```
                if (/}/) {      # end of comment?
                    s|^|$front\{|;
                    redo LINE;
                }
            }
        }
        print;
    }
```

redo cannot be used to retry a block which returns a value such as `eval {}`, `sub {}` or `do {}`, and should not be used to exit a grep() or map() operation.

Note that a block by itself is semantically identical to a loop that executes once. Thus `redo` inside such a block will effectively turn it into a looping construct.

See also continue for an illustration of how `last`, `next`, and `redo` work.

ref EXPR

ref

Returns a non-empty string if EXPR is a reference, the empty string otherwise. If EXPR is not specified, $_ will be used. The value returned depends on the type of thing the reference is a reference to. Builtin types include:

```
SCALAR
ARRAY
HASH
CODE
REF
GLOB
LVALUE
```

If the referenced object has been blessed into a package, then that package name is returned instead. You can think of `ref` as a `typeof` operator.

```
if (ref($r) eq "HASH") {
    print "r is a reference to a hash.\n";
}
unless (ref($r)) {
    print "r is not a reference at all.\n";
}
if (UNIVERSAL::isa($r, "HASH")) {  # for subclassing
    print "r is a reference to something that isa hash.\n";
}
```

See also *perlref*.

rename OLDNAME,NEWNAME

Changes the name of a file; an existing file NEWNAME will be clobbered. Returns true for success, false otherwise.

Behavior of this function varies wildly depending on your system implementation. For example, it will usually not work across file system boundaries, even though the system *mv* command sometimes compensates for this. Other restrictions include whether it works on directories, open files, or pre-existing files. Check *perlport* and either the rename(2) manpage or equivalent system documentation for details.

require VERSION

require EXPR