

Работа с директории и файлове



Perl

Работа с файлове



Отваряне/създаване на файл

Чрез файлов манипулатор (file handle), който се асоциира към конкретен файл чрез `open FILEHANDLE, MODE, EXPR;`

Отваряне на файл за *четене*:

```
open OUTF, " C:/temp/pass.txt "; ( подразбира се '<' )
```

Отваряне на файл за *запис* (файла се създава, ако не съществува):

```
open OUTF, ">", "C:/temp/pass.txt ";
```

Отваряне на файл за *добавяне*

```
open OUTF, ">>", " C:/temp/pass.txt ";
```

При възникване на грешки:

```
open OUTF, " C:/temp/pass.txt " or die "Can't open the file: $!";
```

Четене от файл

`<FILEHANDLE>` - прочита пореден ред от файла, асоцииран с манипулатора `FILEHANDLE`. Файла трябва да е отворен за четене.

```
open INF, "c:/temp/pass.txt";
while ($line = <INF>){
    print "$line";
}
```

! В `$line` ще се съдържа текущия ред плюс символа за нов ред - по дефиниция: `\n`. При необходимост може да се премахне чрез:

```
chomp ($line);
```

! Текущия номер на прочетен ред се съдържа в автоматичната променлива `'$.'`.

```
print " $. - $line";
```

Запис във файл

`print FILEHANDLE EXPR;`

```
open INF, "c:/temp/pass.txt";
open OUTF, ">", "c:/backup/pass.txt";
while ($line = <INF>) {
    print OUTF "$line";
}
```

`close FILEHANDLE;`

затваряне на файл:

Изтриване на файл

`unlink` FULL_FILENAME; - изтрива указания файл, подобно на системната команда 'del'. Изтрива само файлове, но не и директории. Може да се зададе списък с имена на файлове за изтриване - и тогава функцията връща броя на успешно изтритите файла.

```
$dir = "c:/temp";
opendir DIR, $dir or die "Can't open $dir: $!";
foreach $file ( readdir DIR){
    if ( unlink "$dir/$file" ) {
        print "$file successfully deleted!\n";
    }
    else {
        print "Can't delete $file: $!\n"    };
};
```

Статус на файл

- r : дали потребителя може да чете от файла
- w : дали потребителя може да записва във файла.
- x : дали файла може да бъде стартиран от потребителя.
- e : дали файла съществува.
- s : връща размера на файла ('истина', ако е различен от 0, и 'лъжа' ако е 0)
- d : дали файла е директория.
- T : дали файла е текстов.

Perl

Работа с директории



Четене от директория

`opendir` DIRHANDLE, EXPR; - директорията е достъпна единствено за четене !

`closedir` DIRHANDLE ; - затваря съответната директория.

След като сме асоциирали манипулатор към дадена директория, можем да четем съдържанието ѝ чрез функцията:

`readdir` DIRHANDLE;

- в списъчен контекст връща **списък** от всички файлове в съответната директория (включително и "." и "..");

- в скаларен контекст, връща името (само името, без пътя!) на поредния файл от дадената директория. Ако всички файлове са прочетени - връща undef.

Промяна на текущата директория - чрез :

`chdir` EXPR - сменя текущата директория с тази зададена в EXPR.

Например:

```
chdir("c:/temp") or die "Can't cd: $!";
```

Създаване/изтриване на директория

mkdir DIRNAME, MASK; - създава директорията с име DIRNAME, а правата за тази директория се определят от стойността на MASK. Например:

```
mkdir ( "c:/temp/back", 0777 ) || die "cannot mkdir: $!";
```

- създава директорията "c:/temp/back, с пълни права за всички.
- ако MASK = **0666** => само четене/запис
- ако MASK = **0444** => само четене

rmdir DIRNAME; - изтрива посочената директория

Работа с директории - пример

Извеждане на съдържанието на директория "c:/temp" в азбучен ред, без системните файлове ('.' и '..'):

```
$dir = "c:/temp";  
opendir DIR, $dir or die "Can't open $dir: $!";  
foreach $file ( sort readdir DIR){  
    if ($file ne '.' and $file ne '..') {  
        print "$file\n"  
    }  
}
```

Допълнителен материал: Работа с директории

Друг вариант за четене от директория, който наподобява повече системните команди, е чрез глобални символи за избор на файл/ове (wild cards): *, ?, [].

Използват се в изрази заградени в <> или изрази подавани на `glob`.

Следния пример извежда всички файлове (в текущата директория) с разширение 'mp3' или 'mp4' :

```
@mp3 = glob( " *.mp? " );
foreach (@mp3) {
    print "$_ \n";
}
```

А ако искаме да изведем всички текстови файлове в директория "c:/temp" :

```
$dir="c:/temp";
@txt= glob( " $dir/*.txt " )
foreach (@txt) {
    print "$_ \n";
}
```