Globals

Pre-defined variables

The exception information message set by 'raise'.

```
$!
```

Array of backtrace of the last exception thrown.

```
$@
```

The string matched by the last successful match.

```
$&
```

The string to the left of the last successful match.

```
$`
```

The string to the right of the last successful match.

```
$'
```

The highest group matched by the last successful match.

```
$+
```

The Nth group of the last successful match. May be > 1.

```
$1
```

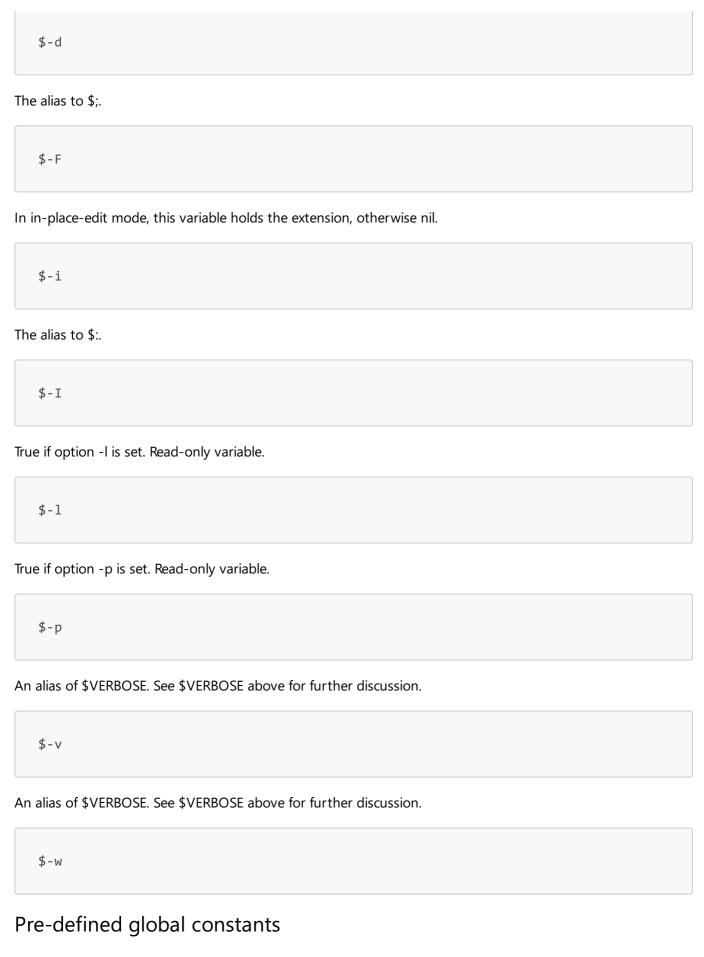$~ The information about the last match in the current scope.

```
$~
```

The flag for case insensitive, nil by default.

```
$=
```

The input record separator, newline by default.

```
$/
```

The output record separator for the print and IO#write. Default is nil.

```
$\
```

The output field separator for the print and Array#join.

```
$,
```

The default separator for String#split.

```
$;
```

The current input line number of the last file that was read.

```
$.
```

The virtual concatenation file of the files given on command line (or from $stdin if no files were given).

```
$<
```

The default output for print, printf. $stdout by default.

```
$>
```

The last input line of string by gets or readline.

```
$_
```

Contains the name of the script being executed. May be assignable.

```
$0
```

Command line arguments given for the script sans args.

```
$*
```

The process number of the Ruby running this script.

```
$$
```

The status of the last executed child process.

```
$?
```

Load path for scripts and binary modules by load or require.

```
$:
```

The array contains the module names loaded by require.

```
$"
```

The debug flag, which is set by the -d switch. Enabling debug output prints each exception raised to $stderr (but not its backtrace). Setting this to a true value enables debug output as if -d were given on the command line. Setting this to a false value disables debug output.

```
$DEBUG
```

Current input file from $<. Same as $<.filename.

```
$FILENAME
```

The alias to the $:.

```
$LOAD_PATH
```

The current standard error output.

```
$stderr
```

The current standard input.

```
$stdin
```

The current standard output.

```
$stdout
```

The verbose flag, which is set by the -w or -v switch. Setting this to a true value enables warnings as if -w or -v were given on the command line. Setting this to nil disables warnings, including from Kernel#warn.

```
$VERBOSE
```

The alias to $/.

```
$-0
```

True if option -a is set. Read-only variable.

```
$-a
```

The alias of $DEBUG. See $DEBUG above for further discusison.

```
$-d
```

The alias to $;.

```
$-F
```

In in-place-edit mode, this variable holds the extension, otherwise nil.

```
$-i
```

The alias to $:.

```
$-I
```

True if option -l is set. Read-only variable.

```
$-l
```

True if option -p is set. Read-only variable.

```
$-p
```

An alias of $VERBOSE. See $VERBOSE above for further discussion.

```
$-v
```

An alias of $VERBOSE. See $VERBOSE above for further discussion.

```
$-w
```

Pre-defined global constants

The typical true value.

```
TRUE
```

The false itself.

```
FALSE
```

The nil itself.

```
NIL
```

The standard input. The default value for $stdin.

```
STDIN
```

The standard output. The default value for $stdout.

```
STDOUT
```

The standard error output. The default value for $stderr.

```
STDERR
```

The hash contains current environment variables.

```
ENV
```

The alias to the $<.

```
ARGF
```

The alias to the $*.

```
ARGV
```

The file object of the script, pointing just after **END**.

```
DATA
```

The ruby version string (VERSION was deprecated).

```
RUBY_VERSION
```

The release date string.

```
RUBY_RELEASE_DATE
```

The platform identifier.

```
RUBY_PLATFORM
```