

v1.6.3	CGI365Lib は Python3 で作った CGI ライブラリで Request と Response , Utility クラスを含む。				
Request クラス					
No	メソッド	パラメータ	戻り値	機能	備考
1	Request()	なし	Request オブジェクト	コンストラクタ	
2	self.RawData		受信した生データ	クライアントから受け取った生データ (文字列あるいはバイト列)	
3	self.QueryString		辞書	GET で受け取った環境変数 QUERY_STRING (文字列)	
4	self.Method		文字列	メソッド (GET,POSTなど)	
5	self.Address		辞書	キー 'Server', 'Client', 'Host' からなる辞書	サーバアドレス、クライアントアドレス、HTTP_HOST
6	self.Cookie		辞書	受信したクッキー	
7	self.Query		辞書	GET のパラメータ	
8	self.Form		辞書	POST のパラメータ	
9	self.PathInfo		文字列	URL の route 部分。 例えば、http://../some.cgi/path1/path2 なら "/path1/path2" が格納される。	nginx では無効。
10	getParam(self, key)	key: パラメータのキー	キーに対応するパラメータの値	キーに対するパラメータの値を返す。 キーがないときは空文字を返す。	GET または POSTに対応
11	getCookie(self, key)	key: クッキーのキー	キーに対するクッキーの値。 キーが存在しないときは ""	self.Cookie のキーに対する値を得る。キーが存在しないときは "" が返される。	
12	parseFormBody(self)	なし	なし	POST されたフォームデータを解析してパラメータの辞書を作り、self.Form に格納する。	このメソッドを実行しないと self.Form はからの辞書なので注意。
13	parseJSON(self)	なし	辞書	sys.stdin から生データを読んで、JSON とみなしてデコードする。 読んだ生データは self.RawData に格納される。	
14	getRawData(self)	なし	self.RawData	生データ (stdin) を BLOB とみなし、self.RawData に格納する	self.RawData はバイト列
15	getRawString(self)	なし	self.RawData	生データ (stdin) を UTF-8 文字列とみなし、self.RawData に格納する。	self.RawData は文字列
16	saveRawData(self, savePath)	savePath: 保存先のファイル	なし	self.RawData を BLOB としてファイル保存する。	保存先には書き込み権限が必要
17	saveAsRawString(self, path)	path: 生データの保存先ファイル	なし	クライアントから受け取った生データを UTF-8 文字列とみなし、self.RawData に格納する。 さらに path が "" でないならファイルのパスとみなしファイル保存する。	saveRawData() と違って stdin から生データを取得する。
18	saveAsBLOB(self, path)	path: 生データの保存先ファイル	なし	クライアントから受け取った生データを BLOB とみなし、self.RawData に格納する。 さらに path が "" でないならファイルのパスとみなしファイル保存する。	saveRawData() と違って stdin から生データを取得する。
19	saveFile(self, name, savedir, binary=False)	name: input[type="file"] の name 属性値 savedir: 保存先のディレクトリ binary=False: テキストで保存するかバイナリーモードで保存するか	成功したら True、失敗したら False	name で指定されたアップロードファイルを保存先ディレクトリにクライアント側のファイル名で保存する。	request.parseFormBody() を事前に実行しておくこと。
20	formdataToString(self)	なし	文字列	辞書型 self.Form の内容を文字列にする。	デバッグ用
21	__getQuery(self)	なし	辞書		内部で使用
22	__getMethod(self)	なし	文字列		内部で使用
23	__getCookie(self)	なし	辞書		内部で使用
24	__getAddress(self)	なし	辞書		内部で使用
25	__getChunk(self, name)	name: HTMLコントロールのnameの内容	タプル(filename, chunk)		内部で使用

	Response クラス				
No	メソッド	パラメータ	戻り値	機能	備考
1	Response()	なし	Response オブジェクト	コンストラクタ	
2	self.Cookie			クッキーのリスト	List
3	self.Headers			sendString() で付加されるヘッダーのリスト	List
4	setCookie(self, cookies)	cookie: 送信するクッキー用の辞書	なし	Set-Cookie 行を作るためのデータとして保存される。	クッキーは HTML を出力するときに同時に出力される。
5	makeCookie(self)	なし	Set-Cookie 行	Set-Cookie 行をHTTPヘッダに追加する。	
6	sendString(self, s:str, mime="", cookie=True, headers=True, embed=None)	s: 出力する文字列 mime="": MIME (Content-Type: 以下の文字列) cookie: True ならクッキーを返す。 headers: True なら self.Headers の内容を付加する。 embed: s に埋め込む変数の辞書	なし	設定に基づいて文字列を出力する。デフォルトは text/html で文字コードの指定なし。	HTTPクッキーと追加されたHTTPヘッダーも出力。
7	sendSimple(self, s:str, charset="")	s:文字列 charset="": 文字コード	なし	デフォルトの設定に基づいて文字列を出力する。	デフォルトは "Content-Type: text/plain"
8	sendBinData(self, data)	data: バイナリーデータ	なし	octet-stream としてデータを出力する。	Content-Type: application/octet-stream"
9	sendJSON(self, data, charset="")	data: オブジェクト charset="": 文字コード	なし	data を文字列に変換して出力する。	デフォルトは "Content-Type: application/plain"
10	sendText(self, path, charset="")	path: テキストファイル charset="": 文字コード	なし	path で指定されたテキストファイルの内容を出力する。	デフォルトは "Content-Type: text/plain"
11	sendHtml(self, path, charset="", cookie=True, embed=None)	path: HTMLファイル charset="": 文字コード cookie=True: Trueならクッキーも出力 embed=None: HTML に埋め込む変数の辞書	なし	path で指定されたHTMLファイルを出力する。 その際、cookie=True ならクッキーを付加する。 また、embed が None でないなら、HTML ファイル内の "{key}" という文字列を embed[key] で置き換える。	デフォルトは "Content-Type: text/html"
12	sendImage(self, path)	path: 画像ファイル	なし	画像ファイルを出力する。	jpg, png, gif, svg
13	sendVideo(self, path)	path: 動画ファイル	なし	動画ファイルを出力する。	mp4, webm, ogv
14	sendAudio(self, path)	path: 音声ファイル	なし	音声ファイルを出力する。	mp3, m4a, wav, ogg
15	sendFile(self, path, mime, filename="")	path: ファイル mime: MIME filename="": ファイル名	なし	mime で指定したファイルを出力する。 filename="" の場合は、ファイル保存ダイアログにファイル名が表示されない。	zip ファイルなど
16	redirect(self, url)	url: URL	なし	url で指定したサイト/ページへリダイレクトする。	Location ヘッダを使用する。
17	header(self, headers)	headers: 完全なヘッダ文のリスト	なし	HTTP ヘッダを出力する。	HTTPヘッダーのみ出力し、本体は出力しない。
18	sendPPrint(self, obj, charset="")	obj: オブジェクト charset="": 文字コード	なし	デバッグ用: pprint() を使ってオブジェクトをシリアル化して文字列として出力する。	デフォルトは "Content-Type: text/plain"
19	status(self, code, message="")	code: ステータスコード message: ブラウザに表示するメッセージ (HTML)	なし	HTTPレスポンスステータスコードを返す。code は "500 Internal Server Error"のようする。	
	Utility クラス				
No	変数または関数	パラメータ	戻り値	機能	備考

1	htmlTable(data, header, table="", tr="", th="", td="")	data: 行(列の配列)のリスト header: ヘッダ行(配列) table: tableタグのクラス tr: trタグのクラス th: thタグのクラス td: tdタグのクラス	HTML table 文字列	HTML テーブル文字列を作成する。	static メソッド
2	htmlList(data, list="ul", ul="", li="")	data: データの配列 list: "ul" または "ol" ul: ul(ol)タグのクラス li: liタグのクラス	HTMLリスト文字列	HTMLリストを作成する。	static メソッド
3	svg(shape, size=32, borderWidth=1, borderColor="black", bgColor="white")	shape: "circle" または "square" size: サイズ(直径または辺の長さ) borderWidth: 境界線の太さ borderColor: 境界線の色 bgColor: 内部の色	SVG 文字列	SVG を作成する。(円と正方形のみ)	static メソッド
4	startProcess(cmd, *args)	cmd: コマンド(パラメータを含まない) *args: パラメータ(可変長引数)	コマンドが返す文字列	コマンドを起動する。	static メソッド
5	readConf(filePath)	filePath: JSONファイル	オブジェクト	JSON ファイルを読んでそのオブジェクトを返す。主に構成ファイルの読み込みに使用する。	static メソッド

クラスに含まれない変数や関数

No	変数または関数	パラメータ	戻り値	機能	備考
1	ENC			文字コード。デフォルトは UTF-8、	
2	LOG			ログの出力先のパス名	ウェブサーバが書き込みできること。
3	isDebug()	なし	デバッグモードのとき True	コマンド引数に "debug" または "debug_get" または "debug_post" が指定されているときに True を返す。また、環境変数 REQUEST_METHOD に "GET" または "POST" を設定する。	コンソールから起動する場合に使用する。
4	setQueryString(qs)	qs: GET メソッドの時の URL パラメータ	なし	デバッグモードのとき、環境変数 QUERY_STRING の内容を設定する。	
5	setHttpCookie(cookie)	cookie: HTTP_COOKIE の内容を設定する。	なし	環境変数 HTTP_COOKIE の内容を設定する。	
6	setRequestMethod(method)	method: REQUEST_METHOD の内容	なし	環境変数 REQUEST_METHOD の内容を設定する。	
7	info(obj)	obj: ログファイルに出力するオブジェクト	なし	LOG で指定したファイルに obj を文字列に変換して追加する。	LOG が空文字の場合は、何もしない。
8	tag(t, s, c="")	t: タグ名 s: タグ内部の文字列 c="": クラス	タグの HTML を返す。	タグの HTML を作成する。	
9	anchor(url, s, target="")	url: href属性 s: タグ内部の文字列 target="": target属性	a タグの HTML を返す。	a タグの HTML を作成する。	
10	BAD_REQUEST			"400 Bad Request"	ステータスコード
11	FORBIDDEN			"403 Forbidden"	ステータスコード
12	METHOD_NOT_ALLOWED			"405 Method Not Allowed"	ステータスコード
13	INTERNAL_SERVER_ERROR			"500 Internal Server Error"	ステータスコード
14	NOT_IMPLEMENTED			"501 Not Implemented"	ステータスコード