

Python / Flask

General Information & Licensing

Code Repository	Flask / Python
License Type	BSD-3-Clause Source
License Description	<ul style="list-style-type: none">THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
License Restrictions	<ul style="list-style-type: none">Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

Parsing HTTP Headers

Start a thread.

<https://github.com/eventlet/eventlet/blob/85290a11dd5c5f511a45f4010b512eedfef87000/eventlet/greenthread.py#L219>

<https://github.com/eventlet/eventlet/blob/85290a11dd5c5f511a45f4010b512eedfef87000/eventlet/greenthread.py#L221>

Go to the processing and handling methods of the request.

if not self.parse_request():

self here is a HttpProtocol object.

<https://github.com/eventlet/eventlet/blob/85290a11dd5c5f511a45f4010b512eedfef87000/eventlet/wsgi.py#L828>

<https://github.com/eventlet/eventlet/blob/85290a11dd5c5f511a45f4010b512eedfef87000/eventlet/wsgi.py#L837>

<https://github.com/eventlet/eventlet/blob/85290a11dd5c5f511a45f4010b512eedfef87000/eventlet/wsgi.py#L350>

<https://github.com/eventlet/eventlet/blob/85290a11dd5c5f511a45f4010b512eedfef87000/eventlet/wsgi.py#L379>

<https://github.com/eventlet/eventlet/blob/85290a11dd5c5f511a45f4010b512eedfef87000/eventlet/wsgi.py#L383>

<https://github.com/eventlet/eventlet/blob/85290a11dd5c5f511a45f4010b512eedfef87000/eventlet/wsgi.py#L408>

<https://github.com/eventlet/eventlet/blob/85290a11dd5c5f511a45f4010b512eedfef87000/eventlet/wsgi.py#L424>

self.headers = http.client.parse_headers(self.rfile, _class=self.MessageClass)

Start parsing headers, Message class encapsulates information: policy, headers, payload, and so on.

<https://github.com/python/cpython/blob/bf26bdf6ac04878fc720e78422991aaedb9808a1/Lib/http/server.py#L341>

def _read_headers(fp):

Read headers (in bytes) line by line from a file pointer (a line is a header) and append each header to the headers list.

<https://github.com/python/cpython/blob/bf26bdf6ac04878fc720e78422991aaedb9808a1/Lib/http/client.py#L234>

<https://github.com/python/cpython/blob/bf26bdf6ac04878fc720e78422991aaedb9808a1/Lib/http/client.py#L206>

hstring = b''.join(headers).decode('iso-8859-1')

Decode headers list we got from the previous step using 'iso-8859-1'.

<https://github.com/python/cpython/blob/bf26bdf6ac04878fc720e78422991aaedb9808a1/Lib/http/client.py#L206>

[1/Lib/http/client.py#L235](https://github.com/python/cpython/blob/bf26bdf6ac04878fc720e78422991aaedb9808a1/Lib/http/client.py#L235)

```
email.parser.Parser(_class=_class).parsestr(hstring)
```

Create a message structure from the headers string we got from the previous step.

```
feedparser = FeedParser(self._class, policy=self.policy)
```

Get the feedparser, policy contains methods that parse the headers.

<https://github.com/python/cpython/blob/bf26bdf6ac04878fc720e78422991aaedb9808a1/Lib/http/client.py#L236>

[1/Lib/http/client.py#L236](https://github.com/python/cpython/blob/bf26bdf6ac04878fc720e78422991aaedb9808a1/Lib/http/client.py#L236)

<https://github.com/python/cpython/blob/bf26bdf6ac04878fc720e78422991aaedb9808a1/Lib/email/parser.py#L56>

[1/Lib/email/parser.py#L56](https://github.com/python/cpython/blob/bf26bdf6ac04878fc720e78422991aaedb9808a1/Lib/email/parser.py#L56)

<https://github.com/python/cpython/blob/bf26bdf6ac04878fc720e78422991aaedb9808a1/Lib/email/parser.py#L64>

[1/Lib/email/parser.py#L64](https://github.com/python/cpython/blob/bf26bdf6ac04878fc720e78422991aaedb9808a1/Lib/email/parser.py#L64)

<https://github.com/python/cpython/blob/bf26bdf6ac04878fc720e78422991aaedb9808a1/Lib/email/parser.py#L41>

[1/Lib/email/parser.py#L41](https://github.com/python/cpython/blob/bf26bdf6ac04878fc720e78422991aaedb9808a1/Lib/email/parser.py#L41)

<https://github.com/python/cpython/blob/bf26bdf6ac04878fc720e78422991aaedb9808a1/Lib/email/parser.py#L49>

[1/Lib/email/parser.py#L49](https://github.com/python/cpython/blob/bf26bdf6ac04878fc720e78422991aaedb9808a1/Lib/email/parser.py#L49)

```
while data := fp.read(8192):
```

```
    feedparser.feed(data)
```

Read all the data (headers string) from the file.

```
self._input.push(data)
```

Push the headers string into the parser.

<https://github.com/python/cpython/blob/bf26bdf6ac04878fc720e78422991aaedb9808a1/Lib/email/parser.py#L53>

[1/Lib/email/parser.py#L53](https://github.com/python/cpython/blob/bf26bdf6ac04878fc720e78422991aaedb9808a1/Lib/email/parser.py#L53)

<https://github.com/python/cpython/blob/1012dc1b4367e05b92d67ea6925a39d50dce31b7/Lib/email/feedparser.py#L173>

[1b7/Lib/email/feedparser.py#L173](https://github.com/python/cpython/blob/1012dc1b4367e05b92d67ea6925a39d50dce31b7/Lib/email/feedparser.py#L173)

<https://github.com/python/cpython/blob/1012dc1b4367e05b92d67ea6925a39d50dce31b7/Lib/email/feedparser.py#L175>

[1b7/Lib/email/feedparser.py#L175](https://github.com/python/cpython/blob/1012dc1b4367e05b92d67ea6925a39d50dce31b7/Lib/email/feedparser.py#L175)

Continue parsing the headers string, go to the parse function.

<https://github.com/python/cpython/blob/1012dc1b4367e05b92d67ea6925a39d50dce31b7/Lib/email/feedparser.py#L176>

[1b7/Lib/email/feedparser.py#L176](https://github.com/python/cpython/blob/1012dc1b4367e05b92d67ea6925a39d50dce31b7/Lib/email/feedparser.py#L176)

<https://github.com/python/cpython/blob/1012dc1b4367e05b92d67ea6925a39d50dce31b7/Lib/email/feedparser.py#L178>

[1b7/Lib/email/feedparser.py#L178](https://github.com/python/cpython/blob/1012dc1b4367e05b92d67ea6925a39d50dce31b7/Lib/email/feedparser.py#L178)

<https://github.com/python/cpython/blob/1012dc1b4367e05b92d67ea6925a39d50dce31b7/Lib/email/feedparser.py#L180>

[1b7/Lib/email/feedparser.py#L180](https://github.com/python/cpython/blob/1012dc1b4367e05b92d67ea6925a39d50dce31b7/Lib/email/feedparser.py#L180)

<https://github.com/python/cpython/blob/1012dc1b4367e05b92d67ea6925a39d50dce31b7/Lib/email/feedparser.py#L164>

[1b7/Lib/email/feedparser.py#L164](https://github.com/python/cpython/blob/1012dc1b4367e05b92d67ea6925a39d50dce31b7/Lib/email/feedparser.py#L164)

```
headers = []
```

```
for line in self._input:
```

```
    headers.append(line)
```

Converting headers string into a list by reading line by line (lines end by ‘\r\n’).

Elements in the list would be like “header_name: header_value\r\n”.

<https://github.com/python/cpython/blob/1012dc1b4367e05b92d67ea6925a39d50dce31b7/Lib/email/feedparser.py#L218>

[1b7/Lib/email/feedparser.py#L218](https://github.com/python/cpython/blob/1012dc1b4367e05b92d67ea6925a39d50dce31b7/Lib/email/feedparser.py#L218)

Continue to call the functions that parse the headers list we got from the previous step.
<https://github.com/python/cpython/blob/1012dc1b4367e05b92d67ea6925a39d50dce31b7/Lib/email/feedparser.py#L240>
<https://github.com/python/cpython/blob/1012dc1b4367e05b92d67ea6925a39d50dce31b7/Lib/email/feedparser.py#L471>

```
def header_source_parse(self, sourcelines):
```

This function is basically what we do in the homework. Split the header on the first occurrence of `:`. Everything before `:` is the header name, after is the header value. Put them into a tuple (header_name, header_value).

```
self._headers.append((name, value))
```

Store the parsed header (self here is the Message object).

<https://github.com/python/cpython/blob/1012dc1b4367e05b92d67ea6925a39d50dce31b7/Lib/email/feedparser.py#L488>
https://github.com/python/cpython/blob/1012dc1b4367e05b92d67ea6925a39d50dce31b7/Lib/email/_policybase.py#L293
<https://github.com/python/cpython/blob/1012dc1b4367e05b92d67ea6925a39d50dce31b7/Lib/email/message.py#L506>

```
self.environ = self.get_environ()
```

Get the environment, packaging request information.

<https://github.com/eventlet/eventlet/blob/85290a11dd5c5f511a45f4010b512eedfef87000/eventlet/wsgi.py#L454>
<https://github.com/eventlet/eventlet/blob/85290a11dd5c5f511a45f4010b512eedfef87000/eventlet/wsgi.py#L455>

Go to the response handling method.

<https://github.com/eventlet/eventlet/blob/85290a11dd5c5f511a45f4010b512eedfef87000/eventlet/wsgi.py#L459>
<https://github.com/eventlet/eventlet/blob/85290a11dd5c5f511a45f4010b512eedfef87000/eventlet/wsgi.py#L467>

```
result = self.application(self.environ, start_response)
```

Get the response data according to the self.environ (this contains the request information). And starts a response.

<https://github.com/eventlet/eventlet/blob/85290a11dd5c5f511a45f4010b512eedfef87000/eventlet/wsgi.py#L569>
https://github.com/pallets/werkzeug/blob/3115aa6a6276939f5fd6efa46282e0256ff21f1a/src/werkzeug/debug/__init__.py#L329
<https://github.com/pallets/flask/blob/066a35dd322f689ec07d7c0e82b19eacadac3c6b/src/flask/app.py#L2498>
<https://github.com/pallets/flask/blob/066a35dd322f689ec07d7c0e82b19eacadac3c6b/src/flask/app.py#L2535>
<https://github.com/pallets/werkzeug/blob/3115aa6a6276939f5fd6efa46282e0256ff21f1a/src/werkzeug/wrappers/response.py#L619>
<https://github.com/pallets/werkzeug/blob/3115aa6a6276939f5fd6efa46282e0256ff21f1>

[a/src/werkzeug/wrappers/response.py#L629](https://github.com/pallets/werkzeug/blob/3115aa6a6276939f5fd6efa46282e0256ff21f1a/src/werkzeug/wrappers/response.py#L629)

[https://github.com/pallets/werkzeug/blob/3115aa6a6276939f5fd6efa46282e0256ff21f1](https://github.com/pallets/werkzeug/blob/3115aa6a6276939f5fd6efa46282e0256ff21f1a/src/werkzeug/wrappers/response.py#L600)

[a/src/werkzeug/wrappers/response.py#L600](https://github.com/pallets/werkzeug/blob/3115aa6a6276939f5fd6efa46282e0256ff21f1a/src/werkzeug/wrappers/response.py#L630)

[https://github.com/pallets/werkzeug/blob/3115aa6a6276939f5fd6efa46282e0256ff21f1](https://github.com/pallets/werkzeug/blob/3115aa6a6276939f5fd6efa46282e0256ff21f1a/src/werkzeug/wrappers/response.py#L630)

[a/src/werkzeug/wrappers/response.py#L630](https://github.com/pallets/werkzeug/blob/3115aa6a6276939f5fd6efa46282e0256ff21f1a/src/werkzeug/wrappers/response.py#L630)

```
def start_response(status, response_headers, exc_info=None):
```

Get the response code and process the response headers (response headers capitalization).

[https://github.com/eventlet/eventlet/blob/85290a11dd5c5f511a45f4010b512eedfef8700](https://github.com/eventlet/eventlet/blob/85290a11dd5c5f511a45f4010b512eedfef87000/eventlet/wsgi.py#L536)

[0/eventlet/wsgi.py#L536](https://github.com/eventlet/eventlet/blob/85290a11dd5c5f511a45f4010b512eedfef87000/eventlet/wsgi.py#L536)

```
towrite.append(six.b('%s %s\r\n' % (self.protocol_version, status)))
```

```
for header in response_headers:
```

```
    towrite.append(six.b('%s: %s\r\n' % header))
```

```
towrite.append(b'Connection: close\r\n')
```

```
towrite.append(data)
```

```
wfile.writelines(towrite)
```

Append response line, headers and body to towrite. And write data in towrite to wfile.

[https://github.com/eventlet/eventlet/blob/85290a11dd5c5f511a45f4010b512eedfef8700](https://github.com/eventlet/eventlet/blob/85290a11dd5c5f511a45f4010b512eedfef87000/eventlet/wsgi.py#L564)

[0/eventlet/wsgi.py#L564](https://github.com/eventlet/eventlet/blob/85290a11dd5c5f511a45f4010b512eedfef87000/eventlet/wsgi.py#L564)

[https://github.com/eventlet/eventlet/blob/85290a11dd5c5f511a45f4010b512eedfef8700](https://github.com/eventlet/eventlet/blob/85290a11dd5c5f511a45f4010b512eedfef87000/eventlet/wsgi.py#L483)

[0/eventlet/wsgi.py#L483](https://github.com/eventlet/eventlet/blob/85290a11dd5c5f511a45f4010b512eedfef87000/eventlet/wsgi.py#L483)

[https://github.com/eventlet/eventlet/blob/85290a11dd5c5f511a45f4010b512eedfef8700](https://github.com/eventlet/eventlet/blob/85290a11dd5c5f511a45f4010b512eedfef87000/eventlet/wsgi.py#L491)

[0/eventlet/wsgi.py#L491](https://github.com/eventlet/eventlet/blob/85290a11dd5c5f511a45f4010b512eedfef87000/eventlet/wsgi.py#L491)

[https://github.com/eventlet/eventlet/blob/85290a11dd5c5f511a45f4010b512eedfef8700](https://github.com/eventlet/eventlet/blob/85290a11dd5c5f511a45f4010b512eedfef87000/eventlet/wsgi.py#L492)

[0/eventlet/wsgi.py#L492](https://github.com/eventlet/eventlet/blob/85290a11dd5c5f511a45f4010b512eedfef87000/eventlet/wsgi.py#L492)

[https://github.com/eventlet/eventlet/blob/85290a11dd5c5f511a45f4010b512eedfef8700](https://github.com/eventlet/eventlet/blob/85290a11dd5c5f511a45f4010b512eedfef87000/eventlet/wsgi.py#L493)

[0/eventlet/wsgi.py#L493](https://github.com/eventlet/eventlet/blob/85290a11dd5c5f511a45f4010b512eedfef87000/eventlet/wsgi.py#L493)

[https://github.com/eventlet/eventlet/blob/85290a11dd5c5f511a45f4010b512eedfef8700](https://github.com/eventlet/eventlet/blob/85290a11dd5c5f511a45f4010b512eedfef87000/eventlet/wsgi.py#L521)

[0/eventlet/wsgi.py#L521](https://github.com/eventlet/eventlet/blob/85290a11dd5c5f511a45f4010b512eedfef87000/eventlet/wsgi.py#L521)

[https://github.com/eventlet/eventlet/blob/85290a11dd5c5f511a45f4010b512eedfef8700](https://github.com/eventlet/eventlet/blob/85290a11dd5c5f511a45f4010b512eedfef87000/eventlet/wsgi.py#L531)

[0/eventlet/wsgi.py#L531](https://github.com/eventlet/eventlet/blob/85290a11dd5c5f511a45f4010b512eedfef87000/eventlet/wsgi.py#L531)

[https://github.com/eventlet/eventlet/blob/85290a11dd5c5f511a45f4010b512eedfef8700](https://github.com/eventlet/eventlet/blob/85290a11dd5c5f511a45f4010b512eedfef87000/eventlet/wsgi.py#L532)

[0/eventlet/wsgi.py#L532](https://github.com/eventlet/eventlet/blob/85290a11dd5c5f511a45f4010b512eedfef87000/eventlet/wsgi.py#L532)

```
wfile.flush()
```

Send the response.

[https://github.com/eventlet/eventlet/blob/85290a11dd5c5f511a45f4010b512eedfef8700](https://github.com/eventlet/eventlet/blob/85290a11dd5c5f511a45f4010b512eedfef87000/eventlet/wsgi.py#L533)

[0/eventlet/wsgi.py#L533](https://github.com/eventlet/eventlet/blob/85290a11dd5c5f511a45f4010b512eedfef87000/eventlet/wsgi.py#L533)