

# Minishell User's Manual

---

This is the Minishell user's manual.

## Table of Contents

1. Introduction .....	2
2. Prerequisites .....	2
3. Installation in 3 steps .....	2
4. Usage.....	3
5. Other Functionalities .....	4
6. Uninstall .....	4

## 1. Introduction

This document is about minishell. It is a simple shell program written in C language that can implement certain functionalities of a Linux shell. This manual will guide you through the necessary steps for the installation and usage of this program.

## 2. Prerequisites

Minishell requires the following:

- An installed GNU Readline library.  
Use the following command to install it to your system.

**sudo apt-get install libreadline6 libreadline6-dev**

*Note: Installation of Minishell also installs the readline library with it. So do not worry if you can't install the library yourself.*

## 3. Installation in 3 steps

Step 1: Go to a Linux shell and login as the system administrator root:

**sudo su (and type your password)**

Step 2: Change your directory to that of where you extracted the folder and type

**tar xvfz abdulsalam-sadik.tar.gz**

The file would be decompressed in the current directory. You can use the option **-C** to decompress in another directory of your choice:

(tar xvfz abdulsalam-sadik.tar.gz -C directory)

Warning: already installed Minish versions get overwritten by this command.

Step 3: Type **make** or **make install** to set up the program to your system.

That's all. Minishell is now installed and ready to use. To use Minishell simply type the command **minishell** on your Linux shell.

Note: You can launch Minishell from your Linux terminal irrespective of your current/working directory.

## 4. Usage

The usage of this Minishell is similar to one in the present in the UNIX environment.

- ❖ Type your command(s) and press enter to execute. You can use **pipes (|)**, **flow redirection (>, >>, <)**, **background (&)**, **single and double quotations (" ", ' ')** etc. No need to type spaces between commands and these symbols. You can also put as much spaces as you want between commands and these symbols.

Examples:

- a. `cmd 1 | cmd2` – connect standard output of cm1 to standard input of cmd2.
  - b. `cmd > file` – writes the output of cmd into file (overwrites file).
  - c. `cmd < file` – puts the contents of file to the standard input of cmd (overwrites file).
  - d. `cmd >> file` – appends the output of command into file.
  - e. `cmd < file1 > file2` - puts the contents of file to the standard input of cmd, executes the command and puts the result in file2.
  - f. `cmd < file1 >> file2` – same as e but appends the result to file2 instead of overwriting it.
- ❖ To create a local variable in the shell, use the syntax ***“variable\_name=variable\_value”***. Type the name of the variable, followed by the equal sign (=), then the value you want to store in the variable. No spaces should be between the variable name, the equal sign and the variable value.
  - ❖ To show the value stores in a variable use the **echo** command followed by a dollar sign (\$), then the name of the variable.
  - ❖ To remove the variable from the shell use the **unset** command followed by the name of the variable. You can unset as much variables as you want.

Examples:

- `dog=barks` (creates a variable with a name “dog” and a value “barks”).
  - `cat=meows` (creates a variable with a name “cat” and a value “meows”).
  - `echo $dog` (Prints the value “barks” on the screen).
  - `echo “$dog”` (Prints the value “barks” on the screen).
  - `echo ‘$dog’` (Prints the value “barks” on the screen).
  - `unset dog cat` (removes the variables dog and cat from the shell).
  - `echo $dog “$cat”` (Prints nothing).
- ❖ To execute a command in background use the ampersand (&) e.g. `gedit &`
  - ❖ Type **exit** to quit the Minishell.

## 5. Other Functionalities

Minishell also takes in to consideration subtle behaviors of the Linux shell.

- You can effortlessly scroll through the history of the last commands you typed by using the up and down arrows on the keyboard. You can use the left and right arrows the keyboard to scroll through a certain command.
- When no command is inputted or when no program is on the foreground, using Ctrl+C is ignored by Minishell.
- Typing Ctrl+D quits the shell.
- Tilde expansions (~) are taken into consideration. So typing `echo ~`, shows you your home path and `cd ~`, changes your current directory to your home directory.

## 6. Uninstall

To uninstall Minishell, in your Linux terminal, go to the extracted Minishell folder and type:  
***make uninstall.***

# THE END