```
TIEHANDLE classname, LIST
READ this, scalar, length, offset
READLINE this
GETC this
WRITE this, scalar, length, offset
PRINT this, LIST
PRINTF this, format, LIST
BINMODE this
EOF this
FILENO this
SEEK this, position, whence
TELL this
OPEN this, mode, LIST
CLOSE this
DESTROY this
UNTIE this
```

A class implementing a scalar should have the following methods:

```
TIESCALAR classname, LIST
FETCH this,
STORE this, value
DESTROY this
UNTIE this
```

Not all methods indicated above need be implemented. See *perltie*, *Tie::Hash*, *Tie::Array*, *Tie::Scalar*, and *Tie::Handle*.

Unlike dbmopen, the `tie` function will not use or require a module for you–you need to do that explicitly yourself. See DB_File or the *Config* module for interesting `tie` implementations.

For further details see *perltie*, §**??**.

tied VARIABLE

Returns a reference to the object underlying VARIABLE (the same value that was originally returned by the `tie` call that bound the variable to a package.) Returns the undefined value if VARIABLE isn't tied to a package.

time

Returns the number of non-leap seconds since whatever time the system considers to be the epoch, suitable for feeding to `gmtime` and `localtime`. On most systems the epoch is 00:00:00 UTC, January 1, 1970; a prominent exception being Mac OS Classic which uses 00:00:00, January 1, 1904 in the current local time zone for its epoch.

For measuring time in better granularity than one second, you may use either the Time::HiRes module (from CPAN, and starting from Perl 5.8 part of the standard distribution), or if you have gettimeofday(2), you may be able to use the `syscall` interface of Perl. See *perlfaq8* for details.

times

Returns a four-element list giving the user and system times, in seconds, for this process and the children of this process.

```
($user,$system,$cuser,$csystem) = times;
```

In scalar context, `times` returns `$user`.

tr///

The transliteration operator. Same as `y///`. See *perlop*.

truncate FILEHANDLE,LENGTH