

Sample Code

Updated : 20/05/2021

Creating a Customer Invoice

```
run
// Create Gateway Object
Gateway oGW = new Gateway();

// Set the location of the Infusion Data folder
oGW.setpath(@"C:\Infusion 8.430\data");

// Login to the Gateway, Use your application's tile as the parameter
// This tells Infusion your application is there and accessing the data.
// In the case Infusion is unable to be accessed for example when a backup is being

// the login will return 1 on successful login.

if ( oGW.login("My Application Name") != 1)
{
    MessageBox.Show("Unable to Login");
    return;
}

String lcInvNum = "";
Int32 lnLink = 0;

// Get the Next Invoice Number from Infusion
lcInvNum = Convert.ToString(oGW.nextinv(0));

// Ensure all Gateway Parameter Variables are set to default
oGW.varreset();

// Set Parameters required for CreateInvoice function
oGW.PACCOUNT= 22;
oGW.PLOCATION = 1;
oGW.PDATE = DateTime.Now.Date;
oGW.PINVNUMBER = lcInvNum;

// Create the Invoice Header
// This will return the LINK number of the Invoice
// This value is used in the AddLine and TotalInvoice functions.
lnLink = oGW.CreateInvoice();

// Ensure all Gateway Parameter Variables are set to default
oGW.varreset();

// Set Parameters required for InvoiceAddLine function
oGW.PPCODE = "27108 ";
oGW.PQTY = 1;
oGW.PAUTOPRICE = 1;
oGW.PLINK = lnLink;
oGW.InvoiceAddLine();

// Alternatively use a GL ID instead of a Product Code
// Auto Pricing is not available when using GL ID's.
```

```

// Ensure you populate pCostRate, pRate and pDiscount.
// pPCODE should be left as an empty string in this case.
oGW.PPCODE = "";
oGW.PGLID = "1050";
oGW.PQTY = 1;
oGW.PRATE = 115.00;
oGW.PDISCOUNT = 0.00;
oGW.PCOSTRATE = 50.00;
oGW.PLINK = lnLink;
oGW.InvoiceAddLine();

// Repeat varreset and InvoiceAddLine as required.
oGW.varreset();

oGW.PPCODE = "41449 ";
oGW.PQTY = 2;
oGW.PAUTOPRICE = 1;
oGW.PLINK = lnLink;
oGW.InvoiceAddLine();

oGW.varreset();

oGW.PPCODE = "41471 ";
oGW.PQTY = 3;
oGW.PAUTOPRICE = 1;
oGW.PLINK = lnLink;
oGW.InvoiceAddLine();

oGW.varreset();

oGW.PPCODE = "41472 ";
oGW.PQTY = 4;
oGW.PAUTOPRICE = 1;
oGW.PLINK = lnLink;
oGW.InvoiceAddLine();

oGW.varreset();

// Finally ensure the TotalInvoice function is called
// This ensures the invoices matches the lines added
// This is good practice, but not mandatory, the InvoiceAddLine
// function does re total the Invoice each time.

// Set the Invoice Link Number Variable
oGW.PLINK = lnLink;

// Total the Invoice
oGW.totalinvoice();

```

Editing a Customer Invoice

Depending on what you want to edit on the Customer Invoice, alters the way in which you would approach it.

- To change something that is on the Invoice Header (Like the Order Number or add a Sales Rep ID)

```
run
// Create Gateway Object
Gateway oGW = new Gateway();

// Set the location of the Infusion Data folder
oGW.setpath(@"C:\Infusion 8.430\data");

// Login to the Gateway, Use your application's tile as the parameter
// This tells Infusion your application is there and accessing the data.
// In the case Infusion is unable to be accessed for example when a backup is being

// the login will return 1 on successful login.

if (oGW.login("My Application Name") != 1)
{
    MessageBox.Show("Unable to Login");
    return;
}

Int32 lnResult = 0;

oGW.varreset();

oGW.PLINK = 12345;
oGW.PINVORDNUM = "NEWORDER";

lnResult = oGW.UpdateInvHeader("INVORDNUM");
```

- To add a line to an existing PENDING Invoice:

```
run
// Create Gateway Object
Gateway oGW = new Gateway();

// Set the location of the Infusion Data folder
oGW.setpath(@"C:\Infusion 8.430\data");

// Login to the Gateway, Use your application's tile as the parameter
// This tells Infusion your application is there and accessing the data.
// In the case Infusion is unable to be accessed for example when a backup is being

// the login will return 1 on successful login.

if (oGW.login("My Application Name") != 1)
{
    MessageBox.Show("Unable to Login");
    return;
}
```

```

Int32 lnResult = 0;

oGW.varreset();

oGW.PLINK = 12345;
oGW.PPCODE = "PRODUCTA";
oGW.PAUTOPRICE = 1;
oGW.PQTY = 1;

lnResult = oGW.InvoiceAddLine();

// Not essential but good practice is to call TotalInvoice function
// after you have completed all the line changes.
oGW.totalinvoice();

```

- Delete a line from a PENDING Invoice

This is a little trickier and requires you to actually locate to the correct row in the database, so should be used VERY carefully.

```

run
// Create Gateway Object
Gateway oGW = new Gateway();

// Set the location of the Infusion Data folder
oGW.setpath(@"C:\Infusion 8.430\data");

// Login to the Gateway, Use your application's tile as the parameter
// This tells Infusion your application is there and accessing the data.
// In the case Infusion is unable to be accessed for example when a backup is being

// the login will return 1 on successful login.

if (oGW.login("My Application Name") != 1)
{
    MessageBox.Show("Unable to Login");
    return;
}

Int32 lnResult = 0;

oGW.Open("invln", "a_invln");
oGW.Select("a_invln");
lnResult = oGW.Locate("code='PRODUCTA' and link=12345 and qty=2");
if (lnResult==1)
{
    // Record Found attempt delete
    lnResult = oGW.InvoiceDeleteLine("a_invln");

    // If lnResult is 1 then the line has been deleted
    // Otherwise check LASTERRORMESSAGE for the reason it was not deleted.
    // ? Invoice is POSTED
}

```

```
oGW.Close("a_invln");
```

- Changing the quantity or rate of an existing line on a PENDING Invoice

Use the InvoiceDeleteLine function first then re add the line using InvoiceAddLine for the new quantity/rate.

Posting a Customer Invoice

Allows the posting of basic Customer Invoices and is available from Gateway 833 onward.

```
run
// Create Gateway Object
Gateway oGW = new Gateway();

// Set the location of the Infusion Data folder
oGW.setpath(@"C:\Infusion 8.430\data");

// Login to the Gateway, Use your application's tile as the parameter
// This tells Infusion your application is there and accessing the data.
// In the case Infusion is unable to be accessed for example when a backup is being

// the login will return 1 on successful login.

if (oGW.login("My Application Name") != 1)
{
    MessageBox.Show("Unable to Login");
    return;
}

Int32 lnLink = 838902;
Int32 lnResult = 0;
lnResult = oGW.CustomerInvoice_Post(lnLink);
```

Creating a Customer Quote

```
run

// Create Gateway Object
Gateway oGW = new Gateway();

// Set the location of the Infusion Data folder
oGW.setpath(@"C:\Infusion 8.430\data");

// Login to the Gateway, Use your application's tile as the parameter
// This tells Infusion your application is there and accessing the data.
// In the case Infusion is unable to be accessed for example when a backup is being

// the login will return 1 on successful login.

if (oGW.login("My Application Name") != 1)
{
    MessageBox.Show("Unable to Login");
    return;
}

String lcQuoteNum = "";
Int32 lnLink = 0;

// Get the Next Quote Number from Infusion
lcQuoteNum = Convert.ToString(oGW.nextquote(0));

// Ensure all Gateway Parameter Variables are set to default
oGW.varreset();

// Set Parameters required for CreateInvoice function
oGW.PACCOUNT = 22;
oGW.PLOCATION = 1;
oGW.PDATE = DateTime.Now.Date;
oGW.PQUOTENUMBER = lcQuoteNum;

// Create the Quote Header
// This will return the LINK number of the Quote
// This value is used in the QuoteAddLine and TotalQuote functions.
lnLink = oGW.CreateQuote();

// Ensure all Gateway Parameter Variables are set to default
oGW.varreset();

// Set Parameters required for InvoiceAddLine function
oGW.PPCODE = "27108 ";
oGW.PQTY = 1;
oGW.PAUTOPRICE = 1;
oGW.PLINK = lnLink;
oGW.QuoteAddLine();

// Repeat varreset and QuoteAddLine as required.
oGW.varreset();

oGW.PPCODE = "41449 ";
oGW.PQTY = 2;
oGW.PAUTOPRICE = 1;
```

```
oGW.PLINK = lnLink;
oGW.QuoteAddLine();

oGW.varreset();

oGW.PPCODE = "41471 ";
oGW.PQTY = 3;
oGW.PAUTOPRICE = 1;
oGW.PLINK = lnLink;
oGW.QuoteAddLine();

oGW.varreset();

oGW.PPCODE = "41472 ";
oGW.PQTY = 4;
oGW.PAUTOPRICE = 1;
oGW.PLINK = lnLink;
oGW.QuoteAddLine();

oGW.varreset();

// Finally ensure the TotalInvoice function is called
// This ensures the invoices matches the lines added
// This is good practice, but not mandatory, the InvoiceAddLine
// function does re total the Invoice each time.

// Set the Invoice Link Number Variable
oGW.PLINK = lnLink;

// Total the Quote
oGW.TotalQuote();
```


Editing a Customer Quote

Depending on what you want to edit on the Customer Quote, alters the way in which you would approach it.

- To change something that is on the Quote Header (Like the Order Number or add a Sales Rep ID)

Currently there is no function like the UpdateInvoiceHeader used for Invoices. You need to do this using the lower level functions.

```
run
// Create Gateway Object
Gateway oGW = new Gateway();

// Set the location of the Infusion Data folder
oGW.setpath(@"C:\Infusion 8.430\data");

// Login to the Gateway, Use your application's tile as the parameter
// This tells Infusion your application is there and accessing the data.
// In the case Infusion is unable to be accessed for example when a backup is being

// the login will return 1 on successful login.

if (oGW.login("My Application Name") != 1)
{
    MessageBox.Show("Unable to Login");
    return;
}

Int32 lnResult = 0;

oGW.Open("quoteidx", "a_quoteidx");
oGW.Select("a_quoteidx");
lnResult = oGW.Locate("link=12345");
if (lnResult==1)
{
    // Record Found attempt lock and check it is pending.
    if (oGW.Get("posted")==0)
    {
        if (oGW.reclock()==1)
        {
            // Record has been locked and can be edited
            oGW.Replace("ordnum", "NEWORDERNUM");

            oGW.recunlock();
        }
        else
        {
            // Record is in use by another user an cannot be edited
            MessageBox.Show("Record cannot be locked at this time.");
        }
    }
    else
    {

```

```

        MessageBox.Show("Quote has been posted / converted to an Invoice");
    }
}
else
{
    MessageBox.Show("Link not found");
}

oGW.Close("a_quoteidx");

```

- To add a line to an existing PENDING Quote:

```

run
// Create Gateway Object
Gateway oGW = new Gateway();

// Set the location of the Infusion Data folder
oGW.setpath(@"C:\Infusion 8.430\data");

// Login to the Gateway, Use your application's tile as the parameter
// This tells Infusion your application is there and accessing the data.
// In the case Infusion is unable to be accessed for example when a backup is being

// the login will return 1 on successful login.

if (oGW.login("My Application Name") != 1)
{
    MessageBox.Show("Unable to Login");
    return;
}

Int32 lnResult = 0;

oGW.varreset();

oGW.PLINK = 12345;
oGW.PPCODE = "PRODUCTA";
oGW.PAUTOPRICE = 1;
oGW.PQTY = 1;

lnResult = oGW.QuoteAddLine();

// Not essential but good practice is to call TotalInvoice function
// after you have completed all the line changes.
oGW.TotalQuote();

```

- Delete a line from a PENDING Quote

This is a little trickier and requires you to actually locate to the correct row in the database, so should be used VERY carefully.

```

// Create Gateway Object
Gateway oGW = new Gateway();

```

```

run
// Set the location of the Infusion Data folder
oGW.setpath(@"C:\Infusion 8.430\data");

// Login to the Gateway, Use your application's tile as the parameter
// This tells Infusion your application is there and accessing the data.
// In the case Infusion is unable to be accessed for example when a backup is being

// the login will return 1 on successful login.

if (oGW.login("My Application Name") != 1)
{
    MessageBox.Show("Unable to Login");
    return;
}

Int32 lnResult = 0;

oGW.Open("quote", "a_quoteLn");
oGW.Select("a_quoteLn ");
lnResult = oGW.Locate("code='PRODUCTA' and link=12345 and qty=2");
if (lnResult==1)
{
    // Record Found attempt delete
    lnResult = oGW.DeleteDeleteLine("a_quoteLn");

    // If lnResult is 1 then the line has been deleted
    // Otherwise check LASTERRORMESSAGE for the reason it was not deleted.
    // ? Quote is POSTED
}

oGW.Close("a_quoteLn");

```

- Changing the quantity or rate of an existing line on a PENDING Quote

Use the QuoteDeleteLine function first then re add the line using QuoteAddLine for the new quantity/rate.

Editing a Customer Account

```
run
// Create Gateway Object
Gateway oGW = new Gateway();

// Set the location of the Infusion Data folder
oGW.setpath(@"C:\Infusion 8.430\data");

// Login to the Gateway, Use your application's tile as the parameter
// This tells Infusion your application is there and accessing the data.
// In the case Infusion is unable to be accessed for example when a backup is being

// the login will return 1 on successful login.

if (oGW.login("My Application Name") != 1)
{
    MessageBox.Show("Unable to Login");
    return;
}

Int32 lnResult = 0;

oGW.Open("customer", "a_customer");
oGW.Select("a_customer");
lnResult = oGW.Locate("account=12345");
if (lnResult==1)
{
    // Record Found attempt lock
    if (oGW.reclock()==1)
    {
        // Record has been locked and can be edited
        oGW.Replace("name", "Infusion Business Software Limited");
        oGW.Replace("add1", "11 Picton Ave");
        oGW.Replace("add2", "Riccarton");
        oGW.Replace("add3", "Christchurch");
        oGW.Replace("postcode", "8013");

        // Update the filtered search sting. This is good practice otherwise
        // a reindex is required in the software.
        oGW.Cust_UpdateSearchPlus(12345);

        oGW.recunlock();
    }
    else
    {
        // Recrod is in use by another user an cannot be edited
        MessageBox.Show("Record cannot be locked at this time.");
    }
}

}
else
{
    MessageBox.Show("Account not found");
}
```

```

}

oGW.Close("a_customer");

```

Reading details from a Customer Account

The same logic used here can be used to read a Supplier, Product, Invoice, Quote etc.

```

run
// Create Gateway Object
Gateway oGW = new Gateway();

// Set the location of the Infusion Data folder
oGW.setpath(@"C:\Infusion 8.430\data");

// Login to the Gateway, Use your application's tile as the parameter
// This tells Infusion your application is there and accessing the data.
// In the case Infusion is unable to be accessed for example when a backup is being

// the login will return 1 on successful login.

if (oGW.login("My Application Name") != 1)
{
    MessageBox.Show("Unable to Login");
    return;
}

Int32 lnResult = 0;
string lcValue = "";

oGW.Open("customer", "a_customer");
oGW.Select("a_customer");
lnResult = oGW.Locate("account=12345");
if (lnResult==1)
{
    // Record Found
    lcValue=oGW.Get("Name");
    lcValue=oGW.Get("Add1");
}
else
{
    MessageBox.Show("Account not found");
}

oGW.Close("a_customer");

```

Creating Job

```
Gateway oGW = new Gateway();
oGW.setpath(@"c:\infusion 8.516 nz\--demo--");
oGW.login("Test Application");

Int32 nNextJob = oGW.nextjob(0);

Int32 nJobLink = oGW.CreateJob(1000, nNextJob.ToString(), "STAFF", "*STD*");

// The job will have been created with the default values from the customer
account.
// Name, Addresses, Phone Numbers, Price Level, as well as the supplied Job Type
and Opened By.
// All other values and these can then be updated using standard replace commands.

oGW.Open("jobidx", "a_jobidx");
oGW.Select("a_jobidx");
oGW.setorder("link");
Int32 lnResult = oGW.Locate("link = " + nJobLink.ToString());

if (lnResult == 1)
{
    if (oGW.reclock() == 1 )
    {
        oGW.Replace("name", "Updated Customer Name Value");
        oGW.Replace("startdate", DateTime.Today.Date);
        oGW.Replace("jobnotes", "Notes for the Job" + Environment.NewLine +
"Second Line of information");

        oGW.recunlock();

    }
    else
    {
        MessageBox.Show("Unable to lock Job for editing.");
    }
}
else
{
    MessageBox.Show("Unable to locate job.");
}

oGW.Close("a_jobidx");
oGW.Logout();

oGW.Destroy();
```

Creating a Job Line

```
Gateway oGW = new Gateway();
oGW.setpath(@"c:\test\infusion 9.106 nz\data");
oGW.login("Test Application");

// Reset Gateway Variables
oGW.varreset();

// Populate Gateway Variables
oGW.PDATE = new DateTime(2022, 3, 21);
oGW.PJOBID = "21729";
oGW.PPCODE = "3020";
oGW.PQTY = 2;
// Let Infusion work out the price.
oGW.PAUTOPRICE = 1;

// Optional way of passing prices.
oGW.PAUTOPRICE = 0;
oGW.PRATE = 123.45;    // This is GST Inclusive or Exclusive based on Infusion Setting.
oGW.PDISCOUNT = 10.00; // This would apply a 10% discount to the 123.45
oGW.PCOSTRATE = 59.95; // This is always GST Exclusive.

// Pass optional start and end time information.
// You must still provide the Quantitiy field as thse are only informational fields.
oGW.PFROM = "11:25";
oGW.PTO = "11:45";

// Create Transaction
oGW.postjobtxn();

oGW.Logout();
```