clonevar — Clone existing variable

Description

clonevar generates `newvar` as an exact copy of an existing variable, `varname`, with the same storage type, values, and display format as `varname`. `varname`’s variable label, value labels, notes, and characteristics will also be copied.

Quick start

Copy contents, label, and value label of `v1` to `newv1`

```
clonevar newv1 = v1
```

Copy observations from `v2` to `newv2` where `v2` is less than 30

```
clonevar newv2 = v2 if v2 < 30
```

Copy the first 20 observations of `v3` to `newv3`

```
clonevar newv3 = v3 in f/20
```

Same as above

```
clonevar newv3 = v3 in 1/20
```

Menu

Data > Create or change data > Other variable-creation commands > Clone existing variable

Syntax

```
clonevar newvar = varname [if] [in]
```

Remarks and examples

clonevar has various possible uses. Programmers may desire that a temporary variable appear to the user exactly like an existing variable. Interactively, you might want a slightly modified copy of an original variable, so the natural starting point is a clone of the original.
Example 1

We have a dataset containing information on modes of travel. These data contain a variable named `mode` that identifies each observation as a specific mode of travel: air, train, bus, or car.

```
. use https://www.stata-press.com/data/r16/travel
. describe mode
    storage  display       value
  variable name  type  format    label    variable label
    mode       byte  %8.0g  travel  travel mode alternatives
. label list travel
  travel:
      1 air
      2 train
      3 bus
      4 car
```

To create an identical variable identifying only observations that contain air or train, we could use `clonevar` with an `if` qualifier.

```
. clonevar airtrain = mode if mode == 1 | mode == 2
(420 missing values generated)
. describe mode airtrain
    storage  display       value
  variable name  type  format    label    variable label
    mode       byte  %8.0g  travel  travel mode alternatives
    airtrain   byte  %8.0g  travel  travel mode alternatives
. list mode airtrain in 1/5

<table>
<thead>
<tr>
<th>mode</th>
<th>airtrain</th>
</tr>
</thead>
<tbody>
<tr>
<td>air</td>
<td>air</td>
</tr>
<tr>
<td>train</td>
<td>train</td>
</tr>
<tr>
<td>bus</td>
<td>.</td>
</tr>
<tr>
<td>car</td>
<td>.</td>
</tr>
<tr>
<td>air</td>
<td>air</td>
</tr>
</tbody>
</table>
```

The new `airtrain` variable has the same storage type, display format, value label, and variable label as `mode`. If `mode` had any characteristics or notes attached to it, they would have been applied to the new `airtrain` variable, too. The only differences in the two variables are their names and values for bus and car.

Technical note

The `if` qualifier used with the `clonevar` command in example 1 referred to the values of `mode` as 1 and 2. Had we wanted to refer to the values by their associated value labels, we could have typed

```
. clonevar airtrain = mode if mode == "air":travel | mode == "train":travel
```

For more details, see [U] 13.11 Label values.
Acknowledgments

clonivar was written by Nicholas J. Cox of the Department of Geography at Durham University, UK, and coeditor of the Stata Journal and author of Speaking Stata Graphics. He in turn thanks Michael Blasnik of Nest Labs and Ken Higbee of StataCorp for very helpful comments on a precursor of this command.

Also see

[D] generate — Create or change contents of variable
[D] separate — Create separate variables