Productive Computing

Exchange Manipulator

Outlook Manipulator

Functions Guide

Revised June 17, 2014

TABLE OF CONTENTS

I. Introduction

Description:

The Outlook/Exchange Manipulator plug-ins are powerful tools used to exchange data between FileMaker® and Microsoft® Outlook (32-bit). With this plug-in FileMaker users are able to bidirectionally exchange data between FileMaker and Outlook. These operations are accomplished using FileMaker function calls from within FileMaker calculations. These calculations are generally determined from within FileMaker "SetField" or "If" script steps. This document describes all available plug-in functions and Outlook fields. Please see the accompanying "Developer's Guide" for a list of the integration steps, features and error handling.

Intended Audience:

FileMaker developers or persons who have knowledge of FileMaker scripting, calculations and relationships as proper use of the plug-in requires that FileMaker integration scripts be created in your FileMaker solution.

Successful Integration Practices:

1) Read the Developer's Guide
2) Read the Functions Guide
3) Watch our video tutorials: http://productivecomputing.com/video/
4) Review our FileMaker demo file
5) Familiarize yourself with Microsoft Outlook

II. Function Descriptions

This section describes the functions that are available with the Outlook/Exchange Manipulator plug-ins.

1) Registration and Other Functions

PCEM_Authenticate(Profile ; optPassword)

Description:

Opens an Outlook profile for reading and writing. If there is an error opening the designated Profile, the user will be prompted to choose a profile. The plug-in will work with the currently opened Outlook profile. If Outlook is not opened, the default profile will be used. Authenticate must be called before any other Outlook function once per FileMaker session.

Parameters:

Profile = the name of the Profile to use for the session (Optional)

Password = optional password for the designated profile (Optional)

Return Values:

0 for success or !!ERROR!!

PCEM_GetLastError(type)

Description:

Returns a textual or numeric description of the last error the plug-in encountered. This gives the developer the option to trap errors as a text or number.

Parameters:

Type = Values are either Text or Number. The default value is text

Return Values:

The desired error description or error number.

PCEM_GetOperatingMode

Description:

Used to identify the operating mode of the plug-in. Operating mode indicates if the plug-in has been properly registered.

Parameters: none

Return Values:

"DEMO" for demo mode, "LIVE" for live mode, "UNREGISTERED" for unregistered mode, or "EXPIRED" for expired mode after 30 minute or 30 day timeout.

PCEM_OutlookVersion

Description:

Once authenticated to a particular Outlook installation, this function can return the Outlook version number.

Parameters:

Return Values:

A version number on success, otherwise an !!ERROR!!

PCEM_Register(ServerName ; ServerPort ; ServerPage ; licenseID)

Description:

Registers the plug-in with Productive Computing's registration servers. Must call Register function once before using the plug-in.

Parameters:

ServerName = "licensing.productivecomputing.com"

ServerPort = "80"

ServerPage = "/PCIReg/pcireg.php"

LicenseID = "your license ID" or "DEMO-OM6" or "DEMO-EM4"

Return Values:

0 for success or !!ERROR!! or error text. Anything other than 0 is an error.

PCEM_Version(type)

Description:

Used to identify the product and product version installed on a machine.

Parameters:

type = 'Short' or 'Long'

The short version string includes only the version number of the plug-in. For example: "5.0.0.5".

The long includes the plug-in name and version number. For example: "Outlook Manipulator 5.0.0.5"

The default version string returned is "long".

Return Values:

Product name and/or version or ?

2) Record Specific Functions

PCEM_AddOneOffMember(Name ; Address ; Type)

Description:

 Adds a name, email address and type to the currently opened distribution list. Currently you can only add One Off Members to NEW distribution lists.

Parameters:

Name = The name for the addressee.

Address = The email address for the addressee.

Type = The type for the address. Currently only SMTP addresses are accepted

Return Values:

0 for success or !!ERROR!!

PCEM_CreateCustomField(Name ; Type)

Description:

Creates a custom field for the specified type and adds it to the currently opened record.

Parameters:

Name = The name for the custom field.

Type = The type of data that will be contained in the custom field.

Type can take the following values: Currency, Date (resolves to DateTime), Time (resolves to DateTime), DateTime (resolves to DateTime), Duration, Keywords, Number, Percent, Text, YesNo, or Integer

Return Values:

0 for success or !!ERROR!!

PCEM_DeleteCurrentRecord(bPermanent)

Description:

Deletes the currently opened record and maintains the index of items. A call to OpenNextRecord will return the next record in the found set of records.

Parameters:

 bPermanent = A "True" value permanently deletes the item. A "False" value moves the item to the deleted items folder in Outlook.

Return Values:

 0 for success or !!ERROR!!

PCEM_DeleteRecord(OutlookID ; Permanent)

Description:

Deletes a record in the currently opened folder.

Parameters:

OutlookID = The Outlook ID of the record to delete

Permanent = "Y" or empty ("")

Return Values:

0 for success or !!ERROR!!

Notes:

Permanent (Parameter 2)

: Use **"Y"** to delete records permanently. This will remove records from outlook WITHOUT moving them to the "Deleted Items" folder. We suggest you do not use this option especially if you have a more than 100 records in a given Outlook module. This option can take several seconds to delete a single record.

: Use **blank** "" to move the item to the "Deleted Items" folder. This option will move the object to the "Deleted Items" folder instantly which is the preferred method when dealing with many records in a given Outlook module. For instance, if you are to delete 1000 items from a set of 3000 items, this method should be used and the second parameter should be left blank. Later you can permanently delete the items from the "Deleted Items" folder simply by calling the command "PCEM_EmptyAllDeletedItems."

PCEM_Display(ShowInForeground)

Description:

Displays the current active message in Outlook opened with the PCEM_OpenRecord, PECM_GetFirstRecord or PCEM_GetNextRecord.

Parameters:

ShowInForeground - Accepts boolean values true and false (or 1 and 0). The default is true. Forces Outlook 2010 to bring the desired message to the foreground. Earlier versions of Outlook (pre 2010) caused the message to be displayed modally in Outlook.

Return Values:

0 for success or !!ERROR!!

Notes:

When performing operations in FileMaker 13, some unexpected behavior in regards to dialog boxes may occur. The following is a list of observed behavior that occurs only in FileMaker 13:

- When using "PCEM_Display" after opening a record and passing a parameter value of "1" or blank, the record will appear in an Outlook modal window, as expected. After a short period of time, FileMaker will display a "Server Busy" status window that will persist until the Outlook modal window is closed. This behavior does not occur if PCEM_Display is called with a parameter value of "0".

- If a dialog window appears in Outlook as a result of a function call from FileMaker, after a short period of time, a "Server Busy" status window will appear and will persist until the dialog window in Outlook is closed.

PCEM_FilterByFilterString(FilterString)

Description:

The function is designed to restrict the records available to the plug-in to those that meet the criteria stated in the FilterString parameter. The filter function should come right after the PCEM_OpenFolder command. The available records are also limited to those in the currently opened folder. For example, passing [Subject] = "Sample Subject"' to the function will restrict the available items to those in the current folder with the subject field set to "Sample Subject."

Filter strings should be constructed in a very specific manner known as the "Restrict Method." The "Restrict Method" is explained in detail on the following Microsoft websites:

http://msdn.microsoft.com/en-us/library/bb220369.aspx

http://msdn2.microsoft.com/en-us/library/aa210275(office.11).aspx

You can also search for the "Restrict Method" available in Visual Basic for Applications. Review the "Reference" section of this link http://msdn.microsoft.com/en-us/library/aa221877.aspx to see which properties are available to which Outlook objects and which properties are NOT supported by the Restrict Method."

Parameters:

FilterString – A string that defines the filter to be applied to the current folder. Please ensure that only valid parameters are passed as invalid parameters can result in a "?" and/or other undesirable results.

Return Values:

0 for success or !!ERROR!!

Live demo of the FiilterByFilterString is available on the Contacts tab of our Filemaker Demo file by selecting the "Apply Filters" button.

PCEM_FilterByLastModified(Timestamp)

Description:

The function is designed to restrict the records available to the plug-in to those that are in the current folder and have a modified timestamp later than that passed into the function. Note that though the function takes a full timestamp, records will only be filtered to the exact minute and not the second.

Parameters:

Timestamp – A FileMaker Timestamp. Please ensure that only valid parameters are passed as invalid parameters can result in a "?" and/or other undesirable results.

Return Values:

0 for success or !!ERROR!!

Live demo of the FiilterByLastModified is available on the Contacts tab of our Filemaker Demo file by selecting the "Apply Filters" button.

PCEM_GetCustomFieldData(FieldName)

Description:

Extracts the value contained in the specified field. The field must exist in the form that the Outlook Item/ Record was created with.

Parameters:

FieldName = The name of the Outlook field to extract the data from

Return Values:

The string representation of the field value or !!ERROR!!

PCEM_GetFieldData(FieldName)

Description:

Extracts the value contained in the specified field.

Parameters:

FieldName = The name of the Outlook field to extract the data from (see "Available Outlook Fields" section)

Return Values:

The string representation of the field value or !!ERROR!!

PCEM_MoveRecord(FolderPath)

Description:

Moves the currently opened Outlook Item from its current folder to another folder. Note that a call to PCEM_OpenFolder must precede this call. Generally when working with a script loop the function is called in order to obtain the source record. However, there are occasions where the user will obtain the record to be moved via the PCEM_OpenRecord function. In this case the user must call PCEM_OpenFolder for either the source folder or the destination folder.

Parameters:

FolderPath = The path of the destination folder

Return Values:

A new Outlook ID for the record or !!ERROR!!

Notes:

Non-Exchange - Must include the full path of the folder to move the item into. The destination folder becomes the currently opened folder.

Exchange - Must include the full path of the folder to move the item into including the desired root folder (Mailbox / Public Folder). The destination folder becomes the currently opened folder.

PCEM_NewRecord(optModule)

Description:

Creates a new record/Item in the current Outlook Folder or creates a new distribution list. This record may be modified using calls to SetField(); Must call OpenFolder first to set the folder that the item will be saved to.

Parameters:

optModule = The type of record to create. This is an optional parameter and if left blank the new record will match the type for the current folder. Valid values are "Post," "Calendar," "Tasks," "Contacts," "Note," "Journal," "Mail," "Distribution List".

Return Values:

0 for success or !!ERROR!!

PCEM_OpenRecord(OutlookID)

Description:

Opens the specified record for editing

Parameters:

OutlookID = The Outlook ID of the item to open.

Return Values:

0 for success or !!ERROR!!

PCEM_OpenSavedItem(strPath)

Description:

Opens the saved MSG file. Please see PCEM_SaveRecordAsMsg (strPath) function in order to save an email as a MSG file on your hard drive. MSG is the file format that Microsoft and Outlook use to save individual messages to a hard disk.

Parameters:

strPath = Absolute path to the MSG file. Must be a full windows path including the drive letter and destination file name. For example: "C:\SomeFolder\MySavedMessage.msg"

Return Values:

0 for success or !!ERROR!!

PCEM_SaveRecord(optionalParam)

Description:

Saves the record to the currently opened Outlook folder and returns the Outlook ID. If the item being saved is an e-mail, then the message will be sent. Sending email using SaveRecord is known as the "long" method to send email and requires the following functions (PCEM_NewRecord, PCEM_SetFieldData, PCEM_SaveRecord). We recommend using the "short" method using PCEM_SendMail function to send email out in a single step. It is far easier to program and supports HTML email via a parameter setting.

The only time you need to create an email using the "long" method is if you need to obtain the Outlook ID for the email being sent, set additional fields such as SendOnBehalfOf, From, Categories, Flag Icon, if you have a calendar/appointment type email or desire to delay sending an email. If you have a calendar/ appointment type email record or wish to delay sending an email, then use the optionalParam explained below.

Parameters:

optionalParam = "Appointment", "ApptResponse" or "DontSend"

The literal string "Appointment" sends the meeting request to Attendees. Use semi colon between multiple attendees. Option to auto invite "Optional Attendees" or "Required Attendees" if using MS Exchange. When calling SaveRecord, use the word "Appointment". For example: PCEM_SaveRecord ("Appointment").

The literal string "DontSend" delays sending the email. For example you may desire to have the plug-in send the email and then open the email in Outlook before the email is sent. This can be accomplished by this pseudo script:

...

Set Variable[$outlookID ; PCEM_SaveRecord("DontSend")]

Set Variable[$result ; PCEM_OpenRecord($outlookID)]

Set Variable[$result ; PCEM_Display("false")]

...

The parameter "ApptResponse" will send out a meeting response to the currently opened meeting invitation event. This will have Outlook create a meeting response message and send the message to the organizer of the meeting with the intended response.

Return Values:

The Outlook ID for the Item or !!ERROR!!

Note:

The "ApptResponse" parameter currently only supports sending a response of "Accepted" via Outlook. It does not yet support "Declined" or "Tentatively Accepted" meeting response messages.

PCEM_SaveRecordAsMsg (strPath)

Description:

Saves the currently opened email as a MSG formatted file. This is the file format that Microsoft and Outlook use to save individual messages to the local hard disk.

Parameters:

strPath = Absolute path to a folder where the email will be saved. Must be a full windows path including the destination file name. For example: "C:\SomeFolder\MySavedMessage.msg"

Return Values:

0 for success or !!ERROR!!

PCEM_SendMail(To ; Cc ; Bcc ; Subject ; Body ; Attachments ; optUseHTML)

Description:

Use to send mail in a single step. This method for sending email is known as the short method and sends an e-mail message to the intended recipients with the given subject, body, attachments and formatting options. All parameters except Subject, Body and OptUseHTML can contain multiple comma separated values.

We recommend using the "PCEM_SendMail" function to send email out in a single step. It is far easier to program and supports HTML email via a parameter setting. The long method of sending emails requires the use of PCEM_NewRecord, PCEM_SetFieldData, PCEM_SaveRecord. Use the long method if you need to obtain the Outlook ID for the email being sent, set additional mail fields or if you have a calendar/ appointment type email. Please see the PCEM_SaveRecord function for additional information.

Parameters:

To	The recipients of the e-mail
Cc	The carbon copy recipients
Bcc	The blind carbon recipients
Subject	The subject of the message
Body	The Body of the message
Attachments	Any attachments for the message. Entire file path is needed including drive letter.
optUseHTML	'Y' for HTML formatted, 'N' or '' for plain text formatting

Return Values:

0 for success or !!ERROR!!

**In our FileMaker demo file we demonstrate sending attachments stored in a container field. However you can also send attachments from any path on your computer or network drive by simply specifying the full path to the attachment including the drive letter. Be sure to include the name of the file when specifying the path and use commas between multiple attachments.

For example: PCEM_SendMail(To ; Cc ; Bcc ; Subject ; Body ; "C:/test.pdf" ; optUseHTML)**

PCEM_SetCustomFieldData(FieldName ; Data ; optType)

Description:

Sets the value of a custom field of an Item created with a custom form. The custom field needs to be part of the Message Class of the Outlook Item.

Parameters:

FieldName = Name of the Outlook field to populate (See 'Available Outlook Fields' section)

Data = The information to populate the Outlook field.

optType = The desired type for the new custom field. Valid values are: Boolean, Text, Percent, Number, Keywords, Duration, Date, Time, DateTime, Currency, or Integer

Return Values:

0 for success or !!ERROR!!

PCEM_SetFieldData(FieldName ; Data)

Description:

Sets the specified field to the specified value.

Parameters:

FieldName = Name of the Outlook field to populate (See 'Available Outlook Fields' section)

Data = The information to populate the Outlook field.

Return Values:

0 for success or !!ERROR!!

PCEM_VerifyFieldData(FieldName ; VerifyData)

Description:

Compare the specified value to the value contained in the specified field

Parameters:

FieldName = The name of the Outlook filed to compare VerifyData to.

VerifyData = The data to compare to the contents of the Outlook field identified by FieldName

Return Values:

0 if VerifyData matches the contents of the Outlook field or !!ERROR!!

3) Attachment Specific Functions

PCEM_AddAttachment(FilePath ; Name)

Description:

Adds an attachment to the currently opened item.

Parameters:

FilePath = The full path to the file to attach including the drive letter. For example: U:\SomeFolder\SomeFile.pdf

Name = The name of the file as it is to appear in Outlook. If this parameter is left blank then the filename will be used.

Return Values:

0 for success or !!ERROR!!

PCEM_GetAttachmentCount

Description:

Returns the number of attachments the currently opened item has. This function returns a text value. In order to perform numeric comparisons the result should be wrapped with the GetAsNumber function. [e.g. GetAsNumber (PCEM_GetAttachmentCount) > 100].

Parameters:

Return Values:

The count or !!ERROR!!

PCEM_GetAttachmentName(Index ; OptTruncated)

Description:

This function returns the name of a given attachment specified by its index. The second parameter "optTruncate" can truncate the name to a shorter length with a unique name if there are naming conflicts. Full name of attachment is returned unless the you set the OptTruncated parameter to "True."

Parameters:

Index = The index of the attachment. Values are 1-n where n is less than or equal to the number of attachments available.

OptTruncated = Allows an 8 character truncated attachment name to be returned. Values are "True" or "False." This parameter defaults to "false", but a value of "true" can be specified to truncate the name.

Return Values:

The name of the attachment or !!ERROR!!

PCEM_GetAttachmentSize(Index)

Description:

This function can return the size of a particular attachment, specified in bytes.

Parameters:

Index = Specifies which attachment to return the size for. Values are 1-n where n is the number of attachments available for a given message.

Return Values:

Some integer specifying the size of the attachment in bytes, otherwise an !!ERROR!!

PCEM_SaveAttachment(FilePath ; Index)

Description:

Saves the specified attachment to a specified location.

Parameters:

FilePath = The full file path where attachments will be saved to when importing email attachments from Outlook into FileMaker. For example: "U:\SomeFolder\SomeFileName" . You can use PCEM_GetAttachmentName to obtain the specific file name and reference the file name retrieved. Forexample: "U:\SomeFolder\\" & $Attachment

Index = The index of the items attachment.

Return Values:

0 for success or !!ERROR!!

4) Folder Specific Functions

PCEM_DeleteAllRecords(ExcludePrivate ; ExcludeDistList)

Description:

This command will delete all records from the specified open folder. This command should be used with extreme caution as it will remove ALL items permanently and it will not save a copy of items in the "Deleted Items" folder in Outlook. There is no "undo" for this command. The optional parameters to exclude "Private" will remove all records except for those marked as "private" in Outlook. The optional parameter to exclude "Distribution List" applies to the Contacts module only. Use this option if you wish exclude any "Distribution List" and its associated members. See the Outlook help file for more information on "Distribution Lists" in Outlook.

Parameters:

ExcludePrivate = excludes Private items (set to "1" to exclude)

ExcludeDistList = exclude Distribution List (set to "1" to exclude) Contacts module

Return Values:

0 for success or !!ERROR!!

PCEM_EmptyAllDeletedItems

Description:

This command will remove all items from the "Deleted Items" folder in Outlook. Keep in mind it will delete items you move there using the "DeleteRecord" command and it will also remove ALL objects that the user moved to this folder manually. This option should be used with caution.

Parameters:

Return Values:

0 for success or !!ERROR!!

PCEM_GetCurrentFolderName

Description:

Returns the name of the currently opened folder

Parameters:

Return Values:

The name of the folder or !!ERROR!!

PCEM_GetFirstFolder

Description:

Returns the full path of the first folder identified by Outlook in the current root folder.

Parameters:

Return Values:

A path to a folder or !!ERROR!!

PCEM_GetFirstRecord

Description:

Opens the first record in the currently opened folder.

Parameters:

Return Values:

The Outlook EntryID or !!ERROR!!

PCEM_GetDefaultFolderName(ItemType)

Description:

Returns the default folder name for the indicated item type.

Parameters:

ItemType = the type of items Valid values are:

    Deleted Items = The deleted items folder
    Outbox = folder containing items queued to be sent
    Sent Mail = items that have been sent
    Inbox = folder where new incoming items are placed
    Calendar = default folder for new calendar/appt items
    Contacts = default folder for new Contacts
    Journal = default folder for new Journal items
    Notes = for Notes
    Tasks = for Tasks
    Drafts = draft items
    All Public Folders = Public Folders
    Junk = Junk Mail
    Root = Default root folder

Return Values:

The name of the desired folder or !!ERROR!!

PCEM_GetFolderID

Description:

This is used to get the OutlookID of the currently opened Folder.

Parameters:

Return Values:

An OutlookID or !!ERROR!!

PCEM_GetFolderType

Description:

Returns a string representation of the default item type for the currently opened folder.

Parameters:

Return Values:

Mail, Contact, Task, Calendar, Note, Post, Journal or !!ERROR!!

PCEM_GetNextFolder

Description:

Returns the full path of the next folder identified by outlook in the current root folder. A call to GetFirstFolder must be made before calling this function for the first time.

Parameters:

Return Values:

A path to a folder, "End" if there are no more folders or !!ERROR!!

PCEM_GetNextRecord

Description:

Opens the next record in the currently opened folder. Requires a single call to GetFirstRecord in order to build the list of available records in the currently opened folder

Parameters:

Return Values:

"End'" if no more records or the OutlookID of the next record or !!ERROR!!

PCEM_GetRecordCount

Description:

Returns the number of available records in the currently opened folder. This function returns a text value. In order to perform numeric comparisons the result should be wrapped with the GetAsNumber function. [e.g. GetAsNumber (PCEM_GetRecordCount) > 100].

Parameters:

Return Values:

The number of records or !!ERROR!!

PCEM_NewFolder(Name ; Type)

Description:

Creates a new folder either in the root folder or the currently opened folder depending on the value of the Name parameter. Prefix the Name string with the '/' character to create in the root folder. The Type parameter is used to specify the default type for the items in the new folder.

Parameters:

Name = The name of the folder to create

Type = The default type for the folder - either "Contact," "Mail," "Calendar," "Tasks," "Note," "Journal" or "Post". Post and Mail are synonymous- the resulting folder will hold both types.

Return Values:

0 or !!ERROR!!

PCEM_OpenFolder(FolderPath)

Description

Opens the specified Folder for editing.

Parameters

FolderPath = The path to the folder from the current root folder.

Return Values:

The Outlook EntryID for the Folder or !!ERROR!!

Notes:

If path starts with "/" then it will search for the folder from the root folder.

If path does not start with "/", it will find the folder from the current folder.

Any call to "NewRecord" will create a record of the same type as the default type for the folder.

5) Root and Shared Folder Functions

PCEM_GetFirstRootFolder

Description:

Returns the Name of the first Mailbox/folder encountered by Outlook at the NameSpace level.

Parameters:

Return Values:

The name of the first root folder or !!ERROR!!

PCEM_GetNextRootFolder

Description:

Returns the Name of the next Mailbox/folder encountered by Outlook at the NameSpace level. Requires a call to GetFirstRootFolder before the first call to GetNextRootFolder.

Parameters:

None

Return Values:

The Folder Name, End or !!ERROR!!

PCEM_OpenRootFolder(RootFolderName)

Description:

Opens the specified mailbox or root folder. Please note that in Outlook 2003 and Outlook 2007, the root mailbox folder name is identified by your name using the format "Mailbox - Your Name." Outlook 2010 now uses your email address as the root folder name. Please account for this change in your FileMaker solution as needed if your users are using various Outlook versions.

Parameters:

RootFolderName = The Name of the root folder (Mailbox) to open

Example: ("Public Folders" , "Mailbox – User" or "support@productivecomputing.com")

Return Values:

0 for success or !!ERROR!!

PCEM_OpenSharedFolder (ContactName; Folder)

Description:

The function allows access to a root folder that is being shared by another user in the Exchange organization. Access permissions to the folder are created by the owner of the folder and the plug-in has the same access permissions as the current user.

This is an Exchange function only. Shared SUB folders are not currently available.

Parameters:

ContactName = This is the name of the user sharing the folder. Normally this is the name of the Exchange user as displayed in the Global Address List.

Folder = An integer representing the folder to be opened. Below is a list of the folders that are available and their corresponding integer values.

Inbox = 6
Calendar = 9
Contacts = 10
Journal = 11
Notes = 12
Tasks = 13
Drafts = 16
Conflicts = 19
SyncIssues = 20
LocalFailures = 21
ServerFailures = 22
Junk = 23

Return Values:

0 for success or !!ERROR!!

6) Recurrence Pattern Functions

PCEM_RPClearRecurrencePattern

Description:

Removes the recurrence settings for the currently opened appointment or task. Returns the item to a single occurrence item.

Parameters:

Return Values:

0 for success or or !!ERROR!!

PCEM_RPCloseOccurrenceItem(optSaveChanges)

Description:

Closes the currently opened occurrence of the recurrence pattern.

Parameters:

optSaveChanges - true or false

Return Values:

0 for success or !!ERROR!!

PCEM_RPCloseRecurrencePattern(optSaveChanges)

Description:

Closes the currently opened recurrence pattern.

Parameters:

optSaveChanges - true or false

Return Values:

0 for success or !!ERROR!!

PCEM_RPDeleteOccurrence(Date)

Description:

Deletes the occurrence for the specified day.

Parameters:

Date - date of occurrence to delete

Return Values:

0 for success or !!ERROR!!

PCEM_RPGetExceptionCount

Description:

Returns the number of exceptions in the currently opened Recurrence Pattern.

Parameters:

Return Values:

The number of exceptions or !!ERROR!!

PCEM_RPOpenOccurrence(Date)

Description:

Opens an occurrence of an Item in a recurrence pattern. The opened item is either a calendar item or a task item depending on which type of item to which the recurrence pattern belongs. The occurrence may be read, written or removed.

Parameters:

Date – the date on which the desired occurrence recurs.

Return Values:

0 for success or !!ERROR!!

PCEM_RPOpenRecurrencePattern

Description:

Opens the recurrence pattern for the currently opened appointment or task.

Parameters:

Return Values:

0 for success or !!ERROR!!

PCEM_RPOpenFirstException

Description:

Opens the first exception in the recurrence pattern's group of exceptions. The exception is available for read, write or deletion. It is of the same class as the original item type. If there are no exceptions in the currently opened recurrence pattern the function returns End.

Parameters:

Return Values:

End, 0 for success or !!ERROR!!

PCEM_RPOpenNextException

Description:

Opens the next exception in the recurrence pattern's group of exceptions. The exception is available for read, write or deletion. It is of the same class as the original item type. If there are no more exceptions in the currently opened recurrence pattern the function returns End.

Parameters:

Return Values:

End, 0 for success or !!ERROR!!

PCEM_RPOpenExceptionAt(Index)

Description:

Opens the specified exception in the recurrence pattern's group of exceptions. The exception is available for read, write or deletion. It is of the same class as the original item type.

Parameters:

Index – the index of the exception to be opened

Return Values:

0 for success or !!ERROR!!

III. Available Outlook Fields

Outlook has various records or item types such as Contact Items, Task Items, Mail Item etc. Each item contains various fields. These fields have values which can either be read or set. In order to read or "get" an Outlook field value you must call the PCEM_GetFieldData function and specify the field you would like to get. In order to write or "set" an Outlook field value you must call the PCEM_SetFieldData function and specify the field you would like to set.

All available fields are listed below and ARE CASE SENSITIVE. The fields are separated by item type and indicate if you can "get" or "set" the fields. If there is a missing field that you would like to access, then please email a request to support@productivecomputing.com

Available Contact Fields

Contact Items represent a contact in a Contacts folder.

Method	Field Name	Acceptable Values/Notes
Get/Set	Account	
Get/Set	Anniversary	Date
Get/Set	Assistant's Name	
Get/Set	Assistant's Phone	
Get/Set	Billing Information	
Get/Set	Birthday	Date
Get/Set	Body	This is the "Notes" field in Outlook
Get	Business Address	Get entire address as a block
Get/Set	Business Address City	
Get/Set	Business Address Country	
Get/Set	Business Address PO Box	
Get/Set	Business Address Postal Code	
Get/Set	Business Address State	
Get/Set	Business Address Street	
Get/Set	Business Fax	
Get/Set	Business Home Page	
Get/Set	Business Phone	
Get/Set	Business Phone 2	
Get/Set	Callback	
Get/Set	Car Phone	
Get/Set	Categories	
Get/Set	Children	
Get/Set	Company	
Get/Set	Company Main Phone	
Get/Set	Computer Network Name	
Get/Set	Customer ID	
Get	Created	
Get/Set	Department	

Method	Field Name	Acceptable Values/Notes

Get/Set	E-mail	
Get/Set	E-mail 2	
Get/Set	E-mail 3	
Get/Set	E-mail Address Type	
Get/Set	E-mail 2 Address Type	
Get/Set	E-mail 3 Address Type	
Get/Set	E-mail Display As	
Get/Set	E-mail 2 Display As	
Get/Set	E-mail 3 Display As	
Get/Set	File As	
Get/Set	First Name	
Get/Set	Form Name	Same as Message Class field. Possible ramifications if setting this field.
Get/Set	FTP Site	
Get/Set	Full Name	
Get/Set	Gender	Female, Male, Unspecified
Get/Set	Government ID Number	
Get/Set	Hobbies	
Get	Home Address	Get entire address as a block
Get/Set	Home Address City	
Get/Set	Home Address Country	
Get/Set	Home Address PO Box	
Get/Set	Home Address Postal Code	
Get/Set	Home Address State	
Get/Set	Home Address Street	
Get/Set	Home Fax	
Get/Set	Home Phone	
Get/Set	Home Phone 2	
Get	ID	OutlookID
Get/Set	IM Address	
Get/Set	Internet Free/Busy Address	
Get/Set	ISDN	
Get/Set	Job Title	
Get/Set	Journal	True or False
Get/Set	Language	
Get/Set	Last Name	
Get/Set	Location	
Get	Mailing Address	Get entire address as a block
Get/Set	Mailing Address City	
Get/Set	Mailing Address Country	
Get/Set	Mailing Address PO Box	
Get/Set	Mailing Address Postal Code	
Get/Set	Mailing Address State	
Get/Set	Mailing Address Street	
Get/Set	Manager's Name	
Get/Set	Message Class	Same as Form Name. Possible ramifications if setting this field.

Method	Field Name	Acceptable Values/Notes
Get/Set	Middle Name	
Get/Set	Mileage	
Get/Set	Mobile Phone	
Get	Modify	Date Modified
Get/Set	Nickname	
Get/Set	Office Location	
Get/Set	Organizational ID Number	
Get	Other Address	Get entire address as a block
Get/Set	Other Address City	
Get/Set	Other Address Country	
Get/Set	Other Address PO Box	
Get/Set	Other Address Postal Code	
Get/Set	Other Address State	
Get/Set	Other Address Street	
Get/Set	Other Fax	
Get/Set	Other Phone	
Get/Set	Pager	
Get/Set	Personal Home Page	
Get/Set	Primary Phone	
Get/Set	Profession	
Get/Set	Radio Phone	
Get/Set	Referred By	
Get/Set	Sensitivity	Confidential, Normal, Personal, Private
Get	Size	
Get/Set	Spouse	
Get/Set	Subject	
Get/Set	Suffix	
Get/Set	Telex	
Get/Set	Title	
Get/Set	TTY/TDD Phone	
Get/Set	User Field 1	
Get/Set	User Field 2	
Get/Set	User Field 3	
Get/Set	User Field 4	
Get/Set	Web Page	

**TIP: Please note that when adding any phones field, the format of the string is as follows:

['+' & country code & " "] & ["(" & area/city code & ") "] & number & [" x" & extension]

 - the country is optional, but if included it is preceded by a '+' character and followed by a space
 - the city/area code is optional, but if included it is enclosed by parenthesis and followed by a space
 - the actual number is required
 - if an extension is included it is prefixed with the 'x' character**

Available Mail Fields

Mail items represent a mail message in an Inbox folder.

Method	Field Name	Acceptable Values/Notes
Get/Set	Auto Forwarded	True or False
Get/Set	Bcc	
Get/Set	Billing Information	
Get/Set	Body	
Get/Set	Body Format	HTML, Plain, Rich Text, Unspecified
Get/Set	Categories	
Get/Set	Cc	
Get	Conversation Index	
Get	Conversation Topic	
Get	Created	Date/Time
Get/Set	Defer Until	Date/Time
Get/Set	Expires	Date/Time
Get/Set	Flag Icon	Available only in Outlook 2003 Text field with the following possible values: Clear, Red, Blue, Yellow, Green, Orange, Purple For example: PCEM_SetFieldData("Flag Icon" ; "Clear")
Get/Set	Flag Status	Complete, Marked, None
Get/Set	Form Name	Same as Message Class field. Possible ramifications if setting this field.
Get/Set	From	"Set" is only available using the Exchange Manipulator.
Get	Header	
Get/Set	HTML Body	
Get	In Folder	Name of Folder that record is in.
Get/Set	Importance	High, Low, Normal or H, L, N
Get	Is Marked As Task*	True or False (available only in Outlook 2007)
Set	Mark As Task*	Acceptable values are Clear, None, Today, Tomorrow, This Week, or Next Week. (available only in Outlook 2007 or later)
Get/Set	Mileage	
Get/Set	Message Class	Same as Form Name. Possible ramifications if setting this field.
Get	Modify	Date/Time
Get/Set	Originator Delivery Request	True or False
Get/Set	Receipt Requested	True or False
Get	Received Time	Date/Time
Get/Set	Remote Status	Marked For Copy, Marked For Delete, Marked For Download, None, Unmarked
Get/Set	RTF Body	
Get/Set	Sensitivity	Confidential, Normal, Personal, Private
Get	Sent On	Date/Time
Set	Send On Behalf Of	
Get	Size	
Get/Set	Subject	

Method	Field Name	Acceptable Values/Notes
Get/Set	Task Completed Date*	Please see "Mark As Task" field
Get/Set	Task Due Date*	Please see "Mark As Task" field
Get/Set	Task Start Date*	Please see "Mark As Task" field
Get/Set	Task Subject*	Please see "Mark As Task" field
Get/Set	To Do Task Ordiinal*	Please see "Mark As Task" field
Get/Set	To	
Get	Unread	True or False
Get/Set	Voting Response	
Get	ID	OutlookID

*When setting the "Is Marked As Task" field, then the the related "Is Marked As Task ," "Task Subject," "Task Start Date," "Task Due Date," "Task Completed Date," and "To Do Task Ordinal" fields will be adjusted accordingly as explained below.

If value is set to "Clear," then the "Is Marked As Task" field is set to False.

If value is set to "None," then the "Is Marked As Task" is set to True. "Task Subject" is set to the value of the Subject property for the Outlook item. "Task Start Date," "Task Due Date," and "Task Completed Date" are set NULL ("None"). "To Do Task Ordinal" is set to the current date.

If set to "Today," then the "Is Marked As Task" is set to True. "Task Subject" is set to the value of the Subject property for the Outlook item. "Task Start Date" and "Task Due Date" are set to the current date. "Task Completed Date" is set NULL ("None"). "To Do Task Ordinal" is set to the current date.

If value is set to "Tomorrow," then the "Is Marked As Task" is set to True. "Task Subject" is set to the value of the Subject property for the Outlook item. "Task Start Date" and "Task Due Date" are set to the one day after current date. "Task Completed Date" is set NULL ("None"). "To Do Task Ordinal" is set to the current date tomorrow.

If value is set to "This Week," then the "Is Marked As Task" is set to True. "Task Subject" is set to the value of the Subject property for the Outlook item. "Task Start Date" is set to a date two working days ahead of the current date. If that value would exceed the value of Task Due Date, then Task Start Date is set to the value of Task Due Date. "Task Due Date" is set to the last working day of the current week. "Task Completed Date" is set NULL ("None"). "To Do Task Ordinal" is set to the current date.

If value is set to "Next Week," then the "Is Marked As Task" is set to True. "Task Subject" is set to the value of the Subject property for the Outlook item. "Task Start Date" the first working day of next week. "Task Due Date" is set to the last working day of next week. "Task Completed Date" is set NULL ("None"). "To Do Task Ordinal" is set to the current date tomorrow.

Available Calendar/Appointment Fields

Appointment items represent a meeting, one-time appointment, recurring appointment or meeting.

Method	Field Name	Acceptable Values/Notes
Get/Set	All Day Event*	True or False (*see notes below)
Get/Set	Billing Information	
Get/Set	Body	This is the "note" field in Outlook
Get/Set	Busy Status*	Busy, Free, OutOfOffice or Tentative (*see notes below)
Get/Set	Categories	
Get/Set	Contacts	
Get	Created	Date/Time
Get/Set	Duration	Time in Minutes
Get/Set	End*	Date/Time - place "End" field AFTER "Start" field in script steps (*see notes below)
Get/Set	Form Name	Same as Message Class field. Possible ramifications if setting this field.
Get	ID	OutlookID
Get/Set	Importance	High, Normal, or Low (H, N, or L)
Get	IsRecurring	True or False
Get/Set	Label	Available only in Outlook 2003 Number field with integers representing the labels. Possible values are: 0= None, 1 = Important 2 = Business, 3 = Personal 4 = Vacation, 5 = Must Attend 6 = Travel Required, 7 = Needs Preparation 8 = Birthday, 9 = Anniversary 10 = Phone Call
Get/Set	Location	
Get/Set	MeetingResponse	Accepted, Declined, Tentative
Get/Set	Meeting Status	Meeting, Canceled, Received, NonMeeting
Get/Set	Message Class	Same as Form Name. Possible ramifications if setting this field.
Get/Set	Mileage	
Get	Modify	Date/Time
Get/Set	Net Meeting Auto Start	True or False
Get/Set	Net Meeting Office Document	Path to file
Get	Net Meeting Organizer Alias	
Get	Net Meeting Server	
Get	Net Meeting Type	
Get	Net Show URL	
Get/Set	Optional Attendees	Use semi colon between multiple attendees. Option to auto invite meeting attendees if using MS Exchange by calling PCEM_SaveRecord ("Appointment").
Get	Organizer	
Get /Get/ SetSet	Remind Beforehand	in Minutes
Get/Set	Reminder	True or False
Get/Set	Reminder Override Default	True or False
Get/Set	Reminder Sound	True or False
Get/Set	Reminder Sound File	Path to sound file

Method	Field Name	Acceptable Values/Notes
Get/Set	Required Attendees	Use semi colon between multiple attendees. Option to auto invite meeting attendees if using MS Exchange by calling PCEM_SaveRecord ("Appointment").
Get/Set	Resources	
Get/Set	Response Requested	True of False
Get/Set	Sensitivity	Confidential, Normal, Personal, Private
Get	Size	
Get/Set	Start*	Date/Time - place "Start" field BEFORE "End" field in script steps (*see notes below)
Get/Set	Subject	

*When setting the "All Day Event," "Busy Status," "End," or "Start" fields, then the fields will be adjusted accordingly to Outlook's rules. The value of one field can override and change the value of another field in Outlook regardless of what has been scripted in FileMaker. Please read examples below.

Example 1: In FileMaker if you set the "All Day Event" field to true BEFORE setting the "Start" and "End" fields then the event will NOT be marked as an all day event, but rather will have a start and end time as the "Start" and "End" fields were set AFTER the "All Day Event" field. The reverse is also true. If you set the "All Day Event" field to True AFTER you set the "Start" and "End" fields, then the event will be marked as an all day event without any specific times as the event is now an all day event. Outlook does not allow times to be applied to an all day event.

Example 2: In Outlook if the event status is marked as "Busy" and you then select the "All Day Event" check box, the event status will automatically be updated to "Free" even though you did not manually change the event status. This is because so many all day events are holidays, birthdays and anniversaries that don't affect your availability, so Outlook automatically sets the status of an all day event to Free. Please open up Outlook by itself too see this behavior for yourself.

Since our plug-in adheres to rules and functionality of Outlook, please be aware of the order you set these fields in your FileMaker script steps. We recommend setting these fields in the order shown below, however the field order of the script steps should be carefully designed and tested by a developer according to your solution and use.

- PCEM_SetFieldData("Start" ; Calendar:: Start)
- PCEM_SetFieldData("End" ; Calendar:: End)
- PCEM_SetFieldData("All Day Event" ; Calendar:: All Day Event)
- PCEM_SetFieldData("Busy Status" ; Calendar:: Busy Status)

Available Task Fields

Represents a task (an assigned, delegated, or self-imposed task to be performed within a specified time frame) in a Tasks folder.

Method	Field Name	Acceptable Values/Notes
Get	% Complete*	Number (no decimals)
Get/Set	Actual Work*	Time in Minutes
Get/Set	Billing Information	
Get/Set	Body	This is the "note" field in Outlook.
Get/Set	Categories	
Get/Set	Company	
Get/Set	Complete*	True or False
Get/Set	Contacts	
Get	Created	Date/Time
Get/Set	Date Completed*	Date
Get/Set	Due Date	Date
Get/Set	Form Name	Same as Message Class field. Possible ramifications if setting this field.
Get/Set	Importance	High, Normal or Low (H, N or L) - Outlook refers to this field as the "Priority" field.
Get/Set	Message Class	Same as Form Name. Possible ramifications if setting this field.
Get/Set	Mileage	
Get	Modify	Date/Time
Get/Set	Owner	
Get/Set	Reminder	True or False
Get/Set	Reminder Override Default	True or False
Get/Set	Reminder Sound	True or False
Get/Set	Reminder Sound File	Name of sound to play
Get/Set	Reminder Time	Date/Time or "none"
Get/Set	Role	
Get/Set	Schedule + Priority	
Get/Set	Sensitivity	Confidential, Normal, Personal, Private
Get	Size	
Get/Set	Start Date	Date
Get/Set	Status*	Complete, Deferred, In Progress, Not Started, Waiting
Get/Set	Subject	
Get/Set	Team Task	True or False
Get/Set	Total Work*	Time in Minutes
Get	ID	OutlookID

*The Outlook fields "% Complete," "Date Complete," "Complete," "Status," "Actual Work," and "Total Work" fields can recalculate their results based off of one another. If using these fields, then we recommend exploring your Outlook version as a guide to understand how these fields relate to each other. For example:
- If "Status" field is set to Complete, then the "% Complete" field will be set to 100.
- If "%Complete" is set to 0, then the "Status" field will be set to NotStarted.
- If you want to update the "% Complete" field independently of the "Actual Work" and/or "Total Work" fields, then set the "% Complete" field last or after these fields.*

Available Note Fields

Note Items represents a note in a Notes folder.

Method	Field Name	Acceptable Values/Notes
Get/Set	Body	This is the "notes" field in Outlook.
Get/Set	Categories	
Get/Set	Color	Integer representing the colors. Possible values are: 0 for Blue, 1 for Green, 2 for Pink, 3 for Yelllow, 4 for White
Get	Created	Date/Time
Get	Form Name	Same as Message Class field.
Get/Set	Message Class	Same as Form Name field. Possible ramifications if setting this field.
Get	Modify	Date/Time
Get	Subject	
Get	ID	OutlookID

Available Distribution List Fields

Distribution List Items represent a distribution list in a Contacts folder. A distribution list can contain multiple recipients and is used to send messages to everyone in the distribution list

Method	Field Name	Acceptable Values/Notes
Get/Set	Body	This is the "Notes" field in Outlook
Get/Set	Categories	
Get	Conversation Index	
Get	Conversation Topic	
Get	Created	Date/Time
Get/Set	DL Name	Name of distribution list
Get/Set	Importance	High, Normal or Low (H, N or L)
Get/Set	Marked for download	
Get	Member Count	Total number of members in distribution list
Get/Set	Message Class	Possible ramifications if setting this field
Get	Modify	Date/Time
Get	One Off Members	"Name,E-mail Address,Type" are returned. For example: Melinda DePalma,melinda@pci.com,SMTP
Set	One Off Members	Use the PCEM_AddOneOffMembers function to set this field
Get/Set	Sensitivity	0 = Normal, 1 = Personal, 2 = Private, 3 = Confidential
Get	Size	

Available Post Fields

Post Items represents a post in a public folder that others may browse.
Unlike a MailItem object, a PostItem object is not sent to a recipient. You use the Post method, which is analogous to the Send method for the MailItem object, to save the PostItem to the target public folder instead of mailing it.

Use the CreateItem or CreateItemFromTemplate method to create a PostItem object that represents a new post.

Use Items(index), where index is the index number of a post or a value used to match the default property of a post, to return a single PostItem object from a public folder.

Method	Field Name	Acceptable Values/Notes
Get/Set	Billing Information	
Get/Set	Body	
Get/Set	Body Format	HTML, Plain, Rich Text
Get/Set	Categories	
Get/Set	Companies	
Get	Conversation Index	
Get	Conversation Topic	
Get	Creation Time	Date/Time
Get	EntryID	
Get/Set	Expires	Date/Time
Get	Form Name	Same as Message Class field.
Get/Set	HTML Body	
Get/Set	Importance	High, Normal or Low (H, N or L)
Get	Is Conflict	True or False
Get	Last Modification Time	Date/Time
Get/Set	Message Class	Same as Form Name. Possible ramifications if setting this field.
Get/Set	Mileage	
Get/Set	No Aging	True or False
Get	Received Time	
Get	Saved	Date/Time
Get	Sender Name	
Get/Set	Sensitivity	Confidential, Normal, Personal, Private
Get	Sent On	
Get	Size	
Get/Set	Subject	
Get/Set	Unread	

Available Journal Fields

Journal Items represents a journal entry in a Journal folder.

Method	Field Name	Acceptable Values/Notes
Get/Set	Billing Information	
Get/Set	Body	
Get/Set	Categories	
Get/Set	Companies	
Get/Set	Contact Names	Contacts
Get	Conversation Index	
Get	Conversation Topic	
Get	Created	Date/Time
Get/Set	Doc Posted	True or False
Get/Set	Doc Printed	True or False
Get/Set	Doc Routed	True or False
Get/Set	Doc Saved	True or False
Get/Set	Duration	Time in Minutes
Get/Set	End	Date/Time
Get	Form Name	Same as Message Class field.
Get/Set	Importance	High, Normal or Low (H, N, or L)
Get	Is Conflict	True or False
Get/Set	Message Class	Same as Form Name. Possible ramifications if setting this field.
Get/Set	Mileage	
Get	Modify	Date/Time
Get/Set	No Aging	True or False
Get	Saved	True or False
Get/Set	Sensitivity	Confidential, Normal, Personal, Private
Get	Size	
Get/Set	Start	Date/Time
Get/Set	Subject	
Get/Set	Type	Contains display name of journalizing application
Get/Set	Unread	True or False

Available Report Fields

Report items represent a mail delivery report in an Inbox folder. The ReportItem object is similar to a Mail Item object and it contains a report (usually the non-delivery report) or error message from the mail transport system. You cannot create report field items as they are automatically created when any report or error in general is received from the mail transport system

Method	Field Name	Acceptable Values/Notes
Get	Billing Information	
Get	Body	
Get	Categories	
Get	Class	
Get	Companies	
Get	Conversation Index	
Get	Conversation Topic	
Get	Created	
Get	Form Name	Same as Message Class field.
Get	From	
Get	From1	
Get	From2	
Get	ID	OutlookID
Get	Importance	High, Low, None
Get	In Folder	
Get	Is Conflict	
Get	Message Class	Same as Form Name.
Get	Mileage	
Get	Modify	Date/Time
Get	Sensitivity	Confidential, Normal, Personal, Private
Get	Size	
Get	Subject	
Get	Unread	

Available Task Request Fields

Task Request Items represents a change to the recipient's Tasks list initiated by another party or as a result of a group tasking.

Unlike other Microsoft Outlook objects, you cannot create this object. When the sender applies the Assign and Send methods to a TaskItem object to assign (delegate) the associated task to another user, the TaskRequestItem object is created when the item is received in the recipient's Inbox.

Method	Field Name	Acceptable Values/Notes
Get	Billing Information	
Get	Body	
Get	Categories	
Get	Class	
Get	Companies	
Get	Conversation Index	
Get	Conversation Topic	
Get	Created	
Get	Form Name	Same as Message Class field.
Get	From	
Get	Is Conflict	
Get	In Folder	
Get	Importance	High, Low, Normal
Get	Message Class	Same as Form Name.
Get	Mileage	
Get	Modify	Date/Time
Get	Sensitivity	Confidential, Normal, Personal, Private
Get	Size	
Get	Subject	
Get	To	
Get	Unread	
Get	ID	OutlookID
Get	From1	
Get	From2	

Available Meeting Fields

Meeting Items represent a change to the recipient's Calendar folder initiated by another party or as a result of a group action. You cannot create Meeting Items as these items are created automatically when you set the MeetingStatus property of an AppointmentItem object and send it to one or more users.

Method	Field Name	Acceptable Values/Notes
Get	Associated Appointment	
Get	Billing Information	
Get	Body	
Get	Categories	
Get	Class	
Get	Companies	
Get	Conversation Index	
Get	Conversation Topic	
Get	Created	
Get	Due Date	
Get	Expiry Time	
Get	Flag Icon	To Be Determined
Get	Flag Request	
Get	Flag Status	Complete, Marked, None
Get	Form Name	Same as Message Class field.
Get	From	
Get	From Name	
Get	Importance	
Get	ID	OutlookID
Get	Is Conflict	
Get	Is Sent	
Get	Is Submitted	
Get	Mileage	
Get	Message Class	Same as Form Name.
Get	Modify	Date/Time
Get	Received	
Get	Reminder Set	
Get	Reminder Time	
Get	Sensitivity	
Get	Sent On	
Get	Size	
Get	Subject	

Available Meeting Request Fields

Method	Field Name	Acceptable Values/Notes
Get	Billing Information	
Get	Body	
Get	Categories	
Get	Companies	
Get	Conversation Index	
Get	Conversation Topic	
Get	Due Date	
Get	Expiry Time	
Get	Flag Icon	To Be Determined
Get	Flag Request	
Get	Flag Status	Complete, Marked, None
Get	Form Name	Same as Message Class field.
Get	Importance	High, Low, Normal
Get	Message Class	Same as Form Name.
Get	Mileage	
Get	Sensitivity	Confidential, Normal, Personal, Private
Get	Subject	

Available Recurrence Pattern Fields

Recurrence pattern represents the pattern of incidence of recurring appointments and tasks for the associated calendar and task item.

Method	Field Name	Acceptable Values/Notes
Get/Set	Day Of Month	The Nth day of the month
Get/Set	Day Of Week Mask	For example: Every Monday, Wednesday and Friday
Get/Set	Duration	Indicating the duration (in minutes). Only valid for appointments.
Get/Set	End Time	Only valid for appointments.
Get/Set	Instance	The Nth Tuesday
Get/Set	Interval	Every N days, N weeks, N Months etc.
Get/Set	Month of Year	For example: February
Get/Set	Occurrences	The number of occurrences of the recurrence pattern.
Get/Set	Pattern End Date	Indicates the end date for the recurrence pattern.
Get/Set	Pattern Start Date	Indicates the start date for the recurrence pattern.
Get/Set	Recurrence Type	Daily, Weekly, Monthly, Nth Month Yeary, Nth Year or Unknown
Get/Set	Regenerate	True to regenerate. Boolean. Used to control the regeneration of the task as each occurrence of a recurring task is completed.
Get/Set	Start Time	Only valid for appointments.
Get/Set	No End Date	True if the recurrence pattern has no end date. Boolean. Must coordinate with other fields when setting this field.

IV. Contact Us

Successful integration of a FileMaker plug-in requires the creation of integration scripts within your FileMaker solution. A working knowledge of FileMaker Pro, especially in the areas of scripting and calculations is necessary. If you need additional support for scripting, customization or setup (excluding registration) after reviewing the videos, documentation, FileMaker demo and sample scripts, then please contact us via the avenues listed below.

> Phone: 760-510-1200
> Email: support@productivecomputing.com
> Forum: www.productivecomputing.com/forum

Please note assisting you with implementing this plug-in (excluding registration) is billable at our standard hourly rate. We bill on a time and materials basis billing only for the time in minutes it takes to assist you. We will be happy to create your integration scripts for you and can provide you with a free estimate if you fill out a Request For Quote (RFQ) at www.productivecomputing.com/rfq. We are ready to assist and look forward to hearing from you!