PHP Fat-Free Framework Quick Reference

Core Pack

F3::allow(string **set);**
Allows functions defined in PHP extensions to be used in templates. The default value for **set** is **'standard|date|pcre'**. Methods in user-defined classes may be added to the **set** by specifying the class name.

F3::call(mixed **processes);**
Convenience method for executing sandboxed **processes**. Used for programmatically calling an anonymous or named function, a single or daisy-chained string of named functions/imported files similar to the **F3::route** and **F3::input** handlers.

F3::clear(string **variable);**
Clear global **variable** and remove from symbol table.

F3::exists(string **variable);**
Return TRUE if global **variable** has been assigned a value, or FALSE if **variable** has not been defined.

F3::get(string **variable);**
Return value of global **variable**.

F3::http404(void **);**
Trigger an HTTP 404 error.

F3::httpstatus(integer **code);**
Send HTTP status header. Return text equivalent of status code.

F3::input(string **field,** mixed **handler, [** string **tags]);**
Assign **handler** to HTML form **field** for validating/manipulating user input. **handler** may be an anonymous or named function, a single or daisy-chained string of named functions/imported files similar to the F3::route command. All HTML/PHP tags are stripped If **tags** is not specified.

F3::map(string **uri,** string **class);**
Assign **class** as route handler for incoming **uri**. Class should contain methods that correspond to an incoming HTTP request: get(), post(), put(), head() or delete().

F3::profile(void **);**
Returns an array of runtime performance analysis data.

F3::reroute(string **location);**
Reroute to specified URI **location**.

F3::resolve(string **expression);**
Evaluate **expression** containing template tokens.

F3::route(string **pattern,** mixed **handler, [** integer **cache_timeout]);**
Assign **handler** to URI **pattern** consisting of an HTTP GET/HEAD/POST/PUT/DELETE command, a space and a URI. If **cache_timeout** is specified, Fat-Free retrieves previously-generated output associated with the URL from cache if timer has not expired. Otherwise, output is refreshed by calling **handler**. Output is not cached if **cache_timeout** is not specified.

F3::run(void **);**
Run the application. Process routes based on incoming URI.

F3::scrub(mixed **value, [** string **tags]);**
Remove all HTML tags to protect against XSS/code injection attacks. Enumerated **tags**, if present, will be preserved. If **value** is an array, HTML tags in all string elements will be scrubbed.

F3::send(string **file);**
Transmit **file** for downloading by HTTP client.

F3::serve(string **template, [** boolean **is_html], [** string **path]);**
Returns a string representing a rendered HTML/XML template. If **is_html** is TRUE (default), template is interpreted as a regular HTML/text file. If FALSE, rendering follows strict XML parsing rules. Specify **path** if **template** is not in the **GUI** folder.

F3::set(string **variable,** mixed **value);**
Bind **value** to global **variable**.

version 1.2.10

Database Pack

F3::qq(mixed **arg1,** mixed **arg2, ...** mixed **argN);**
Enclose arguments (all string elements if array is passed) in double-quotes and add commas if multiple arguments are present.

F3::sql(mixed **statement, [** string **db_identifier]);**
Execute SQL statement. **statement** can be a single string or an array of strings. If an array is specified, it is treated as a single SQL transaction - commit and rollback are dependent on the successful completion of the transaction. If **cache_timeout** is specified, previously-generated output associated with the SQL statement is retrieved from cache (if timer has not expired). Default **db_identifier** is the DB global variable.

Axon ORM

new Axon(string **table, [** string **db_identifier]);**
Instantiate Axon object and map to specified **table**. Default **cache_timeout** is 60 seconds. Adjust if necessary. Default **db_identifier** is DB.

copyfrom(string **name);**
Hydrate Axon with elements from framework array variable, keys of which must be identical to field names in the record.

copyto(string **name);**
Populate framework array variable with Axon properties, keys of which will have names identical to fields in the record.

def(string **name,** string **expression);**
Create a virtual field and assign SQL **expression** for use in queries.

dry(void **);**
Return TRUE if Axon is devoid of values in its properties.

erase(void **);**
Delete record and reset Axon.

find([string **criteria], [** string **order], [** string **limit], [** integer **cache_timeout]);**
Return an array of database records matching specified **criteria**. If **criteria** is omitted or NULL, all records in table are returned. Result is sorted according to the sequence of fields specified in **order** and short-listed by **limit**.

found(string **criteria], [** integer **cache_timeout]);**
Return number of records that match specified **criteria**.

isdef(string **name);**
Return TRUE if specified virtual field exists, FALSE otherwise.

load(string **criteria, [** string **order], [** integer **offset]);**
Retrieve first record that satisfies **criteria**, sort according to comma-separated list of fields specified in **order**, and starting at the relative **offset**.

lookup(string **fields, [** string **criteria], [** string **grouping], [** string **order], [** string **limit], [** integer **cache_timeout]);**
Similar to Axon **find** method but provides fine-grained control over specific fields and grouping of results. The **select** method is an alias of the **lookup** method.

save(void **);**
Insert or update record based on contents of Axon properties.

skip([integer **offset]);**
Retrieve the record at the **offset** relative to current position following the same criteria used to hydrate the Axon. Positive offset moves forward, negative backward. Zero re-syncs Axon. Triggers error if skipping beyond bounds of hydration criteria.

sync(string **table, [** string **db_identifier]);**
Synchronize new Axon and table structure. Default **cache_timeout** is 60 seconds. Called automatically when instantiating new Axon. Default **db_identifier** is DB.

undef(string **name);**
Destroy the specified virtual field.

Axon Event Listeners

beforesync(void **); aftersync(** void **);**
Invoked before/after Axon **sync** method is executed.

beforeload(void **); afterload(** void **);**
Invoked before/after Axon **load** method is executed.

beforesave(void **); aftersave(** void **);**
Invoked before/after Axon **save** method is executed.

beforeerase(void **); aftererase(** void **);**
Invoked before/after Axon **erase** method is executed.

Expansion Pack

F3::captcha(integer **width,** integer **height,** integer **length, [** string **font]);**
Generate CAPTCHA image of specified **width** and **height** (in pixels). **length** represents the number of hex characters to generate. The default TrueType **font** is cube.ttf. The string equivalent of the CAPTCHA image is stored in **$_SESSION['captcha']** and the framework variable **SESSION['captcha']**.

F3::expect(mixed **condition,** string **true_text,** string **false_text);**
Test condition and append result to **TEST** global variable.

F3::fakeimage(integer **width,** integer **height,** string **bgcolor);**
Generate a blank image for use as a placeholder.

F3::http(string **pattern, [** string **content], [** array **headers], [** boolean **follow]);**
Send HTTP/S request to another host; forward **headers** received and return host's response. **pattern** consists of an HTTP GET/HEAD/POST/PUT/DELETE command, a space and an absolute URI. **pattern** should include port number after host name if not using a standard TCP port (80 for HTTP, 443 for HTTPS). **content** is optional URI query string. Respect HTTP 30x redirects if **follow** is TRUE (default).

F3::identicon(string **hash, [** integer **size]);**
Generate icon from a hex-string **hash** value. Default **size** is 64 pixels.

F3::minify(string **path,** array **filenames);**
Compress specified files by removing all whitespaces and comments, then send as a single block of CSS or Javascript.

F3::mock(string **pattern);**
Simulate an incoming HTTP request. Refer to **F3::route** for **pattern** syntax.

F3::sitemap(string **root_url);**
Generate XML sitemap starting at specified **url**.

F3::thumb(string **filename,** integer **width,** integer **height);**
Generate thumbnail image of specified maximum **width** and maximum **height**. Resulting image has the same aspect ratio (proportions) as original.

F3::transpose(string **variable,** boolean **replace);**
Rotate a two-dimensional framework array variable, so rows become columns, and vice versa. Content of variable is replaced if second argument is TRUE (default).

F3::validurl(string **url);**
Return TRUE if **url** is valid, FALSE otherwise.

Template Tokens and Directives

@token
Replace **token** with global **variable** of the same name.

{ scalar **expression }**
Evaluate **expression** and replace with scalar result. **expression** may include template tokens, constants, operators (unary, arithmetic, ternary and relational), parentheses, data type converters, and PHP functions (see F3::allow for restrictions).

<F3:include href="string **filename"/>**
Get contents of **filename** and insert at current position in template.

<F3:exclude> text-block </F3:exclude>
Remove **text-block** at runtime.

```
<F3:check if="{ boolean condition }">
    <F3:true> true-block </F3:true>
    <F3:false> false-block </F3:false>
</F3:check>
```

Evaluate **condition**. If TRUE, **true-block** replaces the F3:check block. Otherwise, **false-block** is used.

```
<F3:repeat
    [ key="{ scalar @key }" ] index="{ mixed @index }"
    group="{ array @group }">
        text-block
</F3:repeat>
```

Repeat **text-block** as many times as there are elements in the array variable **group**. Variables **key** and **index** may be used within **text-block** to represent the key and value of current array iteration element, respectively.

Framework Variables

string **AUTOLOAD**
Location of user-defined PHP classes that the framework will autoload at run-time. If **onLoad** method is defined in an autoloaded class, it is automatically executed. Default is **autoload/** folder under the Web root.

string **CACHE**
Location of the framework cache. Default is **cache/** folder under the Web root.

array **COOKIE,** array **GET,** array **POST,** array **REQUEST,** array **SESSION,** array **FILES**
Mirrors PHP's global variables of the same names for direct use in templates.

array **DB**
The default database identifier for several SQL objects. Assign Data Source Name (DSN) to **DB.dsn** to establish a database connection, Set the values of **DB.user**, **DB.password**, **DB.options** and **DB.attributes** if needed. **DB.pdo** contains PDO instance representing the connection. Result of the last query is saved in **DB.result**. Use **DB.query** for fine-grain control. This is the equivalent of PHP's PDOStatement.

boolean **DEBUG**
Toggle for including framework internals in stack trace. Default is FALSE.

string **DNSBL**
Public blacklist services for looking up IP addresses of spammers.

string **E404,** string **E500**
Contains file names of custom HTTP 404 and 500 error templates. File should be in the path pointed to by **GUI**.

string **ENCODING**
Character set used for document encoding. Default value is **UTF-8**.

array **ERROR**
Holds a profile of the last HTTP error that occurred. **ERROR.code** is the HTTP error number (404 or 500). **ERROR.title** contains a brief description of the error. **ERROR.text** provides greater detail.

string **FONTS**
Path to fonts files used by **F3::captcha**. Default is the Web root.

array **FUNCS**
Contains a list of all functions permitted in templates .

string **GUI**
Location of graphical user interface files used by **F3::serve** and **F3::minify**. May be an absolute path or relative to the Web root folder. Default is the Web root.

string **IMPORTS**
Location of imported files used in the **F3::route**, **F3::input** and **F3::call** methods. May be an absolute path or relative to the Web root folder. Default is the Web root.

integer **MAXSIZE**
Maximum file upload size in bytes.

array **PARAMS**
Holds the captured values of tokens defined in a URI pattern. Use **PARAMS.0** to get the captured URL relative to the Web root.

boolean **QUIET**
Toggle for suppressing output (useful for unit testing).

boolean **RELEASE**
Indicates release or development version of application. If **FALSE** (default), stack trace appears in both HTML error page and server's error log. If **TRUE**, stack trace in HTML output is suppressed.

array **RESPONSE**
The body of the last HTTP response (useful for unit testing).

array **ROUTES**
Contains all the routes and handlers defined by the application.

string **SPAM**
The URI for redirecting a user blacklisted in a **DNSBL**.

string **SYNC**
Cache timeout for Axon **sync** method. Default value is 60 seconds.

array **TEST**
Array containing the results of each **F3::expect** statement.

integer **THROTTLE**
The minimum number of milliseconds to serve a Web page.

float **TIME**
Framework start time (floating point Unix timestamp).