

CityAPI PHP Library v 0.4

Contents

CityAPI PHP Library v 0.4	1
Description	5
Features	5
Status	5
New in v0.4	5
Author Info	6
Getting Started	7
Basic Usage	7
Simple Example.....	7
Encoding	7
Returned Values	7
Parameters	8
Configuration	8
API Signature and User Token.....	8
Debug Message Behavior	8
File Locations	8
Dependencies	8
Rate Limit Support	8
HTTP Response Handling	9
Exceptions.....	10
Function Documentation	10
Function Index	10
CityAPI PHP Library Function	10
Description.....	10
Status in v 0.3.....	10
Function Descriptions	16
set_key.....	16
set_token	16
get_last_headers	16

get_last_response_start_line	17
get_last_status_code.....	17
get_ratelimit_limit_by_account	18
get_ratelimit_limit_by_ip	18
get_ratelimit_remaining_by_account.....	19
get_ratelimit_remaining_by_ip	19
users_index.....	19
users_show.....	20
users_create	20
users_update	21
users_destroy	21
users_count	22
users_memberize	22
users_dememberize	23
users_deactivate.....	23
users_bulk_memberize.....	24
users_bulk_deactivate.....	24
users_addresses_index.....	25
users_addresses_show	25
users_addresses_create	26
users_addresses_update	26
users_addresses_destroy	27
users_admin_privileges_index.....	27
users_admin_privileges_create	27
users_barcodes_index	28
users_barcodes_show	28
users_barcodes_create.....	29
users_barcodes_destroy.....	29
users_invitations_index.....	30
users_invitations_create.....	30
users_family_index.....	31
users_family_create.....	31
users_family_update	32
users_family_destroy.....	32

users_notes_index.....	32
users_notes_show.....	33
users_notes_create	33
users_processes_index.....	34
users_processes_show	34
users_processes_answers_index.....	35
users_processes_notes_index	35
users_roles_index.....	36
users_roles_show	36
users_roles_create	37
users_roles_destroy.....	37
users_roles_activate	37
users_roles_deactivate	37
users_roles_promote.....	37
users_roles_demote	37
users_skills_index	37
users_skills_create.....	38
users_skills_destroy.....	38
groups_index	39
groups_show	39
groups_create.....	40
groups_update.....	40
groups_count.....	41
groups_addresses_index	41
groups_addresses_show.....	42
groups_addresses_create	42
groups_addresses_update.....	43
groups_addresses_destroy	43
groups_exports_index	44
groups_exports_show	44
groups_exports_create.....	44
groups_invitations_index.....	45
groups_invitations_create	45
groups_roles_index	46

groups_roles_show.....	46
groups_roles_create	47
groups_roles_delete	47
groups_roles_activate	47
groups_roles_deactivate	48
groups_roles_promote	48
groups_roles_demote.....	48
groups_roles_deactivate_all.....	48
groups_tags_index.....	48
groups_tags_create	48
groups_tags_destroy	49
addresses_index	49
addresses_show	50
addresses_groups.....	50
barcodes_show.....	51
families_show.....	51
families_create	51
families_update	51
families_destroy	51
invitations_index	51
invitations_bulk_invite	51
metrics_index	51
metrics_show.....	51
metrics_create	51
metrics_measurements_index	52
metrics_measurements_show.....	52
metrics_measurements_create	52
metrics_measurements_destroy	52
metrics_measurements_last	52
web_hooks_index.....	52
web_hooks_create	52
web_hooks_destroy	53
call_city.....	54

Description

CityAPI is a PHP library to simplify use of The City API within a PHP environment. It is designed for stand-alone use or as a building block for other applications and services.

The initial version supports The City Admin API.

Features

It's easy to use. Digitally signing and hashing is taken care of for you. Query string and URL construction is handled. Encoding and HTTP transactions, status, header processing, and error handling are done by the library. You simply configure with your API key and user token and start interacting with The City data.

Results can be returned as JSON or PHP objects as desired.

Access to HTTP response status code and headers is supported.

Rate limit information is easily accessible to make it easier to prevent exceeding rate limits.

Many required parameters are checked locally to prevent unnecessary web service calls.

The `CityApi` class provides a function for each City Admin API web service endpoint. The functions are named like the endpoints. For example, the function for the endpoint GROUPS COUNT is `groups_count()`.

The class also currently provides a lower-level function, `call_city()`, to allow you to call an arbitrary endpoint by specifying the verb, URL (with querystring if applicable), and JSON body (if applicable). This would be useful to call new endpoints that aren't directly supported by the library yet and to help isolate bugs during testing. This may go away in future releases.

Status

The functions included in this release are those that were documented in the API Endpoints section of The City Admin API v1 documentation dated 5/17/2012. Some of the endpoints listed there are not yet included in the detailed endpoint documentation. This library contains only those that have the detailed documentation. Much is working well. Some functions are in progress, some are stubs. Function signatures are well established and should change little. Please check the function index status for details.

New in v0.4

Changed the visibility of some members from private to protected to facilitate extending the class.

Corrected spelling of admin privileges.

Provided default `$msg` argument to `debug_message()`.

Fixed bug in `users_deactivate()`, now complete.

Added `users_bulk_memberize()`.

Added `users_bulk_deactivate()`.

Completed `users_admin_privileges_index()`.

Completed `users_admin_privileges_create()`.

More testing on users_invitations_create(), still some City-side issues.

Completed users_family_destroy().

Changed users_skills_create() to use JSON body.

Author Info

John Roberts

john@john-roberts.net

Comments, bug reports, and suggestions for improvements are most welcome.

Getting Started

Edit `lib/ca-config.php` to add your API key and user token.

Familiarize yourself with the test code in the `test` directory. There is a test script for each API call. For example, to test `GROUPS COUNT`, see `test/test-groups-count.php`. You can simply change the user and group values in these scripts and try it out.

Basic Usage

The PHP script using the API includes the main library file, `ca-main.php`. This file defines the `CityApi` class. Your script creates an instance of this class and calls class methods to get data from The City.

Simple Example

This code gets the number of groups in your City returned as JSON. It assumes the `ca-main.php` file is a lib subdirectory of your PHP file.

```
<?php

require_once(dirname(__FILE__) . '/lib/ca-main.php');

$ca = new CityApi();
$results = $ca->groups_count();
echo $results;

?>
```

Running this code produces this output (assuming you have 352 groups in your City):

```
{"count":352}
```

Encoding

Returned Values

The City Admin API returns JSON. The library can return data in JSON or in PHP array format. The idea is that you can either return JSON to the browser for jQuery consumption or get a PHP array and process it to your heart's content on your server.

You can control whether you get JSON or PHP arrays returned by setting `$json`. If `$json` is true (the default), the JSON result from The City web service is returned, else a PHP associative array is returned.

This code gets the number of groups in your City, returned as a PHP array:

```
<?php

require_once(dirname(__FILE__) . '/lib/ca-main.php');

$ca = new CityApi();
$ca->set_json(false);
$results = $ca->groups_count();
var_dump($results);

?>
```

Running this code produces this output:

```
array(1) {  
    ["count"]=>  
    int(355)  
}
```

Parameters

Many City API web service calls have name/value pair array arguments. The CityApi library functions expect PHP objects (strings, ints, arrays) as input. The function documentation explains and provides examples in the included test directory. The function signatures are designed to make it fairly straightforward to use from PHP. Some reasonable sanity checking of arguments is done by the library functions.

In many cases, known required arguments are single arguments. Multiple optional or required arguments are usually handled by passing them as an associative array parameter, `$args`. I think this will both allow some local sanity checking and also minimize breaking changes should additional parameters be added to the API endpoints.

Configuration

API Signature and User Token

You will need to edit the `ca-config.php` file to configure the API. This allows updates to be easily made to `ca-main.php` without disturbing your settings.

Set the values of the constants representing the default City API signature and user token in `ca-config.php` to the ones you have generated on The City. These can also be set at run time by using `set_key()` and `set_token()`.

Debug Message Behavior

For development purposes, the class has a debug message facility. Debug messages will be output if `$debug` is true. If it's true, the library will call an external function if it exists, `ca_debug_message()`, to do the actual logging. A default version of this function is defined in `ca-config.php`. You can configure the behavior of debug messages from the class by modifying this function in `ca-config.php`.

File Locations

The only requirement is that `ca-main.php` and `ca-config.php` must be located in the same directory.

Dependencies

The library uses cURL and requires PHP 5. It was developed and tested using PHP 5.3.2 and cURL 7.19.7.

Rate Limit Support

The City API web service endpoints return rate limit information in the HTTP response headers. The rate limit is the number of transactions that are available per hour. The rate limit remaining is the number of transactions still available in the current hour. The City API web service currently resets these rate limits at the top of the hour.

The rate limit from the last call is available by using these functions:

```
get_ratelimit_limit_by_account()
```

```
get_ratelimit_limit_by_ip()
```

```
get_ratelimit_remaining_by_account()
```



```
get_ratelimit_remaining_by_ip()
```

Since this information is returned in HTTP response headers, it is not available until you have made at least one City endpoint call on your current CityApi object. Before the first endpoint call, these functions return NULL.

HTTP Response Handling

The last HTTP response status code, response headers and response start line are retained and can be obtained by calling `get_last_status_code()`, `get_last_headers()` and `get_last_response_start_line()`, respectively.

```
<?php

require_once(dirname(__FILE__) . '/ca-main.php');

$ca = new CityApi();
$ca->set_json(false);
$results = $ca->groups_count();
echo $ca->get_last_status_code();

?>
```

Running this code produces this output:

```
200
```

`var_dump(get_last_status_code())` produces:

```
int(200)
```

`var_dump(get_last_response_start_line())` produces:

```
string(15) "HTTP/1.1 200 OK"
```

`var_dump(get_last_headers())` produces:

```
array(10) {
  ["Content-Type"]=>
  string(31) "application/json; charset=utf-8"
  ["Server"]=>
  string(35) "thin 1.3.1 codename Triple Espresso"
  ["X-City-Ratelimit-Limit-By-Account"]=>
  string(4) "2000"
  ["X-City-Ratelimit-Limit-By-Ip"]=>
  string(4) "2200"
  ["X-City-Ratelimit-Remaining-By-Account"]=>
  string(4) "1994"
  ["X-City-Ratelimit-Remaining-By-Ip"]=>
  string(4) "2192"
  ["X-Frame-Options"]=>
  string(10) "sameorigin"
  ["X-Xss-Protection"]=>
  string(13) "1; mode=block"
  ["Content-Length"]=>
  string(2) "13"
  ["Connection"]=>
  string(10) "keep-alive"
```

```
}
```

Since this information is returned in HTTP response headers, it is not available until you have made at least one City endpoint call on your current CityApi object. Before the first endpoint call, these functions return NULL.

Exceptions

This library catches exceptions that may occur, logs an error using `error_log()`, and returns error information to the caller following The City API Web Service error convention. The HTTP last status code is set to 500. The library does not throw exceptions out of its own world.

Function Documentation

Function Index

CityAPI PHP Library Function	Description	Status in v 0.3
<i>Security Configuration</i>		
set_key	Set the API access key	Complete
set_token	Set the API user token	Complete
<i>HTTP Response Code and Headers</i>		
get_last_status_code	Get the HTTP response status code from the last endpoint call	Complete
get_last_headers	Get the HTTP response headers from the last endpoint call	Complete
get_last_response_start_line	Get the HTTP response start line from the last endpoint call	Complete
<i>Rate Limiting</i>		
get_ratelimit_limit_by_account	Get the number of endpoint calls your account is allowed per hour	Complete
get_ratelimit_limit_by_ip	Get the number of endpoint calls your IP address is allowed per hour	Complete
get_ratelimit_remaining_by_account	Get the number of endpoint calls your account has left in the current clock hour	Complete
get_ratelimit_remaining_by_ip	Get the number of endpoint calls your IP has left in the current clock hour	Complete

<i>Users</i>		
users_index	Get a list of users info	Complete There is a City-side bug relating to filters
users_show	Show info for a given user	Complete
users_create	Create an offline user	Complete
users_update	Update info for a user	Complete City is returning a list of users instead of just the updated user
users_destroy	Delete a user	Complete
users_count	Get the number of users that meet selected criteria	Complete
users_memberize	Memberize a user	Complete
users_dememberize	Dememberize a user	Complete
users_deactivate	Deactivate a user	Complete
users_bulk_memberize	Memberize a list of users	Tested Produced the POST per spec, The City returned 204 per spec, but user was not memberized. The City's API test harness fails on signature error. The City has a fix.
users_bulk_deactivate	Deactivate a list of users	Complete
users_addresses_index	Get the list of addresses for a user	Complete
users_addresses_show	Show a user's address	Complete
users_addresses_create	Create an address for a user	Complete
users_addresses_update	Update a user's address	Complete
users_addresses_destroy	Delete a user's address	Complete

users_admin_privileges_index	Show the admin privileges for a user	Complete
users_admin_privileges_create	Add an admin privilege to a user	Complete
users_barcodes_index	List the barcodes for a user	Complete
users_barcodes_create	Add a barcode to a user	
users_barcodes_destroy	Delete a barcode from a user	Complete
users_invitations_index	Get a list of invitations for a user	Complete
users_invitations_create	Create an invitation for a user to join a group	Tested, creates endpoint request per spec, 500 error with no custom message, signature errors with it.
users_family_index	Get a user's family info	Complete
users_family_create	Add a user to a family	Complete
users_family_update	Change the family role for a user	Complete
users_family_destroy	Remove a user from a family	Tested, creates endpoint request per spec, signature error
users_notes_index	Get the index of notes for a user	Complete
users_notes_show	Get a note for a user	Complete
users_notes_create	Create a note for a user	Complete
users_processes_index	Get the list of processes and their state for a user	Complete
users_processes_show	Get a given process info for a user	Complete
users_processes_answers_index	Get the list of answers for a given process for a user	Complete
users_processes_notes_index	Get a list of notes associated with a given process instance for a user	Complete
users_roles_index	Get the groups a user is in, including role in group	Complete

users_roles_show	Get a user's role in a given group	Works, need support for parameters
users_roles_create	Queue adding a role in a group for a user	Tested 404 not found
users_roles_destroy	Delete of a role in a group for a user	In progress
users_roles_activate		Stub
users_roles_deactivate		Stub
users_roles_promote		Stub
users_roles_demote		Stub
users_skills_index	Get the list of skills for a user	Complete
users_skills_create	Add a skill to a user	Complete
users_skills_destroy	Delete a skill from a user	Complete
<i>Groups</i>		
groups_index	Get a list of groups	Complete
groups_show	Show info for a group	Complete
groups_create	Create a group	Complete
groups_update	Update info for a group	Complete
groups_count	Get the number of groups in your City	Complete
groups_addresses_index	Get the list of addresses for a group	Complete
groups_addresses_show	Show a selected address of a group	Complete
groups_addresses_create	Add an address for a group	Complete
groups_addresses_update	Change a group address	Complete
groups_addresses_destroy	Delete a group address	Complete
groups_exports_index	Get the list of exports for a group with export status	Complete
groups_exports_show	Show info for a group export	Complete
groups_exports_create	Create a group export	Complete

groups_invitations_index	Get a list of invitations to a group	Complete
groups_invitations_create	Create an invitation to a group	Complete
groups_roles_index	Get the list of group members, can select by role and active status	Complete
groups_roles_show	Show role info for a given role in a group	Complete
groups_roles_create	Create a group role	Complete
groups_roles_destroy	Delete a group role	In progress
groups_roles_activate	Activate a group role	In progress
groups_roles_deactivate	Deactivate a group role	In progress
groups_roles_promote	Promote a group role	In progress
groups_roles_demote	Demote a group role	In progress
groups_roles_deactivate_all	Deactivate all group roles	In progress
groups_tags_index	List tags for a group	Complete
groups_tags_create	Add an existing tag to a group	Complete
groups_tags_destroy	Remove a tag from a group	Tested, issues
<i>Addresses</i>		
addresses_index	Get a list of addresses	Complete
addresses_show	Show an address	Complete
addresses_groups	Show addresses filtered by group type and parent	Complete
<i>Barcodes</i>		
barcodes_show	Show the user associated with a given barcode	Complete
<i>Families</i>		
families_show	Show a family	In progress
families_create	Create a family	Stub

families_update	Update a family	Stub
families_destroy	Delete a family	Stub
<i>Invitations</i>		
invitations_index	List invitations	Stub
invitations_bulk_invite	Do a bulk invitation	Stub
<i>Metrics</i>		
metrics_index	Show the list of metrics	Stub
metrics_show	Show a single metric	Stub
metrics_create	Crate a metric	Stub
metrics_measurements_index	Show the list of measurements	Stub
metrics_measurements_show	Show a single measurement	Stub
metrics_measurements_create	Create a measurement	Stub
metrics_measurements_destroy	Delete a measurement	Stub
metrics_measurements_last	Get the last measurement	Stub
<i>Web Hooks</i>		
web_hooks_index	List the currently active web hooks	Complete
web_hooks_create	Create a web hook	Complete
web_hooks_destroy	Delete a web hook	Complete
<i>Lower Level</i>		
call_city	<p>Call an arbitrary API endpoint by specifying the verb, URL (with querystring if applicable), and JSON body (if applicable). This could be useful to call new endpoints that aren't directly supported by this library yet.</p> <p>If you call this function, be prepared to handle exceptions.</p> <p>This may go away in later versions.</p>	Complete

Function Descriptions

set_key

Description

set_key (string \$apikey)

Sets the API key on the object

Parameters

apikey The API key generated on the City

Return Values

None

Changelog

Examples

```
$ca = new CityApi();  
$ca->set_key('6b079e7ccf332c5f95a7928854b342d3a637a69a'); // your API key
```

set_token

Description

set_token (string \$apitoken)

Sets the user token on the object

Parameters

string \$apitoken The user token generated on the City

Return Values

None

Changelog

Examples

```
$ca = new CityApi();  
$ca->set_token('5ba04202ae1fc35b'); // your user token
```

get_last_headers

Description

array get_last_headers ()

Gets the HTTP response headers from the last endpoint call.

Parameters

None

Return Values

An associative array containing the headers.

Changelog

Examples

```
$ca = new CityApi();

$results = $ca->groups_count(); // headers are NULL until after the first endpoint call
echo "\n<strong>group_count() results</strong>\n";
var_dump($results);

echo "\n<strong>last_headers</strong>\n";
var_dump($ca->get_last_headers());
```

get_last_response_start_line

Description

string get_last_response_start_line ()

Gets the HTTP response start line from the last endpoint call.

Parameters

None

Return Values

A string containing the last response start line.

Changelog

Examples

```
$ca = new CityApi();

$results = $ca->groups_count(); // headers are NULL until after the first endpoint call
echo "\n<strong>group_count() results</strong>\n";
var_dump($results);

echo "\n<strong>last_response_start_line</strong>\n";
var_dump($ca->get_last_response_start_line());
```

get_last_status_code

Description

int get_last_status_code ()

Gets the HTTP status code from the last endpoint call.

Parameters

None

Return Values

The last HTTP status code.

Changelog

Examples

get_ratelimit_limit_by_account

Description

`get_ratelimit_limit_by_account ()`

Gets the number of endpoint calls your account is allowed per hour.

Parameters

None

Return Values

The rate limit for your account.

Changelog

Examples

```
$ca = new CityApi();  
$results = $ca->get_ratelimit_limit_by_account();
```

get_ratelimit_limit_by_ip

Description

`get_ratelimit_limit_by_ip ()`

Gets the number of endpoint calls your IP address is allowed per hour.

Parameters

None

Return Values

The rate limit for your ip.

Changelog

Examples

```
$ca = new CityApi();  
$results = $ca->get_ratelimit_limit_by_ip();
```

get_ratelimit_remaining_by_account

Description

get_ratelimit_remaining_by_account ()

Gets the number of endpoint calls your account has left in the current clock hour. Resets at the top of the hour.

Parameters

None

Return Values

The number of endpoint calls you have left for the current hour.

Changelog

Examples

```
$ca = new CityApi();  
$results = $ca->get_ratelimit_remaining_by_account();
```

get_ratelimit_remaining_by_ip

Description

get_ratelimit_remaining_by_ip ()

Gets the number of endpoint calls your IP has left in the current clock hour. Resets at the top of the hour.

Parameters

None

Return Values

The number of endpoint calls your IP has left for the current hour.

Changelog

Examples

```
$ca = new CityApi();  
$results = $ca->get_ratelimit_remaining_by_ip();
```

users_index

Description

mixed users_index (array \$args = NULL)

Get a list of users info

Parameters

args Optional associative array of parameters

See the City API spec for now, will update when stabilized

Return Values

User info for the users that satisfy the selection criteria. The format is a PHP array or JSON depending on \$json.

Changelog

Examples

See test/test-users-index.php

users_show

Description

mixed users_show (mixed \$userid = NULL, array \$args = NULL)

Show info for a given user

Parameters

userid The id of the user to show (required)

args Optional associative array of parameters (not implemented yet)

Return Values

The user info. The format is a PHP array or JSON depending on \$json.

Changelog

Examples

See test/test-users-show.php

users_create

Description

mixed users_create (array \$args = NULL)

Create an offline user

Parameters

args Required associative array of field values

See the City API spec for now, will update when stabilized

Return Values

The newly created user info. The format is a PHP array or JSON depending on \$json.

Changelog

Examples

See test/test-users-create.php

users_update

Description

mixed users_update (mixed \$userid = NULL, array \$args = NULL)

Update info for a user

Parameters

userid The id of the user to update (required)

args Associative array of field names and values you want to update

See the City API spec for now, will update when stabilized

Return Values

The updated user info. The format is a PHP array or JSON depending on \$json.

Changelog

Examples

See test/test-users-update.php

users_destroy

Description

users_destroy (mixed \$userid = NULL)

Delete a user

Parameters

userid The id of the user to delete (required)

Return Values

None

The HTTP status code of a successful delete will be 204

Changelog

Examples

See test/test-users-destroy.php

users_count

Description

mixed users_count (string \$filter = NULL)

Get the number of users that meet selected criteria

Parameters

filter The selection filter to apply (optional)

args Associative array of the count of users and filter applied and values you want to update

See the City API spec for now, will update when stabilized

Return Values

The count of users and the filter you used. The format is a PHP array or JSON depending on \$json.

Changelog

Examples

See test/test-users-count.php

users_memberize

Description

mixed users_memberize (mixed \$userid = NULL, array \$args = NULL)

Memberize a user

Parameters

userid The id of the user to memberize (required)

args Optional associative array specifying member-since date in either of two formats:

array('member_since' => '15/02/2012') DD/MM/YYYY

array('member_since' => '2012-02-15') YYYY-MM-DD

Return Values

The user info for the user you memberized. The format is a PHP array or JSON depending on \$json.

Changelog

Examples

See test/test-users-memberize.php

users_dememberize

Description

mixed users_dememberize (mixed \$userid = NULL)

Memberize a user

Parameters

userid The id of the user to dememberize (required)

Return Values

The user info for the user you dememberized. The format is a PHP array or JSON depending on \$json.

Changelog

Examples

See test/test-users-dememberize.php

users_deactivate

Description

mixed users_deactivate (mixed \$userid = NULL, array \$args = NULL)

Deactivate an online user

Parameters

userid The id of the user to deactivate (required)

args Optional associative array specifying the reason:

array('reason' => 'This is the reason')

Return Values

The user info for the user modified, same as users_show(). The format is a PHP array or JSON depending on \$json.

If called on an offline user, The City returns "404 Not Found".

Changelog

Examples

See test/test-users-deactivate.php

users_bulk_memberize

Description

mixed users_bulk_memberize (array \$args = NULL)

Memberize a list of users

Parameters

args Associative array specifying parameters (required):

```
array( 'member_since' => '2002-01-16', // this is the default member_since value
      'members'       => array( array('user_id' => 225262, 'member_since' => '1954-12-01'),
                               array('user_id' => 238801) ) );
```

The user can be identified by any of user_id, email, or external_id_1.

Return Values

The HTTP status code of a successful memberize will be 204

Changelog

Examples

See test/test-users-bulk-memberize.php

users_bulk_deactivate

Description

mixed users_bulk_dememberize (array \$args = NULL)

Deactivate a list of users

Parameters

args Associative array specifying parameters (required):

```
array( 'reason' => 'This is the default reason.',
      'users'   => array( array('user_id' => 225262, 'reason' => 'Extreme coffee breath'),
                          array('user_id' => 238801) ) );
```

The user can be identified by any of user_id, email, or external_id_1.

Return Values

The HTTP status code of a successful deactivate will be 204

Changelog

Examples

See test/test-users-bulk-deactivate.php

users_addresses_index

Description

mixed users_addresses_index (mixed \$userid = NULL, array \$args = NULL)

Get the list of addresses for a user

Parameters

userid The id of the user whose addresses you want to retrieve (required)

args Optional associative array of arguments

array('page' => 1)

Return Values

The user's addresses. The format is a PHP array or JSON depending on \$json.

Changelog

Examples

See test/test-users-addresses-index.php

users_addresses_show

Description

mixed users_addresses_show (mixed \$userid = NULL, mixed \$addressid = NULL)

Show a single selected address for a user

Parameters

userid The id of the user whose addresses you want to retrieve (required)

addressid The id of the address you want (required)

Return Values

The selected address. The format is a PHP array or JSON depending on \$json.

Changelog

Examples

See test/test-users-addresses-show.php

users_addresses_create

Description

mixed users_addresses_create (mixed \$userid = NULL, array \$args = NULL)

Create an address for a user

Parameters

userid The id of the user the address is for (required).

args Required associative array of field values, like:

```
array('street'=>'123 Any Way', 'city'=>'Fulton', 'state'=>'MD', 'zipcode'=>'20759', 'location_type'=>'Home' )
```

Return Values

The user's addresses you just created. The format is a PHP array or JSON depending on \$json.

Changelog

Examples

See test/test-users-addresses-create.php

users_addresses_update

Description

mixed users_addresses_update (mixed \$userid = NULL, mixed \$addressid = NULL, array \$args = NULL)

Update an address for a user

Parameters

userid The id of the address you want to update (required).

args An associative array of field values you want to update, like:

```
array('location_type'=>'Work' )
```

Return Values

The updated user's addresses. The format is a PHP array or JSON depending on \$json.

Changelog

Examples

See test/test-users-addresses-update.php

users_addresses_destroy

Description

mixed users_addresses_update (mixed \$userid = NULL, mixed \$addressid = NULL)

Delete an address for a user

Parameters

userid The id of the user the address is for (required).

userid The id of the address you want to update (required).

Return Values

The remaining addresses for the user. The format is a PHP array or JSON depending on \$json.

Changelog

Examples

See test/test-users-addresses-destroy.php

users_admin_privileges_index

Description

mixed users_admin_privileges_index (mixed \$userid = NULL)

Show the admin privileges for a user

Parameters

userid The id of the user of interest (required).

Return Values

The admin privileges for the user. The format is a PHP array or JSON depending on \$json.

Changelog

Examples

See test/test-users-admin-privileges-index.php

users_admin_privileges_create

Description

mixed users_admin_privileges_create (mixed \$userid = NULL, \$args = NULL)

Add an admin privilege to a user

Parameters

userid The id of the user you're adding the privilege to (required).

args An associative array specifying the privilege title (required), like:

```
array('title'=>'Reporting User' )
```

Return Values

The admin privileges for the user. The format is a PHP array or JSON depending on \$json.

Changelog

Examples

See test/test-users-admin-privileges-create.php

users_barcodes_index

Description

mixed users_barcodes_index (mixed \$userid = NULL, \$args = NULL)

List the barcodes for a user

Parameters

userid The id of the user whose barcodes you want (required).

args Optional associative array of arguments

```
array( 'page' => 1)
```

Return Values

The user's barcode info. The format is a PHP array or JSON depending on \$json.

Changelog

Examples

See test/test-users-barcodes-index.php

users_barcodes_show

Description

mixed users_barcodes_index (mixed \$userid = NULL, \$barcode = NULL)

Show a barcode for a user

Parameters

userid The id of the user whose barcode you want (required).

Barcode The barcode or barcode id of the barcode to retrieve (required).

Return Values

The user's barcode info for the selected barcode. The format is a PHP array or JSON depending on \$json.

Changelog

Examples

See test/test-users-barcodes-show.php

users_barcodes_create

Description

mixed users_barcodes_index (mixed \$userid = NULL, \$barcode = NULL)

Show a barcode for a user

Parameters

userid The id of the user you're adding a barcode to (required).

barcode The barcode to add (required).

Return Values

The user's barcode info. The format is a PHP array or JSON depending on \$json.

Changelog

Examples

See test/test-users-barcodes-create.php

users_barcodes_destroy

Description

mixed users_barcodes_destroy (mixed \$userid = NULL, \$barcode = NULL)

Delete a barcode from a user

Parameters

userid The id of the user whose barcode you are deleting (required).

Barcode The barcode or barcode id of the barcode to delete (required).

Return Values

An object containing the user's barcode info. The format is a PHP array or JSON depending on \$json.

Changelog

Examples

See test/test-users-barcodes-destroy.php

users_invitations_index

Description

mixed users_invitations_index (mixed \$userid = NULL, array \$args = NULL)

Get a list of invitations for a user

Parameters

userid The id of the user whose invitations you want to retrieve (required)

args Optional associative array of arguments

```
array('page' => 1)
```

Return Values

The user's invitations. The format is a PHP array or JSON depending on \$json.

Changelog

Examples

See test/test-users-invitations-index.php

users_invitations_create

Description

mixed users_invitations_create (mixed \$userid = NULL, mixed \$groupid, array \$args = NULL)

Get the list of addresses for a user

Still debugging, there is a City-side bug that is preventing landing. Parameters may change.

Parameters

userid The id of the user who you want to invite (required)

groupid The id of the group you're inviting to (required)

args Optional associative array of arguments

```
$args = array( 'custom_message' => 'Join our awesome group!');
```

Return Values

The invitation info. The format is a PHP array or JSON depending on \$json.

Changelog

Examples

See test/test-users-invitations-create.php

users_family_index

Description

mixed users_invitations_index (mixed \$userid = NULL)

Get a user's family info

Parameters

userid The id of the user whose family info you want to retrieve (required)

Return Values

The user's family info. The format is a PHP array or JSON depending on \$json.

Changelog

Examples

See test/test-users-family-index.php

users_family_create

Description

mixed users_family_create (mixed \$userid = NULL, array \$args = NULL)

Add a user to a family

Parameters

userid The id of the user you're adding to the family (required)

args Optional associative array of arguments (required)

Must contain family_role and either family_id or family_external_id

```
$args = array ( 'family_id' => $familyid, 'family_role' => $role );
```

Return Values

The family info. The format is a PHP array or JSON depending on \$json.

Changelog

Examples

See test/test-users-family-create.php

users_family_update

Description

mixed users_family_update (mixed \$userid = NULL, array \$args = NULL)

Change the family role for a user

Parameters

userid The id of the user whose family role you are changing (required)

args Optional associative array of arguments containing the new family role (required)

```
$args = array ( 'family_role' => $role);
```

Return Values

The updated family info. The format is a PHP array or JSON depending on \$json.

Changelog

Examples

See test/test-users-family-update.php

users_family_destroy

Description

mixed users_family_update (mixed \$userid = NULL)

Remove a user from a family

Parameters

userid The id of the user you are removing from the family (required)

Return Values

An empty object and HTTP status code 200

Changelog

Examples

See test/test-users-family-destroy.php

users_notes_index

Description

mixed users_notes_index (mixed \$userid = NULL, array \$args = NULL)

Get the list of notes for a user

Parameters

userid The id of the user whose note index you want to retrieve (required)

args Optional associative array of arguments

```
array('page' => 1)
```

Return Values

The user's notes info. The format is a PHP array or JSON depending on \$json.

Changelog

Examples

See test/test-users-notes-index.php

users_notes_show

Description

mixed users_notes_show (mixed \$userid = NULL, mixed \$noteid)

Get info for a given note for a user

Parameters

userid The id of the user whose note index you want to retrieve (required)

noteid The id of the note of interest (required)

Return Values

The note info. The format is a PHP array or JSON depending on \$json.

Changelog

Examples

See test/test-users-notes-show.php

users_notes_create

Description

mixed users_notes_create (mixed \$userid = NULL, array \$args = NULL)

Create a note for a user

Parameters

userid The id of the user you're adding to the family (required)

args Optional associative array of arguments (required)

Must contain body and visible_to. visible_to can be a single role or an array of roles the note will be visible to.

```
$args = array(
    'body' => 'Larry is a fantastic guitar player.',
    'visible_to' => array('Designer', 'User Admin')
);
```

Return Values

The note info. The format is a PHP array or JSON depending on \$json.

Changelog

Examples

See test/test-users-notes-create.php

users_processes_index

Description

mixed users_processes_index (mixed \$userid = NULL, array \$args = NULL)

Get the list of processes and their state for a user

Parameters

userid The id of the user whose processes you want to retrieve (required)

args Optional associative array of arguments

```
array('page' => 1)
```

Return Values

The user's processes info. The format is a PHP array or JSON depending on \$json.

Changelog

Examples

See test/test-users-processes-index.php

users_processes_show

Description

mixed users_processes_index (mixed \$userid = NULL, mixed \$processid = NULL)

Get info for a given process info for a user

Parameters

userid The id of the user whose process info you want (required).

processid The process id of the process info you want to retrieve (required).

Return Values

The user's process info for the selected process. The format is a PHP array or JSON depending on \$json.

Changelog

Examples

See test/test-users-processes-show.php

users_processes_answers_index

Description

mixed users_processes_answers_index (mixed \$userid = NULL, mixed \$processid = NULL, array \$args = NULL)

Get the list of answers for a given process for a user

Parameters

userid	The id of the user whose processes you want to retrieve (required)
processid	The process id of the process info you want to retrieve (required).
args	Optional associative array of arguments

`array('page' => 1)`

Return Values

The user's answers for the process. The format is a PHP array or JSON depending on \$json.

Changelog

Examples

See test/test-users-processes-answers-index.php

users_processes_notes_index

Description

mixed users_processes_notes_index (mixed \$userid = NULL, mixed \$processid = NULL, array \$args = NULL)

Get a list of notes associated with a given process instance for a user

Parameters

userid	The id of the user whose process notes you want to retrieve (required)
processid	The process id of the process of interest (required).
args	Optional associative array of arguments

`array('page' => 1)`

Return Values

The user's notes for the process. The format is a PHP array or JSON depending on \$json.

Changelog

Examples

See test/test-users-processes-notes-index.php

users_roles_index

Description

mixed users_roles_index (mixed \$userid = NULL, array \$args = NULL)

Get the groups a user is in, includes role in group

Parameters

userid The id of the user whose roles you want to retrieve (required)

args Optional associative array of arguments

```
array('page' => 1)
```

Return Values

The user's roles. The format is a PHP array or JSON depending on \$json.

Changelog

Examples

See test/test-users-roles-index.php

users_roles_show

Description

mixed users_roles_show (mixed \$userid = NULL, mixed \$roleid = NULL)

Get a user's role info for a given group role

Parameters

userid The id of the user whose roles you want to retrieve (required)

roleid The id of the role of interest (required)

Return Values

The role info. The format is a PHP array or JSON depending on \$json.

Changelog

Examples

See test/test-users-roles-show.php

users_roles_create

mixed users_roles_create (mixed \$userid = NULL, array \$args = NULL)

Queue adding a role in a group for a user

Is this only for offline users?

Parameters

userid The id of the user you're creating a role for (required)

args Associative array of arguments, include group_id and title (role title)

```
$args = array('group_id' => 238801, 'title' => 'Manager');
```

Return Values

The invitation info. The format is a PHP array or JSON depending on \$json.

Changelog

Examples

See test/test-users-roles-create.php

users_roles_destroy

In progress

users_roles_activate

In progress

users_roles_deactivate

In progress

users_roles_promote

In progress

users_roles_demote

In progress

users_skills_index

Description

mixed users_skills_index (mixed \$userid = NULL, array \$args = NULL)

Get a list of a user's skills

Parameters

userid The id of the user whose skills you want to retrieve (required)

args Optional associative array of arguments

```
array('page' => 1)
```

Return Values

The user's skills. The format is a PHP array or JSON depending on \$json.

Changelog

Examples

See test/test-users-skills-index.php

users_skills_create

Description

mixed users_skills_create (mixed \$userid = NULL, array \$args = NULL)

Add a skill to a user

Parameters

userid The id of the user you're adding to (required)

args Associative array of arguments, either name or skill_id is required.

```
array('name' => 'Computer-WordPress')
```

Return Values

The user's skills. The format is a PHP array or JSON depending on \$json.

Changelog

Examples

See test/test-users-skills-create.php

users_skills_destroy

Description

mixed users_skills_destroy (mixed \$userid = NULL, mixed \$skill = NULL)

Add a skill to a user

Parameters

userid The id of the user you're adding to (required)

skill The skill name or skill id you're deleting (required).

Return Values

The user's skills remaining after the delete. The format is a PHP array or JSON depending on \$json.

Changelog

Examples

See test/test-users-skills-destroy.php

groups_index

Description

mixed groups_index (array \$args = NULL)

Get a list of groups

Parameters

args Optional associative array of arguments

```
array('search' => 'Test', 'page' => 1)
```

Return Values

The groups that match. The format is a PHP array or JSON depending on \$json.

Changelog

Examples

See test/test-groups-index.php

groups_show

Description

mixed groups_show (mixed \$groupid = NULL)

Show info for a group

Parameters

groupid The id of the group whose info you want to retrieve (required)

Return Values

The group info. The format is a PHP array or JSON depending on \$json.

Changelog

Examples

See test/test-groups-show.php

groups_create

Description

mixed groups_create (array \$args = NULL)

Create a group

Parameters

args An associative array of arguments providing the group info.

name, parent_id, and group_type are required

```
$args = array(
    'name' => 'Test Service Group 3',
    'parent_id' => '43681',
    'group_type' => 'Service',
    'description' => 'A great group to serve in!',
    'hide_albums' => 'true',
    'target_size' => '20',
    'secure' => 'true',
);
```

Return Values

The new group info, like groups_show(). The format is a PHP array or JSON depending on \$json.

Changelog

Examples

See test/test-groups-create.php

groups_update

Description

mixed groups_update (mixed \$groupid = NULL, array \$args = NULL)

Create a group

Parameters

groupid The id of the group you're updating (required)

args An associative array of arguments providing the changed info.

```
$args = array(
    'name' => 'Test Service Group 3',
    'hide_albums' => 'false',
    'auto_approve_invites' => 'true',
);
```

Return Values

The new group info, like groups_show(). The format is a PHP array or JSON depending on \$json.

Changelog

Examples

See test/test-groups-create.php

groups_count

Description

mixed groups_count ()

Get the number of groups in your City

Parameters

None

Return Values

The count of groups in your City. The format is a PHP array or JSON depending on \$json.

Changelog

Examples

See test/test-groups-count.php

groups_addresses_index

Description

mixed groups_addresses_index (mixed \$groupid = NULL, array \$args = NULL)

Get the list of addresses for a group

Parameters

groupid The id of the group of interest (required)

args Optional associative array of arguments

`array('page' => 1)`

Return Values

The addresses of the group. The format is a PHP array or JSON depending on \$json.

Changelog

Examples

See test/test-groups-addresses-index.php

groups_addresses_show

Description

mixed groups_addresses_show (mixed \$groupid = NULL, mixed \$addressid = NULL)

Show a selected address of a group

Parameters

groupid The id of the group of interest (required)

addressid The id of the address of interest (required)

Return Values

The address info. The format is a PHP array or JSON depending on \$json.

Changelog

Examples

See test/test-groups-addresses-show.php

groups_addresses_create

Description

mixed groups_addresses_create (mixed \$groupid = NULL, array \$args = NULL)

Add an address to a group

Parameters

groupid The id of the group of interest (required)

args An associative array of arguments providing the address info.

```
$args = array(
    'street'=>'123 Any Way',
    'city'=>'Fulton',
    'state'=>'MD',
    'zipcode'=>'20759',
    'location_type'=>'Home',
);
```

Return Values

The group's addresses, like groups_addresses_index(). The format is a PHP array or JSON depending on \$json.

Changelog

Examples

See test/test-groups-addresses-create.php

groups_addresses_update

Description

mixed groups_addresses_update (mixed \$groupid = NULL, mixed \$addressid = NULL, array \$args = NULL)

Change an address of a group

Parameters

- | | |
|-----------|---|
| groupid | The id of the group of interest (required) |
| addressid | The id of the address you're changing (required) |
| args | An associative array of arguments providing the updated address info. |

```
$args = array(
    'street'=>'123 Any Way',
    'city'=>'Fulton',
    'state'=>'MD',
    'zipcode'=>'20759',
    'location_type'=>'Home',
);
```

Return Values

The updated address. The format is a PHP array or JSON depending on \$json.

Changelog

Examples

See test/test-groups-addresses-update.php

groups_addresses_destroy

Description

mixed groups_addresses_destroy (mixed \$groupid = NULL, mixed \$addressid)

Remove an address from a group

Parameters

- | | |
|-----------|--|
| groupid | The id of the group of interest (required) |
| addressid | The id of the address you're deleting (required) |

Return Values

The remaining group addresses, like groups_addresses_index(). The format is a PHP array or JSON depending on \$json.

Changelog

Examples

See test/test-groups-addresses-destroy.php

groups_exports_index

Description

mixed groups_exports_index (mixed \$groupid = NULL, array \$args = NULL)

Get the list of exports for a group with export status

Parameters

groupid The id of the group of interest (required)

args Optional associative array of arguments

```
array('page' => 1)
```

Return Values

The exports for the group. The format is a PHP array or JSON depending on \$json.

Changelog

Examples

See test/test-groups-exports-index.php

groups_exports_show

Description

mixed groups_exports_show (mixed \$groupid = NULL, mixed \$exportsid = NULL)

Show info for a selected export of a group

Parameters

groupid The id of the group of interest (required)

exportsid The id of the export of interest (required)

Return Values

The export info, including state The format is a PHP array or JSON depending on \$json.

Changelog

Examples

See test/test-groups-exports-show.php

groups_exports_create

Description

mixed groups_exports_create (mixed \$groupid = NULL, mixed \$user_type = NULL)

Show info for a selected export of a group

Parameters

groupid The id of the group of interest (required)

user_type The user_type you are exporting, "OfflineUser" or "User" (required)

Return Values

The export info, including state. The format is a PHP array or JSON depending on \$json.

Changelog

Examples

See test/test-groups-exports-create.php

groups_invitations_index

Description

mixed groups_invitations_index (mixed \$groupid = NULL, array \$args = NULL)

Get the list of exports for a group

Parameters

groupid The id of the group of interest (required)

args Optional associative array of arguments

```
array('page' => 1)
```

Return Values

The exports for the group. The format is a PHP array or JSON depending on \$json.

Changelog

Examples

See test/test-groups-invitations-index.php

groups_invitations_create

Description

mixed groups_invitations_create (mixed \$groupid, array \$args = NULL)

Invite someone to a group

Parameters

groupid The id of the group you're inviting to (required)

args An associative array of arguments providing the group info.

At least one of user_id, external_id, or email is required. custom_message is optional.

```
$args = array(
    'userid' => 238801,
    'custom_message' => 'Please join our group!',
);
```

Return Values

The new invitation info. The format is a PHP array or JSON depending on \$json.

Changelog

Examples

See test/test-groups-invitations-create.php

groups_roles_index

Description

mixed groups_roles_index (mixed \$groupid, array \$args = NULL)

Get the list of group members, can select by role and active status

Parameters

groupid The id of the group of interest (required)

args An optional associative array of selection arguments. You can filter on page, title, and include_inactive.

```
$args = array(
    'page' => 1,
    'title' => 'Leaders',
    'include_inactive' => 'false',
);
```

Return Values

The roles satisfying the selection criteria. The format is a PHP array or JSON depending on \$json.

Changelog

Examples

See test/test-groups-roles-index.php

groups_roles_show

Description

mixed groups_roles_show (mixed \$groupid, mixed \$roleid = NULL)

Show role info for a selected group and role

Parameters

groupid The id of the group of interest (required)

roleid The id of the role you want to show (required)

Return Values

The role info. The format is a PHP array or JSON depending on \$json.

Currently The City is returning an error 500.

Changelog

Examples

See test/test-groups-roles-show.php

groups_roles_create

Description

mixed groups_roles_create (mixed \$groupid, array \$args = NULL)

Create a group role

Parameters

groupid The id of the group of interest (required)

args An associative array of arguments providing the role info.

 user_id is required.

```
$args = array(  
    'userid' => 238801,  
    'title' => 'Manager',  
);
```

Return Values

An empty object and HTTP status code 204.

Changelog

Examples

See test/test-groups-roles-create.php

groups_roles_delete

In progress

groups_roles_activate

In progress

groups_roles_deactivate

In progress

groups_roles_promote

In progress

groups_roles_demote

In progress

groups_roles_deactivate_all

In progress

groups_tags_index

Description

mixed groups_tags_index (mixed \$groupid)

List tags for a group

Parameters

groupid The id of the group of interest (required)

Return Values

The group's tag list. The format is a PHP array or JSON depending on \$json.

Changelog

Examples

See test/test-groups-tags-index.php

groups_tags_create

Description

mixed groups_tags_create (mixed \$groupid, array \$args = NULL)

Add an existing tag to a group

Parameters

groupid The id of the group of interest (required)

args An associative array of arguments providing the tag info.

Either name or tag_id is required. The tag must already exist in your City's tag list.

```
$args = array( 'name' => 'College Age' );
```

Return Values

The group's tag list. The format is a PHP array or JSON depending on \$json.

Changelog

Examples

See test/test-groups-tags-create.php

groups_tags_destroy

Description

mixed groups_tags_destroy (mixed \$groupid, array \$args = NULL)

Delete a tag from a group

Parameters

groupid The id of the group of interest (required)

tag The name or tag_id of the tag you're deleting (required)

Return Values

The group's remaining tag list. The format is a PHP array or JSON depending on \$json.

Changelog

Examples

See test/test-groups-tags-destroy.php

addresses_index

Description

mixed addresses_index ()

Get group addresses in your City

Parameters

None

Return Values

The addresses. The format is a PHP array or JSON depending on \$json.

Changelog

Examples

See test/test-addresses-index.php

addresses_show

Description

mixed addresses_show (mixed \$addressid)

Show role info for a selected group and role

Parameters

addressid The id of the address you want to show (required)

Return Values

The address info. The format is a PHP array or JSON depending on \$json.

Changelog

Examples

See test/test-addresses-show.php

addresses_groups

Description

mixed addresses-groups (array \$args = NULL)

Show addresses filtered by group type and parent

Parameters

args An optional associative array of selection arguments.

You can filter on page, group_types, and under_group_id.

```
$args = array(
    'page' => 1,
    'group_types' => 'Service',
);
```

Return Values

The address groups satisfying the selection criteria. The format is a PHP array or JSON depending on \$json.

Changelog

Examples

See test/test-addresses-groups.php

barcodes_show

Description

mixed barcodes_show (mixed \$barcode)

Show the user associated with a given barcode

Parameters

barcode The barcode or barcode_id you want to find the user for (required)

Return Values

The barcode info. The format is a PHP array or JSON depending on \$json.

Changelog

Examples

See test/test-barcodes-show.php

families_show

In progress

families_create

In progress

families_update

In progress

families_destroy

In progress

invitations_index

In progress

invitations_bulk_invite

In progress

metrics_index

In progress

metrics_show

In progress

metrics_create

In progress

metrics_measurements_index

In progress

metrics_measurements_show

In progress

metrics_measurements_create

In progress

metrics_measurements_destroy

In progress

metrics_measurements_last

In progress

web_hooks_index

Description

mixed webhooks_index (\$args = NULL)

List the currently active web hooks

Parameters

args An optional associative array of selection arguments.

You can filter on page.

```
$args = array('page' => 1);
```

Return Values

The currently active web hooks. The format is a PHP array or JSON depending on \$json.

Changelog

Examples

See test/test-web-hooks-index.php

web_hooks_create

Description

mixed web_hooks_create (array \$args = NULL)

Create a group role

Parameters

args An associative array of arguments providing the web hook info.

callback_uri, object, and event are required. For example:

```
$args = array(
    'callback_uri' => https://yoursite.org/wh-target.php,
    'object' => 'User',
    'event' => 'create',
);

callback_uri = ["https://valid.sslurl.com"]
object =      ["User" | "Group" | "Invitation" | "Privilege"]
event =      ["create" | "update" | "destroy" | "expire"]
```

Return Values

The currently active web hooks. The format is a PHP array or JSON depending on \$json.

Changelog

Examples

See test/test-web-hooks-create.php

web_hooks_destroy

mixed webhooks_destroy (mixed \$web_hook_id = NULL)

Delete a web hook

Parameters

web_hook_id The id of the web hook you want to delete (required).

Return Values

The remaining active web hooks. The format is a PHP array or JSON depending on \$json.

Changelog

Examples

See test/test-web-hooks-destroy.php

call_city

mixed call_city (string \$verb, string \$url, string \$body = NULL)

Call an arbitrary City Admin API endpoint by specifying the verb, URL (with querystring if applicable), and JSON body (if applicable). This could be useful to call new endpoints that aren't directly supported by this library yet.

If you call this function, be prepared to handle exceptions.

This may go away in later versions.

Parameters

verb GET, PUT, POST, DELETE

url endpoint URL, like <https://api.onthecity.org/groups/1234>

body JSON encoded body (optional, depends on API endpoint)

Return Values

The results from the endpoint API call. The format is a PHP array or JSON depending on \$json.

Changelog

Examples

See test/test-call-city.php