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
32	<input type="checkbox"/> EmploymentID	32	11783	char	20		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
33	<input type="checkbox"/> RefType	33	11784	smallint			<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
34	<input type="checkbox"/> StartDate	34	11785	timestamp			<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
35	<input type="checkbox"/> EndDate	35	11786	timestamp			<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
36	<input type="checkbox"/> ReferenceConfirmationRequired	36	11787	smallint			<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	0	

### SQL for the database table

Copy and paste this SQL into SQL Tool in the IQX program.

```
CREATE TABLE "pears"."ReferenceRequest" (
    "ReferenceRequestID" CHAR(20) NOT NULL,
    "PersonID" CHAR(20) NOT NULL,
```

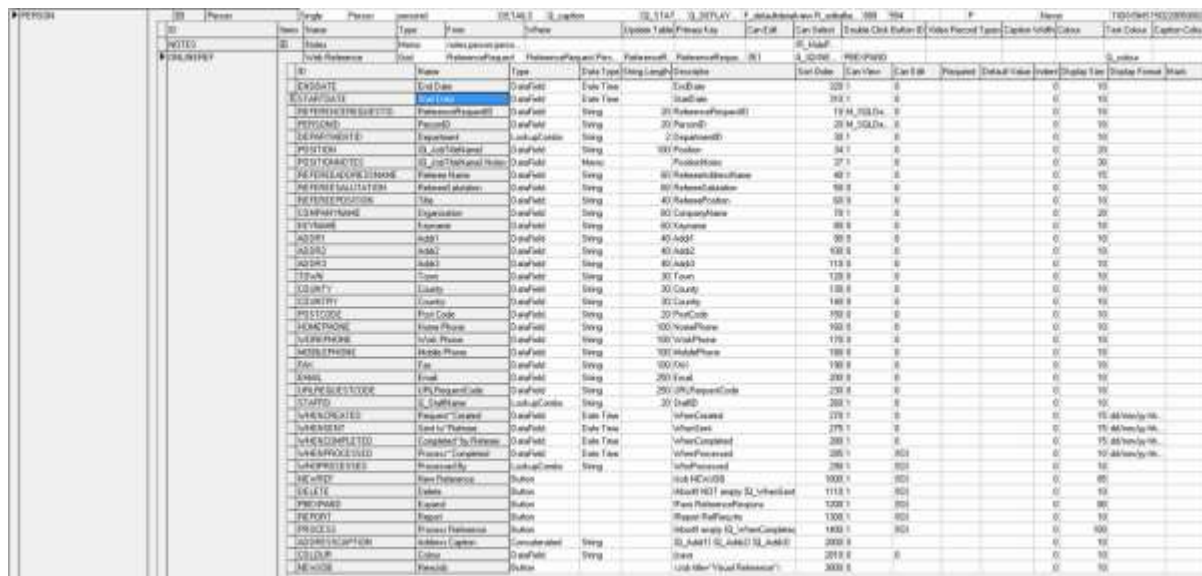
"DepartmentID" CHAR(2) NOT NULL,  
"RefereeAddressName" CHAR(60) NOT NULL,  
"RefereeSalutation" CHAR(60) NOT NULL,  
"RefereePosition" CHAR(40) NOT NULL,  
"CompanyName" CHAR(60) NULL,  
"Keyname" CHAR(60) NOT NULL,  
"Addr1" CHAR(40) NULL,  
"Addr2" CHAR(40) NULL,  
"Addr3" CHAR(40) NULL,  
"Town" CHAR(30) NULL,  
"County" CHAR(30) NULL,  
"Country" CHAR(30) NULL,  
"PostCode" CHAR(20) NULL,  
"HomePhone" CHAR(100) NULL,  
"WorkPhone" CHAR(100) NULL,  
"MobilePhone" CHAR(100) NULL,  
"Fax" CHAR(100) NULL,  
"Email" CHAR(250) NOT NULL,  
"Position" CHAR(100) NOT NULL,  
"PositionNotes" LONG VARCHAR NULL,  
"URLRequestCode" CHAR(250) NULL,  
"Processed" SMALLINT NULL DEFAULT 0,  
"StaffID" CHAR(20) NULL,  
"WhenCreated" TIMESTAMP NULL DEFAULT CURRENT\_TIMESTAMP,  
"WhenCompleted" TIMESTAMP NULL,  
"WhenProcessed" TIMESTAMP NULL,  
"WhoProcessed" CHAR(20) NULL,  
"WhenSent" TIMESTAMP NULL,  
"WhenReturned" TIMESTAMP NULL,  
"EmploymentID" CHAR(20) NULL,  
"RefType" SMALLINT NULL,

```
"StartDate" TIMESTAMP NULL,  
"EndDate" TIMESTAMP NULL,  
"ReferenceConfirmationRequired" SMALLINT DEFAULT 0,  
PRIMARY KEY ( "ReferenceRequestID" ASC )  
 ) IN "system";  
  
ALTER TABLE "pears"."ReferenceRequest" ADD CONSTRAINT "Person" NOT NULL FOREIGN KEY (  
"PersonID" ASC ) REFERENCES "pears"."Person" ( "personid" );  
  
ALTER TABLE "pears"."ReferenceRequest" ADD CONSTRAINT "Department" NOT NULL FOREIGN KEY  
( "DepartmentID" ASC ) REFERENCES "pears"."Department" ( "departmentid" );  
  
ALTER TABLE "pears"."ReferenceRequest" ADD CONSTRAINT "staff" NOT NULL FOREIGN KEY (  
"StaffID" ASC ) REFERENCES "pears"."staff" ( "staffid" );
```

## Woodpecker File Update.

This information is provided for information purposes only. Changes to the woodpecker file should only be carried out by experienced personnel.

A new form is required in Woodpecker. Run the following Xml in WPKMaintain.exe to set up this form.



ID	Name	Type	Sex	Location	Staff	Display	Address	City	State	Zip	Phone	Cell	Home	Fax	Web	Notes	Created	Modified
ENDDATE	End Date	DateTime	End Time	EndDate	300 1	0												
STARTDATE	Start Date	DateTime	Start Time	StartDate	300 1	0												
PERSONID	Person ID	String	Person ID	PersonID	19 4	0												
PERSONTYPE	Person Type	String	Person Type	PersonType	20 4	0												
DEPARTMENT	Department	String	Department	Department	25 1	0												
POSITION	Job Title	String	Job Title	Position	30 1	0												
POSITIONCHARACT	Job Title Character	String	Job Title Character	PositionCharact	30 1	0												
PERSONALID	Person ID	String	Person ID	PersonalID	30 1	0												
PERSONSALUTATION	Person Salutation	String	Person Salutation	PersonSalutation	40 1	0												
PERSONPOSITION	Person Position	String	Person Position	PersonPosition	50 1	0												
COMPANYNAME	Company Name	String	Company Name	CompanyName	80 1	0												
KEYNAME	Key Name	String	Key Name	KeyName	80 1	0												
CITY	City	String	City	City	40 1	0												
STATE	State	String	State	State	40 1	0												
ADDRESS	Address	String	Address	Address	119 1	0												
CITY	City	String	City	City	40 1	0												
COUNTY	County	String	County	County	30 1	0												
ZIP	Zip	String	Zip	Zip	10 1	0												
PHONE	Phone	String	Phone	Phone	30 1	0												
CITY	City	String	City	City	40 1	0												
COUNTY	County	String	County	County	30 1	0												
STATE	State	String	State	State	40 1	0												
ADDRESS	Address	String	Address	Address	119 1	0												
PHONE	Phone	String	Phone	Phone	30 1	0												
CITY	City	String	City	City	40 1	0												
COUNTY	County	String	County	County	30 1	0												
STATE	State	String	State	State	40 1	0												
ADDRESS	Address	String	Address	Address	119 1	0												
PHONE	Phone	String	Phone	Phone	30 1	0												

Job for WoodPecker

```
<Job title="Visual Reference">
```

```
<Parameters>
```

```
<Parameter name="cPersonID" type="String"/>
```

```
</Parameters>
```

```
<SetVariable name="xButton1caption" value="Save/Continue"></SetVariable>
```

```
<SetVariable name="xButton2caption" value="Save/Close"></SetVariable>
```

```
<SetVariable name="xButton1show" value="0"></SetVariable>
```

```
<SetVariable name="xButton2show" value="1"></SetVariable>
```

```
<SetVariable name="xShowSalutation" value="1"></SetVariable>
```

```
<SetVariable name="xShowKeyname" value="1"></SetVariable>
```

```
<SetVariable name="xRefereeAddressName" value=""></SetVariable>
```

```
<SetVariable name="xRefereeSalutation" value=""></SetVariable>
```

```
<SetVariable name="xKeyName" value=""></SetVariable>
```

```
<SetVariable name="xCompanyName" value=""></SetVariable>
```

```
<SetVariable name="xRefereePosition" value=""></SetVariable>
```

```
<SetVariable name="xHomePhone" value=""></SetVariable>
```

```
<SetVariable name="xWorkPhone" value=""></SetVariable>
<SetVariable name="xMobilePhone" value=""></SetVariable>
<SetVariable name="xFax" value=""></SetVariable>
<SetVariable name="xAddr1" value=""></SetVariable>
<SetVariable name="xAddr2" value=""></SetVariable>
<SetVariable name="xAddr3" value=""></SetVariable>
<SetVariable name="xTown" value=""></SetVariable>
<SetVariable name="xCounty" value=""></SetVariable>
<SetVariable name="xCountry" value=""></SetVariable>
<SetVariable name="xPostCode" value=""></SetVariable>
<SetVariable name="xEmail" value=""></SetVariable>
<SetVariable name="xStartDate" value=""></SetVariable>
<SetVariable name="xEndDate" value=""></SetVariable>
<SQLQuery>
  <SQLSelect xml:space="preserve">
    select top 1 b.departmentid as thedepartmentid from search b JOIN Department a On
    b.DepartmentID = a.DepartmentID where (b.permanent =1 or b.temp =1) and a.searchable = 1 and
    b.personid = '{cPersonID}'
  </SQLSelect>
  <SetVariable name="xdepartmentid" value="{thedeartmentid}"></SetVariable>
</SQLQuery>
<SQLQuery>
  <SQLSelect xml:space="preserve">
    select "WPKMaintainGetSwitchValue" ('JOBTITLENAME','Global','P') as theJobTitle
  </SQLSelect>
  <SetVariable name="xJobTitle" value="{theJobTitle}"></SetVariable>
</SQLQuery>
<Form backgroundcolour="15647929" height="780" width="800" top="0" gridx="330" gridy="25"
gridtop="0" gridleft="60" left="400" caption="Online Reference Request" labelwidth="115"
readonlycolour="13487565" labelfontsize="10" fontsize="10" fontcolour="16711680">
<ControlGroup name="ThePosition">
<Control type="shape" shape="spacerline"/>
```

```
<Control name="lblPos" type="label" caption="Position Details" labelwidth="115"
labelfontsize="18" left="57" height="30" />
```

```
<Control name="lblastrix" type="label" caption="* = Required" labelwidth="115"
labelfontsize="10" left="57" height="30" />
```

```
<Control name="cDepartmentID" type="combo" caption="Department *:" value="{xdepartmentid}"
choices="select a.name , b.departmentid from search b JOIN Department a On b.DepartmentID =
a.DepartmentID where (b.permanent =1 or b.temp =1) and a.searchable = 1 and b.personid =
'{cPersonID}'" required="YES" width="200" labelwidth="115" gridrow="3" />
```

```
<Control name="cPosition" type="string" required="YES" caption="{xJobTitle} *:" value=""
readonly="" width="200" labelwidth="115" />
```

```
<Control name="cPositionNotes" type="memo" required="YES" caption="{xJobTitle} Notes *:"
value="" readonly="" width="530" labelwidth="115" height="180"/>
```

```
<Control name="cStartDate" type="datetime" required="YES" caption="Start Date *:"
value="{xStartDate}" readonly="" width="200" gridcolumn="1" labelwidth="100" left="400"
top="75"/>
```

```
<Control name="cEndDate" type="datetime" required="YES" caption="End Date *:"
value="{xEndDate}" readonly="" width="200" gridcolumn="1" labelwidth="100" left="400"
top="100" />
```

```
<Control type="shape" shape="spacerline" />
```

```
</ControlGroup>
```

```
<ControlGroup name="Referee">
```

```
<Control name="lblRef" type="label" caption="Referee Details" labelwidth="100" labelfontsize="18"
left="57" height="30" />
```

```
<Control name="cRefereeAddressName" type="string" required="YES" caption="Referee Name
*:" value="{xRefereeAddressName}" readonly="" width="200" gridcolumn="0" labelwidth="115"
gridrow="16"/>
```

```
<Control name="cRefereeSalutation" type="string" required="YES" caption="Salutation *:"
value="{xRefereeSalutation}" readonly="" width="200" gridcolumn="0" labelwidth="115"
show="{xShowSalutation}" />
```

```
<Control name="cKeyName" type="string" required="YES" caption="Keyname *:"
value="{xKeyName}" readonly="" width="200" gridcolumn="0" labelwidth="115" show="1"/>
```

```
<Control name="cCompanyName" type="string" required="YES" caption="Organisation *:"
value="{xCompanyName}" readonly="" width="200" gridcolumn="0" labelwidth="115" />
```

```
<Control name="cRefereePosition" type="string" required="YES" caption="Referee Job Title *:"
value="{xRefereePosition}" readonly="" width="200" gridcolumn="0" labelwidth="115" />
```

```
<Control name="cHomePhone" type="string" caption="Home Phone:" value="{xHomePhone}"
readonly="" width="200" gridcolumn="0" labelwidth="115"/>
```



```
<Control name="cWorkPhone" type="string" required="YES" caption="Work Phone *:"  
value="{xWorkPhone}" readonly="" width="200" gridColumn="0" labelwidth="115"/>
```

```
<Control name="cMobilePhone" type="string" required="YES" caption="Mobile Phone *:"  
value="{xMobilePhone}" readonly="" width="200" gridColumn="0" labelwidth="115"/>
```

```
<Control name="cFax" type="string" caption="Fax:" value="{xFax}" readonly="" width="200"  
gridColumn="0" labelwidth="115"/>
```

```
<Control type="shape" shape="spacerline" top="670"/>
```

```
<Control name="cAddr1" type="string" required="YES" caption="Address *:" value="{xAddr1}"  
readonly="" gridColumn="1" left="400" width="200" labelwidth="100" gridrow="16"/>
```

```
<Control name="cAddr2" type="string" caption="" value="{xAddr2}" readonly=""  
gridColumn="1" left="500" width="200" labelwidth="100"/>
```

```
<Control name="cAddr3" type="string" caption="" value="{xAddr3}" readonly=""  
gridColumn="1" left="500" width="200" labelwidth="100"/>
```

```
<Control name="cTown" type="string" required="YES" caption="Town *:" value="{xTown}"  
readonly="" gridColumn="1" left="400" width="200" labelwidth="100" />
```

```
<Control name="cCounty" type="string" required="YES" caption="County *:" value="{xCounty}"  
readonly="" width="200" gridColumn="1" left="400" labelwidth="100"/>
```

```
<Control name="cCountry" type="string" required="YES" caption="Country *:"  
value="{xCountry}" readonly="" width="200" gridColumn="1" left="400" labelwidth="100"/>
```

```
<Control name="cPostCode" type="string" required="YES" caption="Postcode *:"  
value="{xPostCode}" readonly="" width="200" gridColumn="1" left="400" labelwidth="100" />
```

```
<Control name="cEmail" type="string" required="YES" caption="E-mail *:" value="{xEmail}"  
readonly="" width="200" gridColumn="1" labelwidth="100" left="400" />
```

```
</ControlGroup>
```

```
<Page name="page1" caption="Referee Wizard - Step 1/3">
```

```
<ControlGroup name="ThePosition" />
```

```
<ControlGroup name="Referee" />
```

```
<ControlGroup>
```

```
<Control type="button" caption="{xButton1caption}" left="210" top="700"  
show="{xButton1show}">
```

```
<SetVariable value="{UniqueKey}" name="RefID" />
```

```
<SQLExec>
```

```
Insert Into ReferenceRequest
```

```
(ReferenceRequestID, PersonID, DepartmentID, RefereeAddressName, RefereeSalutation,  
RefereePosition, CompanyName, Keyname, Addr1, Addr2, Addr3, Town, County, Country, PostCode,
```

HomePhone, WorkPhone, MobilePhone, Fax, Email, Position, PositionNotes, URLRequestCode, StaffID, WhenCreated, StartDate, EndDate)

Values

(:RefID, :cPersonID, :cDepartmentID, :cRefereeAddressName, :cRefereeSalutation, :cRefereePosition, :cCompanyName, :cKeyName, :cAddr1, :cAddr2, :cAddr3, :cTown, :cCounty, :cCountry, :cPostCode, :cHomePhone, :cWorkPhone, :cMobilePhone, :cFax, :cEmail, :cPosition, :cPositionNotes, HASH(':RefID', 'SHA256'), UserStaffID, CURRENT\_TIMESTAMP, :cStartDate, :cEndDate)

</SQLExec>

<SetControl control="cRefereeAddressName" value=""></SetControl>

<SetControl control="cRefereeSalutation" value=""></SetControl>

<SetControl control="cKeyName" value=""></SetControl>

<SetControl control="cCompanyName" value=""></SetControl>

<SetControl control="cRefereePosition" value=""></SetControl>

<SetControl control="cHomePhone" value=""></SetControl>

<SetControl control="cWorkPhone" value=""></SetControl>

<SetControl control="cMobilePhone" value=""></SetControl>

<SetControl control="cFax" value=""></SetControl>

<SetControl control="cAddr1" value=""></SetControl>

<SetControl control="cAddr2" value=""></SetControl>

<SetControl control="cAddr3" value=""></SetControl>

<SetControl control="cTown" value=""></SetControl>

<SetControl control="cCounty" value=""></SetControl>

<SetControl control="cCountry" value=""></SetControl>

<SetControl control="cPostCode" value=""></SetControl>

<SetControl control="cEmail" value=""></SetControl>

<If x1="{xButton1caption}" x2="Save/Email">

<Cancel/>

</If>

</Control>

<Control type="button" caption="{xButton2caption}" left="330" top="700" show="{xButton2show}">

<ValidatePage></ValidatePage>

<SetVariable value="{UniqueKey}" name="RefID" />

<SQLExec>

Insert Into ReferenceRequest

(ReferenceRequestID, PersonID, DepartmentID, RefereeAddressName, RefereeSalutation, RefereePosition, CompanyName, Keyname, Addr1, Addr2, Addr3, Town, County, Country, PostCode, HomePhone, WorkPhone, MobilePhone, Fax, Email, Position, PositionNotes, URLRequestCode, StaffID, WhenCreated, StartDate, EndDate)

Values

(:RefID, :cPersonID, :cDepartmentID, :cRefereeAddressName, :cRefereeSalutation, :cRefereePosition, :cCompanyName, :cKeyName, :cAddr1, :cAddr2, :cAddr3, :cTown, :cCounty, :cCountry, :cPostCode, :cHomePhone, :cWorkPhone, :cMobilePhone, :cFax, :cEmail, :cPosition, :cPositionNotes, HASH(:RefID, 'SHA256'), UserStaffID, CURRENT\_TIMESTAMP, :cStartDate, :cEndDate)

</SQLExec>

<Cancel/>

</Control>

<Control type="button" caption="Close" left="450" top="700">

<Cancel/>

</Control>

</ControlGroup>

</Page>

</Form>

</Job>

## System Administrator Setup

This section includes System Administrator setup activities.

### IQXNet Setup

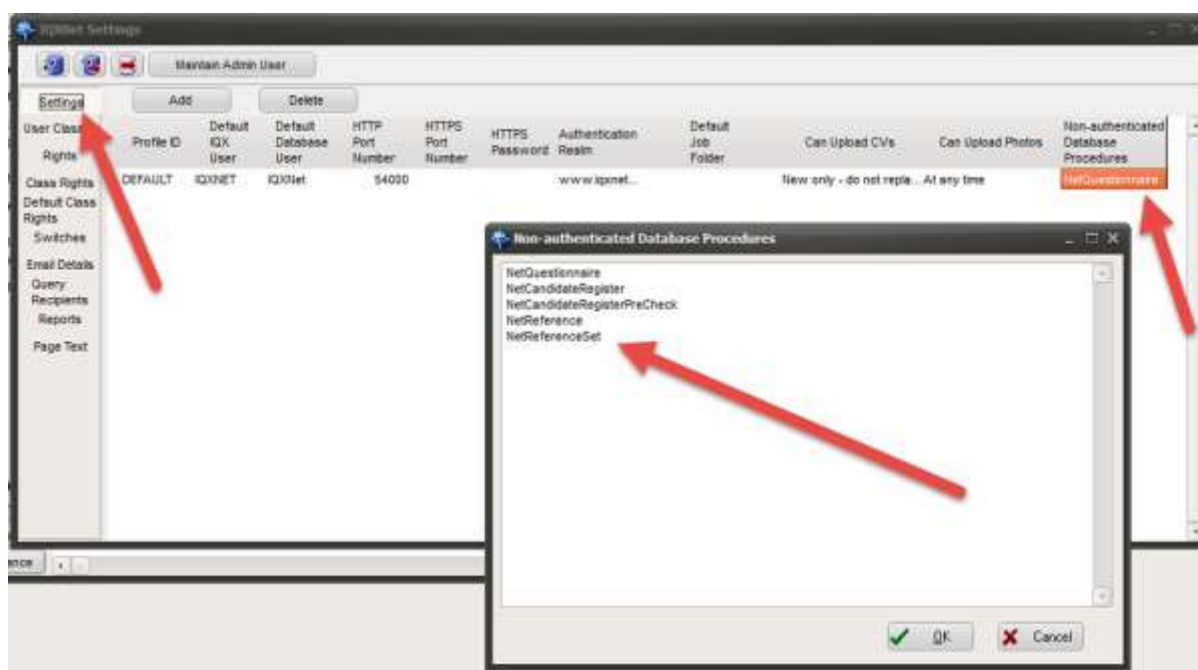
Once the IQXHub etc. has been set up clients will need to set up their IQXWeb Settings.

Under Maintenance > IQXNet Setup

Click on the settings tab then scroll along to Non-authenticated Database Procedures

Double click on the entry and add this add the bottom:

- NetReference
- NetReferenceSet



Whilst in IQXNet Settings set up access rights and other settings.

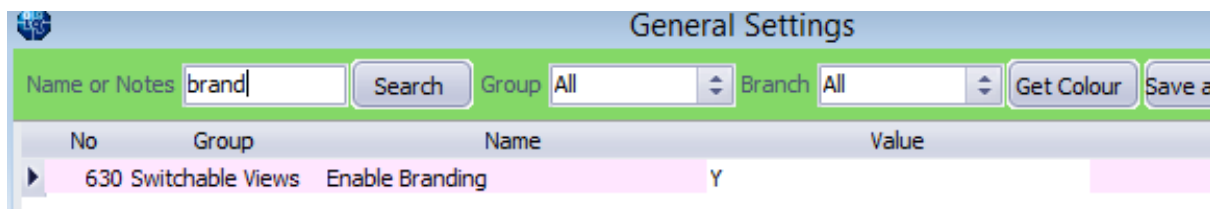
## Switchable Views Setup

Under Maintenance > General Settings

Type in "Name or Notes": On-Line, you should see this:

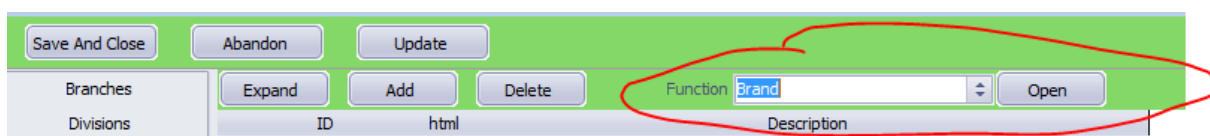
Setting up Branding

- a) Set up the brand using



- b) In Agency Setup, locate the branding view, under 'Vacancy Role'

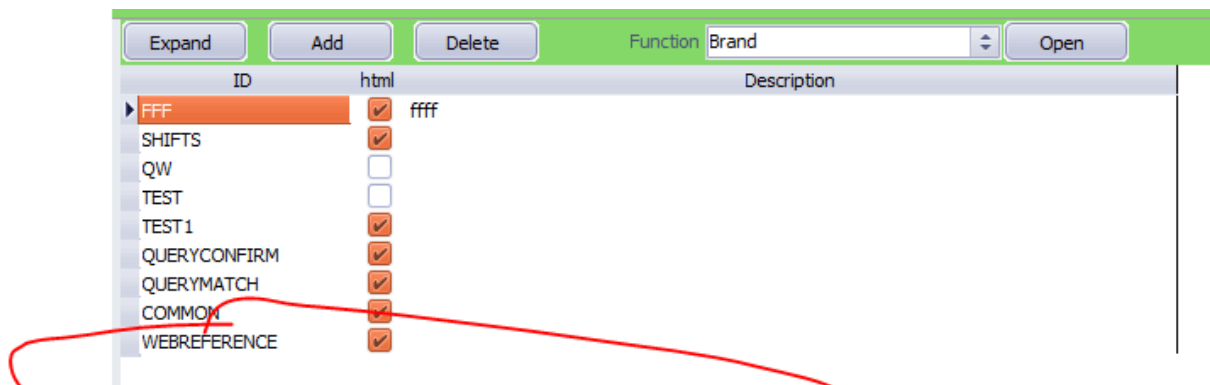
- c) Select the branding function, and press open



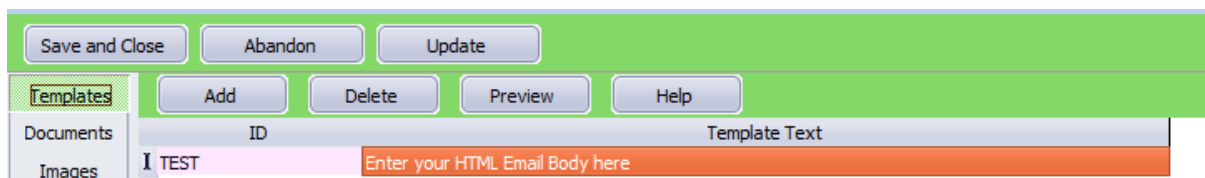
And set up the function as below

```
(
  in persid char(20),in vacid char(20),in ClientOrTemp char(1),in Use char(20) )
returns char(20)
begin
  declare RVal char(20);
  if Use = 'WEBREFERENCE' then
    set rval = 'WEBREFERENCE'
  end if;
  return RVal
end
```

- d) Back to branding, create a new brand called 'WEB REFERENCE'



- e) Double click webreference, and create a new template



f) You need to assign a template to 'WEB REFERENCE', this can be done by selecting



Once open, set up the function as below and save

```
(
  in "BehaviourID" char(1),in "BrndID" char(20),in "EventID" char(1),in "TargetType"
char(1),in "ShiftsID" char(1000),in "PlacementID" char(20) )
returns long varchar
begin
  -- TargetType C or T (client/temp)
  -- behaviourid P, S, E (Post, SMS or Email)
  -- EventID C, X or O (Confirmation, Cancellation or Offer)
  declare "RVAL" long varchar;
  declare "ID" char(20);
  select "visibleid" into "ID" from "EN_brand" where "brandid" = "BrndID";
  if("ID" = 'WEBREFERENCE') and("eventid" = 'D') then
    select "brandtext" into "RVAL" from "en_template" where "brandid" = "BrndID" and
"visibleid" = 'REQUEST';
    return "RVAL"
  end if;
  if("ID" = 'WEBREFERENCE') and("eventid" = 'S') then
    select "brandtext" into "RVAL" from "en_template" where "brandid" = "BrndID" and
"visibleid" = 'SUMMARY';
    return "RVAL"
  end if;
  return "RVAL"
end
```

- g) Xml jobs for sending out reference requests and a summary of the reference to referees will use the above SQL to pull in the standard web signature,
- h) Use the standard html wrap
- i) Convert the brand email body using boiler plate text

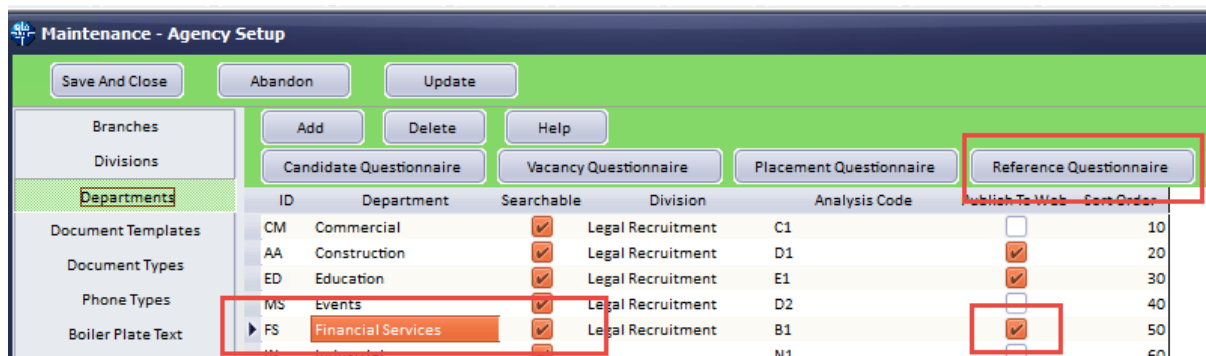


Make sure 'Y' is in the box for the Web Reference to show.

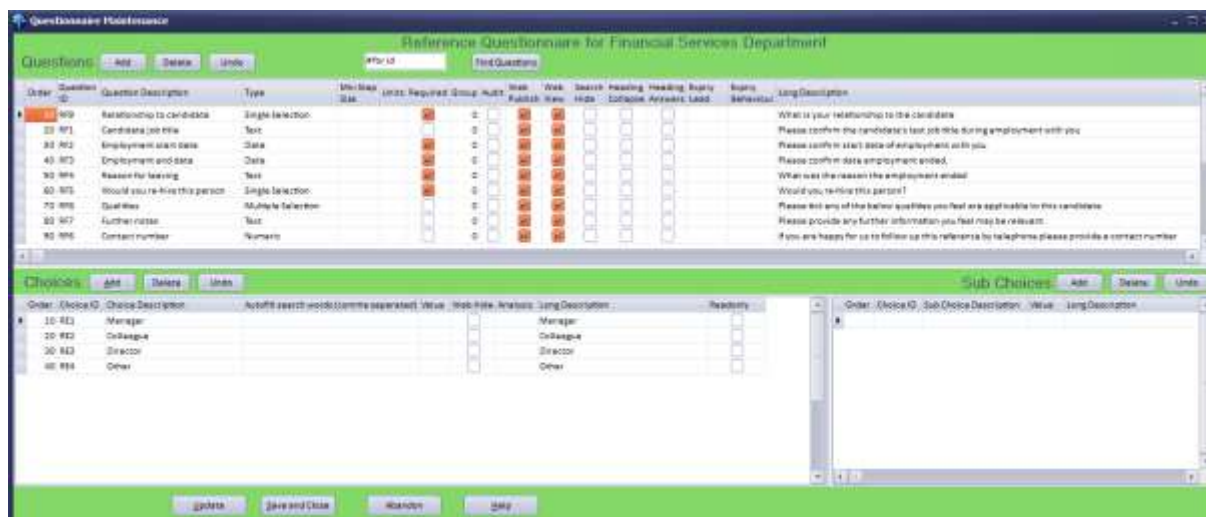
Refer to the IQX Web Referencing User Guide for information on using Web Referencing.

## Reference Questionnaire Setup

The questionnaire must be set up via Maintenance – Agency Setup – Department – Select a department – Click Publish to Web – Reference Questionnaire.



The next window will allow the creation of questions relevant to the Department. Web Publish and Web View must be clicked for this feature to be displayed on the Web. The long description is the question that will be displayed in the reference request.



Each department will need separate Reference Questionnaires set up.

## Job Runner Scheduling

To set up the automatic jobs for emailing the reference request and a summary of their completed reference to the referee got to Maintenance – Database setup.

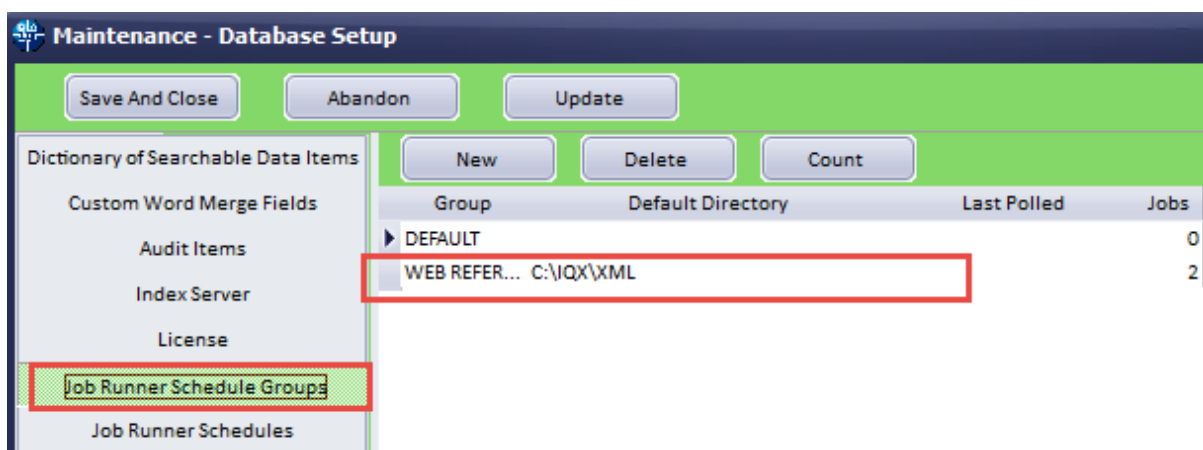
In Job Runner Schedule Groups create a group for web referencing along with the default directory address housing the two xml jobs.

Send Web Reference Emails.xml and

Send Web Reference Summary.xml.

SMTP needs to be set up for this function to work.

The xml code for these are included at the end of this document.



In Job Runner Schedules schedule the two xml jobs. Here you can detail how frequent these jobs will run. Note – Active means the jobs will be run as and when they are required.



For testing each job can be run manually via Maintenance – Run a Job and locating the respective xml jobs.



## Web Reference – Sending out the reference requests

All the web reference requests are stored in the database marked as ready to be emailed out.

The job below is then scheduled to run every night or at intervals set by the system administrator. It finds all the reference requests that need to be processed, creates an email for each request and then emails it out to the referee with a unique URL so that the referee can access an online form to fill in.

## Reference Request Job - Send Web Reference Emails.xml

NOTE some lines (**in bold**) need to be populated by the System Administrator.

```
<?xml version="1.0"?>
<Job title="Email Reference Requests" autoclose="yes" dateformat="dd/mm/yyyy" delimiter=","
stringquote="&#34;">
  <Library>
    <SetEmailBody smb_text="" smb_wrap="">
      <SetVariable name="Emailbodytext" type="S" value="{smb_text}" />
      <If x1="{emailhtml}" x2="Yes">
        <SetVariable name="EmailBodyText" type="S" sql="select replace(:smb_text, char(13)+char(10),
'&lt;BR/&gt;')" />
        <SetVariable name="EmailBodyText" type="S" sql="select replace(:smb_wrap, '{{bodytext}',
:EmailBodyText)" />
        <SetVariable name="EmailBodyText" type="S" sql="select replace(:EmailBodyText,
'{{signature}', :mysignature) " />
        <SetVariable name="EmailBodyText" type="S" sql="select replace(:EmailBodyText,
'{{origmessage}', '')" />
      </If>
    </SetEmailBody>
    <GetBoilerPlateText gbt_text="" gbp_person="" gbp_vacancy="" gbp_company=""
gbp_placement="" gbp_employment="" gbp_brand="">
      <SetVariable name="BPText" value="" />
      <StaticFunction service="CONVERT_BOILERPLATETEXT" person="{gbp_person}"
company="{gbp_company}" employment="{gbp_employment}" placement="{gbp_placement}"
vacancy="{gbp_vacancy}" brand="{gbp_brand}" text="{gbt_text}" />
      <SetVariable name="BPText" type="S"><![CDATA[{{WebServiceResult}}]></SetVariable>
    </GetBoilerPlateText>
```

```

<DoBranding br_person="" br_vacancy="" br_use="" br_type="">
  <!-- locate your brand based on a new notify status-->
  <SetVariable name="brandid" sql="select BrandID from en_Brand where VisibleID =
GetBrand(:br_person, :br_vacancy, :br_type, :br_use)" />
  <!-- work out the template -->
  <SetVariable name="emailbody" sql="select GetNotifyDocuments(:br_type, :brandid, 'D', '', '')"
/>
  <SetVariable name="emailbody" sql="select replace(:emailbody, '{{REFEREEADDRESSNAME}}',
:REFEREEADDRESSNAME)"/>
  <SetVariable name="emailbody" sql="select replace(:emailbody, '{{yourURL}}', :yourURL)"/>
  <SetVariable name="emailbody" sql="select replace(:emailbody, '{{URLRequestCode}}',
:URLRequestCode)"/>
  <!-- convert the body-->
  <Call gbt_text="{emailbody}" gbp_person="{br_person}" gbp_vacancy="{br_vacancy}"
gbp_brand="{brandid}">GetBoilerPlateText</Call>
  <!-- slot body, default signature into html wrap-->
  <Call smb_text="{BPText}" smb_wrap="{shtmlwrap}">SetEmailBody</Call>
  <!-- <messagedialog>{EmailBodyText}</messagedialog>-->
</DoBranding>
</Library>
  <SetVariable type="S" name="CRLF" sql="select string(char(13),char(10))"/>
  <SetVariable type="S" name="TAB" sql="select string(char(9))"/>
  <SetVariable type="S" name="SMTPHost" value="INSERT SMTP SERVER" />
  <SetVariable type="S" name="SMTPPort" value="INSERT PORT VALUE" />
  <SetVariable type="S" name="SMPTTLS" value="YES OR NO" />
  <SetVariable type="S" name="SMTPUserID" value="INSERT SMTP USER ID" />
  <SetVariable type="S" name="SMTPPassword" value="INSERT SMTP PASSWORD" />
  <SetVariable type="S" name="setBrand" value="INSERT BRAND" />
  <SetVariable type="S" name="yourURL" value="INSERT URL" />
  <ChangeDirectory path="{TempDir}"/>
  <SQLExec ignoreerror="YES">drop table zzzIQXREF_TEMP_Refs</SQLExec>
  <SQLExec>DECLARE LOCAL TEMPORARY TABLE zzzIQXREF_TEMP_Refs (
    ReferenceRequestID char(20),

```

```
Email char(250))
on commit preserve rows;
</SQLExec>
<!-- Get default signature -->
<StaticFunction service="GetEmailSignature"/>
<SetVariable name="mysignature" type="S"><![CDATA[{{WebServiceResult}}]></SetVariable>
<!-- get htmlwrap -->
<SetVariable name="emailhtml" type="S" value="Yes"/>
<SetVariable name="shtmlwrap" type="S" sql="select signature from emailsignature where
emailsignatureid='HTML_WRAP'"/>
<IfSQL condition="'shtmlwrap'=''">
  <SetVariable name="emailhtml" type="S" value="No"/>
</IfSQL>
<SQLExec><![CDATA[
insert into zzzIQXREF_TEMP_Refs
SELECT
ReferenceRequestID, replace(trim(Email), ' ', '') as TrimEmail
FROM "pears"."ReferenceRequest"
where whensent is null and whencompleted is null and whenprocessed is null and
TrimEmail like '%@%.%' and
TrimEmail not like '% %' and TrimEmail not like '%..%' and TrimEmail not like '%.@"%' and
TrimEmail not like '%.%' and TrimEmail not like '%.%' and TrimEmail not like '%@%@"%'
]]>
</SQLExec>
<SQLQuery>
<SQLSelect><![CDATA[
select ReferenceRequest.ReferenceRequestID, person.name as PersonName,
RefereeAddressName, RefereePosition, ReferenceRequest.CompanyName as
RefereeCompanyName, z.Email as RefereeEmail, ReferenceRequest.URLRequestCode,
staff.email as SMTPSender, staff."name" as Staffname, person.personid as pid
FROM ReferenceRequest
inner join staff on staff.staffid=ReferenceRequest.staffid
```

```
inner join person on person.personid = ReferenceRequest.personid

inner join department on department.departmentid=ReferenceRequest.departmentid

inner join zzzIQXREF_TEMP_Refs z on (z.ReferenceRequestID =
ReferenceRequest.ReferenceRequestID)

order by ReferenceRequest.WhenCreated

  ]]>
</SQLSelect>
<IfAnyRows>
  <ForEachRow>
    <Block>
      <Call br_person="{pid}" br_use="WEBREFERENCE">DoBranding</Call>

      <SendEmail subject="{Brand} Reference Request" smtpport="{SMTPPort}"
smtppls="{SMTPTLS}" smtphost="{SMTPHost}" smtpuserid="{SMTPUserID}" html="YES"
smtpsender="{SMTPSender}" smtppassword="{SMTPPassword}">

        <EmailRecipients>
          <WriteLn>{RefereeEmail}</WriteLn>
        </EmailRecipients>

        <EmailBody>{EmailBodyText}</EmailBody>

        <EmailPlainBody>
          <WriteLn><![CDATA[{EmailBodyText}]]></WriteLn>
        </EmailPlainBody>
      </SendEmail>

      <SetVariable name="emailFailed" value=""/>
    </Block>
  </Catch>

  <SetVariable name="emailFailed" value="{Exception}"/>
</Catch>

<IfSQL condition="{emailFailed}' = ''">
  <SQLExec>
    update ReferenceRequest
    set WhenSent = current timestamp
    from zzzIQXREF_TEMP_Refs
```

```
        where ReferenceRequest.ReferenceRequestID = zzzIQXREF_TEMP_Refs.ReferenceRequestID
    </SQLExec>
</IfSQL>
</ForEachRow>
</IfAnyRows>
</SQLQuery>
<SQLExec ignoreerror="YES">drop table zzzIQXREF_TEMP_Refs</SQLExec>
</Job>
```

The URL is unique to each reference request. The URL also expires in 7 day. There are no edit or delete features on the form. If there are any errors, the consultant will have to generate a new reference request.

The email text that is being sent out is standard for all requests. There is no functionality for the text of the email to be dynamic.

## The Summary – Send Web Reference Summary.xml.

NOTE some lines (**in bold**) need to be populated by the System Administrator.

```

<?xml version="1.0"?>

<Job title="Email Reference Requests Summary" autoclose="yes" dateformat="dd/mm/yyyy"
delimiter="," stringquote="&#34;">

  <Library>

    <SetEmailBody smb_text="" smb_wrap="">

      <SetVariable name="Emailbodytext" type="S" value="{smb_text}" />

      <If x1="{emailhtml}" x2="Yes">

        <SetVariable name="EmailBodyText" type="S" sql="select replace(:smb_text, char(13)+char(10),
'&lt;BR/&gt;')" />

        <SetVariable name="EmailBodyText" type="S" sql="select replace(:smb_wrap, '{{bodytext}',
:EmailBodyText)" />

        <SetVariable name="EmailBodyText" type="S" sql="select replace(:EmailBodyText,
'{{signature}', :mysignature) " />

        <SetVariable name="EmailBodyText" type="S" sql="select replace(:EmailBodyText,
'{{origmessage}', '')" />

      </If>

    </SetEmailBody>

    <GetBoilerPlateText gbt_text="" gbp_person="" gbp_vacancy="" gbp_company=""
gbp_placement="" gbp_employment="" gbp_brand="">

      <SetVariable name="BPText" value="" />

      <StaticFunction service="CONVERT_BOILERPLATETEXT" person="{gbp_person}"
company="gbp_company" employment="{gbp_employment}" placement="{gbp_placement}"
vacancy="{gbp_vacancy}" brand="{gbp_brand}" text="{gbt_text}" />

      <SetVariable name="BPText" type="S"><![CDATA[{WebServiceResult}]]></SetVariable>

    </GetBoilerPlateText>

    <DoBranding br_person="" br_vacancy="" br_use="" br_type="">

      <!-- locate your brand based on a new notify status -->

      <SetVariable name="brandid" sql="select BrandID from en_Brand where VisibleID =
GetBrand(:br_person, :br_vacancy, :br_type, :br_use)" />

      <!-- work out the template -->
  
```

```

    <SetVariable name="emailbody" sql="select GetNotifyDocuments(:br_type, :brandid, 'S','','')" />
    <SetVariable name="emailbody" sql="select replace(:emailbody,
'{{JOB_Email_Question_and_Answers}}, :Email_Question_and_Answers)"/>
    <SetVariable name="emailbody" sql="select replace(:emailbody, '{{JOB_RefereeAddressName}}',
:RefereeAddressName)"/>
    <!-- convert the body -->
    <Call gbt_text="{emailbody}" gbp_person="{br_person}" gbp_vacancy="{br_vacancy}"
gbp_brand="{brandid}">GetBoilerPlateText</Call>
    <!-- slot body, default signature into html wrap-->
    <Call smb_text="{BPTText}" smb_wrap="{shtmlwrap}">SetEmailBody</Call>
    <!-- <messagedialog>{EmailBodyText}</messagedialog-->
</DoBranding>
</Library>
<SetVariable type="S" name="CRLF" sql="select string(char(13),char(10))"/>
<SetVariable type="S" name="TAB" sql="select string(char(9))"/>
<SetVariable type="S" name="SMTPHost" value="INSERT SMTP SERVER" />
<SetVariable type="S" name="SMTPPort" value="INSERT PORT VALUE" />
<SetVariable type="S" name="SMTPTLS" value="YES OR NO" />
<SetVariable type="S" name="SMTPUserID" value="INSERT SMTP USER ID" />
<SetVariable type="S" name="SMTPPassword" value="INSERT SMTP PASSWORD" />
<SetVariable type="S" name="setBrand" value="INSERT BRAND" />
<SetVariable type="S" name="yourURL" value="INSERT URL" />
<ChangeDirectory path="{TempDir}"/>
<!-- Get default signature -->
<StaticFunction service="GetEmailSignature"/>
<SetVariable name="mysignature" type="S"><![CDATA[{{WebServiceResult}}]]></SetVariable>
<!-- get htmlwrap -->
<SetVariable name="emailhtml" type="S" value="Yes"/>
<SetVariable name="shtmlwrap" type="S" sql="select signature from emailsignature where
emailsignatureid='HTML_WRAP'"/>
<IfSQL condition=""shtmlwrap"="">
    <SetVariable name="emailhtml" type="S" value="No"/>

```

```
</IfSQL>
```

```
<SQLQuery>
```

```
<Message>Fetching all completed references</Message>
```

```
<SQLSelect><![CDATA[
```

```
select ReferenceRequest.DepartmentID, ReferenceRequest.ReferenceRequestID,
ReferenceRequest.Email, ReferenceRequest.RefereeAddressName, staff.name as Staffname,
person.name as PersonName, person.personid as pid
```

```
FROM ReferenceRequest
```

```
Join staff on ReferenceRequest.StaffID = staff.staffid
```

```
Join person on ReferenceRequest.PersonID = person.personid
```

```
WHERE ReferenceRequest.ReferenceConfirmationRequired = 1 AND
ReferenceRequest.WhenCompleted is not null
```

```
order by ReferenceRequest.WhenCreated
```

```
]]>
```

```
</SQLSelect>
```

```
<IfAnyRows>
```

```
<ForEachRow> <!-- for each reference request -->
```

```
<Message>Fetching {PersonName}'s answers</Message>
```

```
<SetVariable type="S" name="Email_Question_and_Answers" value="" />
```

```
<SQLQuery>
```

```
<SQLSelect><![CDATA[
```

```
SELECT isnull( t.LongDescription, t.description) as question,
```

```
(if t.tagtype= 'D' THEN
dateformat(getquestdate(string('R',:DepartmentID),t.tagid,:ReferenceRequestID),'dd/mm/yyyy')
```

```
else getquestanswer(string('R',:DepartmentID),t.tagid,:ReferenceRequestID) endif) as answer
```

```
FROM tag t
```

```
WHERE t.taglocation = 'R{DepartmentID}' order by t.sortorder
```

```
]]>
```

```
</SQLSelect>
```

```
<IfAnyRows>
```

```
<ForEachRow> <!-- for each question + answer -->
```

```
<SetVariable type="S" name="Email_Question_and_Answers" value="
```



```

    {Email_Question_and_Answers}
    &lt;tr&gt;
        &lt;td&gt; &lt;strong&gt;{question}&lt;/strong&gt; &lt;/td&gt;
        &lt;td&gt;{answer}&lt;/td&gt;
    &lt;/tr&gt;
" />
</ForEachRow>
</IfAnyRows>
</SQLQuery>
<Call br_person="{pid}" br_use="WEBREFERENCE">DoBranding</Call>
<Block>
    <Message>Sending Email to {RefereeAddressName} / {Email}</Message>
    <SendEmail subject="{Brand} Reference Summary" smtpport="{SMTPPort}"
smtppls="{SMTPTLS}" smtphost="{SMTPHost}" smtpuserid="{SMTPUserID}" html="YES"
smtpsender="{SMTPSender}" smtppassword="{SMTPPassword}">
        <EmailRecipients>
            <WriteLn>{Email}</WriteLn>
        </EmailRecipients>
        <EmailBody>{EmailBodyText}</EmailBody>
        <EmailPlainBody>
            <WriteLn><![CDATA[{EmailBodyText}]]></WriteLn>
        </EmailPlainBody>
    </SendEmail>
    <SetVariable name="emailFailed" value=""/>
    <Message>Updating Table</Message>
    <SQLExec><![CDATA[UPDATE ReferenceRequest SET ReferenceConfirmationRequired = 0
WHERE ReferenceRequestID = '{ReferenceRequestID}']]></SQLExec>
</Block>
<Catch>
    <SetVariable name="emailFailed" value="{Exception}"/>
</Catch>
</ForEachRow>

```

```
</IfAnyRows>  
</SQLQuery>  
</Job>
```