

### setenv()

```
setenv("MY_VARIABLE",  
"my_value", 1); // set and  
overwrite if exists  
setenv("MY_VARIABLE", " -  
my_value", 0); // set only if  
not already set  
// commonly used to modify PATH  
setenv("PATH", new_path.c -  
_str(), 1);
```

Sets or updates an environment variable for the current process and any child processes it spawns.

Takes three arguments: the variable name, the value, and an overwrite flag.

1 means overwrite if the variable already exists, 0 means leave it unchanged if it exists.

### unsetenv()

```
unsetenv("MY_VARIABLE"); //  
removes MY_VARIABLE from the  
environment  
// verify it was removed  
const char* value = std::getenv(  
"MY_VARIABLE");  
// value == nullptr
```

Removes an environment variable from the current running program and any child processes it spawns.

After calling it the variable no longer exists, so `std::getenv()` would return `nullptr` for it.

Other programs running on the system are completely unaffected — each process has its own copy of environment variables.



By **blakecromar**

[cheatography.com/blakecromar/](https://cheatography.com/blakecromar/)

Not published yet.

Last updated 26th June, 2026.

Page 1 of 1.

Sponsored by **CrosswordCheats.com**

Learn to solve cryptic crosswords!

<http://crosswordcheats.com>