Package: prompt (via r-universe)

June 26, 2024

Title Dynamic 'R' Prompt

Version 1.0.2.9000

Author Gábor Csárdi

Maintainer Gábor Csárdi <csardi.gabor@gmail.com>

Description Set the 'R' prompt dynamically, from a function. The package contains some examples to include various useful dynamic information in the prompt: the status of the last command (success or failure); the amount of memory allocated by the current 'R' process; the name of the R package(s) loaded by 'pkgload' and/or 'devtools'; various 'git' information: the name of the active branch, whether it is dirty, if it needs pushes pulls. You can also create your own prompt if you don't like the predefined examples.

License MIT + file LICENSE

URL https://github.com/gaborcsardi/prompt

BugReports https://github.com/gaborcsardi/prompt/issues

Imports cli

Suggests callr, gert, mockery, pkgload, ps (>= 1.6.0), R6, rstudioapi, testthat, withr

Encoding UTF-8

RoxygenNote 7.2.3

Roxygen list(markdown = TRUE)

Config/testthat/edition 3

Repository https://gaborcsardi.r-universe.dev

RemoteUrl https://github.com/gaborcsardi/prompt

RemoteRef HEAD

RemoteSha 17bd0e1737babf6c6752c31e68ab4ac4bd0f5acd

Contents

`new_prompt_powerline` *This is a Powerline-like prompt*

Description

It is inspired by the https://github.com/powerline/powerline project. This prompt uses some Unicode glyphs that work best with the fonts specifically modified for Powerline: https://github.com/powerline/fonts It also works best on consoles that support ANSI colors.

Usage

```
new_prompt_powerline(
  parts = list("status", "memory", "loadavg", "path", "devtools", "git"),
  colors = powerline_colors(parts)
)
```

Arguments

`parts`	List of strings and functions. Strings are for the built-in powerline pieces, functions are arbitrary functions with four parameters: `expr`, `value`, `ok` and `visible`, and they should return a character string. The builtin pieces are: • `status`: Status of last command, a red or green box. • `memory`: Memory usage of the R process. • `loadavg`: The load average of the system, see `ps::ps_loadavg()`. • `path`: Current working directory. • `devtools`: Package(s) loaded by `pkgload::load_all()` or the same function of devtools. • `git`: git status, see `prompt_git()`.
`colors`	Colors of the parts. Builtin parts have default colors, but you can change them.

Value

`make_prompt_powerline()` returns a function that you can use with `set_prompt()`.

See Also

Other example prompts: prompt_devtools(), prompt_fancy(), prompt_git(), prompt_mem(), prompt_runtime(), prompt_status()

---

prompt *Dynamic R Prompt*

Description

Set the R prompt dynamically, from a function.

See Also

Useful links:

- https://github.com/gaborcsardi/prompt
- Report bugs at https://github.com/gaborcsardi/prompt/issues

---

prompt_devtools *Example prompt that shows the package being developed with devtools*

Description

If git is installed and the current directory is part of a git tree, then also shows all information from prompt_git.

Usage

```
prompt_devtools(...)

devtools_packages()
```

Arguments

...	Ignored.

Value

prompt_devtools() returns the prompt string.

devtools_packages() returns the packages loaded by devtools/pkgload.

See Also

Other example prompts: new_prompt_powerline(), prompt_fancy(), prompt_git(), prompt_mem(), prompt_runtime(), prompt_status()

prompt_fancy *A fancy prompt, showing probably too much information*

Description

It also uses color, on terminals that support it. Is shows:

- Status of last command.
- Memory usage of the R process.
- Load average of the machine.
- Package being developed using devtools, if any.
- Git branch and state of the working tree if within a git tree.

Usage

```
prompt_fancy(expr, value, ok, visible)
```

Arguments

expr	Evaluated expression.
value	Its value.
ok	Whether the evaluation succeeded.
visible	Whether the result is visible.

Value

`prompt_fancy()` returns the prompt string.

See Also

Other example prompts: `new_prompt_powerline()`, `prompt_devtools()`, `prompt_git()`, `prompt_mem()`, `prompt_runtime()`, `prompt_status()`

prompt_git *An example 'git' prompt*

Description

An example 'git' prompt

Usage

```
prompt_git(...)

is_git_dir()

git_branch()

git_arrows()

git_remote_status()

git_dirty()
```

Arguments

...	Unused.

Details

`prompt_git()` is a prompt with information about the git repository in the current working directory. It shows the current branch, whether there are commits to push or pull to the default remote, and whether the working directory is dirty.

`is_git_dir()` checks whether the working directory is in a git tree. If git is not installed, then it always returns `FALSE`.

`git_branch()` returns the name of the current branch.

`git_arrows()` checks the status of the local tree compared to the configured remote.

`git_remote_status()` checks the status of the local tree, compared to a configured remote.

`git_dirty()` checks if the local tree has uncommitted changes. If there are, it returns `"*"`. Note that it also returns `"*"` on a git error, so you might want to use `is_git_dir()` as well.

Value

`prompt_git()` returns the prompt as a string.

`is_git_dir()` returns a logical scalar.

`git_branch()` returns a string. If the repository has no commits, then it returns `"main"`. Note that if git is not available, or fails for any reason, it will also return `"main"`, so you might want to call `is_git_dir()` as well.

`git_arrows()` returns a string that has a down arrow if the remote has extra commits, and a down arrow if the local tree has extra commits compared to the remote. Or both arrows for diverged branches. If it is not the empty string then it adds a leading space character.

`git_remote_status()` returns a numeric vector of length two. The first number is the number of extra commits in the local tree. The second number is the number of extra commits in the remote. If there is no remote, or git errors, it returns a vector of two `NA`s.

`git_dirty()` returns a character string, `"*"` or `""`.

See Also

Other example prompts: new_prompt_powerline(), prompt_devtools(), prompt_fancy(), prompt_mem(), prompt_runtime(), prompt_status()

---

prompt_mem *Example prompt that shows the current memory usage of the R process*

Description

Example prompt that shows the current memory usage of the R process

Usage

```
prompt_mem(...)

memory_usage()
```

Arguments

...	Ignored.

Details

prompt_mem() is a simple example prompt that shows the physical memory allocated by the current process.

memory_usage() is a utility function that shows memory information about the current R process and the system. You can use it to create a custom prompt.

Value

prompt_mem() returns the formatted prompt in a string.

memory_usage() returns a list with entries:

- bytes: the number of bytes of memory the current process uses. This is the 'Resident Set Size', see ps::ps_memory_info().
- formatted: string that formats bytes nicely, with the appropriate unit.
- total: Total physical memory. See ps::ps_system_memory().
- avail: the memory that can be given instantly to processes without the system going into swap. See ps::ps_system_memory().
- percent: Percentage of memory that is taken. See ps::ps_system_memory().

See Also

Other example prompts: new_prompt_powerline(), prompt_devtools(), prompt_fancy(), prompt_git(), prompt_runtime(), prompt_status()

Examples

```
cat(prompt_mem())


memory_usage()
```

`prompt_runtime` *A prompt that shows the CPU time used by the last top level expression*

Description

A prompt that shows the CPU time used by the last top level expression

Usage

```
prompt_runtime(...)
```

Arguments

`...`	Arguments, ignored.

Value

The prompt. ’

See Also

Other example prompts: `new_prompt_powerline()`, `prompt_devtools()`, `prompt_fancy()`, `prompt_git()`, `prompt_mem()`, `prompt_status()`

`prompt_status` *A prompt that shows the status (OK or error) of the last expression*

Description

A prompt that shows the status (OK or error) of the last expression

Usage

```
prompt_status(expr, value, ok, visible)
```

Arguments

expr	Evaluated expression.
value	Its value.
ok	Whether the evaluation succeeded.
visible	Whether the result is visible.

Value

prompt_status() returns the prompt string.

See Also

Other example prompts: new_prompt_powerline(), prompt_devtools(), prompt_fancy(), prompt_git(), prompt_mem(), prompt_runtime()

---

set_prompt *Set and control the prompt*

Description

Set and control the prompt

Usage

```
set_prompt(value)

suspend()

restore()

toggle()
```

Arguments

value	A character string for a static prompt, or a function that is called after the evaluation every expression typed at the R prompt. The function should always return a character scalar.

Details

Function update_prompt() is used to replace the default R prompt with a custom prompt. A custom prompt can be disabled with suspend() and then re-enable with restore(). Function toggle() toggles between the two.

Value

No return value, called for side effects.

Index