

NAME

DirHandle - (obsolete) supply object methods for directory handles

SYNOPSIS

```
# recommended approach since Perl 5.6: do not use DirHandle
if (opendir my $d, '.') {
    while (readdir $d) { something($_); }
    rewind $d;
    while (readdir $d) { something_else($_); }
}

# how you would use this module if you were going to
# use DirHandle;
if (my $d = DirHandle->new(".")) {
    while (defined($_ = $d->read)) { something($_); }
    $d->rewind;
    while (defined($_ = $d->read)) { something_else($_); }
}
```

DESCRIPTION

There is no reason to use this module nowadays.

The "DirHandle" method provide an alternative interface to the **opendir()**, **closedir()**, **readdir()**, and **rewinddir()** functions.

Up to Perl 5.5, **opendir()** could not autovivify a directory handle from "undef", so using a lexical handle required using a function from Symbol to create an anonymous glob, which took a separate step. "DirHandle" encapsulates this, which allowed cleaner code than **opendir()**. Since Perl 5.6, **opendir()** alone has been all you need for lexical handles.