

# Package ‘comFuncs’

July 22, 2025

**Type** Package

**Title** Commonly Used Functions for R Shiny Applications

**Version** 0.0.6

**Author** Hanming Tu [aut, cre]

**Maintainer** Hanming Tu <hanming.tu@gmail.com>

**Description** A set of common functions to be used for displaying messages, checking variables, finding absolute paths, starting applications, etc. More functions will be added later.

**Depends** R (>= 3.5.0)

**License** MIT + file LICENSE

**URL** <https://github.com/TuCai/comFuncs>

**BugReports** <https://github.com/TuCai/comFuncs/issues>

**Encoding** UTF-8

**RoxygenNote** 7.2.1

**Imports** shiny (>= 1.3.2), stats (>= 4.0.3)

**Suggests** testthat (>= 2.1.0),

**NeedsCompilation** no

**Repository** CRAN

**Date/Publication** 2022-08-27 22:00:02 UTC

## Contents

|                        |   |
|------------------------|---|
| cvt_class2df . . . . . | 2 |
| echo_msg . . . . .     | 2 |
| get_abs_path . . . . . | 3 |
| is_empty . . . . .     | 4 |
| start_app . . . . .    | 4 |

|              |          |
|--------------|----------|
| <b>Index</b> | <b>6</b> |
|--------------|----------|

---

|              |                                     |
|--------------|-------------------------------------|
| cvt_class2df | <i>Convert a class to data fram</i> |
|--------------|-------------------------------------|

---

**Description**

Convert class or list to a data frame

**Usage**

```
cvt_class2df(x, exc = "^__", condition = FALSE)
```

**Arguments**

|           |                         |
|-----------|-------------------------|
| x         | a class or list         |
| exc       | exclude pattern         |
| condition | condition for excluding |

**Author(s)**

Hanming Tu

**Examples**

```
r1 <- Sys.getenv()
r2 <- cvt_class2df(r1)
```

---

|          |                     |
|----------|---------------------|
| echo_msg | <i>Echo message</i> |
|----------|---------------------|

---

**Description**

This method displays or writes the message based on debug level. The filehandler is provided through environment variable 'log\_fn', and the outputs are written to the file. This method will display message or a hash array based on debug level ('d\_level'). If 'd\_level' is set to '0', no message or array will be displayed. If 'd\_level' is set to '2', it will only display the message level (lvl) is less than or equal to '2'. If you call this method without providing a message level, the message level (lvl) is default to '0'. Of course, if no message is provided to the method, it will be quietly returned. If 'd\_level' is set to '1', all the messages with default message level, i.e., 0, and '1' will be displayed. The higher level messages will not be displayed.

**Usage**

```
echo_msg(prg, step, msg, lvl = 0, fn = NULL)
```

**Arguments**

|      |                                                                                                                                      |
|------|--------------------------------------------------------------------------------------------------------------------------------------|
| prg  | program name calling from                                                                                                            |
| step | step in the program                                                                                                                  |
| msg  | the message to be displayed. No newline is needed in the end of the message. It will add the newline code at the end of the message. |
| lvl  | the message level is assigned to the message. If it is higher than the debug level, then the message will not be displayed.          |
| fn   | log file name                                                                                                                        |

**Value**

message

**Author(s)**

Hanming Tu

**Examples**

```
echo_msg('comFunc', 1, 'This is a test');
```

---

|              |                          |
|--------------|--------------------------|
| get_abs_path | <i>Get absolute path</i> |
|--------------|--------------------------|

---

**Description**

Get absolute directory

**Usage**

```
get_abs_path(dir, relpath)
```

**Arguments**

|         |               |
|---------|---------------|
| dir     | directory     |
| relpath | relative path |

**Author(s)**

Hanming Tu

**Examples**

```
get_abs_path("/Users/htu/myRepo", "scripts")  
# get "/Users/htu/myRepo/scripts"
```

---

|          |                                                   |
|----------|---------------------------------------------------|
| is_empty | <i>Check if a variable is na or null or space</i> |
|----------|---------------------------------------------------|

---

**Description**

check if string or list is empty (na, null or blank spaces).

**Usage**

```
is_empty(x)
```

**Arguments**

|   |                  |
|---|------------------|
| x | a list or string |
|---|------------------|

**Value**

true or false

**Author(s)**

Hanming Tu

**Examples**

```
is_empty(NULL);  
is_empty('');  
is_empty(NA);
```

---

|           |                          |
|-----------|--------------------------|
| start_app | <i>Start R Shiny app</i> |
|-----------|--------------------------|

---

**Description**

start R Shiny apps included in this package.

**Usage**

```
start_app(  
  app_name = "showenv",  
  n = 1,  
  pkg = "comFuncs",  
  pt = NULL,  
  lb = getOption("shiny.launch.browser", interactive()),  
  ht = getOption("shiny.host", "127.0.0.1"),  
  dm = "normal",
```

```
    msg_lvl = NULL,  
    loc = "local"  
)
```

**Arguments**

|                       |                                                          |
|-----------------------|----------------------------------------------------------|
| <code>app_name</code> | app or script name                                       |
| <code>n</code>        | app number                                               |
| <code>pkg</code>      | package name                                             |
| <code>pt</code>       | Port number                                              |
| <code>lb</code>       | define the browser- shiny.launch.browser                 |
| <code>ht</code>       | define the host or ip address                            |
| <code>dm</code>       | display modes are auto, normal or showcase               |
| <code>msg_lvl</code>  | message level                                            |
| <code>loc</code>      | location of the script: local github; default to 'local' |

**Author(s)**

Hanming Tu

**Examples**

```
# this function start an interactive page so we could not add an runnable  
# example  
NULL;
```

# Index

`cvt_class2df`, [2](#)

`echo_msg`, [2](#)

`get_abs_path`, [3](#)

`is_empty`, [4](#)

`start_app`, [4](#)