

```
do — do ... while (exp)
```

[Description](#)[Syntax](#)[Remarks and examples](#)[Also see](#)

## Description

do executes *stmt* or *stmts* one or more times, until *exp* is zero.

## Syntax

```
do stmt while (exp)
```

```
do {  
    stmts  
} while (exp)
```

where *exp* must evaluate to a real scalar.

## Remarks and examples

[stata.com](#)

One common use of do is to loop until convergence:

```
do {  
    lasta = a  
    a = get_new_a(lasta)  
} while (mreldif(a, lasta)>1e-10)
```

The loop is executed at least once, and the conditioning expression is not executed until the loop's body has been executed.

## Also see

[M-2] [break](#) — Break out of for, while, or do loop

[M-2] [continue](#) — Continue with next iteration of for, while, or do loop

[M-2] [for](#) — for (*exp1*; *exp2*; *exp3*) *stmt*

[M-2] [while](#) — while (*exp*) *stmt*

[M-2] [Intro](#) — Language definition