

# ifs workbook pdf

*AI generated article from Bing*

---

## What is the meaning of IFS=\$'\n' in bash scripting?

At the beginning of a bash shell script is the following line: IFS=\$'\n' What is the meaning behind this collection of symbols?

## What is the "IFS" variable? - Unix & Linux Stack Exchange

I was reading this Q&A: How to loop over the lines of a file? What is the IFS variable? And what is its usage in the context of for-loops?

## shell - Understanding IFS - Unix & Linux Stack Exchange

The following few threads on this site and StackOverflow were helpful for understanding how IFS works: What is IFS in context of for looping? How to loop over the lines of a file Bash, read line by...

## Understanding "IFS= read -r line" - Unix & Linux Stack Exchange

Using IFS= LC\_ALL=C read -r line works around it there. Using var=value cmd syntax makes sure IFS / LC\_ALL are only set differently for the duration of that cmd command. History note The read builtin was introduced by the Bourne shell and was already to read words, not lines. There are a few important differences with modern POSIX shells.

## How to temporarily save and restore the IFS variable properly?

How do I correctly run a few commands with an altered value of the IFS variable (to change the way field splitting works and how "\$\*" is handled), and then restore the original value of I...

## For loop over lines -- how to set IFS only for one `for` statement?

Here is an example of behavior I want to achieve: Suppose I have a list of lines, each line containing space separated values: lines='John Smith James Johnson' And I want to loop over lines echoin...

## Is it a sane approach to "back up" the \$IFS variable?

From this follows: A portable script cannot dependably inherit IFS via the environment. A script that intends to use only the default splitting behavior (or joining, in the case of "\$\*"), but which may run under a shell which initializes IFS from the environment, must explicitly set/unset IFS to defend itself against environmental intrusion.

## How to send a command with arguments without spaces?

Or more generally, contains a space. `cat${IFS}file.txt` The default value of IFS is space, tab, newline. All of these characters are whitespace. If you need a single space, you can use `${IFS%??}`. More precisely, the reason this works has to do with how word splitting works. Critically, it's applied after substituting the value of variables.

## understanding the default value of IFS - Unix & Linux Stack Exchange

Here if the expansion contains any IFS characters, then it split into different 'words' before the command is processed. Effectively this means that these characters split the substituted text into different arguments (including the name of the command if the variable is specified first).

## Why is ``while IFS= read`` used so often, instead of ``IFS=; while read..``?

The `IFS= read -r` line sets the environment variable IFS (to an empty value) specifically for the execution of `read`. This is an instance of the general simple command syntax: a (possibly empty) sequence of variable assignments followed by a command name and its arguments (also, you can throw in redirections at any point).