```
Name:
  wkhtmltopdf 0.12.6 (with patched qt)

Synopsis:
  wkhtmltopdf [GLOBAL OPTION]... [OBJECT]... <output file>

Document objects:
  wkhtmltopdf is able to put several objects into the output file, an object is
  either a single webpage, a cover webpage or a table of contents.  The objects
  are put into the output document in the order they are specified on the
  command line, options can be specified on a per object basis or in the global
  options area. Options from the Global Options section can only be placed in
  the global options area.

  A page objects puts the content of a single webpage into the output document.

  (page)? <input url/file name> [PAGE OPTION]...
  Options for the page object can be placed in the global options and the page
  options areas. The applicable options can be found in the Page Options and
  Headers And Footer Options sections.

  A cover objects puts the content of a single webpage into the output document,
  the page does not appear in the table of contents, and does not have headers
  and footers.

  cover <input url/file name> [PAGE OPTION]...
  All options that can be specified for a page object can also be specified for
  a cover.

  A table of contents object inserts a table of contents into the output
  document.

  toc [TOC OPTION]...
  All options that can be specified for a page object can also be specified for
  a toc, further more the options from the TOC Options section can also be
  applied. The table of contents is generated via XSLT which means that it can
  be styled to look however you want it to look. To get an idea of how to do
  this you can dump the default xslt document by supplying the
  --dump-default-toc-xsl, and the outline it works on by supplying
  --dump-outline, see the Outline Options section.

Description:
  Converts one or more HTML pages into a PDF document, using wkhtmltopdf patched
  qt.

Global Options:
      --collate                      Collate when printing multiple copies
                                     (default)
      --no-collate                   Do not collate when printing multiple
                                     copies
      --cookie-jar <path>            Read and write cookies from and to the
                                     supplied cookie jar file
      --copies <number>              Number of copies to print into the pdf
                                     file (default 1)
  -d, --dpi <dpi>                    Change the dpi explicitly (this has no
                                     effect on X11 based systems) (default 96)
  -H, --extended-help                Display more extensive help, detailing
                                     less common command switches
  -g, --grayscale                    PDF will be generated in grayscale
  -h, --help                         Display help
      --htmldoc                      Output program html help
      --image-dpi <integer>          When embedding images scale them down to
                                     this dpi (default 600)
      --image-quality <integer>      When jpeg compressing images use this
                                     quality (default 94)
      --license                      Output license information and exit
      --log-level <level>            Set log level to: none, error, warn or
                                     info (default info)
  -l, --lowquality                   Generates lower quality pdf/ps. Useful to
                                     shrink the result document space
      --manpage                      Output program man page
  -B, --margin-bottom <unitreal>     Set the page bottom margin
  -L, --margin-left <unitreal>       Set the page left margin (default 10mm)
  -R, --margin-right <unitreal>      Set the page right margin (default 10mm)
  -T, --margin-top <unitreal>        Set the page top margin
  -O, --orientation <orientation>    Set orientation to Landscape or Portrait
                                     (default Portrait)
      --page-height <unitreal>       Page height
  -s, --page-size <Size>             Set paper size to: A4, Letter, etc.
                                     (default A4)
      --page-width <unitreal>        Page width
```

```
      --no-pdf-compression            Do not use lossless compression on pdf
                                      objects
  -q, --quiet                         Be less verbose, maintained for backwards
                                      compatibility; Same as using --log-level
                                      none
      --read-args-from-stdin          Read command line arguments from stdin
      --readme                        Output program readme
      --title <text>                  The title of the generated pdf file (The
                                      title of the first document is used if not
                                      specified)
      --use-xserver                   Use the X server (some plugins and other
                                      stuff might not work without X11)
  -V, --version                       Output version information and exit

Outline Options:
      --dump-default-toc-xsl          Dump the default TOC xsl style sheet to
                                      stdout
      --dump-outline <file>           Dump the outline to a file
      --outline                       Put an outline into the pdf (default)
      --no-outline                    Do not put an outline into the pdf
      --outline-depth <level>         Set the depth of the outline (default 4)

Page Options:
      --allow <path>                  Allow the file or files from the specified
                                      folder to be loaded (repeatable)
      --background                    Do print background (default)
      --no-background                 Do not print background
      --bypass-proxy-for <value>      Bypass proxy for host (repeatable)
      --cache-dir <path>              Web cache directory
      --checkbox-checked-svg <path>   Use this SVG file when rendering checked
                                      checkboxes
      --checkbox-svg <path>           Use this SVG file when rendering unchecked
                                      checkboxes
      --cookie <name> <value>         Set an additional cookie (repeatable),
                                      value should be url encoded.
      --custom-header <name> <value>  Set an additional HTTP header (repeatable)
      --custom-header-propagation     Add HTTP headers specified by
                                      --custom-header for each resource request.
      --no-custom-header-propagation  Do not add HTTP headers specified by
                                      --custom-header for each resource request.
      --debug-javascript              Show javascript debugging output
      --no-debug-javascript           Do not show javascript debugging output
                                      (default)
      --default-header                Add a default header, with the name of the
                                      page to the left, and the page number to
                                      the right, this is short for:
                                      --header-left='[webpage]'
                                      --header-right='[page]/[toPage]' --top 2cm
                                      --header-line
      --encoding <encoding>           Set the default text encoding, for input
      --disable-external-links        Do not make links to remote web pages
      --enable-external-links         Make links to remote web pages (default)
      --disable-forms                 Do not turn HTML form fields into pdf form
                                      fields (default)
      --enable-forms                  Turn HTML form fields into pdf form fields
      --images                        Do load or print images (default)
      --no-images                     Do not load or print images
      --disable-internal-links        Do not make local links
      --enable-internal-links         Make local links (default)
  -n, --disable-javascript            Do not allow web pages to run javascript
      --enable-javascript             Do allow web pages to run javascript
                                      (default)
      --javascript-delay <msec>       Wait some milliseconds for javascript
                                      finish (default 200)
      --keep-relative-links           Keep relative external links as relative
                                      external links
      --load-error-handling <handler> Specify how to handle pages that fail to
                                      load: abort, ignore or skip (default
                                      abort)
      --load-media-error-handling <handler> Specify how to handle media files
                                      that fail to load: abort, ignore or skip
                                      (default ignore)
      --disable-local-file-access     Do not allowed conversion of a local file
                                      to read in other local files, unless
                                      explicitly allowed with --allow (default)
      --enable-local-file-access      Allowed conversion of a local file to read
                                      in other local files.
      --minimum-font-size <int>       Minimum font size
      --exclude-from-outline          Do not include the page in the table of
                                      contents and outlines
      --include-in-outline            Include the page in the table of contents
```

```
                                  and outlines (default)
      --page-offset <offset>      Set the starting page number (default 0)
      --password <password>       HTTP Authentication password
      --disable-plugins           Disable installed plugins (default)
      --enable-plugins            Enable installed plugins (plugins will
                                  likely not work)
      --post <name> <value>       Add an additional post field (repeatable)
      --post-file <name> <path>   Post an additional file (repeatable)
      --print-media-type          Use print media-type instead of screen
      --no-print-media-type       Do not use print media-type instead of
                                  screen (default)
  -p, --proxy <proxy>             Use a proxy
      --proxy-hostname-lookup     Use the proxy for resolving hostnames
      --radiobutton-checked-svg <path> Use this SVG file when rendering checked
                                  radiobuttons
      --radiobutton-svg <path>    Use this SVG file when rendering unchecked
                                  radiobuttons
      --resolve-relative-links    Resolve relative external links into
                                  absolute links (default)
      --run-script <js>           Run this additional javascript after the
                                  page is done loading (repeatable)
      --disable-smart-shrinking   Disable the intelligent shrinking strategy
                                  used by WebKit that makes the pixel/dpi
                                  ratio non-constant
      --enable-smart-shrinking    Enable the intelligent shrinking strategy
                                  used by WebKit that makes the pixel/dpi
                                  ratio non-constant (default)
      --ssl-crt-path <path>       Path to the ssl client cert public key in
                                  OpenSSL PEM format, optionally followed by
                                  intermediate ca and trusted certs
      --ssl-key-password <password>   Password to ssl client cert private key
      --ssl-key-path <path>       Path to ssl client cert private key in
                                  OpenSSL PEM format
      --stop-slow-scripts         Stop slow running javascripts (default)
      --no-stop-slow-scripts      Do not Stop slow running javascripts
      --disable-toc-back-links    Do not link from section header to toc
                                  (default)
      --enable-toc-back-links     Link from section header to toc
      --user-style-sheet <path>   Specify a user style sheet, to load with
                                  every page
      --username <username>       HTTP Authentication username
      --viewport-size <>          Set viewport size if you have custom
                                  scrollbars or css attribute overflow to
                                  emulate window size
      --window-status <windowStatus>  Wait until window.status is equal to this
                                  string before rendering page
      --zoom <float>              Use this zoom factor (default 1)

  Headers And Footer Options:
      --footer-center <text>      Centered footer text
      --footer-font-name <name>   Set footer font name (default Arial)
      --footer-font-size <size>   Set footer font size (default 12)
      --footer-html <url>         Adds a html footer
      --footer-left <text>        Left aligned footer text
      --footer-line               Display line above the footer
      --no-footer-line            Do not display line above the footer
                                  (default)
      --footer-right <text>       Right aligned footer text
      --footer-spacing <real>     Spacing between footer and content in mm
                                  (default 0)
      --header-center <text>      Centered header text
      --header-font-name <name>   Set header font name (default Arial)
      --header-font-size <size>   Set header font size (default 12)
      --header-html <url>         Adds a html header
      --header-left <text>        Left aligned header text
      --header-line               Display line below the header
      --no-header-line            Do not display line below the header
                                  (default)
      --header-right <text>       Right aligned header text
      --header-spacing <real>     Spacing between header and content in mm
                                  (default 0)
      --replace <name> <value>    Replace [name] with value in header and
                                  footer (repeatable)

  TOC Options:
      --disable-dotted-lines      Do not use dotted lines in the toc
      --toc-header-text <text>    The header text of the toc (default Table
                                  of Contents)
      --toc-level-indentation <width> For each level of headings in the toc
                                  indent by this length (default 1em)
      --disable-toc-links         Do not link from toc to sections
```

```
      --toc-text-size-shrink <real>   For each level of headings in the toc the
                                      font is scaled by this factor (default
                                      0.8)
      --xsl-style-sheet <file>        Use the supplied xsl style sheet for
                                      printing the table of contents

Page sizes:
  The default page size of the rendered document is A4, but by using the
  --page-size option this can be changed to almost anything else, such as: A3,
  Letter and Legal.  For a full list of supported pages sizes please see
  <https://doc.qt.io/archives/qt-4.8/qprinter.html#PaperSize-enum>.

  For a more fine grained control over the page size the --page-height and
  --page-width options may be used

Reading arguments from stdin:
  If you need to convert a lot of pages in a batch, and you feel that
  wkhtmltopdf is a bit too slow to start up, then you should try
  --read-args-from-stdin,

  When --read-args-from-stdin each line of input sent to wkhtmltopdf on stdin
  will act as a separate invocation of wkhtmltopdf, with the arguments specified
  on the given line combined with the arguments given to wkhtmltopdf

  For example one could do the following:

  echo "https://doc.qt.io/archives/qt-4.8/qapplication.html qapplication.pdf" >> cmds
  echo "cover google.com https://en.wikipedia.org/wiki/Qt_(software) qt.pdf" >> cmds
  wkhtmltopdf --read-args-from-stdin --book < cmds

Specifying A Proxy:
  By default proxy information will be read from the environment variables:
  proxy, all_proxy and http_proxy, proxy options can also by specified with the
  -p switch

  <type> := "http://" | "socks5://"
  <serif> := <username> (":" <password>)? "@"
  <proxy> := "None" | <type>? <string>? <host> (":" <port>)?

  Here are some examples (In case you are unfamiliar with the BNF):

  http://user:password@myproxyserver:8080
  socks5://myproxyserver
  None

Footers And Headers:
  Headers and footers can be added to the document by the --header-* and
  --footer* arguments respectively.  In header and footer text string supplied
  to e.g. --header-left, the following variables will be substituted.

   * [page]       Replaced by the number of the pages currently being printed
   * [frompage]   Replaced by the number of the first page to be printed
   * [topage]     Replaced by the number of the last page to be printed
   * [webpage]    Replaced by the URL of the page being printed
   * [section]    Replaced by the name of the current section
   * [subsection] Replaced by the name of the current subsection
   * [date]       Replaced by the current date in system local format
   * [isodate]    Replaced by the current date in ISO 8601 extended format
   * [time]       Replaced by the current time in system local format
   * [title]      Replaced by the title of the of the current page object
   * [doctitle]   Replaced by the title of the output document
   * [sitepage]   Replaced by the number of the page in the current site being converted
   * [sitepages]  Replaced by the number of pages in the current site being converted


  As an example specifying --header-right "Page [page] of [topage]", will result
  in the text "Page x of y" where x is the number of the current page and y is
  the number of the last page, to appear in the upper left corner in the
  document.

  Headers and footers can also be supplied with HTML documents. As an example
  one could specify --header-html header.html, and use the following content in
  header.html:

  <!DOCTYPE html>
  <html><head><script>
  function subst() {
      var vars = {};
      var query_strings_from_url = document.location.search.substring(1).split('&');
      for (var query_string in query_strings_from_url) {
          if (query_strings_from_url.hasOwnProperty(query_string)) {
```

```
                var temp_var = query_strings_from_url[query_string].split('=', 2);
                vars[temp_var[0]] = decodeURI(temp_var[1]);
            }
        }
        var css_selector_classes = ['page', 'frompage', 'topage', 'webpage', 'section', 'subsection', 'date', 'isodate', 'time', 'title'
        for (var css_class in css_selector_classes) {
            if (css_selector_classes.hasOwnProperty(css_class)) {
                var element = document.getElementsByClassName(css_selector_classes[css_class]);
                for (var j = 0; j < element.length; ++j) {
                    element[j].textContent = vars[css_selector_classes[css_class]];
                }
            }
        }
    }
    </script></head><body style="border:0; margin: 0;" onload="subst()">
    <table style="border-bottom: 1px solid black; width: 100%">
      <tr>
        <td class="section"></td>
        <td style="text-align:right">
          Page <span class="page"></span> of <span class="topage"></span>
        </td>
      </tr>
    </table>
    </body></html>


    As can be seen from the example, the arguments are sent to the header/footer
    html documents in get fashion.

  Outlines:
    Wkhtmltopdf with patched qt has support for PDF outlines also known as book
    marks, this can be enabled by specifying the --outline switch. The outlines
    are generated based on the <h?> tags, for a in-depth description of how this
    is done see the Table Of Contents section.

    The outline tree can sometimes be very deep, if the <h?> tags where spread to
    generous in the HTML document.  The --outline-depth switch can be used to
    bound this.

  Table Of Contents:
    A table of contents can be added to the document by adding a toc object to the
    command line. For example:

    wkhtmltopdf toc https://doc.qt.io/archives/qt-4.8/qstring.html qstring.pdf

    The table of contents is generated based on the H tags in the input documents.
    First a XML document is generated, then it is converted to HTML using XSLT.

    The generated XML document can be viewed by dumping it to a file using the
    --dump-outline switch. For example:

    wkhtmltopdf --dump-outline toc.xml https://doc.qt.io/archives/qt-4.8/qstring.html qstring.pdf

    The XSLT document can be specified using the --xsl-style-sheet switch. For
    example:

    wkhtmltopdf toc --xsl-style-sheet my.xsl https://doc.qt.io/archives/qt-4.8/qstring.html qstring.pdf

    The --dump-default-toc-xsl switch can be used to dump the default XSLT style
    sheet to stdout. This is a good start for writing your own style sheet

    wkhtmltopdf --dump-default-toc-xsl
    The XML document is in the namespace "http://wkhtmltopdf.org/outline" it has a
    root node called "outline" which contains a number of "item" nodes. An item
    can contain any number of item. These are the outline subsections to the
    section the item represents. A item node has the following attributes:

   * "title" the name of the section.
   * "page" the page number the section occurs on.
   * "link" a URL that links to the section.
   * "backLink" the name of the anchor the section will link back to.

    The remaining TOC options only affect the default style sheet so they will not
    work when specifying a custom style sheet.

  Contact:
    If you experience bugs or want to request new features please visit
    <https://wkhtmltopdf.org/support.html>
```