

```

require 'pry'

class RequestParser
  attr_accessor :verb,
    :path,
    :parameters,
    :protocol,
    :host,
    :port,
    :origin,
    :accept_encoding,
    :accept,
    :accept_language

  def initialize(request_lines)
    @verb, @path, @protocol = request_lines[0].split(" ")
    @path, @parameters = @path.split("?")
    parse_parameters if !@parameters.nil?
    parse_request(request_lines)
  end

  def parse_request(request_lines)
    request_lines[1..-1].each do |line|
      prefix, content = line.split(': ')
      set_variable(prefix, content)
    end
    @origin ||= @host
  end

  def parse_parameters
    @parameters = @parameters.split("&").map do |parameter|
      parameter.split("=")
    end
  end

  def set_variable(prefix, content)
    case prefix.downcase
    when "host" then @host, @port = content.split(":")
    when "origin" then @origin = content
  end
end

```

```
when "accept" then @accept = content
when "accept-encoding" then @accept_encoding = content
when "accept-language" then @accept_language = content
else nil end
end

def debug_info
  "" + ("\n") + ("\t") +
  "Verb: #{@verb}
  Path: #{@path}
  Protocol: #{@protocol}
  Host: #{@host}
  Port: #{@port}
  Origin: #{@origin}
  Accept: #{@accept_encoding},#{@accept};#{@accept_language}" +
  ""
end
end
```

@origin = @host

elsif prefix == "origin"

@origin = content

elsif prefix == "accept"

@accept = content

elsif prefix == "accept-encoding"

@accept_encoding = content

elsif prefix == "accept-language"

@accept_language = content

else

nil

end
