Ephpai

0.01

Generated by Doxygen 1.9.5

Chapter 1

Ephpai Class

1.1 ## A PHP Class to easily interact with the OpenAI API (ChatGPT & Dall-E)

it’s an unofficial class and This project has no commercial link with OpenAI.

In order to use it, you need to have an OpenAI account and valid api key from OpenAI :[Get api key from OpenAI] (`https://openai.com/api/`)

1.1.0.1 MIT License

1.1.1 Ephpai Class

This easy PHP Class interacts with the API provided by OpenAI.

It allows, in a limited but simple way, to make requests and obtain answers for the following functions :

- **Completion (text)**: Query, response and text generation. Support for "text-davinci-003" (cahtGPT), "text-curie-001", "text-babbage-001" and "text-ada-001" models. Possibility to define the maximum number of tokens. Fine temperature management. Can Export results as json, array or text.
- **Image Generation**: Easy switch from completion to image generation. Different image sizes, possibility of saving an image. Can display or save the image in jpeg format. Possibility to get an base64 encoded text or an url
- **Moderation**: Moderate contents and/or images through the API. Ability to automate moderation. Possibility to get the reasons of openAI moderation.

1.1.2 How to use it?

It's easy, just create an Ephpai object, execute the request, and retrieve the response.

In order to use the Ephpai class, the OAIPIKEY environment variable must contain your api key (recommended). If you don't have access to environment variables on your server, you can use the **setApikey($key)** method.

1.1.2.1 here is a simple example

```
<?php
require "Ephpai.php";
$Requestgpt=new Ephpai('Whois is spiderman?'); // Create object
if (!$Requestgpt->executeQuery())
     echo ($$Requestgpt->error());
else
echo "Result :".$Requestgpt->getTextresult(0);
?>
```

Here is a sample answer :

```
Result : Spider-Man is a Marvel Comics superhero created by Stan Lee and Steve Ditko in 1962. He is a
       costumed crime fighter with superhuman strength, agility, and the ability to cling to most surfaces.
       He uses his powers to protect the innocent and fight crime in New York City.
```

By default the request type is set to completion (text), and the model to text-davinci-003 (chatGPT). `About models`

The model can be modified with the method **setModel($model)**.

The number of tokens is fixed by default at 850, it can be changed with the method **setMaxtoken($number)**;

1.1.2.1.1 Modification of the previous query :

```
<?php require "Ephpai.php";
$Requestgpt=new Ephpai('Who is spiderman?'); // Create object
$Requestgpt->setModel('text-curie-001'); //model text-curie
$Requestgpt->setMaxtoken(200); //Maxtoken=200
if (!$Requestgpt->executeQuery())
     echo ($$Requestgpt->error());
else
echo "Result :".$Requestgpt->getTextresult(0);
?>
```

Sample Answer :

```
Result : Spiderman is a superhero who first appeared in the comic book series "Amazing Fantasy" in 1962. He
       is known for his webshooters, Spiderman suits, and his ability to climb walls.
```

There is a constructor with the model and maxtoken as parameters and allows us to make our task easier :

```
<?php require "Ephpai.php";
$Requestgpt=new Ephpai('Whois is spiderman?','text-curie-001',200); // Create object with model type
       'text-curie-001' and maxtoken=200
if (!$Requestgpt->executeQuery())
     echo ($$Requestgpt->error());
else
echo "Result :".$Requestgpt->getTextresult(0);
?>
```

The result can be retrieved as a json format text,an array or as a text.

There are several methods for modifying the query parameters, such as temperature management, or retrieving values...

You can find them in the documentation, available in the "doc" directory (HTML & pdf)

1.1.3 Image generation

To generate images using Dall-E and the OpenAI API, you need to enable image generation with the **generateImage(true)** method. To return to completion mode (text), just run this method with false as argument.

1.1.3.0.1 Example of image display using the GD library :

Note: pay attention to unnecessary spaces. In this example, we ask a "cat blue" image and display it. It s a simple example.

```
<?php
header("Content-type: image/jpeg");
require "../Ephpai.php";
$Requestgpt=new Ephpai('blue cat');
$Requestgpt->generateImage(true); //Activate image generation
if((!$Requestgpt->setImgsize('1024x1024'))||(!$Requestgpt->executeQuery()))
     die ($Requestgpt->error());
else
{
$imgdata=$Requestgpt->getTextresult(0);
$image = @imagecreatefromstring($imgdata);
imagejpeg($image,null, 75);
//echo $ImageData;
}
?>
```

If you don't want to bother with functions from the GD library, there is the **displayImg($nb,$quality)** method, which does all the work.

1.1.3.0.2 The same example with displayImg method

```
<?php
header("Content-type: image/jpeg");
require "../Ephpai.php";
$Requestgpt=new Ephpai('blue cat');
$Requestgpt->generateImage(true);//
if((!$Requestgpt->setImgsize('1024x1024'))||(!$Requestgpt->executeQuery()))
     die ($Requestgpt->error());
else
{
    if (!$Requestgpt->displayImg(0,100))
    die ($Requestgpt->error());
}
?>
```

In the same way, the **saveImgtojpeg($nb,$filename,$quality) allows to save image on your server in jpeg format.

```
<?php
require "../Ephpai.php";
$string='blue cat and pink mouse playing card';
$Requestgpt=new Ephpai($string);
$Requestgpt->generateImage(true);//
if((!$Requestgpt->setImgsize('1024x1024'))||(!$Requestgpt->executeQuery()))
     die ($Requestgpt->error());
else
{
    if (!$Requestgpt->saveImgtojpeg(0,urlencode($string).'.jpg',100))
    die ($Requestgpt->error());
    else
    echo 'Image '.urlencode($string).'.jpg saved with success';
}
?>
```

Image size: the OpenAI API allows a limited number of image sizes: 256x256,512x512 and 1024x1024.

For more information on the methods related to the generation of images, you will find more information in the documentation present in the doc directory.

1.1.4 Moderation

It is recommended to use the moderation API provided by OpenAI in order not to submit requests that could be against OpenAI rules, especially if yours users can submit content.

Without making the request, It is possible to check that the content is in compliance with moderation.

Note that the moderation API is valid for any type of text, whether for an image or text.

This is a functional but limited implementation of the functionalities of moderation.

If it is possible to know the reasons for a moderation in the form of an array, the respective scores of each reason (numerical values) are not returned.

1.1.4.0.1 Example of moderation api. This request should be moderated :

```
<?php
require "../Ephpai.php";
$Requestgpt=new Ephpai('I want to hit you. i hate you'); // Create object
//Asks to moderation api to evaluate query.
if ($Requestgpt-> ModerateQuery())
     //Depending the result, the moderation status will be set to true (moderated) or false (no moderated).
     if ($Requestgpt->moderation_status())
        echo "I'm moderated";
     else
          echo "I'm correct";
     else
          echo  $Requestgpt->error();
?>
```

This sentence will be moderated. "I'm moderated" will be displayed.

Note : If moderation works correctly for the English language, support for other languages is limited.

It is possible to know the reason(s) for the moderation in the form of an array. Replace the line in the example above

```
echo "I'm moderated";
```

by

```
{
         echo "I'm moderated. Reason(s) :";
         foreach($Requestgpt->moderated_categories() as $reason)
              echo $reason;
       }
```

Here is the result, this request will be moderated for violence :

```
I'm moderated. Reason(s) :violence
```

1.1.4.1 Auto moderation

It is possible to automatically moderate requests before sending them. The **Moderation_auto($status)** method enables this feature. If the query is moderated, the executeQuery() method will return **false** and moderation_status() will be **true**.

1.1.4.2 Example of auto moderation

```
<?php
require "../Ephpai.php";
$Requestgpt=new Ephpai('enter your bad words..'); // Create object
$Requestgpt->Moderation_auto(true);
if (!$Requestgpt->executeQuery())
{
     if($Requestgpt->moderation_status())
          echo "I m moderated";
     else echo ($$Requestgpt->error());
}
else
echo "Result :".$Requestgpt->getTextresult(0);
?>
```

∗∗Thanks for reading

I hope this code can be useful for you.

Don't hesitate to contact me to show me your creations :)

Thomas Missonier

Chapter 2

Data Structure Index

2.1 Data Structures

Here are the data structures with brief descriptions:

Chapter 3

Data Structure Documentation

3.1 Ephpai Class Reference

Public Member Functions

- __construct ()
- __construct1 ($query)
- __construct2 ($query, $model)
- __construct3 ($query, $model, $maxtoken)
- __destruct ()
- setQuery ($query)
- getQuery ()
- setTemperature ($temp)
- getTemperature ()
- increaseTemperature ($step)
- decreaseTemperature ($step)
- getModel ()
- setModel ($model)
- setApikey ($key)
- error ()
- setMaxtoken ($maxtoken)
- setNbresults ($nbresults)
- generateImage ($imgstatus)
- setImgsize ($size)
- getImgsize ()
- setImgtype ($imagetype)
- getImgtype ()
- displayImg ($nb, $quality)
- saveImgtojpeg ($nb, $filename, $quality)
- Moderation_auto ($choice)
- moderation_status ()
- moderated_categories ()
- ModerateQuery ()
- executeQuery ()
- getJsonresults ()
- getArrayresults ()
- getTextresult ($n)

3.1.1 Detailed Description

Ephpai Class

3.1.2 Constructor & Destructor Documentation

3.1.2.1 __construct()

```
__construct ( )
```

Constructor of Ephpai Class. Redirects to the sub-constructors depending on the number of arguments given.

Parameters

string	query
string	model [optionnal except if maxtoken is set]
number	maxtoken

See also

__construct1,__construct2,__construct3

3.1.2.2 __destruct()

```
__destruct ( )
```

Destructor of Ephpai

3.1.3 Member Function Documentation

3.1.3.1 __construct1()

```
__construct1 (
              $query )
```

Constructor with only Query as argument.

Parameters

string	query

See also

__construct()

3.1.3.2 __construct2()

```
__construct2 (
              $query,
              $model )
```

Constructor with query and model as arguments.

Parameters

string	*$query*	
string	*$model*	

$model needs to be in ['text-davinci-003','text-curie-001','text-babbage-001','text-ada-001'].

See also

https://platform.openai.com/docs/models/overview

3.1.3.3 __construct3()

```
__construct3 (
              $query,
              $model,
              $maxtoken )
```

Constructor with query,model and maxtoken as arguments.

Parameters

string	*$query*	
string	*$model*	
number	*$maxtoken*	

$model needs to be in ['text-davinci-003','text-curie-001','text-babbage-001','text-ada-001'].

See also

https://platform.openai.com/docs/models/overview

3.1.3.4 decreaseTemperature()

```
decreaseTemperature (
              $step )
```

Decreases temperature by step for completion. Note : Temperature is an important setting to control how much randommes is the result of a query

if (temperature-STEP)<0 temperature will be set to 0

Parameters

number	step

Returns

true of false;

3.1.3.5 displayImg()

```
displayImg (
              $nb,
              $quality )
```

Render image to screen (Jpeg format 100% quality)

Headers need to be set . ex :header("Content-type: image/jpeg");

Parameters

number	*$number*	
number	*$quality*	(between 0-100, or -1 to use the default IJG quality (about 75)

Returns

true or false

3.1.3.6 error()

```
error ( )
```

Returns error message.

Parameters

none	

Returns

string

3.1.3.7 executeQuery()

```
executeQuery ( )
```

Executes query. It sends query to Openai api using curl. Note : if auto Moderation is enabled, a query will be send to moderation before and will fail if moderated

See also

Moderation_auto($choice)

Parameters

none	

Returns

true or false

3.1.3.8 generateImage()

```
generateImage (
              $imgstatus )
```

Activates image generation. Set to true to perform image generation.

Parameters

true	or false

Returns

no return

3.1.3.9 getArrayresults()

```
getArrayresults ( )
```

Returns the answer to query from Openai api in array form.

Parameters

none	

Returns

string

3.1.3.10 getImgsize()

```
getImgsize ( )
```

Returns the image size.

Parameters

none	

Returns

string

3.1.3.11 getImgtype()

```
getImgtype ( )
```

Returns the type of image.

Parameters

none	

Returns

string

3.1.3.12 getJsonresults()

```
getJsonresults ( )
```

Returns the answer to query from Openai api in Json format.

Parameters

none	

Returns

string

3.1.3.13 getModel()

```
getModel ( )
```

Returns the model type.

Parameters

none	

Returns

string

3.1.3.14 getQuery()

```
getQuery ( )
```

Returns the query.

Parameters

None	

Returns

string containing the query

3.1.3.15 getTemperature()

```
getTemperature ( )
```

Returns temperature. Note : Temperature is an important setting to control how much randommes is the result of a query. It needs to be between 0 and 1.

See also

https://platform.openai.com/docs/api-reference/completions/create

Parameters

none	

Returns

temperature

3.1.3.16 getTextresult()

```
getTextresult (
              $n )
```

Returns the answer ($n) to query from Openai api in string format

Parameters

number	*$n*	

Returns

string

3.1.3.17 increaseTemperature()

```
increaseTemperature (
              $step )
```

Increases temperature by step for completion. Note: : Temperature is an important setting to control how much randommes is the result of a query

if (temperature+STEP)>1 temperature will be set to 1.

Parameters

number	step

Returns

true of false

3.1.3.18 moderated_categories()

```
moderated_categories ( )
```

Reasons of moderation. If a query has been moderated, an array containing the different reasons is returned.

It lacks moderation results by categories in order to keep it simple to use

Parameters

none	

Returns

array or false

3.1.3.19 ModerateQuery()

```
ModerateQuery ( )
```

Moderate the query. Allows to moderate the content before making a real request. If the query is moderated, the query is flagged.

Parameters

none	

Returns

true or false

3.1.3.20 Moderation_auto()

```
Moderation_auto (
              $choice )
```

Enable or disable auto moderation.

Parameters

bool	*$choice*	

Returns

no return

3.1.3.21 moderation_status()

```
moderation_status ( )
```

Returns moderation flag If a query has been moderated, returns true otherwise false

Parameters

none	

Returns

true or false

3.1.3.22 saveImgtojpeg()

```
saveImgtojpeg (
              $nb,
              $filename,
              $quality )
```

Save image to jpeg file

Parameters

number	number
string	filename (not empty)
number	quality (between 0-100, or -1 to use the default IJG quality (about 75)

3.1.3.23 setApikey()

```
setApikey (
              $key )
```

Defines the environment variable containing the API key.

Parameters

string	*$key*	

Returns

true or false

3.1.3.24 setImgsize()

```
setImgsize (
             $size )
```

Sets the image size

Allowed sizes :'256x256','512x512','1024x1024' generateImage(true) needs to be called before

Parameters

string	*$size*	

Returns

true or false

See also

generateImage

3.1.3.25 setImgtype()

```
setImgtype (
             $imagetype )
```

Sets the type of image. Image can be an url or a bs64-json encoded string.

Parameters

string	*$type*	

Returns

true or false

3.1.3.26 setMaxtoken()

```
setMaxtoken (
              $maxtoken )
```

Returns Maxtoken value.

Parameters

none	

Returns

number

function getMaxtoken() { return $this->maxtoken; }

/** Sets maxtoken value.

Parameters

number	*$maxtoken*	

Returns

true or false

3.1.3.27 setModel()

```
setModel (
              $model )
```

Set the model of image($img_type).

Parameters

string	*$model*	

Returns

true or false;

$model needs to be in ['text-davinci-003','text-curie-001','text-babbage-001','text-ada-001']

See also

https://platform.openai.com/docs/models/overview

3.1.3.28 setNbresults()

```
setNbresults (
              $nbresults )
```

Defines the desired number of results.

Parameters

number	*$results*	

Returns

true or false

3.1.3.29 setQuery()

```
setQuery (
              $query )
```

Sets the query.

Parameters

string	*$query*	(utf-8 encoded)

Returns

true of false

3.1.3.30 setTemperature()

```
setTemperature (
              $temp )
```

Set temperature. Note : Temperature is an important setting to control how much randommes is the result of a query. It needs to be between 0 and 1.

See also

`https://platform.openai.com/docs/api-reference/completions/create`

Parameters

number	temp

Returns

ture or false

The documentation for this class was generated from the following file:

- src/Ephpai.php

Index