> **dyntext —** Process Stata dynamic tags in text file

Description Syntax Options Remarks and examples References
Also see

Description

dyntext converts a dynamic text file—a file containing both plain text and Stata commands—to an output file in text format. Stata processes the Stata dynamic tags (see [P] **dynamic tags**) in the dynamic text file and creates the output text file.

If you want to convert a dynamic text file to an HTML document, see [P] **dyndoc**. If you want to convert a Markdown document to an HTML document, see [P] **markdown**.

Syntax

> dyntext *srcfile* [*arguments*], <u>sav</u>ing(*targetfile*) [*options*]

srcfile is a plain text file containing Stata dynamic tags. *srcfile* and *targetfile* may be any text format (.txt, .html, .do).

arguments are stored in the local macros '1', '2', and so on for use in *srcfile*; see [U] **16.4.1 Argument passing**.

You may enclose *srcfile* and *targetfile* in double quotes and must do so if they contain blanks or other special characters.

options	Description
* <u>sav</u>ing(*targetfile*)	specify the target file to be saved
<u>replace</u>	replaces the target file if it already exists
<u>noremove</u>	do not process <<dd_remove>> and <</dd_remove>> dynamic tags
<u>nostop</u>	do not stop when an error occurs

*saving(*targetfile*) is required.

Options

saving(*targetfile*) specifies the target file to be saved. saving() is required.

replace specifies that the target file be replaced if it already exists.

noremove specifies that <<dd_remove>> and <</dd_remove>> tags should not be processed.

nostop allows the document to continue being processed even if an error occurs. By default, dyndoc stops processing the document if an error occurs. The error can be caused either by a malformed dynamic tag or by executing Stata code within the tag.

Remarks and examples

A dynamic document contains both static narrative and dynamic tags. Dynamic tags are instructions for dyntext to perform certain actions, such as run a block of Stata code, insert the result of a Stata expression in text, export a Stata graph to an image file and include a link to the image file, etc. Any changes in the data or in Stata will change the output as the document is created. The main advantages of using dynamic documents are

- results in the document come from executing commands instead of being copied from Stata and pasted into the document;

- no need to maintain parallel do-files; and

- any changes in data or in Stata are reflected in the final document when it is created.

▷ Example 1

Let us consider an example. Suppose that we have dyntext_ex.txt with the following text that includes Stata dynamic tags. Because we are writing in plain text, we use - to indicate Stata command names, variable names, etc.

—— begin dyntext_ex.txt ————————

```
<<dd_version: 1>>
Using Stata dynamic tags in a text file with the -dyntext- command
==================================================================
Let us consider an example where we study the -mpg- and -weight- variables in
-auto.dta-.  In our examples below, we will first write the commands so that
they will be displayed in our output text file.  Then, we will write the
commands so that Stata will process the Stata dynamic tags, displaying the
results of the Stata commands in the output text file.

We first use the -sysuse- command to load the dataset and then describe
the data using the -describe- command.

<<dd_ignore>>
<<dd_do>>
sysuse auto, clear
describe
<</dd_do>>
<</dd_ignore>>

This produces the following Stata results:

<<dd_do>>
sysuse auto, clear
describe
<</dd_do>>

Now, we want to check if -mpg- is always greater than 0 and less than 100.
We use the -assert- command to perform the check.  In this case, we do not
want to include any output in the output text file, so we use the -quietly-
attribute to modify the behavior of the -dd_do- Stata dynamic tag.

<<dd_ignore>>
<<dd_do:quietly>>
assert mpg > 0 & mpg < 100
<</dd_do>>
<</dd_ignore>>
```

```
<<dd_do:quietly>>
assert mpg > 0 & mpg < 100
<</dd_do>>
```

If the data do not satisfy the conditions, -dyntext- will fail with an error message, which will occur if we run the same -assert- command in a do-file.

Next, we want to summarize the -weight- variable:

```
<<dd_ignore>>
<<dd_do>>
summarize weight
<</dd_do>>
<</dd_ignore>>
```

This produces the following in the output text file:

```
<<dd_do>>
summarize weight
<</dd_do>>
```

We want to use the minimum and maximum values of -weight- in a sentence. Instead of copying and pasting the numbers from the -summarize- output, we can use the -dd_display- Stata dynamic tag with the -r(min)- and -r(max)- stored results

```
<<dd_ignore>>
The variable weight has minimum value <<dd_display: %4.2f 'r(min)'>> and
has maximum value <<dd_display: %4.2f 'r(max)'>>.
<</dd_ignore>>
```

which produces the following in the output text file:

```
> The variable weight has minimum value <<dd_display: %4.2f 'r(min)'>>
and has maximum value <<dd_display: %4.2f 'r(max)'>>.
```

The -dd_display- dynamic tag uses Stata's -display- command to evaluate expressions. It can be used as a calculator. For example, if we want to include the range = max - min in a sentence, instead of calculating the number and then copying and pasting it, we can use

```
<<dd_ignore>>
The variable weight has range <<dd_display: %4.2f 'r(max)'-'r(min)'>>.
<</dd_ignore>>
```

which produces the following in the output text file:

```
> The variable weight has range <<dd_display: %4.2f 'r(max)'-'r(min)'>>.
```

── end dyntext_ex.txt ────────────

To generate the output file in Stata, we then type

```
. dyntext dyntext_ex.txt, saving(dyntext_res.txt)
```

which produces the following:

── begin dyntext_res.txt ───────────

```
Using Stata dynamic tags in a text file with the -dyntext- command
==================================================================
Let us consider an example where we study the -mpg- and -weight- variables in
-auto.dta-.  In our examples below, we will first write the commands so that
they will be displayed in our output text file.  Then, we will write the
commands so that Stata will process the Stata dynamic tags, displaying the
results of the Stata commands in the output text file.

We first use the -sysuse- command to load the dataset and then describe
the data using the -describe- command.

<<dd_do>>
sysuse auto, clear
describe
<</dd_do>>

This produces the following Stata results:

. sysuse auto, clear
(1978 Automobile Data)
. describe
Contains data from /usr/local/stata15/ado/base/a/auto.dta
  obs:            74                          1978 Automobile Data
 vars:            12                          13 Apr 2016 17:45
 size:         3,182                          (_dta has notes)
-------------------------------------------------------------------------------
              storage   display    value
variable name   type    format     label      variable label
-------------------------------------------------------------------------------
make            str18   %-18s                 Make and Model
price           int     %8.0gc                Price
mpg             int     %8.0g                 Mileage (mpg)
rep78           int     %8.0g                 Repair Record 1978
headroom        float   %6.1f                 Headroom (in.)
trunk           int     %8.0g                 Trunk space (cu. ft.)
weight          int     %8.0gc                Weight (lbs.)
length          int     %8.0g                 Length (in.)
turn            int     %8.0g                 Turn Circle (ft.)
displacement    int     %8.0g                 Displacement (cu. in.)
gear_ratio      float   %6.2f                 Gear Ratio
foreign         byte    %8.0g      origin     Car type
-------------------------------------------------------------------------------
Sorted by: foreign

Now, we want to check if -mpg- is always greater than 0 and less than 100.
We use the -assert- command to perform the check.  In this case, we do not
want to include any output in the output text file, so we use the -quietly-
attribute to modify the behavior of the -dd_do- Stata dynamic tag.

<<dd_do:quietly>>
assert mpg > 0 & mpg < 100
<</dd_do>>
```

> If the data do not satisfy the conditions, -dyntext- will fail with an error
> message, which will occur if we run the same -assert- command in a do-file.

> Next, we want to summarize the -weight- variable:

```
<<dd_do>>
summarize weight
<</dd_do>>
```

> This produces the following in the output text file:

```
. summarize weight
    Variable |        Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
      weight |         74    3019.459    777.1936       1760       4840
```

> We want to use the minimum and maximum values of -weight- in a sentence.
> Instead of copying and pasting the numbers from the -summarize- output, we can
> use the -dd_display- Stata dynamic tag with the -r(min)- and -r(max)-
> stored results

> The variable weight has minimum value <<dd_display: %4.2f `r(min)'>> and
> has maximum value <<dd_display: %4.2f `r(max)'>>.

> which produces the following in the output text file:

> > The variable weight has minimum value 1760.00
> and has maximum value 4840.00.

> The -dd_display- dynamic tag uses Stata's -display- command to evaluate
> expressions. It can be used as a calculator. For example, if we want to
> include the range = max - min in a sentence, instead of calculating the
> number and then copying and pasting it, we can use

> The variable weight has range <<dd_display: %4.2f `r(max)'-`r(min)'>>.

> which produces the following in the output text file:

> > The variable weight has range 3080.00.

—————— end dyntext_res.txt ——————

◁

❑ Technical note

Because `quietly` and `capture` suppress the results of the command from being produced, you should not use these prefix commands with Stata code to be converted by `dyntext`.

❑

References

Gillman, M. S. 2018. Some commands to help produce Rich Text Files from Stata. *Stata Journal* 18: 197–205.

Jann, B. 2016. Creating LaTeX documents from within Stata using texdoc. *Stata Journal* 16: 245–263.

Also see

[P] **dynamic tags** — Dynamic tags for Markdown documents

[P] **dyndoc** — Convert dynamic Markdown document to an HTML file

[P] **markdown** — Convert Markdown document to an HTML file