

---

# HTTP Libraries Documentation

*Release 1.0*

**Dylan Hackers**

December 10, 2018



## CONTENTS

<b>1</b>	<b>HTTP Server</b>	<b>3</b>
1.1	Dylan Web in 60 Seconds . . . . .	3
1.2	Configuration . . . . .	8
1.3	URL Routing . . . . .	8
1.4	Dylan Server Pages (DSP) . . . . .	10
1.5	Stand-Alone Server App . . . . .	12
<b>2</b>	<b>HTTP Client</b>	<b>13</b>
<b>3</b>	<b>Reference</b>	<b>15</b>
3.1	The HTTP-SERVER library . . . . .	15
3.2	The HTTP-COMMON library . . . . .	27
3.3	The DSP library . . . . .	52
3.4	The HTTP-CLIENT library . . . . .	59
<b>4</b>	<b>Copyright</b>	<b>65</b>
<b>5</b>	<b>Indices and tables</b>	<b>67</b>
	<b>API Index</b>	<b>69</b>



Dylan provides libraries for writing HTTP clients and servers.



## HTTP SERVER

### Dylan Web in 60 Seconds

This is a series of examples showing how to do web programming in Dylan. It was inspired by a version of [the same thing written for Twisted Python](#). In Twisted, it is often the case that very little code is required to get something basic running, which has also been an inspiration to create similarly simple APIs for the Dylan web infrastructure.

#### Serving Static Content

Next: [Serving Dynamic Content](#)

[Back to top](#)

This example will show you how to use the Dylan web server (`http-server`) to serve static content from the file-system. First I'll go through the process line by line and then will show the complete code.

First, we set up the usual library and module definitions. “`http-server`” is all you need other than `common-dylan`.

```
Module: dylan-user

define library web60-static-content
  use common-dylan;
  use http-server;
end;

define module web60-static-content
  use common-dylan;
  use http-server;
end;
```

Next we need an HTTP server. We'll make it listen to all interfaces on port 8888:

```
let server = make(<http-server>,
                 listeners: list("0.0.0.0:8888"));
```

Instances of `<resource>` are responsible for generating HTTP responses. To serve static content we create a `<directory-resource>` whose file-system directory is `/tmp` and which allows directory listings:

```
let resource = make(<directory-resource>,
                   directory: "/tmp", // c:\tmp on Windows
                   allow-directory-listing?: #t);
```

Next we connect the resource to a specific URL on the server:

```
add-resource(server, "/", resource);
```

Last, we start the server:

```
start-server(server);
```

If you wanted to start the server in a separate thread you could say this instead:

```
start-server(server, background: #t);
```

The entire example, including the library and module definitions, looks like this:

```
---- File: library.dylan ----  
Module: dylan-user  
  
define library web60-static-content  
  use common-dylan;  
  use http-server;  
end;  
  
define module web60-static-content  
  use common-dylan;  
  use http-server;  
end;  
  
---- File: main.dylan ----  
Module: web60-static-content  
  
define function main ()  
  let server = make(<http-server>,  
                   listeners: list("0.0.0.0:8888"));  
  let resource = make(<directory-resource>,  
                     directory: "c:/tmp",  
                     allow-directory-listing?: #t);  
  add-resource(server, "/", resource);  
  start-server(server);  
end;  
  
main();
```

Run this example and point your browser at <http://127.0.0.1:8888/>.

Note that serving static content is one of the things built into the `http-server` library itself, so if that's all you want to do this will accomplish the same thing:

```
http-server --listen 0.0.0.0:8888 --directory /tmp
```

Next: [Serving Dynamic Content](#)

[Back to top](#)

## Serving Dynamic Content

Previous: [Serving Static Content](#)

[Back to top](#)

This example will show how to dynamically generate the contents of a web page.



First, the ever-exciting library and module definitions. In addition to `common-dylan` and `http-server` we need `streams` for writing data to the response and `date` so we can show something dynamic happening:

```
define library web60-dynamic-content
  use common-dylan;
  use io, import: { streams };
  use http-common;
  use http-server;
  use system, import: { date };
end;

define module web60-dynamic-content
  use common-dylan;
  use date, import: { as-iso8601-string, current-date };
  use http-common;
  use http-server;
  use streams, import: { write };
end;
```

A web page is a resource mapped to a URL inside the web server. To create a resource we subclass `<resource>`:

```
define class <clock-page> (<resource>) end;
```

To make our resource do something we define a method on `respond`. (If we only wanted to implement the GET request method we could define a method on `respond-to-get` instead.)

```
define method respond (page :: <clock-page>, #key)
  let stream = current-response();
  set-header(stream, "Content-Type", "text/html");
  let date = as-iso8601-string(current-date());
  write(stream, concatenate("<html><body>", date, "</body></html>"));
end;
```

`current-response` returns the active `<response>` object. To send data back to the client we write to the current response.

```
let server = make(<http-server>,
  listeners: list("0.0.0.0:8888"));
add-resource(server, "/", make(<clock-page>));
start-server(server);
```

In the previous example we already saw how to create and start a server, so the new bit here is using `add-resource` to map a URL to a `<resource>`. The first argument to `add-resource` is the URL router. (In Routes terminology it would be a “mapper”.) For convenience, an `<http-server>` is a kind of router so we can add resources directly to the server. In a future example, I will show how to do more complex URL routing, which will explain the reason for the mysterious `#key` in the `respond` definition above.

Here’s the complete code:

```
---- File: library.dylan ----
Module: dylan-user

define library web60-dynamic-content
  use common-dylan;
  use io, import: { streams };
  use http-common;
  use http-server;
  use system, import: { date };
end;
```

```
define module web60-dynamic-content
  use common-dylan;
  use date, import: { as-iso8601-string, current-date };
  use http-common;
  use http-server;
  use streams, import: { write };
end;

---- File: main.dylan ----
Module: web60-dynamic-content

define class <clock-page> (<resource>)
end;

define method respond (page :: <clock-page>, #key)
  let stream = current-response();
  set-header(stream, "Content-Type", "text/html");
  let date = as-iso8601-string(current-date());
  write(stream, concatenate("<html><body>", date, "</body></html>"));
end;

let server = make(<http-server>,
                 listeners: list("0.0.0.0:8888"));
add-resource(server, "/", make(<clock-page>));
start-server(server);
```

Previous: [Serving Static Content](#)

Next: [Static URL Routing](#)

[Back to top](#)

## Static URL Routing

Previous: [Serving Dynamic Content](#)

[Back to top](#)

This example will show how URL routing works in the Dylan web server, and how to handle optional URL elements for your resources.

I will skip the library and module definitions since they're essentially the same as in the previous examples, but they are included in the full code listing at the end.

In the Dylan web server, *add-resource* maps a URL to a resource. The default implementation of *add-resource* builds up a tree structure whose paths are defined by URL path elements and whose leaves are *<resource>* objects. (Idea stolen from *twisted.web*. I hope to add a simple regular expression based router in the future, for comparison.)

For this example we'll use a hypothetical wiki as our web application and add three different URLs for it. First, we need a *\$wiki-app* resource that will be the root of all wiki URLs, and specialized resource classes to provide behavior. We'll implement page, user and group resources for the wiki:

```
define constant $wiki-app = make(<resource>);
define class <page> (<resource>) end;
```

```
define class <user> (<resource>) end;
define class <group> (<resource>) end;
```

Now wiki resources can be added as children of `$wiki-app`:

```
add-resource($wiki-app, "page/{action}/{title}/{version?}", make(<page>));
add-resource($wiki-app, "user/{action}/{name}", make(<user>));
add-resource($wiki-app, "group/{action}/{name}", make(<group>))
```

The URL path elements surrounded by curly braces are “path variables”. Let’s decompose the first URL above: `page/{action}/{title}/{version?}`. The first element, “page” must be matched literally. The `{action}` and `{title}` elements are required path variables; if either is missing 404 is returned. The last element, `{version?}` is optional, as indicated by the “?” character. (Two more path variable types that aren’t shown here are available: `{v*}` matches zero or more path elements and `{v+}` matches one or more.)

In order to define the behavior of our various resources we define methods on the `respond` generic function. Note that each path variable in the URL passed to `add-resource` corresponds to a keyword in the `respond` method for the resource being added. (For our purposes the behavior will be to simply display the values of all the path variables.)

```
define method respond
  (resource :: <page>, #key action, title, version)
  set-header(current-response(), "Content-Type", "text/html");
  output("<html><body>action = %s, title = %s, version = %s</body></html>",
        action, title, version);
end;
```

The `respond` methods for `<user>` and `<group>` are similar. Notice that `version` may be `#f` but `action` and `title` will always be strings.

Lastly, we’ll connect `$wiki-app` to the root URL (`/`) and start the server:

```
define constant $server = make(<http-server>, listeners: #("0.0.0.0:8888"));
add-resource($server, "/", $wiki-app);
start-server($server);
```

That’s it. Run the server and click on some of these URLs to see the corresponding behavior:

- `http://127.0.0.1:8888/page/view/Foo/3`
- `http://127.0.0.1:8888/page/view/Foo`
- `http://127.0.0.1:8888/user/add/cgay`
- `http://127.0.0.1:8888/group/remove/administrators`

Here’s the full code listing:

```
-----library.dylan-----
Module: dylan-user

define library web60-static-routing
  use common-dylan;
  use http-common;
  use http-server;
end;

define module web60-static-routing
  use common-dylan;
  use http-common;
  use http-server;
end;
```

```
-----static-routing.dylan-----
Module: web60-static-routing

define constant $wiki-app = make(<resource>);

define class <page> (<resource>) end;
define class <user> (<resource>) end;
define class <group> (<resource>) end;

add-resource($wiki-app, "page/{action}/{title}/{version?}", make(<page>));
add-resource($wiki-app, "user/{action}/{name}", make(<user>));
add-resource($wiki-app, "group/{action}/{name}", make(<group>));

define method respond
  (resource :: <page>, #key action, title, version)
  set-header(current-response(), "Content-Type", "text/html");
  output("<html><body>action = %s, title = %s, version = %s</body></html>",
        action, title, version);
end;

define method respond
  (resource :: type-union(<user>, <group>), #key action, name)
  set-header(current-response(), "Content-Type", "text/html");
  output("<html><body>action = %s, name = %s</body></html>",
        action, name);
end;

define constant $server = make(<http-server>, listeners: #("0.0.0.0:8888"));
add-resource($server, "/", $wiki-app);
start-server($server);
```

Previous: [Serving Dynamic Content](#)

[Back to top](#)

## Configuration

### Programmatic Configuration

### Command Line Configuration

### XML Configuration File

## URL Routing

- A URL path is represented internally as a sequence of path elements. For example the URL string `"/foo/bar/"` is `#("", "foo", "bar", "")`. Note that this is exactly equivalent to `split("/foo/bar/", '/')`.
- URL paths are mapped to `<resource>` instances. The `<resource>` instances are arranged in a tree structure whose paths are the URL path elements.
- `add-resource` accepts a `<uri>`, or a string path such as `"/foo/bar/"`, or a sequence of path elements such as `#("", "foo", "bar", "")`.

- `add-resource` divides the path into two parts: the prefix and suffix. The suffix begins where the first path variable, if any, is found. For example, given the path

```
/foo/bar/{id}/{tag}
```

the prefix is `#("foo", "bar")` and the suffix is `#("{id}", "{tag}")` because path elements of the form `{...}` define path variables.

- The child resource is mapped to the prefix, and the suffix (the path variables) are passed as keyword arguments to the `respond*` methods, bound to the corresponding parts of the request URL. For example:

```
add-resource(container, "/foo/bar/{id}/{tag}", make(<my-resource>));

define method respond-to-get (res :: <my-resource>, #key id, tag)
  ...
end;
```

When the request URL is `/foo/bar/22/dylan`, the `respond-to-get` method will be called like this:

```
respond-to-get(my-resource, id: "22", tag: "dylan")
```

If any path variable isn't provided in the request URL path then the request query values are used. If the variable value isn't found in either the path or the query values it is not passed to respond. Example:

Request URL	respond Keyword Args
<code>/foo/bar</code>	None
<code>/foo/bar/abc</code>	<code>id: "abc"</code>
<code>/foo/bar/baz?tag=x</code>	<code>id: "baz", tag: "x"</code>

- There is one case that can only be represented with a sequence of path elements rather than a string: a URL that ends with a `/` but also has path variables. That is, because `"/foo/bar/{x}/{tag}"` is split on `'/'`, the path elements are:

```
#("", "foo", "bar", "{x}", "{tag}")
```

whereas for a trailing slash on `/foo/bar/` the path elements should be:

```
#("", "foo", "bar", "", "{x}", "{tag}")
```

So when adding a resource with a trailing slash on the URL prefix and path variables, you must pass the above sequence to `add-resource` explicitly. See the next point for one way to avoid this issue.

- There seems to be fairly broad agreement that having distinct meanings for `/foo` and `/foo/` is a Bad Thing, which makes sense since it would be difficult for a user to predict the difference. However, it is often useful to ensure that they both map to the same resource rather than having one work and the other get a 404 error.

`add-resource` accepts a `slash:` keyword argument. When true (the default), `add-resource` will automatically map the child resource to both the given URL prefix and to the same URL with a trailing slash added. That is,

```
add-resource(parent, "foo/bar", child)
```

will map `child` to both `foo/bar` and `foo/bar/`. To change this behavior pass the `slash:` keyword argument:

**slash: #`"canonical"`** Make the trailing slash URL prefix canonical (by redirecting the one without the trailing slash). Think “slash is canonical”.

**slash: #`"copy"`** Map the given resource to both the given URL and the URL with the trailing slash. Think “slash is a copy”.

**slash:** **#"redirect" (the default)** Redirect the trailing slash version of the URL to the one with no trailing slash. Think “slash is redirected”.

**slash:** **#f** Don’t map the trailing slash URL prefix to any resource. Use this carefully since it can cause confusing results for users.

## Dylan Server Pages (DSP)

### Overview

Dylan Server Pages (DSP) is a template engine for providing dynamic web site content using the Dylan language. They are similar in spirit to JavaServer Pages, but without the ability to mix program logic and page content, which is a bad idea anyway. All dynamic content is invoked via DSP tag calls such as `<xx:show-current-username/>`.

(Okay, opinions can vary on the wisdom of closely mixing content and implementation, as with JSP scriptlets, but I contend that it’s only ok for one-off hacks and small one-person projects. This isn’t necessary in DSP because with Dylan’s powerful language features writing custom tags is trivial and concise compared to JSP. And with [Open Dylan](#), you often don’t even need to restart your application to compile new custom tags and try them out.)

Currently, a DSP application is implemented as a Dylan project that uses the HTTP server library. This effectively means that each DSP application has to run on a different server and therefore on a different HTTP port. (The plan is to eventually fix it to load DSP application libraries at runtime, based on configuration files.)

### DSP Content Authoring

#### Authoring Overview

DSP templates contain normal HTML plus DSP tag calls. DSP tag calls generate the dynamic content of you web pages. They use standard XML syntax. For example, `<mylib:mytag arg="foo"/>` is tag call with no body that passes one argument, `arg`, to `mytag` when it is invoked. “mylib” tells the DSP template engine what tag library (taglib for short) “mytag” will be found in.

There are several special tags called *DSP Directives* defined that couldn’t easily have been defined by the user. DSP directives use the same syntax as other tags, but they use the special tag library name `%dsp`. For example, the include directive might look like this: `<%dsp:include location="foo.dsp"/>`.

Each top-level template file must have a corresponding instance of `<dylan-server-page>` associated with it. This is accomplished with the define page macro, which also publishes the URLs associated with the page.

DSP template files may have any filename extension, but the extension “.dsp” may be treated specially in the future. For example, .dsp files may eventually be automatically exported as `<dylan-server-page>s`.

#### DSP Directives

A DSP directive is used much like a normal DSP tag, but it couldn’t be written as a normal tag without special support from the DSP engine. DSP directives are distinguished by the `%dsp` tag prefix. There are two built in DSP directives:

**`%dsp:include`** Includes another DSP template (or plain HTML file) in the current page. Example usage:

```
<%dsp:include location="header.dsp"/>
```

Note that if the location given is absolute (i.e., begins with a slash) then the document is looked up relative to the document root directory. Otherwise it is looked up relative to the directory containing the current document.

**%dsp:taglib** Declares another tag library to be in effect for the remainder of the page. Taglib directives are cumulative. That is, using two or more %dsp:taglib directives on the same page means that tags in either taglib may be used on that page. Example usage:

```
<%dsp:taglib name="my-taglib" prefix="xx"/>
...
<xx:my-tag/>
...
```

The value of the name argument must be the same as the name in a *taglib-definer* form. The prefix may contain any characters except for < and : and may not be %dsp.

## The “dsp” Tag Library

The “dsp” taglib defines a few tags that are generally useful for any web application.

**dsp:if** Conditionally includes its body content if a predicate returns true. Example usage:

```
<dsp:if test="my-predicate?">
  <dsp:then>...true part...</dsp:then>
  <dsp:else>...false part...</dsp:else>
</dsp:if>
```

Note that there may be multiple dsp:then and dsp:else tags inside the same dsp:if tag body. If there is any plain HTML in the body of the dsp:if, and outside of any dsp:then or dsp:else tags, it will always be displayed.

**dsp:then** Executes its body only if the test predicate of the containing dsp:if tag returned true. When not contained in the body of a dsp:if tag its body will not be executed.

**dsp:else** Executes its body only if the test predicate of the containing dsp:if tag returned false. When not contained in the body of a dsp:if tag its body will not be executed.

**dsp:table** TBD. Haven’t quite settled on a design here yet.

**dsp:table-row-number** Displays the one-based number of the row currently being displayed.

## DSP Programming

### Tag Libraries

Tags can be organized into separate tag libraries if needed (e.g., for large web apps). Each DSP page may use the %dsp:taglib directive to specify which tag libraries are active for that page. The built-in “dsp” tag library is automatically available to all DSP pages, without having to use the %dsp:taglib directive to make it active. The “dsp” taglib includes some tags that are useful for almost all web pages.

Taglibs are fairly uninteresting as far as programming a DSP application goes. They are only used when defining tags and named methods, to specify which taglib those objects belongs to. They are defined as follows:

```
define taglib demo ()
end;
```

The above defines a taglib named “demo”. See the *Tags* section for how the taglib is specified when defining a tag. This taglib would be included in a page with the following directive:

```
<%dsp:taglib name="demo" prefix="xyz"/>
```

and its tags would then be used like this:

```
<xyz:tag-one/>
```

Note that `prefix` is optional, and defaults to the value of `name`.

### Tags

Tags are defined with the `define tag` macro. The syntax is:

```
define [body] tag tag-name [in taglib-name]
  (method-parameters)
  (tag-parameters)
  ...code...
end;
```

The following example tag should clear things up a bit:

```
define tag current-time in demo
  (page :: <dylan-server-page>)
  (style)
  write(output-stream(current-response()), current-time(style));
end;
```

The above defines a tag called `current-time` in the `demo` taglib which outputs the current time in the DSP page. See the `tag-definer` macro for a full description tag definition. The above tag would be called like this:

```
<%dsp:taglib name="demo" prefix="xyz"/>
<xyz:current-time style="24hr"/>
```

Note that `style` defines a parameter for the tag call such that the `style` variable is bound to the value of that parameter in the body of the tag definition. The tag function must always accept one argument: `page`, an instance of `<dylan-server-page>`.

## Stand-Alone Server App



## HTTP CLIENT

This doc is just some simple examples for now. See *http-request* for additional arguments such as *headers:* and *follow-redirects:*.

Fetch the content of a web page and display it on `*standard-output*`.

```
let response :: <http-response> = http-get("http://opendylan.org");
write(*standard-output*, response.response-content);
```

Fetch a big file and copy it to a local file.

```
with-open-file(out = "/tmp/big", direction: #"output")
  http-get("http://host/big-file", stream: out)
end;
```

Post to a URL. Table data is automatically `application/x-www-form-urlencoded`.

```
http-post("http://some/url",
  content: table(<string-table>,
    "full_name" => "Dylan Thomas",
    "login" => "dthomas"))
```

Send multiple requests on a single connection via the lower-level API.

```
with-http-connection (conn = "opendylan.org")
  send-request(conn, "GET", "/")
  let response :: <http-response> = read-response(conn);
  ...content is in response.response-content...

  send-request(conn, "POST", "/blah", content: "...");
  let response :: <http-response> = read-response(conn);
  ...
end;
```

Send streaming data.

```
start-request(conn, "PUT", "/huge-file.gz");
...write(conn, "foo")...
finish-request(conn);
let response = read-response(conn);
```

How to handle errors.

```
block ()
  http-get(...)
exception (ex :: <resource-not-found-error>)
  ...
exception (ex :: <http-error>)
```

```
...last resort handler...  
end;
```

## The HTTP-SERVER library

### The HTTP-SERVER module

**\*command-line-parser\*** Variable

**\*log-content?\*** Variable

**<abstract-resource>** Open Abstract Class

Superclasses <object>

**<abstract-rewrite-rule>** Open Abstract Class

Superclasses <object>

Init-Keywords

- **redirect-code** –
- **terminal?** –

**<abstract-router>** Open Abstract Class

Superclasses <object>

**<configuration-error>** Class

Superclasses <http-server-api-error>

**<directory-resource>** Open Class

Superclasses <resource>

Init-Keywords

- **allow-directory-listing?** –
- **allow-multi-views?** –
- **default-documents** –
- **directory** –
- **follow-symlinks?** –

**<function-resource>** Open Class

Superclasses <resource>

Init-Keywords

- `function` –
- `methods` –

**<http-server> Open Class**

**Superclasses** <multi-logger-mixin>:httpi, <abstract-router>

**Init-Keywords**

- `clients-shutdown-notification` –
- `debug` –
- `default-virtual-host` –
- `listeners` –
- `listeners-shutdown-notification` –
- `lock` –
- `media-type-map` –
- `server-root` –
- `session-id` –
- `session-max-age` –
- `use-default-virtual-host?` –
- `virtual-hosts` –

**<http-server-api-error> Open Class**

**Superclasses** <http-server-error>:httpi

**<page-context> Class**

**Superclasses** <attributes-mixin>:http-common:http-common

**<redirecting-resource> Class**

**Superclasses** <resource>

**Init-Keywords**

- `target` –

**<request> Open Primary Class**

**Superclasses** <chunking-input-stream>:http-common:http-common, <base-http-request>:http-common:http-common

**<resource> Open Class**

**Superclasses** <abstract-resource>, <abstract-router>

**<response> Open Primary Class**

**Superclasses** <string-stream>:streams:io, <base-http-response>:http-common:http-common

**Init-Keywords**

- `direction` –

**<rewrite-rule> Class**

**Superclasses** <abstract-rewrite-rule>

**Init-Keywords**

- **regex** –
- **replacement** –

**<session> Open Primary Class**

**Superclasses** <attributes-mixin>:http-common:http-common

**Init-Keywords**

- **id** –
- **server** –

**<virtual-host> Class**

**Superclasses** <multi-logger-mixin>:httpi, <abstract-router>, <abstract-resource>

**Init-Keywords**

- **router** –

**add-cookie Generic function**

**Signature** add-cookie (response name value) => (#rest results)

**Parameters**

- **response** – An instance of <object>.
- **name** – An instance of <object>.
- **value** – An instance of <object>.

**Values**

- **#rest results** – An instance of <object>.

**add-resource Open Generic function**

**Signature** add-resource (router url resource #key #all-keys) => (#rest results)

**Parameters**

- **router** – An instance of <abstract-router>.
- **url** – An instance of <object>.
- **resource** – An instance of <abstract-resource>.

**Values**

- **#rest results** – An instance of <object>.

**add-resource-name Open Generic function**

**Signature** add-resource-name (name resource) => (#rest results)

**Parameters**

- **name** – An instance of <string>.
- **resource** – An instance of <resource>.

**Values**

- **#rest results** – An instance of <object>.

**add-virtual-host Open Generic function**

**Signature** add-virtual-host (server fqdn vhost) => ()

**Parameters**

- **server** – An instance of <http-server>.
- **fqdn** – An instance of <string>.
- **vhost** – An instance of <virtual-host>.

**clear-session** Generic function

**Signature** clear-session (request) => (#rest results)

**Parameters**

- **request** – An instance of <object>.

**Values**

- **#rest results** – An instance of <object>.

**configure-server** Generic function

**Signature** configure-server (server config-file) => (#rest results)

**Parameters**

- **server** – An instance of <object>.
- **config-file** – An instance of <object>.

**Values**

- **#rest results** – An instance of <object>.

**count-query-values** Function

**Signature** count-query-values () => (count)

**Values**

- **count** – An instance of <integer>.

**current-request** Function

**Signature** current-request () => (request)

**Values**

- **request** – An instance of <request>.

**current-response** Function

**Signature** current-response () => (response)

**Values**

- **response** – An instance of <response>.

**current-server** Function

**Signature** current-server () => (server)

**Values**

- **server** – An instance of <http-server>.

**debugging-enabled?** Generic function

**Signature** debugging-enabled? (object) => (#rest results)

**Parameters**

- **object** – An instance of `<object>`.

**Values**

- **#rest results** – An instance of `<object>`.

**debugging-enabled?-setter Generic function**

**Signature** `debugging-enabled?-setter (value object) => (#rest results)`

**Parameters**

- **value** – An instance of `<object>`.
- **object** – An instance of `<object>`.

**Values**

- **#rest results** – An instance of `<object>`.

**default-content-type Open Generic function**

**Signature** `default-content-type (resource) => (content-type)`

**Parameters**

- **resource** – An instance of `<abstract-resource>`.

**Values**

- **content-type** – An instance of `type-union (<mime-type>, <string>)`.

**default-documents Generic function**

**Signature** `default-documents (object) => (#rest results)`

**Parameters**

- **object** – An instance of `<object>`.

**Values**

- **#rest results** – An instance of `<object>`.

**default-virtual-host Generic function**

**Signature** `default-virtual-host (object) => (#rest results)`

**Parameters**

- **object** – An instance of `<object>`.

**Values**

- **#rest results** – An instance of `<object>`.

**default-virtual-host-setter Generic function**

**Signature** `default-virtual-host-setter (value object) => (#rest results)`

**Parameters**

- **value** – An instance of `<object>`.
- **object** – An instance of `<object>`.

**Values**

- **#rest results** – An instance of `<object>`.

**do-query-values** Generic function

**Signature** do-query-values (f) => (#rest results)

**Parameters**

- **f** – An instance of <object>.

**Values**

- **#rest results** – An instance of <object>.

**do-resources** Open Generic function

**Signature** do-resources (router function #key seen) => ()

**Parameters**

- **router** – An instance of <abstract-router>.
- **function** – An instance of <function>.
- **seen (#key)** – An instance of <object>.

**ensure-session** Generic function

**Signature** ensure-session (request) => (#rest results)

**Parameters**

- **request** – An instance of <object>.

**Values**

- **#rest results** – An instance of <object>.

**file-contents** Function

**Signature** file-contents (filename #key error?) => (contents)

**Parameters**

- **filename** – An instance of <pathname>:file-system:system.
- **error? (#key)** – An instance of <boolean>.

**Values**

- **contents** – An instance of false-or (<string>).

**find-resource** Open Generic function

**Signature** find-resource (router url) => (resource prefix suffix)

**Parameters**

- **router** – An instance of <abstract-router>.
- **url** – An instance of <object>.

**Values**

- **resource** – An instance of <abstract-resource>.
- **prefix** – An instance of <list>.
- **suffix** – An instance of <list>.

**find-virtual-host** Open Generic function

**Signature** find-virtual-host (server fqdn) => (vhost)



**Parameters**

- **server** – An instance of `<http-server>`.
- **fqdn** – An instance of `<string>`.

**Values**

- **vhost** – An instance of `<virtual-host>`.

**function-resource Function**

**Signature** `function-resource (function #key methods) => (resource)`

**Parameters**

- **function** – An instance of `<function>`.
- **methods** (*#key*) – An instance of `<object>`.

**Values**

- **resource** – An instance of `<resource>`.

**generate-url Open Generic function**

**Signature** `generate-url (router name #key #all-keys) => (url)`

**Parameters**

- **router** – An instance of `<abstract-router>`.
- **name** – An instance of `<string>`.

**Values**

- **url** – An instance of `<object>`.

**get-attr Generic function**

**Signature** `get-attr (node attrib) => (#rest results)`

**Parameters**

- **node** – An instance of `<object>`.
- **attrib** – An instance of `<object>`.

**Values**

- **#rest results** – An instance of `<object>`.

**get-query-value Function**

**Signature** `get-query-value (key #key as) => (value)`

**Parameters**

- **key** – An instance of `<string>`.
- **as** (*#key*) – An instance of `false-or (<type>)`.

**Values**

- **value** – An instance of `<object>`.

**get-session Generic function**

**Signature** `get-session (request) => (#rest results)`

**Parameters**

- **request** – An instance of <object>.

**Values**

- **#rest results** – An instance of <object>.

**http-server-main Function**

**Signature** http-server-main (#key server description before-startup) => ()

**Parameters**

- **server** (#key) – An instance of false-or (<http-server>).
- **description** (#key) – An instance of false-or (<string>).
- **before-startup** (#key) – An instance of false-or (<function>).

**log-content Generic function**

**Signature** log-content (content) => (#rest results)

**Parameters**

- **content** – An instance of <object>.

**Values**

- **#rest results** – An instance of <object>.

**log-debug Constant**

**log-error Constant**

**log-info Constant**

**log-trace Constant**

**log-warning Constant**

**output Function**

**Signature** output (format-string #rest format-args) => (#rest results)

**Parameters**

- **format-string** – An instance of <object>.
- **format-args** (#rest) – An instance of <object>.

**Values**

- **#rest results** – An instance of <object>.

**page-context Generic function**

**Signature** page-context () => (#rest results)

**Values**

- **#rest results** – An instance of <object>.

**process-config-element Open Generic function**

**Signature** process-config-element (server node name) => (#rest results)

**Parameters**

- **server** – An instance of <http-server>.
- **node** – An instance of <object>.

- **name** – An instance of <object>.

**Values**

- **#rest results** – An instance of <object>.

**process-request-content Open Generic function**

**Signature** process-request-content (request content-type) => (#rest results)

**Parameters**

- **request** – An instance of <request>.
- **content-type** – An instance of <object>.

**Values**

- **#rest results** – An instance of <object>.

**redirect-temporarily-to Open Generic function**

**Signature** redirect-temporarily-to (object) => (#rest results)

**Parameters**

- **object** – An instance of <object>.

**Values**

- **#rest results** – An instance of <object>.

**redirect-to Open Generic function**

**Signature** redirect-to (object) => (#rest results)

**Parameters**

- **object** – An instance of <object>.

**Values**

- **#rest results** – An instance of <object>.

**request-absolute-url Generic function**

**Signature** request-absolute-url (request) => (#rest results)

**Parameters**

- **request** – An instance of <object>.

**Values**

- **#rest results** – An instance of <object>.

**request-content-type Function**

**Signature** request-content-type (request) => (#rest results)

**Parameters**

- **request** – An instance of <request>.

**Values**

- **#rest results** – An instance of <object>.

**request-host Generic function**

**Signature** request-host (object) => (#rest results)

**Parameters**

- **object** – An instance of <object>.

**Values**

- **#rest results** – An instance of <object>.

**request-query-values Generic function**

**Signature** request-query-values (object) => (#rest results)

**Parameters**

- **object** – An instance of <object>.

**Values**

- **#rest results** – An instance of <object>.

**request-url-path-prefix Generic function**

**Signature** request-url-path-prefix (object) => (#rest results)

**Parameters**

- **object** – An instance of <object>.

**Values**

- **#rest results** – An instance of <object>.

**request-url-path-suffix Generic function**

**Signature** request-url-path-suffix (object) => (#rest results)

**Parameters**

- **object** – An instance of <object>.

**Values**

- **#rest results** – An instance of <object>.

**respond Open Generic function**

**Signature** respond (resource #key #all-keys) => (#rest results)

**Parameters**

- **resource** – An instance of <abstract-resource>.

**Values**

- **#rest results** – An instance of <object>.

**respond-to-connect Open Generic function**

**Signature** respond-to-connect (resource #key #all-keys) => (#rest results)

**Parameters**

- **resource** – An instance of <abstract-resource>.

**Values**

- **#rest results** – An instance of <object>.

**respond-to-delete Open Generic function**

**Signature** respond-to-delete (resource #key #all-keys) => (#rest results)

**Parameters**

- **resource** – An instance of `<abstract-resource>`.

**Values**

- **#rest results** – An instance of `<object>`.

**respond-to-get Open Generic function**

**Signature** `respond-to-get (resource #key #all-keys) => (#rest results)`

**Parameters**

- **resource** – An instance of `<abstract-resource>`.

**Values**

- **#rest results** – An instance of `<object>`.

**respond-to-head Open Generic function**

**Signature** `respond-to-head (resource #key #all-keys) => (#rest results)`

**Parameters**

- **resource** – An instance of `<abstract-resource>`.

**Values**

- **#rest results** – An instance of `<object>`.

**respond-to-options Open Generic function**

**Signature** `respond-to-options (resource #key #all-keys) => (#rest results)`

**Parameters**

- **resource** – An instance of `<abstract-resource>`.

**Values**

- **#rest results** – An instance of `<object>`.

**respond-to-post Open Generic function**

**Signature** `respond-to-post (resource #key #all-keys) => (#rest results)`

**Parameters**

- **resource** – An instance of `<abstract-resource>`.

**Values**

- **#rest results** – An instance of `<object>`.

**respond-to-put Open Generic function**

**Signature** `respond-to-put (resource #key #all-keys) => (#rest results)`

**Parameters**

- **resource** – An instance of `<abstract-resource>`.

**Values**

- **#rest results** – An instance of `<object>`.

**respond-to-trace Open Generic function**

**Signature** `respond-to-trace (resource #key #all-keys) => (#rest results)`

**Parameters**

- **resource** – An instance of `<abstract-resource>`.

**Values**

- **#rest results** – An instance of `<object>`.

**rewrite-url** Open Generic function

**Signature** `rewrite-url (url rule) => (url extra)`

**Parameters**

- **url** – An instance of `<string>`.
- **rule** – An instance of `<object>`.

**Values**

- **url** – An instance of `<string>`.
- **extra** – An instance of `<object>`.

**route-request** Generic function

**Signature** `route-request (server request) => (#rest results)`

**Parameters**

- **server** – An instance of `<object>`.
- **request** – An instance of `<object>`.

**Values**

- **#rest results** – An instance of `<object>`.

**serve-static-file** Generic function

**Signature** `serve-static-file (policy locator) => (#rest results)`

**Parameters**

- **policy** – An instance of `<object>`.
- **locator** – An instance of `<object>`.

**Values**

- **#rest results** – An instance of `<object>`.

**server-root** Generic function

**Signature** `server-root (object) => (#rest results)`

**Parameters**

- **object** – An instance of `<object>`.

**Values**

- **#rest results** – An instance of `<object>`.

**start-server** Open Generic function

**Signature** `start-server (server #key background wait) => (started?)`

**Parameters**

- **server** – An instance of `<http-server>`.

- **background** (*#key*) – An instance of `<boolean>`.
- **wait** (*#key*) – An instance of `<boolean>`.

**Values**

- **started?** – An instance of `<boolean>`.

**stop-server** Open Generic function

**Signature** stop-server (server *#key* abort) => (*#rest results*)

**Parameters**

- **server** – An instance of `<http-server>`.
- **abort** (*#key*) – An instance of `<object>`.

**Values**

- **#rest results** – An instance of `<object>`.

**unmatched-url-suffix** Open Generic function

**Signature** unmatched-url-suffix (resource unmatched-path) => (*#rest results*)

**Parameters**

- **resource** – An instance of `<abstract-resource>`.
- **unmatched-path** – An instance of `<sequence>`.

**Values**

- **#rest results** – An instance of `<object>`.

**use-default-virtual-host?** Generic function

**Signature** use-default-virtual-host? (object) => (*#rest results*)

**Parameters**

- **object** – An instance of `<object>`.

**Values**

- **#rest results** – An instance of `<object>`.

**with-query-values** Macro

## The HTTP-COMMON library

### The HTTP-COMMON module

`$application-error` Constant

`$bad-gateway-error` Constant

`$bad-header-error` Constant

`$bad-request-error` Constant

`$conflict-error` Constant

`$content-length-required-error` Constant

`$default-cookie-version` Constant

`$expectation-failed-error` Constant  
`$forbidden-error` Constant  
`$found-redirect` Constant  
`$gateway-timeout-error` Constant  
`$gone-error` Constant  
`$header-too-large-error` Constant  
`$http-version-not-supported-error` Constant  
`$internal-server-error` Constant  
`$method-not-allowed-error` Constant  
`$mime-wild` Constant  
`$moved-permanently-redirect` Constant  
`$moved-temporarily-redirect` Constant  
`$not-acceptable-error` Constant  
`$not-implemented-error` Constant  
`$not-modified-redirect` Constant  
`$payment-required-error` Constant  
`$precondition-failed-error` Constant  
`$proxy-authentication-required-error` Constant  
`$request-entity-too-large-error` Constant  
`$request-timeout-error` Constant  
`$request-uri-too-long-error` Constant  
`$requested-range-not-satisfiable-error` Constant  
`$resource-not-found-error` Constant  
`$see-other-redirect` Constant  
`$service-unavailable-error` Constant  
`$status-accepted` Constant  
`$status-application-error` Constant  
`$status-bad-gateway` Constant  
`$status-bad-request` Constant  
`$status-conflict` Constant  
`$status-continue` Constant  
`$status-created` Constant  
`$status-expectation-failed` Constant  
`$status-forbidden` Constant  
`$status-found` Constant  
`$status-gateway-timeout` Constant



---

`$status-gone` Constant  
`$status-http-version-not-supported` Constant  
`$status-internal-server-error` Constant  
`$status-length-required` Constant  
`$status-method-not-allowed` Constant  
`$status-moved-permanently` Constant  
`$status-multiple-choices` Constant  
`$status-no-content` Constant  
`$status-non-authoritative-information` Constant  
`$status-not-acceptable` Constant  
`$status-not-found` Constant  
`$status-not-implemented` Constant  
`$status-not-modified` Constant  
`$status-ok` Constant  
`$status-partial-content` Constant  
`$status-payment-required` Constant  
`$status-precondition-failed` Constant  
`$status-proxy-authentication-required` Constant  
`$status-request-entity-too-large` Constant  
`$status-request-timeout` Constant  
`$status-request-uri-too-long` Constant  
`$status-requested-range-not-satisfiable` Constant  
`$status-reset-content` Constant  
`$status-see-other` Constant  
`$status-service-unavailable` Constant  
`$status-switching-protocols` Constant  
`$status-temporary-redirect` Constant  
`$status-unauthorized` Constant  
`$status-unsupported-media-type` Constant  
`$status-use-proxy` Constant  
`$unauthorized-error` Constant  
`$unsupported-media-type-error` Constant  
`$use-proxy-redirect` Constant  
`*http-common-log*` Variable  
`<application-error>` Class  
    Superclasses `<http-server-protocol-error>`

**<attributes-mixin> Open Class**

**Superclasses** <object>

**Init-Keywords**

- **attributes** –

**<avalue> Class**

**Superclasses** <explicit-key-collection>

**Init-Keywords**

- **alist** –
- **value** –

**<bad-gateway-error> Class**

**Superclasses** <http-server-protocol-error>

**<bad-header-error> Class**

**Superclasses** <http-parse-error>

**<bad-request-error> Class**

**Superclasses** <http-parse-error>

**<base-http-request> Open Class**

**Superclasses** <message-headers-mixin>:http-common-internals

**Init-Keywords**

- **content** –
- **method** –
- **raw-url** –
- **url** –
- **version** –

**<base-http-response> Open Class**

**Superclasses** <message-headers-mixin>:http-common-internals

**Init-Keywords**

- **chunked** –
- **code** –
- **reason-phrase** –
- **request** –

**<chunking-input-stream> Open Class**

**Superclasses** <wrapper-stream>:streams:io

**<conflict-error> Class**

**Superclasses** <http-client-protocol-error>

**<content-length-required-error> Class**

**Superclasses** <http-client-protocol-error>

**<expectation-failed-error> Class**  
 Superclasses <http-client-protocol-error>

**<expiring-mixin> Open Class**  
 Superclasses <object>  
 Init-Keywords  
 • **duration** –

**<forbidden-error> Class**  
 Superclasses <http-client-protocol-error>

**<found-redirect> Class**  
 Superclasses <http-redirect-condition>

**<gateway-timeout-error> Class**  
 Superclasses <http-server-protocol-error>

**<gone-error> Class**  
 Superclasses <http-client-protocol-error>

**<header-table> Class**  
 Superclasses <table>

**<header-too-large-error> Class**  
 Superclasses <http-client-protocol-error>

**<http-client-protocol-error> Class**  
 Superclasses <http-protocol-condition>

**<http-error> Open Class**  
 Superclasses <format-string-condition>:dylan-extensions:dylan, <error>

**<http-parse-error> Class**  
 Superclasses <http-client-protocol-error>

**<http-protocol-condition> Open Class**  
 Superclasses <http-error>  
 Init-Keywords  
 • **code** –  
 • **headers** –

**<http-redirect-condition> Class**  
 Superclasses <http-protocol-condition>

**<http-server-protocol-error> Class**  
 Superclasses <http-protocol-condition>

**<http-version-not-supported-error> Class**  
 Superclasses <http-server-protocol-error>

**<internal-server-error> Class**

**Superclasses** <http-server-protocol-error>

**<media-type> Class**

**Superclasses** <attributes-mixin>, <mime-type>:mime:mime

**<method-not-allowed-error> Class**

**Superclasses** <http-client-protocol-error>

**<moved-permanently-redirect> Class**

**Superclasses** <http-redirect-condition>

**<moved-temporarily-redirect> Class**

**Superclasses** <http-redirect-condition>

**<not-acceptable-error> Class**

**Superclasses** <http-client-protocol-error>

**<not-implemented-error> Class**

**Superclasses** <http-server-protocol-error>

**<not-modified-redirect> Class**

**Superclasses** <http-redirect-condition>

**<payment-required-error> Class**

**Superclasses** <http-client-protocol-error>

**<precondition-failed-error> Class**

**Superclasses** <http-client-protocol-error>

**<proxy-authentication-required-error> Class**

**Superclasses** <http-client-protocol-error>

**<request-entity-too-large-error> Class**

**Superclasses** <http-client-protocol-error>

**<request-timeout-error> Class**

**Superclasses** <http-client-protocol-error>

**<request-uri-too-long-error> Class**

**Superclasses** <http-client-protocol-error>

**<requested-range-not-satisfiable-error> Class**

**Superclasses** <http-client-protocol-error>

**<resource-not-found-error> Class**

**Superclasses** <http-client-protocol-error>

**<see-other-redirect> Class**

**Superclasses** <http-redirect-condition>

**<service-unavailable-error> Class**

**Superclasses** <http-server-protocol-error>

**<unauthorized-error> Class**

**Superclasses** <http-client-protocol-error>

**<unsupported-media-type-error> Class**

**Superclasses** <http-client-protocol-error>

**<use-proxy-redirect> Class**

**Superclasses** <http-redirect-condition>

**application-error Function**

**Signature** application-error (#key headers header-name header-value message) => (#rest results)

**Parameters**

- **headers** (#key) – An instance of false-or (<header-table>).
- **header-name** (#key) – An instance of false-or (<string>).
- **header-value** (#key) – An instance of false-or (<string>).
- **message** (#key) – An instance of <object>.

**Values**

- **#rest results** – An instance of <object>.

**avalue-alist Generic function**

**Signature** avalue-alist (object) => (#rest results)

**Parameters**

- **object** – An instance of <object>.

**Values**

- **#rest results** – An instance of <object>.

**avalue-value Generic function**

**Signature** avalue-value (object) => (#rest results)

**Parameters**

- **object** – An instance of <object>.

**Values**

- **#rest results** – An instance of <object>.

**bad-gateway-error Function**

**Signature** bad-gateway-error (#key headers header-name header-value) => (#rest results)

**Parameters**

- **headers** (#key) – An instance of false-or (<header-table>).
- **header-name** (#key) – An instance of false-or (<string>).
- **header-value** (#key) – An instance of false-or (<string>).

**Values**

- **#rest results** – An instance of <object>.

**bad-header-error Function**

**Signature** bad-header-error (#key headers header-name header-value message) => (#rest results)

**Parameters**

- **headers** (*#key*) – An instance of `false-or(<header-table>)`.
- **header-name** (*#key*) – An instance of `false-or(<string>)`.
- **header-value** (*#key*) – An instance of `false-or(<string>)`.
- **message** (*#key*) – An instance of `<object>`.

**Values**

- **#rest results** – An instance of `<object>`.

**bad-request-error Function**

**Signature** `bad-request-error (#key headers header-name header-value reason) => (#rest results)`

**Parameters**

- **headers** (*#key*) – An instance of `false-or(<header-table>)`.
- **header-name** (*#key*) – An instance of `false-or(<string>)`.
- **header-value** (*#key*) – An instance of `false-or(<string>)`.
- **reason** (*#key*) – An instance of `<object>`.

**Values**

- **#rest results** – An instance of `<object>`.

**chunked-transfer-encoding? Generic function**

**Signature** `chunked-transfer-encoding? (headers) => (#rest results)`

**Parameters**

- **headers** – An instance of `<object>`.

**Values**

- **#rest results** – An instance of `<object>`.

**condition-class-for-status-code Function**

**Signature** `condition-class-for-status-code (code) => (class)`

**Parameters**

- **code** – An instance of `<integer>`.

**Values**

- **class** – An instance of `<class>`.

**conflict-error Function**

**Signature** `conflict-error (#key headers header-name header-value) => (#rest results)`

**Parameters**

- **headers** (*#key*) – An instance of `false-or(<header-table>)`.
- **header-name** (*#key*) – An instance of `false-or(<string>)`.
- **header-value** (*#key*) – An instance of `false-or(<string>)`.

**Values**

- **#rest results** – An instance of `<object>`.

**content-length Open Generic function****Signature** content-length (object) => (length)**Parameters**

- **object** – An instance of <object>.

**Values**

- **length** – An instance of false-or (<integer>).

**content-length-required-error Function****Signature** content-length-required-error (#key headers header-name header-value) => (#rest results)**Parameters**

- **headers** (#key) – An instance of false-or (<header-table>).
- **header-name** (#key) – An instance of false-or (<string>).
- **header-value** (#key) – An instance of false-or (<string>).

**Values**

- **#rest results** – An instance of <object>.

**cookie-comment Generic function****Signature** cookie-comment (object) => (#rest results)**Parameters**

- **object** – An instance of <object>.

**Values**

- **#rest results** – An instance of <object>.

**cookie-domain Generic function****Signature** cookie-domain (object) => (#rest results)**Parameters**

- **object** – An instance of <object>.

**Values**

- **#rest results** – An instance of <object>.

**cookie-max-age Generic function****Signature** cookie-max-age (object) => (#rest results)**Parameters**

- **object** – An instance of <object>.

**Values**

- **#rest results** – An instance of <object>.

**cookie-name Generic function****Signature** cookie-name (object) => (#rest results)**Parameters**

- **object** – An instance of <object>.

**Values**

- **#rest results** – An instance of `<object>`.

**cookie-path Generic function**

**Signature** `cookie-path (object) => (#rest results)`

**Parameters**

- **object** – An instance of `<object>`.

**Values**

- **#rest results** – An instance of `<object>`.

**cookie-value Generic function**

**Signature** `cookie-value (object) => (#rest results)`

**Parameters**

- **object** – An instance of `<object>`.

**Values**

- **#rest results** – An instance of `<object>`.

**cookie-version Generic function**

**Signature** `cookie-version (object) => (#rest results)`

**Parameters**

- **object** – An instance of `<object>`.

**Values**

- **#rest results** – An instance of `<object>`.

**date-modified Open Generic function**

**Signature** `date-modified (object) => (date)`

**Parameters**

- **object** – An instance of `<object>`.

**Values**

- **date** – An instance of `false-or (<date>)`.

**date-modified-setter Open Generic function**

**Signature** `date-modified-setter (new-date object) => (new-date)`

**Parameters**

- **new-date** – An instance of `false-or (<date>)`.
- **object** – An instance of `<object>`.

**Values**

- **new-date** – An instance of `false-or (<date>)`.

**expectation-failed-error Function**

**Signature** `expectation-failed-error (#key headers header-name header-value) => (#rest results)`

**Parameters**



- **headers** (*#key*) – An instance of `false-or(<header-table>)`.
- **header-name** (*#key*) – An instance of `false-or(<string>)`.
- **header-value** (*#key*) – An instance of `false-or(<string>)`.

**Values**

- **#rest results** – An instance of `<object>`.

**expired? Generic function**

**Signature** `expired? (thing) => (#rest results)`

**Parameters**

- **thing** – An instance of `<object>`.

**Values**

- **#rest results** – An instance of `<object>`.

**forbidden-error Function**

**Signature** `forbidden-error (#key headers header-name header-value) => (#rest results)`

**Parameters**

- **headers** (*#key*) – An instance of `false-or(<header-table>)`.
- **header-name** (*#key*) – An instance of `false-or(<string>)`.
- **header-value** (*#key*) – An instance of `false-or(<string>)`.

**Values**

- **#rest results** – An instance of `<object>`.

**found-redirect Function**

**Signature** `found-redirect (#key headers header-name header-value location) => (#rest results)`

**Parameters**

- **headers** (*#key*) – An instance of `false-or(<header-table>)`.
- **header-name** (*#key*) – An instance of `false-or(<string>)`.
- **header-value** (*#key*) – An instance of `false-or(<string>)`.
- **location** (*#key*) – An instance of `<object>`.

**Values**

- **#rest results** – An instance of `<object>`.

**gateway-timeout-error Function**

**Signature** `gateway-timeout-error (#key headers header-name header-value) => (#rest results)`

**Parameters**

- **headers** (*#key*) – An instance of `false-or(<header-table>)`.
- **header-name** (*#key*) – An instance of `false-or(<string>)`.
- **header-value** (*#key*) – An instance of `false-or(<string>)`.

**Values**

- **#rest results** – An instance of `<object>`.

### **get-attribute** Generic function

**Signature** get-attribute (this key) => (value)

#### **Parameters**

- **this** – An instance of <attributes-mixin>.
- **key** – An instance of <string>.

#### **Values**

- **value** – An instance of <object>.

### **get-header** Open Generic function

**Signature** get-header (object header-name #key parsed) => (header-value)

#### **Parameters**

- **object** – An instance of <object>.
- **header-name** – An instance of <byte-string>.
- **parsed** (#key) – An instance of <boolean>.

#### **Values**

- **header-value** – An instance of <object>.

### **gone-error** Function

**Signature** gone-error (#key headers header-name header-value) => (#rest results)

#### **Parameters**

- **headers** (#key) – An instance of false-or (<header-table>).
- **header-name** (#key) – An instance of false-or (<string>).
- **header-value** (#key) – An instance of false-or (<string>).

#### **Values**

- **#rest results** – An instance of <object>.

### **grow-header-buffer** Function

**Signature** grow-header-buffer (old len) => (#rest results)

#### **Parameters**

- **old** – An instance of <byte-string>.
- **len** – An instance of <integer>.

#### **Values**

- **#rest results** – An instance of <object>.

### **has-attribute?** Generic function

**Signature** has-attribute? (this key) => (has-it?)

#### **Parameters**

- **this** – An instance of <attributes-mixin>.
- **key** – An instance of <string>.

#### **Values**

- **has-it?** – An instance of `<boolean>`.

### **header-too-large-error Function**

**Signature** `header-too-large-error (#key headers header-name header-value max-size) => (#rest results)`

#### **Parameters**

- **headers** (*#key*) – An instance of `false-or (<header-table>)`.
- **header-name** (*#key*) – An instance of `false-or (<string>)`.
- **header-value** (*#key*) – An instance of `false-or (<string>)`.
- **max-size** (*#key*) – An instance of `<object>`.

#### **Values**

- **#rest results** – An instance of `<object>`.

### **http-error-headers Generic function**

**Signature** `http-error-headers (error) => (headers)`

#### **Parameters**

- **error** – An instance of `<error>`.

#### **Values**

- **headers** – An instance of `false-or (<header-table>)`.

### **http-error-message-no-code Generic function**

**Signature** `http-error-message-no-code (error) => (#rest results)`

#### **Parameters**

- **error** – An instance of `<object>`.

#### **Values**

- **#rest results** – An instance of `<object>`.

### **http-status-code Generic function**

**Signature** `http-status-code (error) => (code)`

#### **Parameters**

- **error** – An instance of `<error>`.

#### **Values**

- **code** – An instance of `<integer>`.

### **http-version-not-supported-error Function**

**Signature** `http-version-not-supported-error (#key headers header-name header-value version) => (#rest results)`

#### **Parameters**

- **headers** (*#key*) – An instance of `false-or (<header-table>)`.
- **header-name** (*#key*) – An instance of `false-or (<string>)`.
- **header-value** (*#key*) – An instance of `false-or (<string>)`.
- **version** (*#key*) – An instance of `<object>`.

**Values**

- **#rest results** – An instance of `<object>`.

**internal-server-error Function**

**Signature** `internal-server-error (#key headers header-name header-value) => (#rest results)`

**Parameters**

- **headers** (*#key*) – An instance of `false-or (<header-table>)`.
- **header-name** (*#key*) – An instance of `false-or (<string>)`.
- **header-value** (*#key*) – An instance of `false-or (<string>)`.

**Values**

- **#rest results** – An instance of `<object>`.

**match-media-types Generic function**

**Signature** `match-media-types (type1 type2) => (#rest results)`

**Parameters**

- **type1** – An instance of `<object>`.
- **type2** – An instance of `<object>`.

**Values**

- **#rest results** – An instance of `<object>`.

**media-type-exact? Generic function**

**Signature** `media-type-exact? (mr) => (#rest results)`

**Parameters**

- **mr** – An instance of `<object>`.

**Values**

- **#rest results** – An instance of `<object>`.

**media-type-level Generic function**

**Signature** `media-type-level (media-type) => (#rest results)`

**Parameters**

- **media-type** – An instance of `<object>`.

**Values**

- **#rest results** – An instance of `<object>`.

**media-type-more-specific? Generic function**

**Signature** `media-type-more-specific? (type1 type2) => (#rest results)`

**Parameters**

- **type1** – An instance of `<object>`.
- **type2** – An instance of `<object>`.

**Values**

- **#rest results** – An instance of `<object>`.

**media-type-quality Generic function**

**Signature** `media-type-quality (media-type) => (#rest results)`

**Parameters**

- **media-type** – An instance of `<object>`.

**Values**

- **#rest results** – An instance of `<object>`.

**method-not-allowed-error Function**

**Signature** `method-not-allowed-error (#key headers header-name header-value request-method) => (#rest results)`

**Parameters**

- **headers** (*#key*) – An instance of `false-or (<header-table>)`.
- **header-name** (*#key*) – An instance of `false-or (<string>)`.
- **header-value** (*#key*) – An instance of `false-or (<string>)`.
- **request-method** (*#key*) – An instance of `<object>`.

**Values**

- **#rest results** – An instance of `<object>`.

**moved-permanently-redirect Function**

**Signature** `moved-permanently-redirect (#key headers header-name header-value location) => (#rest results)`

**Parameters**

- **headers** (*#key*) – An instance of `false-or (<header-table>)`.
- **header-name** (*#key*) – An instance of `false-or (<string>)`.
- **header-value** (*#key*) – An instance of `false-or (<string>)`.
- **location** (*#key*) – An instance of `<object>`.

**Values**

- **#rest results** – An instance of `<object>`.

**moved-temporarily-redirect Function**

**Signature** `moved-temporarily-redirect (#key headers header-name header-value location) => (#rest results)`

**Parameters**

- **headers** (*#key*) – An instance of `false-or (<header-table>)`.
- **header-name** (*#key*) – An instance of `false-or (<string>)`.
- **header-value** (*#key*) – An instance of `false-or (<string>)`.
- **location** (*#key*) – An instance of `<object>`.

**Values**

- **#rest results** – An instance of `<object>`.

**not-acceptable-error Function**

**Signature** not-acceptable-error (#key headers header-name header-value) => (#rest results)

**Parameters**

- **headers** (#key) – An instance of false-or (<header-table>).
- **header-name** (#key) – An instance of false-or (<string>).
- **header-value** (#key) – An instance of false-or (<string>).

**Values**

- **#rest results** – An instance of <object>.

**not-implemented-error Function**

**Signature** not-implemented-error (#key headers header-name header-value what) => (#rest results)

**Parameters**

- **headers** (#key) – An instance of false-or (<header-table>).
- **header-name** (#key) – An instance of false-or (<string>).
- **header-value** (#key) – An instance of false-or (<string>).
- **what** (#key) – An instance of <object>.

**Values**

- **#rest results** – An instance of <object>.

**not-modified-redirect Function**

**Signature** not-modified-redirect (#key headers header-name header-value) => (#rest results)

**Parameters**

- **headers** (#key) – An instance of false-or (<header-table>).
- **header-name** (#key) – An instance of false-or (<string>).
- **header-value** (#key) – An instance of false-or (<string>).

**Values**

- **#rest results** – An instance of <object>.

**note-bytes-received Open Generic function**

**Signature** note-bytes-received (stream byte-count) => (#rest results)

**Parameters**

- **stream** – An instance of <chunking-input-stream>.
- **byte-count** – An instance of <integer>.

**Values**

- **#rest results** – An instance of <object>.

**parse-header-value Open Generic function**

**Signature** parse-header-value (field-name field-values) => (parsed-field-value)

**Parameters**

- **field-name** – An instance of <symbol>.
- **field-values** – An instance of <field-type>:http-common-internals.

**Values**

- **parsed-field-value** – An instance of `<object>`.

**parse-http-date Function**

**Signature** `parse-http-date (str bpos epos) => (date)`

**Parameters**

- **str** – An instance of `<byte-string>`.
- **bpos** – An instance of `<integer>`.
- **epos** – An instance of `<integer>`.

**Values**

- **date** – An instance of `false-or (<date>)`.

**parsed-headers Generic function**

**Signature** `parsed-headers (object) => (#rest results)`

**Parameters**

- **object** – An instance of `<object>`.

**Values**

- **#rest results** – An instance of `<object>`.

**payment-required-error Function**

**Signature** `payment-required-error (#key headers header-name header-value) => (#rest results)`

**Parameters**

- **headers (#key)** – An instance of `false-or (<header-table>)`.
- **header-name (#key)** – An instance of `false-or (<string>)`.
- **header-value (#key)** – An instance of `false-or (<string>)`.

**Values**

- **#rest results** – An instance of `<object>`.

**precondition-failed-error Function**

**Signature** `precondition-failed-error (#key headers header-name header-value) => (#rest results)`

**Parameters**

- **headers (#key)** – An instance of `false-or (<header-table>)`.
- **header-name (#key)** – An instance of `false-or (<string>)`.
- **header-value (#key)** – An instance of `false-or (<string>)`.

**Values**

- **#rest results** – An instance of `<object>`.

**proxy-authentication-required-error Function**

**Signature** `proxy-authentication-required-error (#key headers header-name header-value) => (#rest results)`

**Parameters**

- **headers** (*#key*) – An instance of `false-or(<header-table>)`.
- **header-name** (*#key*) – An instance of `false-or(<string>)`.
- **header-value** (*#key*) – An instance of `false-or(<string>)`.

**Values**

- **#rest results** – An instance of `<object>`.

**quote-html Function**

**Signature** `quote-html (text #key stream) => (#rest results)`

**Parameters**

- **text** – An instance of `<string>`.
- **stream** (*#key*) – An instance of `<object>`.

**Values**

- **#rest results** – An instance of `<object>`.

**raw-headers Generic function**

**Signature** `raw-headers (object) => (#rest results)`

**Parameters**

- **object** – An instance of `<object>`.

**Values**

- **#rest results** – An instance of `<object>`.

**read-header-line Function**

**Signature** `read-header-line (stream buffer bpos peek-ch require-crlf?) => (buffer bpos epos peek-ch)`

**Parameters**

- **stream** – An instance of `<stream>:common-extensions:common-dylan`.
- **buffer** – An instance of `<byte-string>`.
- **bpos** – An instance of `<integer>`.
- **peek-ch** – An instance of `false-or(<byte-character>)`.
- **require-crlf?** – An instance of `<boolean>`.

**Values**

- **buffer** – An instance of `<byte-string>`.
- **bpos** – An instance of `<integer>`.
- **epos** – An instance of `<integer>`.
- **peek-ch** – An instance of `false-or(<byte-character>)`.

**read-http-line Generic function**

**Signature** `read-http-line (stream) => (#rest results)`

**Parameters**

- **stream** – An instance of `<object>`.



**Values**

- **#rest results** – An instance of <object>.

**read-message-headers Function**

**Signature** read-message-headers (stream #key buffer start headers require-crlf?) => (headers buffer epos)

**Parameters**

- **stream** – An instance of <stream>:common-extensions:common-dylan.
- **buffer** (#key) – An instance of <byte-string>.
- **start** (#key) – An instance of <integer>.
- **headers** (#key) – An instance of <header-table>.
- **require-crlf?** (#key) – An instance of <boolean>.

**Values**

- **headers** – An instance of <header-table>.
- **buffer** – An instance of <byte-string>.
- **epos** – An instance of <integer>.

**remove-attribute Generic function**

**Signature** remove-attribute (this key) => (#rest results)

**Parameters**

- **this** – An instance of <attributes-mixin>.
- **key** – An instance of <string>.

**Values**

- **#rest results** – An instance of <object>.

**request-content Generic function**

**Signature** request-content (object) => (#rest results)

**Parameters**

- **object** – An instance of <object>.

**Values**

- **#rest results** – An instance of <object>.

**request-content-setter Generic function**

**Signature** request-content-setter (value object) => (#rest results)

**Parameters**

- **value** – An instance of <object>.
- **object** – An instance of <object>.

**Values**

- **#rest results** – An instance of <object>.

**request-entity-too-large-error Function**

**Signature** request-entity-too-large-error (#key headers header-name header-value max-size) => (#rest results)

**Parameters**

- **headers** (#key) – An instance of false-or(<header-table>).
- **header-name** (#key) – An instance of false-or(<string>).
- **header-value** (#key) – An instance of false-or(<string>).
- **max-size** (#key) – An instance of <object>.

**Values**

- **#rest results** – An instance of <object>.

**request-method Generic function**

**Signature** request-method (object) => (#rest results)

**Parameters**

- **object** – An instance of <object>.

**Values**

- **#rest results** – An instance of <object>.

**request-method-setter Generic function**

**Signature** request-method-setter (value object) => (#rest results)

**Parameters**

- **value** – An instance of <object>.
- **object** – An instance of <object>.

**Values**

- **#rest results** – An instance of <object>.

**request-raw-url-string Generic function**

**Signature** request-raw-url-string (object) => (#rest results)

**Parameters**

- **object** – An instance of <object>.

**Values**

- **#rest results** – An instance of <object>.

**request-raw-url-string-setter Generic function**

**Signature** request-raw-url-string-setter (value object) => (#rest results)

**Parameters**

- **value** – An instance of <object>.
- **object** – An instance of <object>.

**Values**

- **#rest results** – An instance of <object>.

**request-timeout-error Function**

**Signature** request-timeout-error (#key headers header-name header-value seconds) => (#rest results)

**Parameters**

- **headers** (#key) – An instance of false-or (<header-table>).
- **header-name** (#key) – An instance of false-or (<string>).
- **header-value** (#key) – An instance of false-or (<string>).
- **seconds** (#key) – An instance of <object>.

**Values**

- **#rest results** – An instance of <object>.

**request-uri-too-long-error Function**

**Signature** request-uri-too-long-error (#key headers header-name header-value max-size) => (#rest results)

**Parameters**

- **headers** (#key) – An instance of false-or (<header-table>).
- **header-name** (#key) – An instance of false-or (<string>).
- **header-value** (#key) – An instance of false-or (<string>).
- **max-size** (#key) – An instance of <object>.

**Values**

- **#rest results** – An instance of <object>.

**request-url Generic function**

**Signature** request-url (object) => (#rest results)

**Parameters**

- **object** – An instance of <object>.

**Values**

- **#rest results** – An instance of <object>.

**request-url-setter Generic function**

**Signature** request-url-setter (value object) => (#rest results)

**Parameters**

- **value** – An instance of <object>.
- **object** – An instance of <object>.

**Values**

- **#rest results** – An instance of <object>.

**request-version Generic function**

**Signature** request-version (object) => (#rest results)

**Parameters**

- **object** – An instance of <object>.

**Values**

- **#rest results** – An instance of `<object>`.

#### **request-version-setter Generic function**

**Signature** request-version-setter (value object) => (#rest results)

##### **Parameters**

- **value** – An instance of `<object>`.
- **object** – An instance of `<object>`.

##### **Values**

- **#rest results** – An instance of `<object>`.

#### **requested-range-not-satisfiable-error Function**

**Signature** requested-range-not-satisfiable-error (#key headers header-name header-value) => (#rest results)

##### **Parameters**

- **headers** (#key) – An instance of `false-or (<header-table>)`.
- **header-name** (#key) – An instance of `false-or (<string>)`.
- **header-value** (#key) – An instance of `false-or (<string>)`.

##### **Values**

- **#rest results** – An instance of `<object>`.

#### **resource-not-found-error Function**

**Signature** resource-not-found-error (#key headers header-name header-value url) => (#rest results)

##### **Parameters**

- **headers** (#key) – An instance of `false-or (<header-table>)`.
- **header-name** (#key) – An instance of `false-or (<string>)`.
- **header-value** (#key) – An instance of `false-or (<string>)`.
- **url** (#key) – An instance of `<object>`.

##### **Values**

- **#rest results** – An instance of `<object>`.

#### **response-chunked? Generic function**

**Signature** response-chunked? (object) => (#rest results)

##### **Parameters**

- **object** – An instance of `<object>`.

##### **Values**

- **#rest results** – An instance of `<object>`.

#### **response-chunked?-setter Generic function**

**Signature** response-chunked?-setter (value object) => (#rest results)

##### **Parameters**

- **value** – An instance of `<object>`.

- **object** – An instance of <object>.

**Values**

- **#rest results** – An instance of <object>.

**response-code Generic function**

**Signature** response-code (object) => (#rest results)

**Parameters**

- **object** – An instance of <object>.

**Values**

- **#rest results** – An instance of <object>.

**response-code-setter Generic function**

**Signature** response-code-setter (value object) => (#rest results)

**Parameters**

- **value** – An instance of <object>.
- **object** – An instance of <object>.

**Values**

- **#rest results** – An instance of <object>.

**response-reason-phrase Generic function**

**Signature** response-reason-phrase (object) => (#rest results)

**Parameters**

- **object** – An instance of <object>.

**Values**

- **#rest results** – An instance of <object>.

**response-reason-phrase-setter Generic function**

**Signature** response-reason-phrase-setter (value object) => (#rest results)

**Parameters**

- **value** – An instance of <object>.
- **object** – An instance of <object>.

**Values**

- **#rest results** – An instance of <object>.

**response-request Generic function**

**Signature** response-request (object) => (#rest results)

**Parameters**

- **object** – An instance of <object>.

**Values**

- **#rest results** – An instance of <object>.

**see-other-redirect Function**

**Signature** see-other-redirect (#key headers header-name header-value location) => (#rest results)

**Parameters**

- **headers** (#key) – An instance of false-or (<header-table>).
- **header-name** (#key) – An instance of false-or (<string>).
- **header-value** (#key) – An instance of false-or (<string>).
- **location** (#key) – An instance of <object>.

**Values**

- **#rest results** – An instance of <object>.

**service-unavailable-error Function**

**Signature** service-unavailable-error (#key headers header-name header-value) => (#rest results)

**Parameters**

- **headers** (#key) – An instance of false-or (<header-table>).
- **header-name** (#key) – An instance of false-or (<string>).
- **header-value** (#key) – An instance of false-or (<string>).

**Values**

- **#rest results** – An instance of <object>.

**set-attribute Generic function**

**Signature** set-attribute (this key value) => (#rest results)

**Parameters**

- **this** – An instance of <attributes-mixin>.
- **key** – An instance of <string>.
- **value** – An instance of <object>.

**Values**

- **#rest results** – An instance of <object>.

**set-header Open Generic function**

**Signature** set-header (object header value #key if-exists?) => (#rest results)

**Parameters**

- **object** – An instance of <object>.
- **header** – An instance of <byte-string>.
- **value** – An instance of <object>.
- **if-exists?** (#key) – An instance of one-of ("replace", "append", "ignore", "error").

**Values**

- **#rest results** – An instance of <object>.

**token-end-position Function**

**Signature** token-end-position (buf bpos epos) => (#rest results)

**Parameters**

- **buf** – An instance of `<byte-string>`.
- **bpos** – An instance of `<integer>`.
- **epos** – An instance of `<integer>`.

**Values**

- **#rest results** – An instance of `<object>`.

**unauthorized-error Function**

**Signature** `unauthorized-error (#key headers header-name header-value) => (#rest results)`

**Parameters**

- **headers** (*#key*) – An instance of `false-or (<header-table>)`.
- **header-name** (*#key*) – An instance of `false-or (<string>)`.
- **header-value** (*#key*) – An instance of `false-or (<string>)`.

**Values**

- **#rest results** – An instance of `<object>`.

**unsupported-media-type-error Function**

**Signature** `unsupported-media-type-error (#key headers header-name header-value) => (#rest results)`

**Parameters**

- **headers** (*#key*) – An instance of `false-or (<header-table>)`.
- **header-name** (*#key*) – An instance of `false-or (<string>)`.
- **header-value** (*#key*) – An instance of `false-or (<string>)`.

**Values**

- **#rest results** – An instance of `<object>`.

**use-proxy-redirect Function**

**Signature** `use-proxy-redirect (#key headers header-name header-value location) => (#rest results)`

**Parameters**

- **headers** (*#key*) – An instance of `false-or (<header-table>)`.
- **header-name** (*#key*) – An instance of `false-or (<string>)`.
- **header-value** (*#key*) – An instance of `false-or (<string>)`.
- **location** (*#key*) – An instance of `<object>`.

**Values**

- **#rest results** – An instance of `<object>`.

**validate-http-status-code Generic function**

**Signature** `validate-http-status-code (status-code) => (#rest results)`

**Parameters**

- **status-code** – An instance of `<object>`.

**Values**

- **#rest results** – An instance of `<object>`.

**validate-http-version** Generic function

**Signature** `validate-http-version (version) => (#rest results)`

**Parameters**

- **version** – An instance of `<object>`.

**Values**

- **#rest results** – An instance of `<object>`.

## The DSP library

### The DSP module

**<dylan-server-page>** Open Primary Class

**Superclasses** `<expiring-mixin>`, `<resource>`

**Init-Keywords**

- **source** –

**<page-link>** Open Class

**Superclasses** `<object>`

**Init-Keywords**

- **label** –
- **page-number** –

**<paginator>** Open Class

**Superclasses** `<sequence>`

**Init-Keywords**

- **current-page-number** –
- **page-size** –
- **sequence** –

**<taglib>** Class

**Superclasses** `<object>`

**Init-Keywords**

- **name** –
- **prefix** –

**add-field-error** Generic function

**Signature** `add-field-error (field-name message #rest format-arguments) => (#rest results)`

**Parameters**

- **field-name** – An instance of `<object>`.



- **message** – An instance of `<object>`.
- **format-arguments** (*#rest*) – An instance of `<object>`.

**Values**

- **#rest results** – An instance of `<object>`.

**add-page-error Generic function**

**Signature** `add-page-error (format-string #rest format-arguments) => (#rest results)`

**Parameters**

- **format-string** – An instance of `<object>`.
- **format-arguments** (*#rest*) – An instance of `<object>`.

**Values**

- **#rest results** – An instance of `<object>`.

**add-page-note Generic function**

**Signature** `add-page-note (format-string #rest format-arguments) => (#rest results)`

**Parameters**

- **format-string** – An instance of `<object>`.
- **format-arguments** (*#rest*) – An instance of `<object>`.

**Values**

- **#rest results** – An instance of `<object>`.

**current-page-number Generic function**

**Signature** `current-page-number (object) => (#rest results)`

**Parameters**

- **object** – An instance of `<object>`.

**Values**

- **#rest results** – An instance of `<object>`.

**current-page-number-setter Generic function**

**Signature** `current-page-number-setter (value object) => (#rest results)`

**Parameters**

- **value** – An instance of `<object>`.
- **object** – An instance of `<object>`.

**Values**

- **#rest results** – An instance of `<object>`.

**current-row Function**

**Signature** `current-row () => (#rest results)`

**Values**

- **#rest results** – An instance of `<object>`.

**current-row-number Function**

**Signature** `current-row-number () => (#rest results)`

**Values**

- **#rest results** – An instance of `<object>`.

**get-field-errors** Generic function

**Signature** `get-field-errors (field-name) => (#rest results)`

**Parameters**

- **field-name** – An instance of `<object>`.

**Values**

- **#rest results** – An instance of `<object>`.

**get-named-method** Generic function

**Signature** `get-named-method (taglib name) => (#rest results)`

**Parameters**

- **taglib** – An instance of `<object>`.
- **name** – An instance of `<object>`.

**Values**

- **#rest results** – An instance of `<object>`.

**get-tag-call-attribute** Function

**Signature** `get-tag-call-attribute (attr #key as test) => (attribute-value)`

**Parameters**

- **attr** – An instance of `<object>`.
- **as (#key)** – An instance of `<type>`.
- **test (#key)** – An instance of `<object>`.

**Values**

- **attribute-value** – An instance of `<object>`.

**loop-index** Generic function

**Signature** `loop-index () => (#rest results)`

**Values**

- **#rest results** – An instance of `<object>`.

**loop-value** Generic function

**Signature** `loop-value () => (#rest results)`

**Values**

- **#rest results** – An instance of `<object>`.

**map-tag-call-attributes** Function

**Signature** `map-tag-call-attributes (f #key exclude) => (#rest results)`

**Parameters**

- **f** – An instance of `<function>`.

- **exclude** (*#key*) – An instance of `<sequence>`.

**Values**

- **#rest results** – An instance of `<object>`.

**named-method-definer Macro****next-page-number Generic function**

**Signature** `next-page-number (paginator) => (#rest results)`

**Parameters**

- **paginator** – An instance of `<object>`.

**Values**

- **#rest results** – An instance of `<object>`.

**page-count Open Generic function**

**Signature** `page-count (paginator) => (count)`

**Parameters**

- **paginator** – An instance of `<paginator>`.

**Values**

- **count** – An instance of `<integer>`.

**page-has-errors? Generic function**

**Signature** `page-has-errors? () => (#rest results)`

**Values**

- **#rest results** – An instance of `<object>`.

**page-link-label Generic function**

**Signature** `page-link-label (object) => (#rest results)`

**Parameters**

- **object** – An instance of `<object>`.

**Values**

- **#rest results** – An instance of `<object>`.

**page-link-page-number Generic function**

**Signature** `page-link-page-number (object) => (#rest results)`

**Parameters**

- **object** – An instance of `<object>`.

**Values**

- **#rest results** – An instance of `<object>`.

**page-links Open Generic function**

**Signature** `page-links (paginator #key ellipsis prev next center-span min-pages) => (page-links)`

**Parameters**

- **paginator** – An instance of `<paginator>`.

- **ellipsis** (*#key*) – An instance of `false-or(<string>)`.
- **prev** (*#key*) – An instance of `false-or(<string>)`.
- **next** (*#key*) – An instance of `false-or(<string>)`.
- **center-span** (*#key*) – An instance of `false-or(<integer>)`.
- **min-pages** (*#key*) – An instance of `false-or(<integer>)`.

**Values**

- **page-links** – An instance of `<sequence>`.

**page-size** Generic function

**Signature** `page-size (object) => (#rest results)`

**Parameters**

- **object** – An instance of `<object>`.

**Values**

- **#rest results** – An instance of `<object>`.

**page-source** Generic function

**Signature** `page-source (page) => (#rest results)`

**Parameters**

- **page** – An instance of `<object>`.

**Values**

- **#rest results** – An instance of `<object>`.

**page-template** Generic function

**Signature** `page-template (object) => (#rest results)`

**Parameters**

- **object** – An instance of `<object>`.

**Values**

- **#rest results** – An instance of `<object>`.

**page-template-setter** Generic function

**Signature** `page-template-setter (value object) => (#rest results)`

**Parameters**

- **value** – An instance of `<object>`.
- **object** – An instance of `<object>`.

**Values**

- **#rest results** – An instance of `<object>`.

**paginator-sequence** Generic function

**Signature** `paginator-sequence (object) => (#rest results)`

**Parameters**

- **object** – An instance of `<object>`.

**Values**

- **#rest results** – An instance of `<object>`.

**previous-page-number Generic function**

**Signature** `previous-page-number (paginator) => (#rest results)`

**Parameters**

- **paginator** – An instance of `<object>`.

**Values**

- **#rest results** – An instance of `<object>`.

**process-template Generic function**

**Signature** `process-template (page) => (#rest results)`

**Parameters**

- **page** – An instance of `<object>`.

**Values**

- **#rest results** – An instance of `<object>`.

**register-tag Generic function**

**Signature** `register-tag (tag taglib) => (#rest results)`

**Parameters**

- **tag** – An instance of `<object>`.
- **taglib** – An instance of `<object>`.

**Values**

- **#rest results** – An instance of `<object>`.

**show-tag-call-attributes Function**

**Signature** `show-tag-call-attributes (stream #key exclude) => (#rest results)`

**Parameters**

- **stream** – An instance of `<object>`.
- **exclude** (`#key`) – An instance of `<sequence>`.

**Values**

- **#rest results** – An instance of `<object>`.

**tag-definer Macro**

Defines a new tag in the given tag library.

**Signature** `define [modifiers] tag tag-name [in taglib-name] (method-parameters) (tag-call-parameters) body end`

**Parameters**

- **modifiers** – The only valid modifier is `body`, which must be used if the tag allows nested body elements. If `body` is not specified then the tag call must end in `/>` or an error will be signalled when the DSP template is parsed. If `body` is specified, `method-parameters` must have a third parameter (see below).
- **tag-name** – The name of the tag, as it will appear in the `.dsp` file.

- **taglib-name** – The name of the taglib the tag should be added to.
- **method-parameters** – Each tag definition creates a method that will be called when the tag is invoked. This is the parameter list for that method. The basic form of the parameter list is (page[, process-body]). page is an instance of `<dylan-server-page>`. process-body is an instance of `<function>`. The process-body argument should be specified if and only if the body modifier is supplied.
- **tag-call-parameters** – tag-call-parameters allows you to receive named keyword arguments from a tag call. For example, if your tag call looks like `<xx:foo arg1="one" arg2="2">` then tag-call-parameters might look like (arg1, arg2) in which case body code may refer to arg1 and arg2. If the tag call doesn't specify a given tag-call-parameter then #f will be used. If a tag-call-parameter has a type specifier, then the passed argument will be parsed into the appropriate type before it is passed. See the parse-tag-arg generic function. Note that this means specifying a type of which #f is not a member effectively makes the tag-call-parameter required. For example, (arg1, arg2 :: <integer>) specifies that arg1 is optional (it will be a <string> if supplied) and arg2 is required and must be parsable to an <integer>.
- **body** – The body of the tag definition. method-parameter and tag-call-parameters are bound within the body.

### Discussion

Defines a new tag named tag-name in the taglib-name tag library. For simple DSP tags with no body elements, the body code normally just does output to the output stream of the current response, generating dynamic output in place of the literal tag call in the source file. Tags that have body elements may additionally want to setup state for nested tags to use. This may be done, for example, through the use of dynamically bound thread variables or storing information in the session or page context.

When the DSP engine invokes the tag to generate dynamic content it passes arguments that match method-parameters. tag-call-parameters receive arguments specified in the tag call, in the DSP source file, after they have been parsed to the specified types.

**Example** A simple tag in the “demo” taglib that displays “Hello, world!” in the page. It is invoked with `<demo:hello/>`:

```
define tag hello in demo
  (page :: <dylan-server-page>)
  ()
  format(output-stream(current-response()), "Hello, world!");
end;
```

A tag that allows body elements, and processes the body elements three times. It is invoked with `<demo:three-times>...whatever...</demo:three-times>`:

```
define body tag three-times in demo
  (page :: <dylan-server-page>,
   do-body :: <function>)
  ()
  for (i from 1 to 3)
    do-body();
  end;
end;
```

### taglib-definer Macro

Defines a new tag library.

**Signature** `define taglib taglib-name () end`

**Parameters**

- **taglib-name** – The name of the tag library.

**validate-form-field** Generic function

**Signature** `validate-form-field (field-name validator) => (#rest results)`

**Parameters**

- **field-name** – An instance of `<object>`.
- **validator** – An instance of `<object>`.

**Values**

- **#rest results** – An instance of `<object>`.

## The HTTP-CLIENT library

### The HTTP-CLIENT module

#### Requests

**http-request** Sealed Generic function

Perform a complete HTTP request and response.

**Signature** `http-request (request-method url #key headers parameters content follow-redirects stream) => (response)`

**Parameters**

- **request-method** – An instance of `<byte-string>`
- **url** – A string or a URI.
- **headers** (*#key*) – A `<table>` of strings mapping header names to values.
- **parameters** (*#key*) – A `<table>` of strings mapping query parameter names to values.
- **content** (*#key*) – An instance of `<string>` or `<table>`. If given a `<table>` it will be converted to `application/x-www-urlencoded-form`.
- **follow-redirects** (*#key*) – A `<boolean>` or `<integer>`. If an integer, this is the maximum number of redirects to follow. `#t` means no limit.
- **stream** (*#key*) – A stream to which response content will be copied. If this is provided the response data will not be stored in the `response-content` slot.

**Values**

- **response** – An instance of `<http-response>`.

**http-delete** Constant

Convenience function for performing a DELETE request, equivalent to `curry(http-request, "DELETE")`.

**http-get** Constant

Convenience function for performing a GET request, equivalent to `curry(http-request, "GET")`.

**http-head** Constant

Convenience function for performing a HEAD request, equivalent to `curry(http-request, "HEAD")`.

### **http-options Constant**

Convenience function for performing an OPTIONS request, equivalent to `curry(http-request, "OPTIONS")`.

### **http-post Constant**

Convenience function for performing a POST request, equivalent to `curry(http-request, "POST")`.

### **http-put Constant**

Convenience function for performing a PUT request, equivalent to `curry(http-request, "PUT")`.

### **send-request Generic function**

Send an HTTP request over an existing connection. This is a low-level API.

**Signature** `send-request (conn request-method url #rest start-request-args #key content #all-keys) => ()`

#### **Parameters**

- **conn** – An instance of `<http-connection>`.
- **request-method** – An instance of `<byte-string>`.
- **url** – An instance `<string>` or `<uri>`.
- **start-request-args** (*#rest*) – An instance of `<object>`.
- **content** (*#key*) – An instance of `<byte-string>`.

### **start-request Generic function**

Send the request line and request headers over an existing connection but do not send any content. This is a low-level API.

**Signature** `start-request (conn request-method url #key headers standard-headers http-version) => ()`

#### **Parameters**

- **conn** – An instance of `<http-connection>`.
- **request-method** – An instance of `<byte-string>`.
- **url** – An instance `<string>` or `<uri>`.
- **headers** (*#key*) – An instance of `<object>`.
- **standard-headers** (*#key*) – An instance of `<object>`.
- **http-version** (*#key*) – An instance of `<byte-string>`.

### **finish-request Generic function**

Finish sending a request over an existing connection by, for example, sending a zero-length chunk in a chunked request. This is a low-level API.

**Signature** `finish-request (conn) => ()`

#### **Parameters**

- **conn** – An instance of `<http-connection>`.

## Responses

### **<http-response> Open Primary Class**

All HTTP requests result in an instance of this class being created.

**Superclasses** `<chunking-input-stream>`, `<base-http-response>`,  
`<message-headers-mixin>`

#### **Init-Keywords**



- **content** –

### **read-response** Open Generic function

Read the content of a response from an existing connection after headers have already been read. This is a low-level API.

**Signature** `read-response (conn #key read-content response-class) => (response)`

#### **Parameters**

- **conn** – An instance of `<http-connection>`.
- **read-content** (*#key*) – An instance of `<boolean>`.
- **response-class** (*#key*) – A subclass of `<http-response>`. The default is `<http-response>`.

#### **Values**

- **response** – An instance of `<http-response>`.

### **response-content** Generic function

Fetch the content of a response from a `<http-response>`.

**Signature** `response-content (response) => (content)`

#### **Parameters**

- **response** – An instance of `<http-response>`.

#### **Values**

- **content** – `#f` or an instance of `<byte-string>`.

## **Errors**

### **<maximum-redirects-exceeded>** Open Class

Signaled when an HTTP request results in too many redirects, based on the `follow-redirects:` keyword argument.

**Superclasses** `<http-error>`

### **<redirect-loop-detected>** Open Class

Signaled when an HTTP request results in a redirect loop.

**Superclasses** `<http-error>`

## **Connections**

### **with-http-connection** Macro

Open an HTTP connection and keep it open by automatically adding “Connection: Keep-alive” headers. For most uses, this should be the only connection-related API needed.

Example:

```
with-http-connection (conn = "opendylan.org", port: 80, outgoing-chunk-size: 8192)
  send-request (conn, "GET", "/")
  let response :: <http-response> = read-response (conn);
  ...content is in response.response-content...

  send-request (conn, "POST", "/blah", content: "...");
  let response :: <http-response> = read-response (conn);
```

```
...
end;
```

Note that the port and outgoing-chunk-size specified above are the default values. It is also possible to supply a <uri> instead of a hostname and port number:

```
with-http-connection (conn = uri) ... end
```

### <http-connection> Open Class

**Superclasses** <basic-stream>:streams:io

#### Init-Keywords

- **host** –
- **outgoing-chunk-size** –

#### connection-host Generic function

**Signature** connection-host (object) => (#rest results)

#### Parameters

- **object** – An instance of <object>.

#### Values

- **#rest results** – An instance of <object>.

#### connection-port Generic function

**Signature** connection-port (conn) => (#rest results)

#### Parameters

- **conn** – An instance of <object>.

#### Values

- **#rest results** – An instance of <object>.

#### make-http-connection Function

**Signature** make-http-connection (host-or-url #rest initargs #key port #all-keys) => (#rest results)

#### Parameters

- **host-or-url** – An instance of <object>.
- **initargs** (#rest) – An instance of <object>.
- **port** (#key) – An instance of <object>.

#### Values

- **#rest results** – An instance of <object>.

#### note-bytes-sent Open Generic function

**Signature** note-bytes-sent (conn byte-count) => (#rest results)

#### Parameters

- **conn** – An instance of <http-connection>.
- **byte-count** – An instance of <integer>.

#### Values

- **#rest results** – An instance of `<object>`.

**outgoing-chunk-size** Generic function

**Signature** `outgoing-chunk-size (object) => (#rest results)`

**Parameters**

- **object** – An instance of `<object>`.

**Values**

- **#rest results** – An instance of `<object>`.

**outgoing-chunk-size-setter** Generic function

**Signature** `outgoing-chunk-size-setter (value object) => (#rest results)`

**Parameters**

- **value** – An instance of `<object>`.
- **object** – An instance of `<object>`.

**Values**

- **#rest results** – An instance of `<object>`.

**Miscellaneous**

**\*http-client-log\*** Variable



**COPYRIGHT**

Copyright © 2001-2012 Dylan Hackers.

Companies, names and data used in examples herein are fictitious unless otherwise noted.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the “Software”), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED “AS IS”, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Other brand or product names are the registered trademarks or trademarks of their respective holders.



## INDICES AND TABLES

- genindex
- search





## A

\$application-error (*constant*), 27  
 <abstract-resource> (*class*), 15  
 <abstract-rewrite-rule> (*class*), 15  
 <abstract-router> (*class*), 15  
 <application-error> (*class*), 29  
 <attributes-mixin> (*class*), 29  
 <a-value> (*class*), 30  
 add-cookie (*generic function*), 17  
 add-field-error (*generic function*), 52  
 add-page-error (*generic function*), 53  
 add-page-note (*generic function*), 53  
 add-resource (*generic function*), 17  
 add-resource-name (*generic function*), 17  
 add-virtual-host (*generic function*), 17  
 application-error (*function*), 33  
 avalue-alist (*generic function*), 33  
 avalue-value (*generic function*), 33

## B

\$bad-gateway-error (*constant*), 27  
 \$bad-header-error (*constant*), 27  
 \$bad-request-error (*constant*), 27  
 <bad-gateway-error> (*class*), 30  
 <bad-header-error> (*class*), 30  
 <bad-request-error> (*class*), 30  
 <base-http-request> (*class*), 30  
 <base-http-response> (*class*), 30  
 bad-gateway-error (*function*), 33  
 bad-header-error (*function*), 33  
 bad-request-error (*function*), 34

## C

\$conflict-error (*constant*), 27  
 \$content-length-required-error (*constant*),  
 27  
 \*command-line-parser\* (*variable*), 15  
 <chunking-input-stream> (*class*), 30  
 <configuration-error> (*class*), 15  
 <conflict-error> (*class*), 30  
 <content-length-required-error> (*class*), 30

chunked-transfer-encoding? (*generic function*), 34  
 clear-session (*generic function*), 18  
 condition-class-for-status-code (*function*),  
 34  
 configure-server (*generic function*), 18  
 conflict-error (*function*), 34  
 connection-host (*generic function*), 62  
 connection-port (*generic function*), 62  
 content-length (*generic function*), 34  
 content-length-required-error (*function*), 35  
 cookie-comment (*generic function*), 35  
 cookie-domain (*generic function*), 35  
 cookie-max-age (*generic function*), 35  
 cookie-name (*generic function*), 35  
 cookie-path (*generic function*), 36  
 cookie-value (*generic function*), 36  
 cookie-version (*generic function*), 36  
 count-query-values (*function*), 18  
 current-page-number (*generic function*), 53  
 current-page-number-setter (*generic function*), 53  
 current-request (*function*), 18  
 current-response (*function*), 18  
 current-row (*function*), 53  
 current-row-number (*function*), 53  
 current-server (*function*), 18

## D

\$default-cookie-version (*constant*), 27  
 <directory-resource> (*class*), 15  
 <dylan-server-page> (*class*), 52  
 date-modified (*generic function*), 36  
 date-modified-setter (*generic function*), 36  
 debugging-enabled? (*generic function*), 18  
 debugging-enabled?-setter (*generic function*),  
 19  
 default-content-type (*generic function*), 19  
 default-documents (*generic function*), 19  
 default-virtual-host (*generic function*), 19  
 default-virtual-host-setter (*generic function*), 19

do-query-values (*generic function*), 19  
do-resources (*generic function*), 20

## E

\$expectation-failed-error (*constant*), 27  
<expectation-failed-error> (*class*), 30  
<expiring-mixin> (*class*), 31  
ensure-session (*generic function*), 20  
expectation-failed-error (*function*), 36  
expired? (*generic function*), 37

## F

\$forbidden-error (*constant*), 28  
\$found-redirect (*constant*), 28  
<forbidden-error> (*class*), 31  
<found-redirect> (*class*), 31  
<function-resource> (*class*), 15  
file-contents (*function*), 20  
find-resource (*generic function*), 20  
find-virtual-host (*generic function*), 20  
finish-request (*generic function*), 60  
forbidden-error (*function*), 37  
found-redirect (*function*), 37  
function-resource (*function*), 21

## G

\$gateway-timeout-error (*constant*), 28  
\$gone-error (*constant*), 28  
<gateway-timeout-error> (*class*), 31  
<gone-error> (*class*), 31  
gateway-timeout-error (*function*), 37  
generate-url (*generic function*), 21  
get-attr (*generic function*), 21  
get-attribute (*generic function*), 37  
get-field-errors (*generic function*), 54  
get-header (*generic function*), 38  
get-named-method (*generic function*), 54  
get-query-value (*function*), 21  
get-session (*generic function*), 21  
get-tag-call-attribute (*function*), 54  
gone-error (*function*), 38  
grow-header-buffer (*function*), 38

## H

\$header-too-large-error (*constant*), 28  
\$http-version-not-supported-error (*constant*), 28  
\*http-client-log\* (*variable*), 63  
\*http-common-log\* (*variable*), 29  
<header-table> (*class*), 31  
<header-too-large-error> (*class*), 31  
<http-client-protocol-error> (*class*), 31  
<http-connection> (*class*), 62

<http-error> (*class*), 31  
<http-parse-error> (*class*), 31  
<http-protocol-condition> (*class*), 31  
<http-redirect-condition> (*class*), 31  
<http-response> (*class*), 60  
<http-server-api-error> (*class*), 16  
<http-server-protocol-error> (*class*), 31  
<http-server> (*class*), 16  
<http-version-not-supported-error> (*class*), 31  
has-attribute? (*generic function*), 38  
header-too-large-error (*function*), 39  
http-delete (*constant*), 59  
http-error-headers (*generic function*), 39  
http-error-message-no-code (*generic function*), 39  
http-get (*constant*), 59  
http-head (*constant*), 59  
http-options (*constant*), 59  
http-post (*constant*), 60  
http-put (*constant*), 60  
http-request (*generic function*), 59  
http-server-main (*function*), 22  
http-status-code (*generic function*), 39  
http-version-not-supported-error (*function*), 39

## I

\$internal-server-error (*constant*), 28  
<internal-server-error> (*class*), 31  
internal-server-error (*function*), 40

## L

\*log-content?\* (*variable*), 15  
log-content (*generic function*), 22  
log-debug (*constant*), 22  
log-error (*constant*), 22  
log-info (*constant*), 22  
log-trace (*constant*), 22  
log-warning (*constant*), 22  
loop-index (*generic function*), 54  
loop-value (*generic function*), 54

## M

\$method-not-allowed-error (*constant*), 28  
\$mime-wild (*constant*), 28  
\$moved-permanently-redirect (*constant*), 28  
\$moved-temporarily-redirect (*constant*), 28  
<maximum-redirects-exceeded> (*class*), 61  
<media-type> (*class*), 32  
<method-not-allowed-error> (*class*), 32  
<moved-permanently-redirect> (*class*), 32  
<moved-temporarily-redirect> (*class*), 32  
make-http-connection (*function*), 62

map-tag-call-attributes (*function*), 54  
 match-media-types (*generic function*), 40  
 media-type-exact? (*generic function*), 40  
 media-type-level (*generic function*), 40  
 media-type-more-specific? (*generic function*), 40  
 media-type-quality (*generic function*), 40  
 method-not-allowed-error (*function*), 41  
 moved-permanently-redirect (*function*), 41  
 moved-temporarily-redirect (*function*), 41

## N

\$not-acceptable-error (*constant*), 28  
 \$not-implemented-error (*constant*), 28  
 \$not-modified-redirect (*constant*), 28  
 <not-acceptable-error> (*class*), 32  
 <not-implemented-error> (*class*), 32  
 <not-modified-redirect> (*class*), 32  
 named-method-definer (*macro*), 55  
 next-page-number (*generic function*), 55  
 not-acceptable-error (*function*), 41  
 not-implemented-error (*function*), 42  
 not-modified-redirect (*function*), 42  
 note-bytes-received (*generic function*), 42  
 note-bytes-sent (*generic function*), 62

## O

outgoing-chunk-size (*generic function*), 63  
 outgoing-chunk-size-setter (*generic function*), 63  
 output (*function*), 22

## P

\$payment-required-error (*constant*), 28  
 \$precondition-failed-error (*constant*), 28  
 \$proxy-authentication-required-error (*constant*), 28  
 <page-context> (*class*), 16  
 <page-link> (*class*), 52  
 <paginator> (*class*), 52  
 <payment-required-error> (*class*), 32  
 <precondition-failed-error> (*class*), 32  
 <proxy-authentication-required-error> (*class*), 32  
 page-context (*generic function*), 22  
 page-count (*generic function*), 55  
 page-has-errors? (*generic function*), 55  
 page-link-label (*generic function*), 55  
 page-link-page-number (*generic function*), 55  
 page-links (*generic function*), 55  
 page-size (*generic function*), 56  
 page-source (*generic function*), 56  
 page-template (*generic function*), 56  
 page-template-setter (*generic function*), 56

paginator-sequence (*generic function*), 56  
 parse-header-value (*generic function*), 42  
 parse-http-date (*function*), 43  
 parsed-headers (*generic function*), 43  
 payment-required-error (*function*), 43  
 precondition-failed-error (*function*), 43  
 previous-page-number (*generic function*), 57  
 process-config-element (*generic function*), 22  
 process-request-content (*generic function*), 23  
 process-template (*generic function*), 57  
 proxy-authentication-required-error (*function*), 43

## Q

quote-html (*function*), 44

## R

\$request-entity-too-large-error (*constant*), 28  
 \$request-timeout-error (*constant*), 28  
 \$request-uri-too-long-error (*constant*), 28  
 \$requested-range-not-satisfiable-error (*constant*), 28  
 \$resource-not-found-error (*constant*), 28  
 <redirect-loop-detected> (*class*), 61  
 <redirecting-resource> (*class*), 16  
 <request-entity-too-large-error> (*class*), 32  
 <request-timeout-error> (*class*), 32  
 <request-uri-too-long-error> (*class*), 32  
 <request> (*class*), 16  
 <requested-range-not-satisfiable-error> (*class*), 32  
 <resource-not-found-error> (*class*), 32  
 <resource> (*class*), 16  
 <response> (*class*), 16  
 <rewrite-rule> (*class*), 16  
 raw-headers (*generic function*), 44  
 read-header-line (*function*), 44  
 read-http-line (*generic function*), 44  
 read-message-headers (*function*), 45  
 read-response (*generic function*), 61  
 redirect-temporarily-to (*generic function*), 23  
 redirect-to (*generic function*), 23  
 register-tag (*generic function*), 57  
 remove-attribute (*generic function*), 45  
 request-absolute-url (*generic function*), 23  
 request-content (*generic function*), 45  
 request-content-setter (*generic function*), 45  
 request-content-type (*function*), 23  
 request-entity-too-large-error (*function*), 45  
 request-host (*generic function*), 23  
 request-method (*generic function*), 46

request-method-setter (*generic function*), 46  
 request-query-values (*generic function*), 24  
 request-raw-url-string (*generic function*), 46  
 request-raw-url-string-setter (*generic function*), 46  
 request-timeout-error (*function*), 46  
 request-uri-too-long-error (*function*), 47  
 request-url (*generic function*), 47  
 request-url-path-prefix (*generic function*), 24  
 request-url-path-suffix (*generic function*), 24  
 request-url-setter (*generic function*), 47  
 request-version (*generic function*), 47  
 request-version-setter (*generic function*), 48  
 requested-range-not-satisfiable-error (*function*), 48  
 resource-not-found-error (*function*), 48  
 respond (*generic function*), 24  
 respond-to-connect (*generic function*), 24  
 respond-to-delete (*generic function*), 24  
 respond-to-get (*generic function*), 25  
 respond-to-head (*generic function*), 25  
 respond-to-options (*generic function*), 25  
 respond-to-post (*generic function*), 25  
 respond-to-put (*generic function*), 25  
 respond-to-trace (*generic function*), 25  
 response-chunked? (*generic function*), 48  
 response-chunked?-setter (*generic function*), 48  
 response-code (*generic function*), 49  
 response-code-setter (*generic function*), 49  
 response-content (*generic function*), 61  
 response-reason-phrase (*generic function*), 49  
 response-reason-phrase-setter (*generic function*), 49  
 response-request (*generic function*), 49  
 rewrite-url (*generic function*), 26  
 route-request (*generic function*), 26

## S

\$see-other-redirect (*constant*), 28  
 \$service-unavailable-error (*constant*), 28  
 \$status-accepted (*constant*), 28  
 \$status-application-error (*constant*), 28  
 \$status-bad-gateway (*constant*), 28  
 \$status-bad-request (*constant*), 28  
 \$status-conflict (*constant*), 28  
 \$status-continue (*constant*), 28  
 \$status-created (*constant*), 28  
 \$status-expectation-failed (*constant*), 28  
 \$status-forbidden (*constant*), 28  
 \$status-found (*constant*), 28  
 \$status-gateway-timeout (*constant*), 28  
 \$status-gone (*constant*), 28

\$status-http-version-not-supported (*constant*), 29  
 \$status-internal-server-error (*constant*), 29  
 \$status-length-required (*constant*), 29  
 \$status-method-not-allowed (*constant*), 29  
 \$status-moved-permanently (*constant*), 29  
 \$status-multiple-choices (*constant*), 29  
 \$status-no-content (*constant*), 29  
 \$status-non-authoritative-information (*constant*), 29  
 \$status-not-acceptable (*constant*), 29  
 \$status-not-found (*constant*), 29  
 \$status-not-implemented (*constant*), 29  
 \$status-not-modified (*constant*), 29  
 \$status-ok (*constant*), 29  
 \$status-partial-content (*constant*), 29  
 \$status-payment-required (*constant*), 29  
 \$status-precondition-failed (*constant*), 29  
 \$status-proxy-authentication-required (*constant*), 29  
 \$status-request-entity-too-large (*constant*), 29  
 \$status-request-timeout (*constant*), 29  
 \$status-request-uri-too-long (*constant*), 29  
 \$status-requested-range-not-satisfiable (*constant*), 29  
 \$status-reset-content (*constant*), 29  
 \$status-see-other (*constant*), 29  
 \$status-service-unavailable (*constant*), 29  
 \$status-switching-protocols (*constant*), 29  
 \$status-temporary-redirect (*constant*), 29  
 \$status-unauthorized (*constant*), 29  
 \$status-unsupported-media-type (*constant*), 29  
 \$status-use-proxy (*constant*), 29  
 <see-other-redirect> (*class*), 32  
 <service-unavailable-error> (*class*), 32  
 <session> (*class*), 17  
 see-other-redirect (*function*), 49  
 send-request (*generic function*), 60  
 serve-static-file (*generic function*), 26  
 server-root (*generic function*), 26  
 service-unavailable-error (*function*), 50  
 set-attribute (*generic function*), 50  
 set-header (*generic function*), 50  
 show-tag-call-attributes (*function*), 57  
 start-request (*generic function*), 60  
 start-server (*generic function*), 26  
 stop-server (*generic function*), 27

## T

<taglib> (*class*), 52  
 tag-definer (*macro*), 57

taglib-definer (*macro*), 58  
token-end-position (*function*), 50

## U

\$unauthorized-error (*constant*), 29  
\$unsupported-media-type-error (*constant*),  
29  
\$use-proxy-redirect (*constant*), 29  
<unauthorized-error> (*class*), 32  
<unsupported-media-type-error> (*class*), 33  
<use-proxy-redirect> (*class*), 33  
unauthorized-error (*function*), 51  
unmatched-url-suffix (*generic function*), 27  
unsupported-media-type-error (*function*), 51  
use-default-virtual-host? (*generic function*),  
27  
use-proxy-redirect (*function*), 51

## V

<virtual-host> (*class*), 17  
validate-form-field (*generic function*), 59  
validate-http-status-code (*generic function*),  
51  
validate-http-version (*generic function*), 52

## W

with-http-connection (*macro*), 61  
with-query-values (*macro*), 27