

PROJECT PRESENTATION

MINISHELL - UNIX COMMAND INTERPRETER

LANGUAGE: C

COMPILATION: MAKEFILE

This project is divided into two progressive stages — MiniShell1 and MiniShell2 — with the final goal of creating a fully **functional UNIX-like command-line interpreter**, inspired by **TCSH**, written in pure C. This shell aims to replicate basic behaviors of traditional shells, then extend its features with more complex parsing and execution logic (such as redirections and pipes).

Features:

- Displays a custom **prompt (\$>)** and waits for validated command input.
- Executes **system commands** located in the directories from the PATH variable or via absolute/relative paths.
- Implements **built-in commands** from scratch:
 - cd: Change directory
 - setenv: Add or update environment variables
 - unsetenv: Remove environment variables
 - env: Display environment variables
 - exit: Terminate the shell
- **Handles error output** and status codes properly (stderr + correct return value).
- Restores and **manages the environment** (envp) internally.
- **Semicolon (;)** chaining: execute multiple commands sequentially.
- **Pipes (|)**: connect the output of one command to the input of another.
- Output **redirection** (> and >>)
- Input redirection (< and <<)
- **Priority** and execution flow for combined operations (e.g., cmd1 | cmd2 > file; cmd3).
- And more,...

```
B-PSU-200-PAR-2-2-minishell12-johana.gaba git:(main) x ./mysh
$> ls
include Makefile mysh src
$> pwd
/home/ivsgabi/delivery/EPITECH-PROJECTS/B-PSU-200-PAR-2-2-minishell12-johana.gaba
$> ls -a
. .. .git include Makefile mysh src
$> cd ..
$> lz
Reading directory of standard input.
gzip: compressed data not read from a terminal. Use -f to force decompression.
For help, type: gzip -h
tar: This does not look like a tar archive
tar: Exiting with failure status due to previous errors
$> ls
B-PSU-200-PAR-2-2-minishell12-johana.gaba BSQ
$>
```

```
B-PSU-200-PAR-2-2-minishell12-johana.gaba git:(main) x ./mysh
$> setenv TOTO
$> setenv TATA 7
$> setenv TITI
$> unsetenv TOTO
$> env
SHELL=/usr/bin/zsh
LS_COLORS=Gxfxcxdxbxegedabagacad
SESSION_MANAGER=local/unix:@/tmp/.ICE-unix/4111,unix/unix:/tmp/.ICE-unix/4111
NVM_RC_VERSION=
COLORTERM=truecolor
LESS=-R
TERM_PROGRAM=vscode
TERM_PROGRAM_VERSION=1.99.3
GIT_ASKPASS=/usr/share/code/resources/app/extensions/git/dist/askpass.sh
VSCODE_GIT_ASKPASS_NODE=/usr/share/code/code
VSCODE_GIT_ASKPASS_EXTRA_ARGS=
VSCODE_GIT_ASKPASS_MAIN=/usr/share/code/resources/app/extensions/git/dist/askpass-main.js
VSCODE_GIT_IPC_HANDLE=/run/user/1000/vscode-git-f80cc5f939.sock
VSCODE_INJECTION=1
ZDOTDIR=/home/ivsgabi
USER_ZDOTDIR=/home/ivsgabi
TATA=7
TITI=
$>
```

```
B-PSU-200-PAR-2-2-minishell12-johana.gaba git:(main) x ./mysh
$> mkdir test ; cd test ; ls -a ; ls | cat | wc -c > tutu ; cat tutu
mkdir: cannot create directory 'test': File exists
. .. tutu
```

Why I like this project ?

- Mastery of **low-level UNIX programming and process management**.
- Deep **understanding of how shells work** under the hood.
- Implementation of **real-world system behaviors** (error handling, I/O, process forking).
- **Incremental design**: building a solid base and scaling up functionality progressively.
- Clean code organization with **modular logic** and memory-safe practices.



[Click here to see the project repository](#)