

Команда SETLOCAL - изменение переменных среды для текущего командного файла.

Команда **SETLOCAL** используется для временного изменения значений переменных среды в командных файлах Windows. Изменения среды, внесенные после выполнения команды **SETLOCAL**, являются локальными относительно текущего пакетного файла. Для восстановления их прежних значений необходимо выполнить команду **ENDLOCAL**. При достижении конца пакетного файла автоматически выполняется команда **ENDLOCAL** для всех действующих команд **SETLOCAL**, выполненных в этом пакетном файле и значения всех переменных окружения восстанавливаются к состоянию до выполнения команды **SETLOCAL**.

Формат командной строки:

SETLOCAL | ENABLEDELAYEDEXPANSION | DISABLEDELAYEDEXPANSION

Изменение команды **SETLOCAL** при включении расширенной обработки команд:

Команда **SETLOCAL** принимает необязательные аргументы: **ENABLEEXTENSIONS** или **DISABLEEXTENSIONS**. Это позволяет включить или отключить расширенную обработку команд до выполнения команды. См. **CMD /?** для подробного объяснения.

ENABLEDELAYEDEXPANSION / DISABLEDELAYEDEXPANSION включает или отключает отложенное расширение переменной среды. См. **SET /?** для подробного объяснения. Эти изменения действуют до совпадения команды **ENDLOCAL**, независимо от исходного состояния до вызова команды **SETLOCAL**.

Если команда **SETLOCAL** вызывается с аргументом, она устанавливает код ошибки **ERRORLEVEL**. Если указан один из двух допустимых аргументов, код ошибки будет равен нулю, иначе возвращается значение 1. Это свойство можно использовать в пакетных файлах, чтобы определить доступность расширенной обработки команд, например:

VERIFY OTHER 2>nul

SETLOCAL ENABLEEXTENSIONS

IF ERRORLEVEL 1 echo Не удается включить расширенную обработку

Эта группа команд работает, так как в прежних версиях **CMD.EXE** команда **SETLOCAL** не устанавливает значение **ERRORLEVEL**. Команда **VERIFY** с недопустимым аргументом необходима для установки ненулевого значения **ERRORLEVEL**.

При использовании переменных окружения в командных файлах существует определенное ограничение, связанное с тем фактом, что присваиваемое значение остается без изменения при его модификации внутри группы команд, задаваемой скобками, например в командах **IF** или **FOR**. Для обхода данного ограничения используется запуск командного процессора с параметром **/V:ON** и вместо знаков процента, для получения принимаемого переменной значения, используются восклицательные знаки. Кроме того, существует возможность использовать стандартный запуск командного процессора, но с локальным включением данного режима командой :

Setlocal EnableDelayedExpansion

Разница в результатах использования значений переменных довольно наглядно демонстрируется следующим командным файлом:

Setlocal EnableDelayedExpansion

```
@ECHO OFF
set VAR=before
if "%VAR%" == "before" (
set VAR=after
if "!VAR!" == "after" @echo Со знаком процента=%VAR% , Со знаком
вопроса=!VAR!
)
```

Команда **set VAR=after** выполняется внутри подпрограммы, ограниченной скобками и, если убрать команду **Setlocal EnableDelayedExpansion** или не использовать для получения значения переменной **VAR** восклицательные знаки, ее значение останется старым (тем, что было установлено до входа в подпрограмму). Аналогичная же проблема наблюдается и тогда, когда значение переменной изменяется внутри цикла команды **FOR**. Например, для получения списка файлов текущего каталога такой командный файл не будет работать:

```
set LIST=
for %%i in (*) do set LIST=%LIST% %%i
echo %LIST%
```

Значение переменной **LIST** внутри цикла изменено не будет. Для того, чтобы это произошло, командный файл нужно изменить следующим образом:

Setlocal EnableDelayedExpansion

```
set LIST=
for %%i in (*) do set LIST=!LIST! %%i
echo %LIST%
```

Теперь, значение переменной **LIST** внутри цикла **FOR** будет изменяться, последовательно принимая значения имен файлов, разделенных пробелом (**set LIST=!LIST! %%i**)