



# Functions Guide

Revised August 29, 2017

950 Boardwalk, Suite 205, San Marcos, CA 92078 • (760) 510-1200 • [www.productivecomputing.com](http://www.productivecomputing.com)

© Copyright 2017 Productive Computing, Inc.

# Table of Contents

I. Introduction .....	4
I. Function Descriptions.....	5
1) Error and Registration Related Functions.....	5
PCCC_Register( ServerName ; ServerPort ; ServerPage ; LicenseID ) .....	5
PCCC_Version( type ) .....	6
PCCC_GetOperatingMode.....	6
PCCC_GetLastError( type ) .....	7
2) Global Gateway Related Functions.....	8
PCCC_SetProcessMode( Mode ) .....	8
PCCC_GetProcessMode .....	9
PCCC_GetResp( Field ).....	9
PCCC_GetGateway .....	10
PCCC_UpdateGatewayURL( Gateway ; URL ).....	10
PCCC_UseGateway( Gateway ) .....	11
3) Processing Functions.....	12
PCCC_AuthorizeCapture( GatewayID ; Amount ; CardNum ; ExpDate ; optMerchantID ; optTxnKey ; optUserName ; optPassword [ ; optParams ... ] ).....	12
PCCC_AuthorizeOnly( GatewayID ; Amount ; CardNum ; ExpDate ; optMerchantID ; optTxnKey ; optUserName ; optPassword [ ; optParams ... ] ).....	14
PCCC_CaptureOnly( GatewayID ; optMerchantID ; optTxnKey ; optTxnID ; optUserName ; optPassword [ ; optParams ... ] ).....	15
PCCC_Credit( GatewayID ; Amount ; CardNum ; optMerchantID ; optTxnKey ; optTxnID ; optUserName ; optPassword [ ; optParams ... ] ).....	17
PCCC_Void( GatewayID ; optMerchantID ; optTxnKey ; optTxnID ; optUserName ; optPassword [ ; optParams ... ] ).....	18
PCCC_RawPost( GatewayID ; optMerchantID ; optTxnKey ; optUserName ; optPassword [ ; optParams ... ] ).....	20
PCCC_PP_VoiceAuth( Partner ; Vendor ; User ; Password ; AuthCode ; Amount ; CardNum ; ExpDate; OptParam ).....	22
PCCC_PP_Inquire( Partner ; Vendor ; User ; Password ; OrigID; OptParam ).....	23
PCCC_CP_AuthorizeCapture( GatewayID ; Amount ; optTrack1 ; optTrack2 ; optMerchantID ; optTxnKey ; optUserName ; optPassword [ ; optParams ... ] ).....	24
PCCC_CP_AuthorizeOnly( GatewayID ; Amount ; optTrack1 ; optTrack2 ; optMerchantID ; optTxnKey ; optUserName ; optPassword [ ; optParams ... ] ).....	25

PCCC_CP_CaptureOnly( GatewayID ; optMerchantID ; optTxnKey ; optTxnID ; optUserName ; optPassword [ ; optParams ... ] ) .....	26
4) Deprecated Gateway-Specific Functions .....	27
PCCC_AN_AuthCapt( APILoginID ; TransactionKey ; Amount ; CardNum ; ExpDate; OptParam ).....	27
PCCC_AN_Authorize( APILoginID ; TransactionKey ; Amount ; CardNum ; ExpDate; OptParam ) .....	28
PCCC_AN_Capture( APILoginID ; TransactionKey ; TransID ; Amount; OptParam ) .....	29
PCCC_AN_CaptOnly( APILoginID ; TransactionKey ; AuthCode ; Amount ; CardNum ; ExpDate; OptParam ) .....	30
PCCC_AN_Credit( APILoginID ; TransactionKey ; TransID ; Amount ; CardNum; OptParam ) .....	31
PCCC_AN_UCredit( APILoginID ; TransactionKey ; Amount ; CardNum ; ExpDate; OptParam ).....	32
PCCC_AN_Void( APILoginID ; TransactionKey ; TransID; OptParam ) .....	33
PCCC_AN_RawPost( APILoginID ; TransactionKey; OptParam ) .....	34
PCCC_PP_Sale( Partner ; Vendor ; User ; Password ; Amount ; CardNum ; ExpDate; OptParam ).....	35
PCCC_PP_Authorize( Partner ; Vendor ; User ; Password ; Amount ; CardNum ; ExpDate; OptParam )	36
PCCC_PP_Capture( Partner ; Vendor ; User ; Password ; OrigID; OptParam ).....	37
PCCC_PP_Void( Partner ; Vendor ; User ; Password ; OrigID; OptParam ) .....	38
PCCC_PP_Credit( Partner ; Vendor ; User ; Password ; OrigID; OptParam ).....	39
PCCC_PP_UCredit( Partner ; Vendor ; User ; Password ; Amount ; CardNum ; ExpDate; OptParam )....	40
PCCC_PP_RawPost( Partner ; Vendor ; User ; Password; OptParam ).....	41
III. Available Keys for PCCC_GetResp .....	42
Authorize.Net, PayItForward, and Authorize.Net Emulator Keys.....	42
PayPal Keys.....	45
IV. Links for using “OptParam” .....	47
V. Contact Us.....	48

## I. Introduction

### **Description**

FM Credit Card is a credit card processing plug-in for FileMaker. With this plug-in you are able to automatically process secure SSL encrypted credit card payments through your gateway all from within FileMaker. The plug-in works with an array of gateways to meet all your needs and provide you with more options. These operations are accomplished using FileMaker function calls from within FileMaker calculations. These calculations are generally determined from within FileMaker "SetField," "SetVariable" or "If" script steps. For a list of the basic integration steps, please see the accompanying Developer's Guide.

### **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) Review our FileMaker demo and video tutorials

Demo and video tutorials: <http://www.productivecomputing.com/credit-card-processing>

- 4) Familiarize yourself with your credit card gateway

### **Error Handling:**

Any of the plug-in functions may encounter an error during processing. When an error occurs during processing, immediately call the PCCC\_GetLastError function in order to obtain a full description of the error or error number. This function returns the error message or error number associated with the last error in order to troubleshoot script or logic failures. Please see PCCC\_GetLastError function description in the Functions Guide and the "Handling Errors" section in the Developer's Guide for further clarification on how to properly trap for errors.

## I. Function Descriptions

This section describes the functions that are available with the FM Credit Card plug-in.

### 1) Error and Registration Related Functions

#### **PCCC Register( ServerName ; ServerPort ; ServerPage ; LicenseID )**

**Purpose:**

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

**Dependencies:**

None

**Parameters:**

Parameter Name	Purpose	Values	Default Value
ServerName	The name of Productive Computing's registration server.	"licensing.productivecomputing.com"	
ServerPort	The port on Productive Computing's registration server to access the registration page.	"80"	
ServerPage	The page that the registration code resides on.	"/PCIReg/pcireg.php"	
LicenseID	The license ID provided with the purchase of a plug-in, otherwise the demo code.	"your license ID" or "DEMO-CC1"	

**Return Values:**

Returns 0 on success, otherwise an !!ERROR!! or error text. Anything other than a 0 is an error.

**Notes & Examples:**

Example: PCCC\_Register("licensing.productivecomputing.com" ; "80" ; "/PCIReg/pcireg.php" ; "DEMO-CC1")

## **PCCC\_Version( type )**

### **Purpose:**

Used to identify the plug-in version and/or plug-in name installed on a machine.

### **Dependencies:**

None

### **Parameters:**

<b>Parameter Name</b>	<b>Purpose</b>	<b>Values</b>	<b>Default Value</b>
type	Determines the version string to be returned. The "short" version string includes only the version number of the plug-in. Example: "1.0.1.0". The "long" includes the name of the plug-in and version number. Example: "FM Credit Card 1.0.1.0"	"long" or "short"	short

### **Return Values:**

A short or long version string, otherwise an !!ERROR!!

### **Notes & Examples:**

Example: PCCC\_Version( "long" )

If parameter is not specified, then the default value returned is short.

## **PCCC\_GetOperatingMode**

### **Purpose:**

Returns a string representing the current operating mode of the plug-in.

### **Dependencies:**

None

### **Parameters:**

None

### **Return Values:**

"LIVE", "DEMO", "EXPIRED", "UNREGISTERED", otherwise an !!ERROR!!

### **Notes & Examples:**

None

## **PCCC\_GetLastError( type )**

### **Purpose:**

This function retrieves the last error caused by the previously executed function.

### **Dependencies:**

Must have called any other function from the plug-in.

### **Parameters:**

<b>Parameter Name</b>	<b>Purpose</b>	<b>Values</b>	<b>Default Value</b>
type	Determines the format of the error returned.	"text" or "number"	"number"

### **Return Values:**

Error number or text depending on parameter, otherwise an !!ERROR!!

### **Notes & Examples:**

Example: PCCC\_GetLastError("Number")

## 2) Global Gateway Related Functions

### **PCCC SetProcessMode( Mode )**

**Purpose:**

This function sets the process mode of the plug-in. Payment gateways often provide a test account to make sure everything is working before going live with customers. Setting "test" mode will utilize the chosen gateway provider's test url. Setting "live" mode should be used when processing real credit card transactions.

**Dependencies:**

None

**Parameters:**

Parameter Name	Purpose	Values	Default Value
Mode	To set the operating mode of the gateway provider.	"test" or "live"	"test"

**Return Values:**

0 on success, otherwise an !!ERROR!!.

**Notes & Examples:**

Setting the process mode will immediately update all applicable gateways' URLs to their Test or Live equivalent. This URL can still be overridden by a later call to PCCC\_UpdateGatewayURL. This is a simple way to restore the default gateway URLs.



## **PCCC\_GetProcessMode**

**Purpose:**

This function gets the current process mode of the plug-in.

**Dependencies:**

None

**Parameters:**

None

**Return Values:**

"test", "live", otherwise an !!ERROR!!.

**Notes & Examples:**

None

## **PCCC\_GetResp( Field )**

**Purpose:**

This function can retrieve the last response from a gateway transaction. It is capable of returning both the raw response from the gateway and a specific parsed out value that corresponds to the provided field parameter.

**Dependencies:**

A previous call to one of the gateway functions.

**Parameters:**

<b>Parameter Name</b>	<b>Purpose</b>	<b>Values</b>	<b>Default Value</b>
Field	This is the key for a corresponding value in the gateway's response.	Dependent upon the provider. Please see "Available Keys" at the end of this document.	

**Return Values:**

The response from the gateway, otherwise an !!ERROR!!

**Notes & Examples:**

The use of GetResp is highly specific to the utilized gateway provider. Possible key values are provided in the documentation, and can also be directly referenced in the gateway provider's api documentation. Example: PCCC\_GetResp( "email" ) would return the customer email from an Authorize.Net response.

Example: PCCC\_GetResp would return the raw response from whatever the gateway provider last called.

## **PCCC GetGateway**

### **Purpose:**

This function returns the currently-used gateway's URL. This is useful for validating that the correct gateway is being utilized.

### **Dependencies:**

None

### **Return Values:**

The current gateway's URL, otherwise an !!ERROR!!

### **Notes & Examples:**

The URL format is different between operating systems. On macOS, the gateway URL requires the "http://" or "https://" (as applicable) protocol prior to the main URL part. On Windows, the protocol is not needed.

## **PCCC UpdateGatewayURL( Gateway ; URL )**

### **Purpose:**

This function can retrieve the last response from a gateway transaction. It is capable of returning both the raw response from the gateway and a specific parsed out value that corresponds to the provided field parameter.

### **Dependencies:**

A previous call to one of the gateway functions.

### **Parameters:**

<b>Parameter Name</b>	<b>Purpose</b>	<b>Values</b>	<b>Default Value</b>
Gateway	This is the name of the gateway to update.	Acceptable values are: "Authorize.Net", "PayPal", "PayItForward", or "Emulator"	
URL	This is the URL that will be set for the specified gateway.	Any valid URL text.	

### **Return Values:**

0 for success, otherwise an !!ERROR!!

### **Notes & Examples:**

The URL format is different between operating systems. On macOS, the gateway URL requires the "http://" or "https://" (as applicable) protocol prior to the main URL part. On Windows, the protocol is not needed.

For more information on which URL your merchant gateway requires, please refer to the gateway's documentation.

## **PCCC UseGateway( Gateway )**

### **Purpose:**

This function prepares the plug-in to use a specific gateway. Any calls to the gateway (Authorize and Capture, Authorize, Capture, Credit, Void, or Raw Post) will be sent to the gateway specified in PCCC\_UseGateway, using the associated gateway's URL.

### **Dependencies:**

A previous call to one of the gateway functions.

### **Parameters:**

<b>Parameter Name</b>	<b>Purpose</b>	<b>Values</b>	<b>Default Value</b>
Gateway	This is the name of the gateway to update.	Acceptable values are: "Authorize.Net", "PayPal", "PayItForward", or "Emulator"	

### **Return Values:**

0 for success, otherwise an !!ERROR!!

### **Notes & Examples:**

By default, the plug-in is initialized and prepared to communicate with Authorize.Net's Live gateway.

### 3) Processing Functions

#### **PCCC AuthorizeCapture( GatewayID ; Amount ; CardNum ; ExpDate ; optMerchantID ; optTxnKey ; optUserName ; optPassword [ ; optParams ... ] )**

**Purpose:**

This function does an authorization and capture for funds in one action. There are four minimum parameters and five optional parameters. The optParams parameter(s) can append any key-value pair that matches the format of the gateway currently being used. For more information on key-value pair format, please refer to the developer documentation of your chosen gateway.

Not all parameters are required for every gateway, see Notes & Examples below.

**Dependencies:**

Must have called PCCC\_UseGateway.

**Parameters:**

Parameter Name	Purpose	Values	Default Value
GatewayID	This is the payment gateway's primary indicator.	For Authorize.Net and its emulators, this is the "API Login ID". For PayPal, this is the "Partner".	
Amount	The desired amount of the transaction.	In the format of "Dollars.Cents"	
CardNum	The credit card number used in the transaction.	Some credit card number (no spaces)	
ExpDate	The expiration date of the supplied credit card.	In the format "MMYYYY"	
optMerchantID	The identifier of the merchant account.	PayPal only. Corresponds to the merchant's "Vendor" identifier.	
optTxnKey	The transaction key.	This is applicable only for Authorize.Net and its emulators.	
optUserName	The merchant account's username.	PayPal only.	
optPassword	The merchant account's password.	PayPal only.	
optParams	Append key value pairs.	This can be a single parameter of key-value pairs separated by the ";" character, or a series of additional parameters added. See example below.	

**Return Values:**

A transaction ID to be used for future reference, otherwise an !!ERROR!!

**Notes & Examples:**

Authorize.Net Example: PCCC\_AuthorizeCapture( "ANetLogin12345" ; "1.00" ; "4444333322221111" ; "122020" ; "" ; "1234567890ABCDEF" ; "" ; "" )

PayPal Example: PCCC\_AuthorizeCapture( "PayPal" ; "1.00" ; "4444333322221111" ; "122020" ; "TestVendor" ; "" ; "Tester" ; "123456" )

Authorize.Net Example using Key-Values in the optParam, single optParam value:

PCCC\_AuthorizeCapture( "ANetLogin12345" ; "1.00" ; "4444333322221111" ; "122020" ; "" ; "1234567890ABCDEF" ; "" ; "" ; "x\_email=test@test.com;x\_address=123 Fake St;x\_delim\_char=|" )

Authorize.Net Example using Key-Values in the optParam, multiple optParam values:

PCCC\_AuthorizeCapture( "ANetLogin12345" ; "1.00" ; "4444333322221111" ; "122020" ; "" ; "1234567890ABCDEF" ; "" ; "" ; "x\_email=test@test.com" ; "x\_address=123 Fake St" ; "x\_delim\_char=|" )

**PCCC AuthorizeOnly( GatewayID ; Amount ; CardNum ; ExpDate ; optMerchantID ; optTxnKey ; optUserName ; optPassword [ ; optParams ... ] )**

**Purpose:**

This function does an authorization for the specified amount of funds on a provided credit card. The funds are not captured until the PCCC\_CaptureOnly function is called with the returned transaction ID. There are four minimum parameters and five optional parameters. The optParams parameter(s) can append any key-value pair that matches the format of the gateway currently being used. For more information on key-value pair format, please refer to the developer documentation of your chosen gateway.

Not all parameters are required for every gateway, see Notes & Examples below.

**Dependencies:**

Must have called PCCC\_UseGateway.

**Parameters:**

Parameter Name	Purpose	Values	Default Value
GatewayID	This is the payment gateway's primary indicator.	For Authorize.Net and its emulators, this is the "API Login ID". For PayPal, this is the "Partner".	
Amount	The desired amount of the transaction.	In the format of "Dollars.Cents"	
CardNum	The credit card number used in the transaction.	some credit card number (no spaces)	
ExpDate	The expiration date of the supplied credit card.	In the format "MMYYYY"	
optMerchantID	The identifier of the merchant account.	PayPal only. Corresponds to the merchant's "Vendor" identifier.	
optTxnKey	The transaction key.	This is applicable only for Authorize.Net and its emulators.	
optUserName	The merchant account's username.	PayPal only.	
optPassword	The merchant account's password.	PayPal only.	
optParams	Append key value pairs	This can be a single parameter of key-value pairs separated by the ";" character, or a series of additional parameters added. See example below.	

**Return Values:**

A transaction ID to be used for future reference, otherwise an !!ERROR!!

**Notes & Examples:**

Authorize.Net Example: `PCCC_AuthorizeOnly( "ANetLogin12345" ; "1.00" ; "4444333322221111" ; "122020" ; "" ; "1234567890ABCDEF" ; "" ; "" )`

PayPal Example: `PCCC_AuthorizeOnly( "PayPal" ; "1.00" ; "4444333322221111" ; "122020" ; "TestVendor" ; "" ; "Tester" ; "123456" )`

Authorize.Net Example using Key-Values in the optParam, single optParam value:

`PCCC_AuthorizeOnly( "ANetLogin12345" ; "1.00" ; "4444333322221111" ; "122020" ; "" ; "1234567890ABCDEF" ; "" ; "" ; "x_email=test@test.com;x_address=123 Fake St;x_delim_char=|" )`

Authorize.Net Example using Key-Values in the optParam, multiple optParam values:

`PCCC_AuthorizeOnly( "ANetLogin12345" ; "1.00" ; "4444333322221111" ; "122020" ; "" ; "1234567890ABCDEF" ; "" ; "" ; "x_email=test@test.com" ; "x_address=123 Fake St" ; "x_delim_char=|" )`

**PCCC\_CaptureOnly( GatewayID ; optMerchantID ; optTxnKey ; optTxnID ; optUserName ; optPassword [ ; optParams ... ] )**

**Purpose:**

This function does an authorization for the specified amount of funds on a provided credit card. The funds are not captured until the PCCC\_CaptureOnly function is called with the returned transaction ID. There is one minimum parameter and six optional parameters. The optParams parameter(s) can append any key-value pair that matches the format of the gateway currently being used. For more information on key-value pair format, please refer to the developer documentation of your chosen gateway.

Not all parameters are required for every gateway, see Notes & Examples below.

**Dependencies:**

Must have called PCCC\_UseGateway.

**Parameters:**

Parameter Name	Purpose	Values	Default Value
GatewayID	This is the payment gateway's primary indicator.	For Authorize.Net and its emulators, this is the "API Login ID". For PayPal, this is the "Partner".	
optMerchantID	The identifier of the merchant account.	PayPal only. Corresponds to the merchant's "Vendor" identifier.	

optTxnKey	The transaction key.	This is applicable only for Authorize.Net and its emulators.	
optTxnID	The transaction ID of a previously authorized and uncaptured transaction	For Authorize.Net and its emulators, this is the "TransID" response key. For PayPal, this is the "ORIGID" response key.	
optUserName	The merchant account's username.	PayPal only.	
optPassword	The merchant account's password.	PayPal only.	
optParams	Append key value pairs	This can be a single parameter of key-value pairs separated by the ";" character, or a series of additional parameters added. See example below.	

**Return Values:**

A transaction ID to be used for future reference, otherwise an !!ERROR!!

**Notes & Examples:**

Authorize.Net Example: PCCC\_CaptureOnly( "ANetLogin12345" ; "" ; "1234567890ABCDEF" ; "123ABC" ; "" ; "" )

PayPal Example: PCCC\_CaptureOnly( "PayPal" ; "TestVendor" ; "" ; "123ABC" ; "Tester" ; "123456" )

Authorize.Net Example using Key-Values in the optParam, single optParam value:

PCCC\_CaptureOnly( "ANetLogin12345" ; "" ; "1234567890ABCDEF" ; "123ABC" ; "" ; "" ; "x\_email=test@test.com;x\_address=123 Fake St;x\_delim\_char=|" )

Authorize.Net Example using Key-Values in the optParam, multiple optParam values:

PCCC\_CaptureOnly( "ANetLogin12345" ; "" ; "1234567890ABCDEF" ; "123ABC" ; "" ; "" ; "x\_email=test@test.com" ; "x\_address=123 Fake St" ; "x\_delim\_char=|" )



**PCCC Credit( GatewayID ; Amount ; CardNum ; optMerchantID ; optTxnKey ; optTxnID ; optUserName ; optPassword [ ; optParams ... ] )**

**Purpose:**

This function can credit or refund a customer for a previous transaction performed on the gateway. There are three minimum parameters and six optional parameters. The optParams parameter(s) can append any key-value pair that matches the format of the gateway currently being used. For more information on key-value pair format, please refer to the developer documentation of your chosen gateway.

Not all parameters are required for every gateway, see Notes & Examples below.

**Dependencies:**

Must have called PCCC\_UseGateway.

**Parameters:**

Parameter Name	Purpose	Values	Default Value
GatewayID	This is the payment gateway's primary indicator.	For Authorize.Net and its emulators, this is the "API Login ID". For PayPal, this is the "Partner".	
Amount	The desired amount of the transaction.	In the format of "Dollars.Cents"	
CardNum	The credit card number used in the transaction.	some credit card number (no spaces)	
optMerchantID	The identifier of the merchant account.	PayPal only. Corresponds to the merchant's "Vendor" identifier.	
optTxnKey	The transaction key.	This is applicable only for Authorize.Net and its emulators.	
optTxnID	The transaction ID of a previously authorized and uncaptured transaction	For Authorize.Net and its emulators, this is the "TransID" response key. For PayPal, this is the "ORIGID" response key.	
optUserName	The merchant account's username.	PayPal only.	
optPassword	The merchant account's password.	PayPal only.	
optParams	Append key value pairs	This can be a single parameter of key-value pairs separated by the ";" character, or a series of additional parameters added. See example below.	

**Return Values:**

A transaction ID to be used for future reference, otherwise an !!ERROR!!

**Notes & Examples:**

Authorize.Net Example: `PCCC_Credit( "ANetLogin12345" ; "" ; "1234567890ABCDEF" ; "123ABC" ; "" ; "" )`

PayPal Example: `PCCC_Credit( "PayPal" ; "TestVendor" ; "" ; "123ABC" ; "Tester" ; "123456" )`

Authorize.Net Example using Key-Values in the optParam, single optParam value:

`PCCC_Credit( "ANetLogin12345" ; "" ; "1234567890ABCDEF" ; "123ABC" ; "" ; "" ;  
"x_email=test@test.com;x_address=123 Fake St;x_delim_char=|" )`

Authorize.Net Example using Key-Values in the optParam, multiple optParam values:

`PCCC_Credit( "ANetLogin12345" ; "" ; "1234567890ABCDEF" ; "123ABC" ; "" ; "" ; "x_email=test@test.com" ;  
"x_address=123 Fake St" ; "x_delim_char=|" )`

**PCCC Void( GatewayID ; optMerchantID ; optTxnKey ; optTxnID ; optUserName ;  
optPassword [ ; optParams ... ] )**

**Purpose:**

This function can be used to void a previous transaction that has not yet been cleared by the bank, which usually happens at the end of the day. It cannot void partial amounts, only the entire amount of the transaction. There are three minimum parameters and six optional parameters. The optParams parameter(s) can append any key-value pair that matches the format of the gateway currently being used. For more information on key-value pair format, please refer to the developer documentation of your chosen gateway.

Not all parameters are required for every gateway, see Notes & Examples below.

**Dependencies:**

Must have called PCCC\_UseGateway.

**Parameters:**

Parameter Name	Purpose	Values	Default Value
GatewayID	This is the payment gateway's primary indicator.	For Authorize.Net and its emulators, this is the "API Login ID". For PayPal, this is the "Partner".	
optMerchantID	The identifier of the merchant account.	PayPal only. Corresponds to the merchant's "Vendor" identifier.	
optTxnKey	The transaction key.	This is applicable only for Authorize.Net and its emulators.	

optTxnID	The transaction ID of a previously authorized and uncaptured transaction	For Authorize.Net and its emulators, this is the "TransID" response key. For PayPal, this is the "ORIGID" response key.	
optUserName	The merchant account's username.	PayPal only.	
optPassword	The merchant account's password.	PayPal only.	
optParams	Append key value pairs	This can be a single parameter of key-value pairs separated by the ";" character, or a series of additional parameters added. See example below.	

**Return Values:**

A transaction ID to be used for future reference, otherwise an !!ERROR!!

**Notes & Examples:**

Authorize.Net Example: `PCCC_Void( "ANetLogin12345" ; "" ; "1234567890ABCDEF" ; "123ABC" ; "" ; "" )`

PayPal Example: `PCCC_Void( "PayPal" ; "TestVendor" ; "" ; "123ABC" ; "Tester" ; "123456" )`

Authorize.Net Example using Key-Values in the optParam, single optParam value:

`PCCC_Void( "ANetLogin12345" ; "" ; "1234567890ABCDEF" ; "123ABC" ; "" ; "" ;  
"x_email=test@test.com;x_address=123 Fake St;x_delim_char=|" )`

Authorize.Net Example using Key-Values in the optParam, multiple optParam values:

`PCCC_Void( "ANetLogin12345" ; "" ; "1234567890ABCDEF" ; "123ABC" ; "" ; "" ; "x_email=test@test.com" ;  
"x_address=123 Fake St" ; "x_delim_char=|" )`

**PCCC RawPost( GatewayID ; optMerchantID ; optTxnKey ; optUserName ; optPassword [ ; optParams ... ] )**

**Purpose:**

This is an advanced function that allows for a completely customized post request. There is 1 minimum parameters and five optional parameters. The optParams parameter(s) can append any key-value pair that matches the format of the gateway currently being used. For more information on key-value pair format, please refer to the developer documentation of your chosen gateway.

Not all parameters are required for every gateway, see Notes & Examples below.

**Dependencies:**

Must have called PCCC\_UseGateway.

**Parameters:**

Parameter Name	Purpose	Values	Default Value
GatewayID	This is the payment gateway's primary indicator.	For Authorize.Net and its emulators, this is the "API Login ID". For PayPal, this is the "Partner".	
optMerchantID	The identifier of the merchant account.	PayPal only. Corresponds to the merchant's "Vendor" identifier.	
optTxnKey	The transaction key.	This is applicable only for Authorize.Net and its emulators.	
optTxnID	The transaction ID of a previously authorized and uncaptured transaction	For Authorize.Net and its emulators, this is the "TransID" response key. For PayPal, this is the "ORIGID" response key.	
optUserName	The merchant account's username.	PayPal only.	
optPassword	The merchant account's password.	PayPal only.	
optParams	Append key value pairs	This can be a single parameter of key-value pairs separated by the ";" character, or a series of additional parameters added. See example below.	

**Return Values:**

A transaction ID to be used for future reference, otherwise an !!ERROR!!

**Notes & Examples:**

Authorize.Net Example: `PCCC_RawPost( "ANetLogin12345" ; "" ; "1234567890ABCDEF" ; "123ABC" ; "" ; "" )`

PayPal Example: `PCCC_RawPost( "PayPal" ; "TestVendor" ; "" ; "123ABC" ; "Tester" ; "123456" )`

Authorize.Net Example using Key-Values in the optParam, single optParam value:

```
PCCC_RawPost( "ANetLogin12345" ; "" ; "1234567890ABCDEF" ; "123ABC" ; "" ; "" ;  
"x_email=test@test.com;x_address=123 Fake St;x_delim_char=|" )
```

Authorize.Net Example using Key-Values in the optParam, multiple optParam values:

```
PCCC_RawPost( "ANetLogin12345" ; "" ; "1234567890ABCDEF" ; "123ABC" ; "" ; "" ;  
"x_email=test@test.com" ;  
"x_address=123 Fake St" ; "x_delim_char=|" )
```

**PCCC\_PP\_VoiceAuth( Partner ; Vendor ; User ; Password ; AuthCode ; Amount ; CardNum ; ExpDate; OptParam )**

**Purpose:**

This function can capture payment for a transaction that was voice authorized with the customer's bank. The bank provides an authorization code that is then supplied as one of the parameters to the function. Besides its eight minimum parameters, it can append any of the key value pairs from the PP\_PayflowPro\_Guide provided by PayPal.

**Dependencies:**

None

**Parameters:**

Parameter Name	Purpose	Values	Default Value
Partner	PayflowPro login info.	Provided by PayPal	
Vendor	PayflowPro login info.	Provided by PayPal	
User	PayflowPro login info.	Provided by PayPal	
Password	PayflowPro login info.	Provided by PayPal	
AuthCode	An authorization code provided over the phone by the customer's bank.	some authorization code	
Amount	The amount to be captured from the customer's card.	The format should be "dollars.cents", with no commas or dollar signs.	
CardNum	The customer's card number.	The number should have no dashes "-", only numbers.	
ExpDate	The expiration date on the customer's card.	The date should be formatted as MMYYYY.	
OptParam	Append key value pairs.	Please see "PP_Payflowpro_Guide"	

**Return Values:**

A transaction ID on success, otherwise an !!ERROR!!

**Notes & Examples:**

Example: PCCC\_PP\_VoiceAuth( "PayPal" ; "ProductiveComputing" ; "somUser1" ; "somePassword" ; "123456" ; "10.00" ; "5105105105105100" ; "122020" )

Example using KeyValues in the OptParam: PCCC\_PP\_VoiceAuth( "PayPal" ; "ProductiveComputing" ; "somUser1" ; "somePassword" ; "123456" ; "10.00" ; "5105105105105100" ; "122020" ; "STREET=123 Fake St." ; "EMAIL=test@test.com" )

## **PCCC\_PP\_Inquire( Partner ; Vendor ; User ; Password ; OrigID; OptParam )**

### **Purpose:**

This function can inquire as to the status of a previous transaction. Simply provide it with the ID from any previous call to a PayPal function. Besides its five minimum parameters, it can append any of the key value pairs from the PP\_PayflowPro\_Guide provided by PayPal.

### **Dependencies:**

Any previous call to a paypal function that returned a transaction ID.

### **Parameters:**

<b>Parameter Name</b>	<b>Purpose</b>	<b>Values</b>	<b>Default Value</b>
Partner	PayflowPro login info.	Provided by PayPal	
Vendor	PayflowPro login info.	Provided by PayPal	
User	PayflowPro login info.	Provided by PayPal	
Password	PayflowPro login info.	Provided by PayPal	
OrigID	The ID returned from a previous transaction.	some ID	
OptParam	Append key value pairs.	Please see "PP_Payflowpro_Guide"	

### **Return Values:**

A transaction ID on success, otherwise an !!ERROR!!

### **Notes & Examples:**

Example: PCCC\_PP\_Inquire( "PayPal" ; "ProductiveComputing" ; "someUser1" ; "somePassword" ; "123456" )

Example using KeyValues in the OptParam: PCCC\_PP\_Inquire( "PayPal" ; "ProductiveComputing" ; "someUser1" ; "somePassword" ; "123456"; "STARTTIME=20081220202020" ; "ENDTIME=20091220202020" )

**PCCC CP AuthorizeCapture( GatewayID ; Amount ; optTrack1 ; optTrack2 ; optMerchantID ; optTxnKey ; optUserName ; optPassword [ ; optParams ... ] )**

**Purpose:**

This function does an authorization and capture for funds in one action using Card Present (or “Swipe”) methods. There are three minimum parameters and five optional parameters. The optParams parameter(s) can append any key-value pair that matches the format of the gateway currently being used. For more information on key-value pair format, please refer to the developer documentation of your chosen gateway.

For any Card Present transaction, either Track 1 or Track 2 of the swiped card needs to be submitted to the gateway. If both are provided to the function, the plug-in will use Track 1.

Not all parameters are required for every gateway.

**Dependencies:**

Must have called PCCC\_UseGateway.

**Parameters:**

Parameter Name	Purpose	Values	Default Value
GatewayID	This is the payment gateway’s primary indicator.	For Authorize.Net and its emulators, this is the “API Login ID”. For PayPal, this is the “Partner”.	
Amount	The desired amount of the transaction.	In the format of "Dollars.Cents"	
optTrack1	The Track 1 data of the magnetic strip		
optTrack2	The Track 2 data of the magnetic strip		
optMerchantID	The identifier of the merchant account.	PayPal only. Corresponds to the merchant’s “Vendor” identifier.	
optTxnKey	The transaction key.	This is applicable only for Authorize.Net and its emulators.	
optUserName	The merchant account’s username.	PayPal only.	
optPassword	The merchant account’s password.	PayPal only.	
optParams	Append key value pairs	This can be a single parameter of key-value pairs separated by the “;” character, or a series of additional parameters added. See example below.	

**Return Values:**

A transaction ID to be used for future reference, otherwise an !!ERROR!!



**PCCC\_CP\_AuthorizeOnly( GatewayID ; Amount ; optTrack1 ; optTrack2 ; optMerchantID ; optTxnKey ; optUserName ; optPassword [ ; optParams ... ] )**

**Purpose:**

This function does an authorization for the specified amount of funds on a provided credit card using Card Present (or “Swipe”) methods. The funds are not captured until the PCCC\_CaptureOnly function is called with the returned transaction ID. There are three minimum parameters and five optional parameters. The optParams parameter(s) can append any key-value pair that matches the format of the gateway currently being used. For more information on key-value pair format, please refer to the developer documentation of your chosen gateway. For any Card Present transaction, either Track 1 or Track 2 of the swiped card needs to be submitted to the gateway. If both are provided to the function, the plug-in will use Track 1.

Not all parameters are required for every gateway.

**Dependencies:**

Must have called PCCC\_UseGateway.

**Parameters:**

Parameter Name	Purpose	Values	Default Value
GatewayID	This is the payment gateway’s primary indicator.	For Authorize.Net and its emulators, this is the “API Login ID”. For PayPal, this is the “Partner”.	
Amount	The desired amount of the transaction.	In the format of "Dollars.Cents"	
optTrack1	The Track 1 data of the magnetic strip		
optTrack2	The Track 2 data of the magnetic strip		
optMerchantID	The identifier of the merchant account.	PayPal only. Corresponds to the merchant’s “Vendor” identifier.	
optTxnKey	The transaction key.	This is applicable only for Authorize.Net and its emulators.	
optUserName	The merchant account’s username.	PayPal only.	
optPassword	The merchant account’s password.	PayPal only.	
optParams	Append key value pairs	This can be a single parameter of key-value pairs separated by the “;” character, or a series of additional parameters added. See example below.	

**Return Values:**

A transaction ID to be used for future reference, otherwise an !!ERROR!!

**PCCC\_CP\_CaptureOnly( GatewayID ; optMerchantID ; optTxnKey ; optTxnID ; optUserName ; optPassword [ ; optParams ... ] )**

**Purpose:**

This function does an authorization for the specified amount of funds on a provided credit card. The funds are not captured until the PCCC\_CaptureOnly function is called with the returned transaction ID. There is one minimum parameter and six optional parameters. The optParams parameter(s) can append any key-value pair that matches the format of the gateway currently being used. For more information on key-value pair format, please refer to the developer documentation of your chosen gateway.

The difference between this function and PCCC\_CaptureOnly is that this function submits Capture requests for previously-authorized Card Present transactions.

**Dependencies:**

Must have called PCCC\_UseGateway.

**Parameters:**

Parameter Name	Purpose	Values	Default Value
GatewayID	This is the payment gateway's primary indicator.	For Authorize.Net and its emulators, this is the "API Login ID". For PayPal, this is the "Partner".	
optMerchantID	The identifier of the merchant account.	PayPal only. Corresponds to the merchant's "Vendor" identifier.	
optTxnKey	The transaction key.	This is applicable only for Authorize.Net and its emulators.	
optTxnID	The transaction ID of a previously authorized and uncaptured transaction	For Authorize.net and its emulators, this is the "TransID" response key. For PayPal, this is the "ORIGID" response key.	
optUserName	The merchant account's username.	PayPal only.	
optPassword	The merchant account's password.	PayPal only.	
optParams	Append key value pairs	This can be a single parameter of key-value pairs separated by the ";" character, or a series of additional parameters added. See example below.	

**Return Values:**

A transaction ID to be used for future reference, otherwise an !!ERROR!!

## 4) Deprecated Gateway-Specific Functions

### **PCCC\_AN AuthCapt( APILoginID ; TransactionKey ; Amount ; CardNum ; ExpDate; OptParam )**

**Purpose:**

This function does an authorization and capture for funds in one action. There are five minimum parameters and one optional parameter. The OptParam can append any of the key value pairs from the Advanced Integration Method (AIM) developer guide provided by Authorize.Net.

**Dependencies:**

None

**Parameters:**

Parameter Name	Purpose	Values	Default Value
APILoginID	This is the login provided by Authorize.Net after signing up for their payment gateway.	Provided by Authorize.Net	
TransactionKey	This is the transaction key provided by Authorize.Net, to verify ownership of the gateway account.	Provided by Authorize.Net	
Amount	The desired amount of the transaction.	In the format of "Dollars.Cents"	
CardNum	The Credit card number used in the transaction.	some credit card number (no spaces)	
ExpDate	The expiration date of the supplied credit card.	In the format "MMYYYY"	
OptParam	Append key value pairs	Please see "AIM_guide"	

**Return Values:**

A transaction ID to be used for future reference, otherwise an !!ERROR!!

**Notes & Examples:**

Example: PCCC\_AN\_AuthCapt( "PCI3u3zM6b" ; "92cFg7PCIG89cY" ; "3.50" ; "601100000000012" ; "122020" )

Example using KeyValues in the OptParam: PCCC\_AN\_AuthCapt( "PCI3u3zM6b" ; "92cFg7PCIG89cY" ; "3.50" ; "601100000000012" ; "122020" ; "x\_email=test@test.com" ; "x\_address=123 Fake St" )

**PCCC\_AN\_Authorize( APILoginID ; TransactionKey ; Amount ; CardNum ; ExpDate; OptParam )**

**Purpose:**

This function does an authorization for the specified amount of funds on a provided credit card. The funds are not captured until the Capture function is called with the returned transID. There are five minimum parameters and one optional parameter. The OptParam can append any of the key value pairs from the Advanced Integration Method (AIM) developer guide provided by Authorize.Net.

**Dependencies:**

None

**Parameters:**

Parameter Name	Purpose	Values	Default Value
APILoginID	This is the login provided by Authorize.Net, after signing up for their payment gateway.	Provided by Authorize.Net	
TransactionKey	This is the transaction key provided by Authorize.Net to verify ownership of the gateway account.	Provided by Authorize.Net	
Amount	The desired amount of the transaction.	In the format of "Dollars.Cents"	
CardNum	The credit card number used in the transaction.	some credit card number (no spaces)	
ExpDate	The expiration date of the supplied credit card.	In the format "MMYYYY"	
OptParam	Append key value pairs	Please see "AIM_guide"	

**Return Values:**

A transaction ID upon a successful authorization, otherwise an !!ERROR!!

**Notes & Examples:**

Example: PCCC\_AN\_Authorize( "PCI3u3zM6b" ; "92cFg7PCIG89cY" ; "3.50" ; "601100000000012" ; "122020" )

Example using KeyValues in the OptParam: PCCC\_AN\_Authorize( "PCI3u3zM6b" ; "92cFg7PCIG89cY" ; "3.50" ; "601100000000012" ; "122020" ; "x\_email=test@test.com" ; "x\_address=123 Fake St" )

## **PCCC AN Capture( APILoginID ; TransactionKey ; TransID ; Amount; OptParam )**

### **Purpose:**

This function can capture funds from a previous call to Authorize. Simply supply the previous transaction ID and an amount less than or equal to the authorized amount. There are four minimum parameters and one optional parameter. The OptParam can append any of the key value pairs from the Advanced Integration Method (AIM) developer guide provided by Authorize.Net.

### **Dependencies:**

None

### **Parameters:**

<b>Parameter Name</b>	<b>Purpose</b>	<b>Values</b>	<b>Default Value</b>
APILoginID	This is the login provided by Authorize.Net, after signing up for their payment gateway.	Provided by Authorize.Net	
TransactionKey	This is the transaction key provided by Authorize.Net, to verify ownership of the gateway account.	Provided by Authorize.Net	
TransID	This is the transaction ID from a previous call to PCCC_AN_Capture.	some transaction ID	
Amount	An amount less than or equal to the previously authorized amount.	In the format of "Dollars.Cents"	
OptParam	Append key value pairs	Please see "AIM_guide"	

### **Return Values:**

The provided transaction ID on success, otherwise an !!ERROR!!

### **Notes & Examples:**

Example: PCCC\_AN\_Capture( "PCI3u3zM6b" ; "92cFg7PCIG89cY" ; "123456" ; "20.00" )

Example using KeyValues in the OptParam: PCCC\_AN\_AuthCapt( "PCI3u3zM6b" ; "92cFg7PCIG89cY" ; "123456" ; "3.50" ; "x\_email=test@test.com" ; "x\_ship\_to\_address=123 Fake St")

**PCCC\_AN\_CaptOnly( APILoginID ; TransactionKey ; AuthCode ; Amount ; CardNum ; ExpDate; OptParam )**

**Purpose:**

This function can capture payment on a transaction that was not authorized through the gateway, but was authorized directly by the bank. In this case, the bank or authorizing party must supply an authorization code to be providing in the function parameter. There are six minimum parameters and one optional parameter. The OptParam can append any of the key value pairs from the Advanced Integration Method (AIM) developer guide provided by Authorize.Net.

**Dependencies:**

A previously authorized transaction, by means other than the gateway.

**Parameters:**

Parameter Name	Purpose	Values	Default Value
APILoginID	This is the login provided by Authorize.Net, after signing up for their payment gateway.	Provided by Authorize.Net	
TransactionKey	This is the transaction key provided by Authorize.Net, to verify ownership of the gateway account.	Provided by Authorize.Net	
AuthCode	A code provided by a bank for a previously authorized amount.	some Auth code	
Amount	An amount less than or equal to the previously authorized amount.	In the format of "Dollars.Cents"	
CardNum	The Credit card number used in the transaction.	some credit card number (no spaces)	
ExpDate	The expiration date of the supplied credit card.	In the format "MMYYYY"	
OptParam	Append key value pairs	Please see "AIM_guide"	

**Return Values:**

A transaction ID on success, otherwise an !!ERROR!!

**Notes & Examples:**

Example: PCCC\_AN\_CaptOnly( "PCI3u3zM6b" ; "92cFg7PCIG89cY" ; "123456" ; "10.00" ; "601100000000012" ; "122020" )

Example using KeyValues in the OptParam: PCCC\_AN\_CaptOnly( "PCI3u3zM6b" ; "92cFg7PCIG89cY"; "123456" ; "3.50" ; "601100000000012" ; "122020" ; "x\_email=test@test.com" ; "x\_address=123 Fake St")

## **PCCC AN Credit( APILoginID ; TransactionKey ; TransID ; Amount ; CardNum; OptParam )**

### **Purpose:**

This function can credit or refund a customer for a previous transaction performed on the gateway. There are five minimum parameters and one optional parameter. The OptParam can append any of the key value pairs from the Advanced Integration Method (AIM) developer guide provided by Authorize.Net.

### **Dependencies:**

Must have a transaction ID from a previous call to AuthCapt or Capture, and that payment must have cleared.

### **Parameters:**

<b>Parameter Name</b>	<b>Purpose</b>	<b>Values</b>	<b>Default Value</b>
APILoginID	This is the login provided by Authorize.Net, after signing up for their payment gateway.	Provided by Authorize.Net	
TransactionKey	This is the transaction key provided by Authorize.Net, to verify ownership of the gateway account.	Provided by Authorize.Net	
TransID	This is the transaction ID from a previous call to PCCC_AN_Capture.	some transaction ID	
Amount	An amount less than or equal to the previously authorized amount.	some amount	
CardNum	The Credit card number used in the transaction.	Either the entire credit card number, or just the last four digits.	
OptParam	Append key value pairs	Please see "AIM_guide"	

### **Return Values:**

A transaction ID on success, otherwise an !!ERROR!!

### **Notes & Examples:**

Example: PCCC\_AN\_Credit( "PCI3u3zM6b" ; "92cFg7PCIG89cY" ; "123456" ; "15.00" ; "601100000000012" )

Example using KeyValues in the OptParam: PCCC\_AN\_Credit( "PCI3u3zM6b" ; "92cFg7PCIG89cY" ; "123456" ; "15.00" ; "601100000000012" ; "x\_email=test@test.com" ; "x\_address=123 Fake St" )

## **PCCC\_AN\_UCredit( APILoginID ; TransactionKey ; Amount ; CardNum ; ExpDate; OptParam )**

### **Purpose:**

This function can refund any credit card without a previous transaction ID. The "U" stands for "Unlinked". Meaning it does not require a previous transaction through the gateway, and it requires special privileges from the merchant's bank to use. Also that is why it requires extra parameters like the full card number, amount and expiration date. There are five minimum parameters and one optional parameter. The OptParam can append any of the key value pairs from the Advanced Integration Method (AIM) developer guide provided by Authorize.Net.

### **Dependencies:**

None

### **Parameters:**

<b>Parameter Name</b>	<b>Purpose</b>	<b>Values</b>	<b>Default Value</b>
APILoginID	This is the login provided by Authorize.Net, after signing up for their payment gateway.	Provided by Authorize.Net	
TransactionKey	This is the transaction key provided by Authorize.Net, to verify ownership of the gateway account.	Provided by Authorize.Net	
Amount	The amount to be refunded to a customer's card.	In the format of "Dollars.Cents"	
CardNum	The Credit card number used in the transaction.	some credit card number (no spaces)	
ExpDate	The expiration date of the supplied credit card.	In the format "MMYYYY"	
OptParam	Append key value pairs	Please see "AIM_guide"	

### **Return Values:**

A transaction ID on success, otherwise an !!ERROR!!

### **Notes & Examples:**

Example: PCCC\_AN\_UCredit( "PCI3u3zM6b" ; "92cFg7PCIG89cY" ; "1.00" ; "601100000000012" ; "122020" )

Example using KeyValues in the OptParam: PCCC\_AN\_UCredit( "PCI3u3zM6b" ; "92cFg7PCIG89cY" ; "1.00" ; "601100000000012" ; "122020" ; "x\_email=test@test.com" ; "x\_address=123 Fake St" )



## **PCCC AN Void( APILoginID ; TransactionKey ; TransID; OptParam )**

### **Purpose:**

This function can be used to void a previous transaction that has not yet been cleared by the bank, which usually happens at the end of the day. It cannot void partial amounts, only the entire amount of the transaction. There are three minimum parameters and one optional parameter. The OptParam can append any of the key value pairs from the Advanced Integration Method (AIM) developer guide provided by Authorize.Net.

### **Dependencies:**

A previous call to AuthCapt, Authorize, or Capture, as long as it is before the payment is cleared by the bank.

### **Parameters:**

<b>Parameter Name</b>	<b>Purpose</b>	<b>Values</b>	<b>Default Value</b>
APILoginID	This is the login provided by Authorize.Net, after signing up for their payment gateway.	Provided by Authorize.Net	
TransactionKey	This is the transaction key provided by Authorize.Net, to verify ownership of the gateway account.	Provided by Authorize.Net	
TransID	This is the transaction ID from a previous call to PCCC_AN_AuthCapt, PCCC_AN_Authorize, or PCCC_AN_Capture.	some transaction key	
OptParam	Append key value pairs	Please see "AIM_guide"	

### **Return Values:**

A transaction ID on success, otherwise an !!ERROR!!

### **Notes & Examples:**

It is always a good idea to call this function before calling PCCC\_AN\_Credit, if there is any uncertainty about whether the bank has cleared the payment.

Example: PCCC\_AN\_Void( "PCI3u3zM6b" ; "92cFg7PCIG89cY"; "someTransID" )

Example using KeyValues in the OptParam: PCCC\_AN\_Void( "PCI3u3zM6b" ; "92cFg7PCIG89cY"; "someTransID" ; "x\_email=test@test.com" ; "x\_address=123 Fake St" )

## **PCCC AN RawPost( APILoginID ; TransactionKey; OptParam )**

### **Purpose:**

This is an advanced function that allows for a completely customized post request. There are two minimum parameters and one optional parameter. The OptParam can append any of the key value pairs from the Advanced Integration Method (AIM) developer guide provided by Authorize.Net.

### **Dependencies:**

None

### **Parameters:**

<b>Parameter Name</b>	<b>Purpose</b>	<b>Values</b>	<b>Default Value</b>
APILoginID	The authorize merchant ID	Provided by Authorize.Net	
TransactionKey	The transaction key provided by authorize to validate to gateway posts.	Provided by Authorize.Net	
OptParam	Append key value pairs	Please see "AIM_guide"	

### **Return Values:**

A transaction ID on success, otherwise an !!ERROR!!

### **Notes & Examples:**

```
PCCC_AN_RawPost( "PCI3u3zM6b" ; "92cFg7PCIG89cY" ; "x_method=ECHECK" ;  
"x_bank_aba_code=123456789" ; "x_bank_acct_num=8675309" ; "x_bank_acct_type=CHECKING" ;  
"x_bank_name=chase" ; "x_bank_acct_name=mine" ; "x_echeck_type=ARC" ; "x_bank_check_number=1234" )
```

This is an example of implementing an electronic check transaction that cannot be implemented with the standard functions.

**PCCC\_PP\_Sale( Partner ; Vendor ; User ; Password ; Amount ; CardNum ; ExpDate;  
OptParam )**

**Purpose:**

This function does an authorization and capture for funds in one action. Besides its seven minimum parameters, it can append any of the key value pairs from the PP\_PayflowPro\_Guide provided by PayPal.

**Dependencies:**

None

**Parameters:**

Parameter Name	Purpose	Values	Default Value
Partner	PayflowPro login info.	Provided by PayPal	
Vendor	PayflowPro login.	Provided by PayPal	
User	PayflowPro login info.	Provided by PayPal	
Password	PayflowPro login info.	Provided by PayPal	
Amount	The amount of the transaction.	The format should be "dollars.cents", with no commas or dollar signs.	
CardNum	The customer's credit card number.	The number should have no dashes "-", only numbers.	
ExpDate	The expiration date on the card.	The date should be formatted as MMYYYY.	
OptParam	Append key value pairs.	Please see "PP_PayflowPro_Guide"	

**Return Values:**

A transaction ID on success, otherwise an !!ERROR!!

**Notes & Examples:**

Example: PCCC\_PP\_Sale( "PayPal" ; "ProductiveComputing" ; "someUser1" ; "somePassword" ; "10.01" ; "5105105105105100" ; "1220" )

Example using KeyValues in the OptParam: PCCC\_PP\_Sale( "PayPal" ; "ProductiveComputing" ; "someUser1" ; "somePassword" ; "10.01" ; "5105105105105100" ; "1220" ; "STREET=123 Fake St." ; "EMAIL=test@test.com" )

**PCCC\_PP\_Authorize( Partner ; Vendor ; User ; Password ; Amount ; CardNum ; ExpDate; OptParam )**

**Purpose:**

This function does an authorization for the specified amount of funds on a provided credit card. The funds are not captured until the Capture function is called with the returned origID. Besides its seven minimum parameters, it can append any of the key value pairs from the PP\_PayflowPro\_Guide provided by PayPal.

**Dependencies:**

None

**Parameters:**

Parameter Name	Purpose	Values	Default Value
Partner	PayflowPro login info.	Provided by PayPal	
Vendor	PayflowPro login info.	Provided by PayPal	
User	PayflowPro login info.	Provided by PayPal	
Password	PayflowPro login info.	Provided by PayPal	
Amount	The amount of the transaction.	The format should be "dollars.cents", with no commas or dollar signs.	
CardNum	The customer's credit card number.	The number should have no dashes "-", only numbers.	
ExpDate	The expiration date on the card.	The date should be formatted as MMYYYY.	
OptParam	Append key value pairs	Please see "PP_Payflowpro_Guide"	

**Return Values:**

A transaction ID on success, otherwise an !!ERROR!!

**Notes & Examples:**

Example: PCCC\_PP\_Authorize( "PayPal" ; "ProductiveComputing" ; "someUser1" ; "somePassword" ; "10.01" ; "5105105105105100" ; "1220" )

Example using KeyValues in the OptParam: PCCC\_PP\_Authorize( "PayPal" ; "ProductiveComputing" ; "someUser1" ; "somePassword" ; "10.01" ; "5105105105105100" ; "1220" ; "STREET=123 Fake St." ; "EMAIL=test@test.com" )

## **PCCC\_PP\_Capture( Partner ; Vendor ; User ; Password ; OrigID; OptParam )**

### **Purpose:**

This function can capture funds from a previous call to Authorize. Simply supply the original ID. Besides its five minimum parameters, it can append any of the key value pairs from the PP\_PayflowPro\_Guide provided by PayPal.

### **Dependencies:**

Must have called PCCC\_PP\_Authorize and retained the ID returned.

### **Parameters:**

<b>Parameter Name</b>	<b>Purpose</b>	<b>Values</b>	<b>Default Value</b>
Partner	PayflowPro login info.	Provided by PayPal	
Vendor	PayflowPro login info.	Provided by PayPal	
User	PayflowPro login info.	Provided by PayPal	
Password	PayflowPro login info.	Provided by PayPal	
OrigID	The ID returned from an original PCCC_PP_Authorize transaction.	some ID	
OptParam	Append key value pairs.	Please see "PP_Payflowpro_Guide"	

### **Return Values:**

A transaction ID on success, otherwise an !!ERROR!!

### **Notes & Examples:**

A particularly important key value pair for this function is "AMT". This represents the amount. If less than the initial authorized amount is to be captured this is the parameter used to do it, as shown in the example.

Example: PCCC\_PP\_Capture( "PayPal" ; "ProductiveComputing" ; "someUser2" ; "somePassword" ; "123" )Or with an amount less than the originally authorized amount:PCCC\_PP\_Capture( "PayPal" ; "ProductiveComputing" ; "ProductiveComputing" ; "PcProgrammer1" ; "123" )

Example using KeyValues in the OptParam: PCCC\_PP\_Capture( "PayPal" ; "ProductiveComputing" ; "someUser2" ; "somePassword" ; "123" )Or with an amount less than the originally authorized amount:PCCC\_PP\_Capture( "PayPal" ; "ProductiveComputing" ; "ProductiveComputing" ; "PcProgrammer1" ; "123" ; "AMT=10.50" )

## **PCCC\_PP\_Void( Partner ; Vendor ; User ; Password ; OrigID; OptParam )**

### **Purpose:**

This function can void a previous call to Authorize. Simply supply the original ID. Besides its five minimum parameters, it can append any of the key value pairs from the PP\_PayflowPro\_Guide provided by PayPal.

### **Dependencies:**

Must have called PCCC\_PP\_Authorize and be able to provide the original ID from that transaction.

### **Parameters:**

<b>Parameter Name</b>	<b>Purpose</b>	<b>Values</b>	<b>Default Value</b>
Partner	PayflowPro login info.	Provided by PayPal	
Vendor	PayflowPro login info.	Provided by PayPal	
User	PayflowPro login info.	Provided by PayPal	
Password	PayflowPro login info.	Provided by PayPal	
OrigID	The ID returned from an original PCCC_PP_Authorize transaction.	some ID	
OptParam	Append key value pairs.	Please see "PP_Payflowpro_Guide"	

### **Return Values:**

A transaction ID on success, otherwise an !!ERROR!!

### **Notes & Examples:**

Example: PCCC\_PP\_Void( "PayPal" ; "ProductiveComputing" ; "someUser1" ; "somePassword" ; "123456" )

Example using KeyValues in the OptParam: PCCC\_PP\_Void( "PayPal" ; "ProductiveComputing" ; "someUser1" ; "somePassword" ; "123456" ; "STREET=123 Fake St." ; "EMAIL=test@test.com" )

## **PCCC\_PP\_Credit( Partner ; Vendor ; User ; Password ; OrigID; OptParam )**

### **Purpose:**

This function can credit or refund a customer for a previous transaction performed on the gateway. Besides its five minimum parameters, it can append any of the key value pairs from the PP\_PayflowPro\_Guide provided by PayPal.

### **Dependencies:**

A previous successful call to either PCCC\_PP\_Sale or PCCC\_PP\_Capture.

### **Parameters:**

<b>Parameter Name</b>	<b>Purpose</b>	<b>Values</b>	<b>Default Value</b>
Partner	PayflowPro login info.	Provided by PayPal	
Vendor	PayflowPro login info.	Provided by PayPal	
User	PayflowPro login info.	Provided by PayPal	
Password	PayflowPro login info.	Provided by PayPal	
OrigID	The ID from a previously captured payment.	some ID	
OptParam	Append key value pairs.	Please see "PP_Payflowpro_Guide"	

### **Return Values:**

A transaction ID on success, otherwise an !!ERROR!!

### **Notes & Examples:**

Example: PCCC\_PP\_Credit( "PayPal" ; "ProductiveComputing" ; "someUser1" ; "somePassword" ; "123456" )

Example using KeyValues in the OptParam: PCCC\_PP\_Credit( "PayPal" ; "ProductiveComputing" ; "someUser1" ; "somePassword" ; "123456" ; "STREET=123 Fake St." ; "EMAIL=test@test.com" )

**PCCC\_PP\_UCredit( Partner ; Vendor ; User ; Password ; Amount ; CardNum ; ExpDate; OptParam )**

**Purpose:**

This function can credit or refund a customer for a previous transaction NOT performed on the gateway. The 'U' stands for unlinked, meaning no previous transaction is required to add funds to a card. This is why all of the card information is required. Also, to utilize this function requires special permission from the merchant's bank. Besides its seven minimum parameters, it can append any of the key value pairs from the PP\_PayflowPro\_Guide provided by PayPal.

**Dependencies:**

None

**Parameters:**

Parameter Name	Purpose	Values	Default Value
Partner	PayflowPro login info.	Provided by PayPal	
Vendor	PayflowPro login info.	Provided by PayPal	
User	PayflowPro login info.	Provided by PayPal	
Password	PayflowPro login info.	Provided by PayPal	
Amount	The amount to be refunded to a customer's credit card.	The format should be "dollars.cents", with no commas or dollar signs.	
CardNum	The customer's card number.	The number should have no dashes "-", only numbers.	
ExpDate	The expiration date on the customer's card.	The date should be formatted as MMYYYY.	
OptParam	Append key value pairs.	Please see "PP_Payflowpro_Guide"	

**Return Values:**

A transaction ID on success, otherwise an !!ERROR!!

**Notes & Examples:**

Example: PCCC\_PP\_UCredit( "PayPal" ; "ProductiveComputing" ; "somUser1" ; "somePassword" ; "10.00" ; "5105105105105100" ; "122020" )

Example using KeyValues in the OptParam: PCCC\_PP\_UCredit( "PayPal" ; "ProductiveComputing" ; "somUser1" ; "somePassword" ; "10.00" ; "5105105105105100" ; "122020" ; "STREET=123 Fake St." ; "EMAIL=test@test.com" )



## **PCCC\_PP\_RawPost( Partner ; Vendor ; User ; Password; OptParam )**

### **Purpose:**

This is an advanced function that allows for a custom post to be built. It only requires the login information for a merchant's account, and the rest must be constructed from key value pairs as outlined in the PP\_PayflowPro\_Guide provided by PayPal. Any of the other built in functions can be recreated with this function as well as being able to use some advanced capabilities not achievable with the built in functions.

### **Dependencies:**

Depends on the transaction built.

### **Parameters:**

<b>Parameter Name</b>	<b>Purpose</b>	<b>Values</b>	<b>Default Value</b>
Partner	PayflowPro login info.	Provided by PayPal	
Vendor	PayflowPro login info.	Provided by PayPal	
User	PayflowPro login info.	Provided by PayPal	
Password	PayflowPro login info.	Provided by PayPal	
OptParam	Append key value pairs.	Please see "PP_Payflowpro_Guide"	

### **Return Values:**

A transaction ID on success, otherwise an !!ERROR!!

### **Notes & Examples:**

```
PCCC_PP_RawPost( "PayPal"; "ProductiveComputing" ; "someUser2" ; "somePassword" ; "AMT=100.00" ;  
"TRXTYPE=S" ; "TENDER=D" ; "ACCT=4444333322221111" ; "EXPDATE=1012" )
```

This is an example of a pinless debit transaction. Notice the "TENDER" tag, which represents the method of payment for PayPal.

### III. Available Keys for PCCC\_GetResp

PCCC\_GetResp can be called with no parameters, which will return the raw response from the gateway. PCCC\_GetResp can also be called with one of the following key values relative to the gateway provider. The set of key values used corresponds directly to the last gateway call made. If a call was made to Authorize.Net, then one of the Authorize.Net keys should be used. If a wrong key is ever used, then !!ERROR!! will be returned. The Authorize.Net, PayItForward, and Authorize.Net Emulator keys are NOT case sensitive, while the PayPal keys ARE CASE SENSITIVE. All available keys are listed below. If there is a missing key that you would like to access, then please email a request to [support@productivecomputing.com](mailto:support@productivecomputing.com)

#### Authorize.Net, PayItForward, and Authorize.Net Emulator Keys

The “API Return Field” is provided purely for reference. Only the Key should be used with the PCCC\_GetResp function. Authorize responds with a comma separated list instead of a key value list. The “API Return Field” then allows the developer to identify which key corresponds to a given value in the list, when examining the raw return string.

Key	API Return Field Number
RespCode	1
RespSubCode	2
RespReasonCode	3
RespReasonText	4
AuthCode	5
AVSResp	6
TransID	7
InvoiceNum	8
Description	9
Amount	10
Method	11
TransType	12
CustID	13
FirstName	14
LastName	15

Company	16
Address	17
City	18
State	19
Zip	20
Country	21
Phone	22
Fax	23
Email	24
ShipToFirstName	25
ShipToLastName	26
ShipToCompany	27
ShipToAddress	28
ShipToCity	29
ShipToState	30
ShipToZip	31
ShipToCountry	32
Tax	33
Duty	34
Freight	35
TaxExempt	36
PONum	37
MD5Hash	38
CCVResp	39
CAVVRsp	40
AccountNum	51

CardType	52
SplitTendID	53
RequestedAmount	54
BalanceOnCard	55

## PayPal Keys

Possible transaction return values are as follows:

PNREF –	Payflow Transaction ID, a unique number that identifies the transaction. Character length and limitations: 12 alphanumeric characters
PPREF –	Unique transaction ID of the payment. Character length and limitations: 17-character string
RESULT –	The outcome of the attempted transaction. RESULT=0 means the transaction was approved. NOTE: For Account Verification transactions, RESULT=0 with RESPMSG=Verified means a zero dollar authorization has been successfully performed. Any other value for RESULT indicates a decline or error. Character length and limitations: variable length, numeric
CVV2MATCH –	Result of the card security code (CVV2) check. The issuing bank may decline the transaction if there is a mismatch. In other cases, the transaction may be approved despite a mismatch. Character length and limitations: 1 alpha character (Y, N, X, or no response)
RESPMSG –	The response message returned with the transaction result. Exact wording varies. Sometimes a colon appears after the initial RESPMSG followed by more detailed information. NOTE: For Account Verification transactions, RESULT=0 with RESPMSG=Verified means a zero dollar authorization has been successfully performed. Character length and limitations: variable, alphanumeric characters
AUTHCODE –	Returned for Sale, Authorization, and Voice Authorization credit card transactions. AUTHCODE is the approval code obtained over the telephone from the processing network. AUTHCODE is required when submitting a Force (F) transaction. Character length and limitations: 6 alphanumeric characters
AVSADDR –	Address Verification Service address response returned if you are using Address Verification Service. Address Verification Service address responses are for advice only. This process does not affect the outcome of the authorization. Character length and limitations: 1 alpha character (Y, N, X, or no response)
AVSZIP –	Address Verification Service address response returned if you are using Address Verification Service. Address Verification Service address responses are for advice only. This process does not affect the outcome of the authorization. Character length and limitations: 1 alpha character (Y, N, X, or no response)
IAVS –	International Address Verification Service address responses may be returned if you are using Address Verification Service. IAVS responses are for advice only. This value does not affect the outcome of the transaction. Indicates whether Address Verification Service response is international (Y), US (N), or cannot be determined (X). Client version 3.06 or later is required. Character length and limitations: 1 alpha character (Y, N, X, or no response)

- PROCAVS – Address Verification Service response from the processor when you use Address Verification Service and send a VERBOSITY request parameter value of MEDIUM. Character length and limitations: 1 character
- PROCCV2 – CVV2 response from the processor when you send a VERBOSITY request parameter value of MEDIUM. Character length and limitations: 1 character
- CORRELATIONID – Value used for tracking this Direct Payment transaction. Character length and limitations: 13 alphanumeric characters
- AMEXID – Unique transaction ID returned when VERBOSITY=MEDIUM or VERBOSITY=HIGH for tracking American Express CAPN transactions. NOTE: American Express CAPN transactions only: used by merchants who authorize transactions through the payflow gateway but settle through a third-party solution. Character length and limitations: 15 numeric characters
- AMEXPOSTDATA – Value returned when VERBOSITY=MEDIUM or VERBOSITY=HIGH. Character length and limitations: 12 alphanumeric characters

## IV. Links for using “OptParam”

Many of the functions allow you to append key value pairs by using the “OptParam.” This optional parameter is an advanced function and will require that you reference the following guides below depending on your gateway provider.

Advanced Integration Method (AIM) Developer Guide provided by Authorize.Net:

[http://www.productivecomputing.com/docs/docs\\_library/FM\\_CreditCard/AIM\\_guide.pdf](http://www.productivecomputing.com/docs/docs_library/FM_CreditCard/AIM_guide.pdf)

PP\_PayflowPro\_Guide provided by PayPal:

[http://www.productivecomputing.com/docs/docs\\_library/FM\\_CreditCard/PP\\_PayflowPro\\_Guide.pdf](http://www.productivecomputing.com/docs/docs_library/FM_CreditCard/PP_PayflowPro_Guide.pdf)

## V. 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](mailto:support@productivecomputing.com)

Forum: [www.productivecomputing.com/forum](http://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](http://www.productivecomputing.com/rfq). We are ready to assist and look forward to hearing from you!