

---

*new series name = algebraic formula ;*  
GENR (SILENT, STATIC) *new series name = algebraic formula ;*  
GENR (SILENT, STATIC) *equation name [new series name] ;*

---

## Function:

GENR computes transformations of one or more series over the current SMPL and stores the result as a new series with the name specified. A previously created TSP equation may also be GENRed and the result stored as a series. TSP equations can be created with FRML, EQSUB, IDENT, FORM or DIFFER.

## Usage:

The first two forms of GENR are the most commonly used: GENR followed by the new variable name, an equal (=) sign, and a formula which should be composed according the rules for TSP equations given in the Basic Rules section of this Reference Manual. Note that the GENR keyword is not required. This formula can involve any of the legal TSP functions and as many variables as desired (subject to overall TSP limits on space).

The third form of GENR is usually used to compute predicted values after a nonlinear estimation (LSQ or FIML). It consists of GENR followed by an equation name and then the name of the variable where you want to place the computed values of the equation. If no such name appears, GENR will put the series in data storage under the name given on the left hand side of the equation. If the equation was implicit (there is no left hand side variable), you must supply a name for GENR to store the results.

If there are any missing values in the input series for GENR or arithmetic errors during computation, missing values will be stored for the affected observations of the output series and warnings will be printed (unless the missing data is part of the argument to a MISS() function). Missing values can be generated directly with the internal names @MVAL, @MISS, @NA or @MV.

If the series appearing on the left hand side of the equation also appears on the right with a lag or lags or leads, it may be updated dynamically as it is computed. A warning message "dynamic GENR" is printed, since versions prior to 4.1 did not update the right hand side lagged dependent variables dynamically. Use the SILENT option or the SUPRES SMPL; command to suppress this message. This is much more efficient than SET with subscripts. For example,  $U = RHO * U(-1) + E$ ; creates an AR(1) variable U.  $PDV = REV + PDV(1)/(1+R)$ ; does a reverse dynamic GENR (future observations are evaluated first). The STATIC option can be used to prevent such dynamic evaluation.

Expressions like X(-1) and X(I) are treated as lags/leads if X is a series; otherwise they are treated as subscripts (like @COEF(1)). Expressions like X(76:2) and M(1,2) are subscripts (evaluate to scalars).

## Options:

SILENT/NOSILENT suppresses the "dynamic GENR" message. It does not suppress error or warning messages.

STATIC/NOSTATIC prevents dynamic evaluation of equations with lagged or led dependent variables.

## GENR

### Examples:

```
GNPL1 = GNP(-1) ;
GENR DP = LOG(PRICE/PRICE(-1)) ;
GENR WAVE = GAMMA*SIN(TREND) ;
```

```
DUM = X > 0;
```

This makes a dummy variable equal to one when X is greater than zero (the logical expression has a value of one when true), and equal to zero otherwise.

```
SCLEVEL = 1*(SC<=6) + 2*(SC>6 & SC<=9) + 3*(SC>9 & SC<=12) + 4*(SC>12) ;
```

This makes SCLEVEL equal to 1, 2, 3, or 4, depending on which range the value of SC falls into (this is essentially a recode operation).

```
GENR CONSEQ CONSFIT ;
GENR IDENT12 ;
```

The last two examples compute the series defined by equations; the first computes a fitted consumption from a previously defined (FRML or IDENT) or estimated consumption equation (FORM) and the second generates an identity to define the variable on the left hand side. This ensures that the identity will hold in the data, which is necessary for proper estimation.

```
SMPL 1,10;
A = 1; B = 1;
SMPL 2,10;
GENR A = A(-1) + 1;
GENR(STATIC) B = B(-1) + 1;
```

This creates A as a time trend (just like the TREND A; command). B is 1,2,2,2,2,2,2,2,2,2,2 .

### Output:

GENR produces no printed output, except a note when a dynamic GENR is being performed. It stores one new or replacement series in data storage.