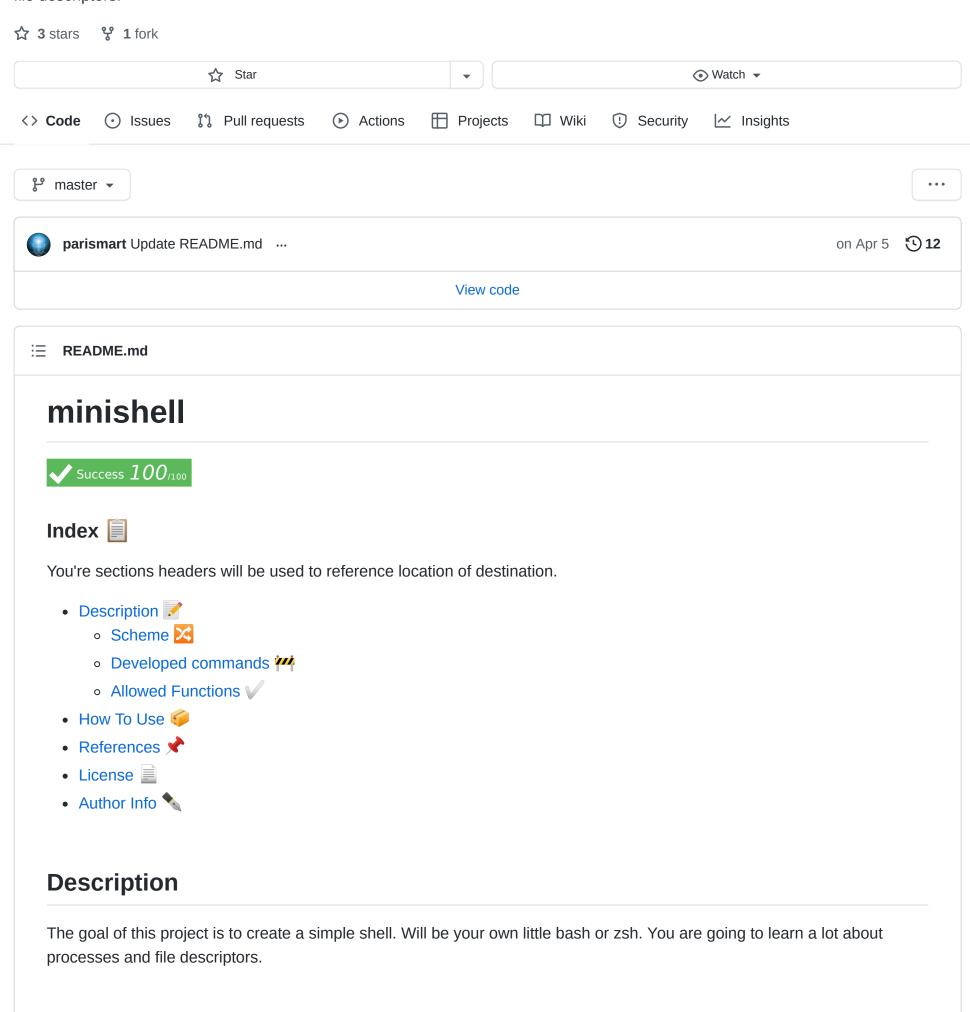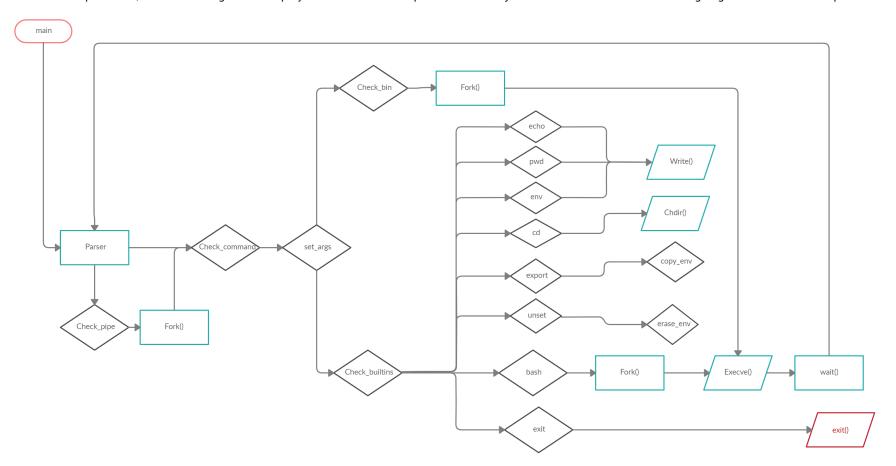🖥 **parismart** / **minishell** `Public`

The goal of this project is to create a simple shell. Will be your own little bash or zsh. You are going to learn a lot about processes and file descriptors.

☆ **3** stars   ⑂ **1** fork

☆ Star ▾	⊙ Watch ▾

‹› **Code**   ⊙ Issues   ⑂ Pull requests   ▶ Actions   ▦ Projects   📖 Wiki   ⊘ Security   📈 Insights

⑂ master ▾   ···

🌐 **parismart** Update README.md   ···   on Apr 5   🕐 **12**

View code

≡ **README.md**

minishell

✔ Success *100*/100

Index 📋

You're sections headers will be used to reference location of destination.

- Description 📝
  - Scheme 🔀
  - Developed commands 🚧
  - Allowed Functions ✔️
- How To Use 📦
- References 📌
- License 📄
- Author Info 🖋

Description

The goal of this project is to create a simple shell. Will be your own little bash or zsh. You are going to learn a lot about processes and file descriptors.

Scheme

[Back To The Top](#)

Developed commands


```
- Find and launch the appropriate executable.
- Builtins:
     - echo (-n flag included)
     - cd (with absolute and relative path)
     - pwd
     - export
     - unset
     - env
     - exit
- ; will have to separate the commands on the command line.
- Redirects >, ">>" y < should work as in bash, except in fd aggregations.
- Environment variables ($ followed by characters) should work like in bash.
- Pipes | should work like in bash.
- $? it should work like in bash.
- ctrl-C, ctrl-D y ctrl-\ should show the same result as in the bash.
```


[Back To The Top](#)

Allowed Functions

- **malloc**
- **free**
- **write**
- **open**
- **read**
- **close**
- **fork** = Creates a child process.
- **wait, waitpid** = Stops the parent process until the child process exit.
- **wait3, wait4** = Are similar to waitpid, but additionally return resource usage information about the child.
- **signal** = Sets a function to handle a signal.
- **kill** = Sends a signal to a process or a group of processes.
- **exit** = Terminates a process immediately, special handle for child processes.
- **getcwd** = Saves the pathname of your current working directory in a string.
- **chdir** = Changes your current working directory.
- **stat, lstat, fstat** = Returns information about a file.
  - stat = Standar.
  - lstat = For symbolic links.
  - fstat = Gets the file from a file descriptor.

- **execve** = Executes a program referred by a variable.
- **dup** = Creates a copy of a file descriptor using the lowest numbereded unused descriptor.
- **dup2** = Creates a copy of a file descriptor using the descriptor number given by the user.
- **pipe** = It's used to create inter-process communication.
- **opendir** = Opens a directory stream.
- **readdir** = Returns a pointer to a dirent structure representing the next directory entry in the directory stream.
- **closedir** = Closes the directory stream.
- **strerror** = Returns an error message.
- **errno** = Number of last error, its a variable.

Back To The Top

How To Use

> Deployment

```
git clone https://github.com/j53rran0/minishell.git
make && ./minishell
```

References

- Variables de entorno
- Sintaxis del shell
- Evaluation

Functions

- fork: parte 1
- fork: parte 2
- fork & pipe
- signal: parte 1
- signal: parte 2
- exec
- stat
- readdir & opendir
- pipe
- wait
- dup & dup2

Back To The Top

License

MIT License

Copyright [2020] [Paris Martínez Ruiz] [Jaime Serrano González]

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Back To The Top

Author Info

Paris

- Twitter - @TroyaParisde
- Instagram - @parisdetroya42
- 42 Madrid - parmarti

Jaime

- 42 Madrid - Jserrano

Back To The Top

Releases

No releases published

Packages

No packages published

Languages

● **C** 97.0%   ● **Makefile** 3.0%