

# Package ‘VarBundle’

August 17, 2018

**Type** Package

**Title** Read-Only Variable Bundles

**Version** 0.3.0

**Author** Brandon C. Loudermilk

**Maintainer** Brandon C. Loudermilk <b.c.loudermilk@gmail.com>

**Description** Easily create list-like structures with constant, read-only variables. Combines the flexibility of lists with read-only fields. Supports defensive programming by throwing errors on field assignment, helping mitigate the introduction of logical runtime bugs.

**License** MIT + file LICENSE

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 6.1.0

**Depends** R (>= 3.5.0)

**Imports** R6 (>= 2.2.2), glue (>= 1.3.0), tibble (>= 1.4.2), magrittr (>= 1.5), dplyr (>= 0.7.6), stringr (>= 1.3.1), crayon (>= 1.3.4)

**Suggests** testthat (>= 2.0.0), purrr (>= 0.2.5), knitr (>= 1.20), rmarkdown (>= 1.10), readr (>= 1.1.1)

**VignetteBuilder** knitr

**NeedsCompilation** no

**Repository** CRAN

**Date/Publication** 2018-08-17 08:40:10 UTC

## R topics documented:

field_names . . . . .	2
print.VarBundle . . . . .	2
varbundle . . . . .	3

<b>Index</b>	<b>4</b>
--------------	----------

field\_names                      *Names of VarBundle*

---

### **Description**

Returns field names of VarBundle Object

### **Usage**

```
field_names(x)
```

### **Arguments**

x                                      (VarBundle)

### **Value**

(character vector)

---

print.VarBundle                      *Print VarBundle*

---

### **Description**

Prints a VarBundle object.

### **Usage**

```
## S3 method for class 'VarBundle'  
print(x, ...)
```

### **Arguments**

x                                      (VarBundle)  
...                                    (additional args)

---

`varbundle`*Create bundle of variables.*

---

## Description

VarBundle objects store read-only variables in a list-like structure. Variable values can be accessed via the '\$' operator (`vb$foobar`) or via field name (`vb[["foobar"]]`). After a VarBundle object has been created, new fields cannot be added and existing field values cannot be modified.

Field names can be accessed via [field\\_names](#).

Simple VarBundle objects that only contain atomic, scalar values are useful for storing configuration information. The object method `$as.data.frame()` will return a `data.frame` version of the VarBundle. This can be persisted to disk and subsequently modified by the end user to change system parameters.

## Usage

```
varbundle(x)
```

## Arguments

`x` (list or `data.frame`)

## Value

(VarBundle)

## Examples

```
# Access via $
sales <- varbundle(list(min = 1, max = 10))
sales$min

# Access via name
my_var <- field_names(sales)[1]
sales[[my_var]]

# Create data.frame
df <- sales$as.data.frame()

# Create VarBundle from data.frame
corp_sales <- varbundle(df)
class(corp_sales)
```

# Index

`field_names`, [2](#), [3](#)

`print.VarBundle`, [2](#)

`varbundle`, [3](#)