

# Package: dplyrAssist (via r-universe)

October 30, 2024

**Type** Package

**Title** RStudio Addin for Teaching and Learning Data Manipulation Using 'dplyr'

**Version** 0.1.1

**Imports** shiny (>= 0.13), miniUI (>= 0.1.1), rstudioapi (>= 0.5), shinyWidgets, shinyAce, tidyverse, tidyr, stringr, magrittr, plyr, DT

**Suggests** nycflights13, knitr, rmarkdown

**URL** <https://github.com/cardiomoon/dplyrAssist>

**BugReports** <https://github.com/cardiomoon/dplyrAssist/issues>

**Description** An RStudio addin for teaching and learning data manipulation using the 'dplyr' package. You can learn each steps of data manipulation by clicking your mouse without coding. You can get resultant data (as a 'tibble') and the code for data manipulation.

**License** GPL-3

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 6.0.1

**VignetteBuilder** knitr

**Repository** <https://cardiomoon.r-universe.dev>

**RemoteUrl** <https://github.com/cardiomoon/dplyrassist>

**RemoteRef** HEAD

**RemoteSha** 4593563a09b41fb1f8edb2191946b5e067313883

## Contents

checkInvalid . . . . .	2
codes2kind . . . . .	2
dplyrAssist . . . . .	3

findData . . . . .	4
makeValid . . . . .	4
makeValidCode . . . . .	5

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

---

checkInvalid	<i>Check the validity of code</i>
--------------	-----------------------------------

---

### Description

Check the validity of code

### Usage

```
checkInvalid(temp)
```

### Arguments

temp            A character string to check validity

### Value

The position of the first invalid code or NULL

### Examples

```
checkInvalid(c("iris", "irissss"))
```

---

codes2kind	<i>Differentiate the R code</i>
------------	---------------------------------

---

### Description

Differentiate the R code

### Usage

```
codes2kind(vcodes)
```

### Arguments

vcodes            A character vector to differentiate

### Value

A character vector indicating whether the code returns text or plot

## Examples

```
require(tidyverse)
temp <-c("iris %>% group_by(Species) %>% summarize_all(mean)","table1")
codes2kind(temp)
```

---

dplyrAssist

*A shiny app for learn dplyr*

---

## Description

Interactively manipulate a `data.frame` or a `tibble`. The resulting code will be emitted as an R code.

## Usage

```
dplyrAssist(df = NULL, right = NULL)
```

## Arguments

`df` A `tibble` or a `tbl_df` or a `data.frame` to manipulate  
`right` Optional second data(A `tibble` or a `tbl_df` or a `data.frame`) to join

## Details

This addin can be used to interactively manipulate a `data.frame` or a `tibble` using `dplyr`. The intended way to use this is as follows:

1. Highlight a symbol naming a `data.frame` or a `tibble` in your R session, e.g. `mtcars`,
2. Execute this addin, to interactively manipulate it.

When you're done, the code for data manipulation will be emitted at the cursor position.

## Value

A manipulated `tibble` or `NULL`

## Examples

```
library(tidyverse)
library(nycflights13)

## Not run:
dplyrAssist(band_members,band_instruments)
dplyrAssist(flights)
result<-dplyrAssist(iris)
cat(attr(result,"code"))

## End(Not run)
```

findData *Detect the valid data*

---

**Description**

Detect the valid data

**Usage**

```
findData(codes)
```

**Arguments**

codes            A character vector to detect

**Value**

The last data in codes

**Examples**

```
require(tidyverse)
temp <- "iris %>%
group_by(Species) %>%
summarize_all(mean)

table1"
findData(temp)
```

---

makeValid *Make valid R code*

---

**Description**

Make valid R code

**Usage**

```
makeValid(temp)
```

**Arguments**

temp            A character vector to make a valid code

**Value**

A valid R codes

**Examples**

```
require(tidyverse)
temp <- "iris %>%
group_by(Species) %>%
summarize_all(mean)

table1"
makeValid(temp)
```

---

`makeValidCode`*Make valid R code*

---

**Description**

Make valid R code

**Usage**

```
makeValidCode(codes)
```

**Arguments**

`codes`            A character vector to make a valid code

**Value**

A valid R codes

**Examples**

```
require(tidyverse)
temp <- "iris %>%
group_by(Species) %>%
summarize_all(mean)

table1"
makeValidCode(temp)
```

# Index

`checkInvalid`, [2](#)

`codes2kind`, [2](#)

`dplyrAssist`, [3](#)

`findData`, [4](#)

`makeValid`, [4](#)

`makeValidCode`, [5](#)