

## mi reshape — Reshape mi data

Description

Remarks and examples

Menu

Also see

Syntax

Options

## Description

mi reshape is Stata's reshape for mi data; see [D] [reshape](#).

## Menu

Statistics > Multiple imputation

## Syntax

## Overview

(The words long and wide in what follows have nothing to do with mi styles mlong, flong, flongsep, and wide; they have to do with reshape's concepts.)

<i>long</i>				<i>wide</i>		
<i>i</i>	<i>j</i>	<i>stub</i>	← reshape →	<i>i</i>	<i>stub1</i>	<i>stub2</i>
1	1	4.1		1	4.1	4.5
1	2	4.5		2	3.3	3.0
2	1	3.3				
2	2	3.0				

To go from long to wide:

```
mi reshape wide stub, i(i) j(j)
```

/ *j* existing variable

To go from wide to long:

```
mi reshape long stub, i(i) j(j)
```

\ *j* new variable

## Basic syntax

Convert mi data from long form to wide form

```
mi reshape wide stubnames, i(varlist) j(varname) [options]
```

Convert mi data from wide form to long form

```
mi reshape long stubnames, i(varlist) j(varname) [options]
```

<i>options</i>	Description
<i>i</i> ( <i>varlist</i> )	<i>i</i> variable(s)
<i>j</i> ( <i>varname</i> [ <i>values</i> ])	long→wide: <i>j</i> , existing variable wide→long: <i>j</i> , new variable optionally specify values to subset <i>j</i>
<u>s</u> tr <i>ing</i>	<i>j</i> is string variable (default is numeric)

---

where *values* is #[-#] [...] if *j* is numeric (the default)  
"string" ["string" ...] if *j* is string

and where *stubnames* are variable names (long→wide), or stubs of variable names (wide→long). Unlike `reshape` (see [D] [reshape](#)), *stubnames* may not contain @ to denote where *j* appears in the name; all *stubnames* must follow the style *stub#*.

## Options

See [D] [reshape](#) for descriptions of the other options.

## Remarks and examples

[stata.com](http://www.stata.com)

The `reshape` command you specify is carried out on the  $m = 0$  data, and then the result is duplicated in  $m = 1, m = 2, \dots, m = M$ .

In `mi reshape`, all variables corresponding to the same *stubnames* must be registered of the same `mi` type: `imputed`, `passive`, or `regular`.

## Also see

[MI] [intro](#) — Introduction to `mi`

[MI] [mi replace0](#) — Replace original data

[D] [reshape](#) — Convert data from wide to long form and vice versa