

# Package ‘envstocker’

November 9, 2018

**Title** stocking and loading execution environments of functions

**Version** 0.0.0.9000

**Description** For usage example, see descripton for ``.ee.l'' function.

**Depends** R (>= 3.4.4)

**License** GPL-3

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 6.1.0

## R topics documented:

.ee. . . . .	2
.ee.a . . . . .	2
.ee.append . . . . .	2
.ee.backup . . . . .	3
.ee.clear . . . . .	3
.ee.e2list . . . . .	4
.ee.get . . . . .	4
.ee.l . . . . .	5
.ee.look . . . . .	7
.ee.ls.str . . . . .	7
.ee.r . . . . .	8
.ee.recover . . . . .	8
.ee.set . . . . .	9
.ee.str . . . . .	9
.ee.subt_time . . . . .	10

<b>Index</b>	<b>11</b>
--------------	-----------

---

`.ee.`                      *a short-name version of ".ee.ls.str()".*

---

### **Description**

This function is a short-name version of ".ee.ls.str()".

### **Usage**

`.ee.(n = 1)`

---

`.ee.a`                      *a short-name version of ".ee.append()".*

---

### **Description**

This function is a short-name version of ".ee.append()".

### **Usage**

`.ee.a(fname, env, i)`

---

`.ee.append`                *register execution environment of focal function*

---

### **Description**

This function registers the execution environment of the focal function to stock list ".ee".

### **Usage**

`.ee.append(fname, env, i)`

### **Arguments**

<code>fname</code>	character : function name.
<code>env</code>	environment : usually "environment()".
<code>i</code>	integer : used to additional information to environment names.

### **Author(s)**

Hiroshi C. Ito

### Examples

```
testfunc <- function(x){  
  .ee.append("testfunc",environment())  
  return(x+1)}  
.ee.set()  
testfunc(3)  
.ee.str(0)
```

---

*.ee.backup*                      *set temporal .GlobalEnv*

---

### Description

This function copies objects in the original `.GlobalEnv` to the environment named `".ee.b"`, and remove all objects in `.GlobalEnv` except `".ee*"`.

### Usage

```
.ee.backup()
```

### Author(s)

Hiroshi C. Ito

### Examples

```
.ee.set()  
.ee.backup()
```

---

*.ee.clear*                      *removes objects in .GlobalEnv.*

---

### Description

This function removes all objects in `.GlobalEnv` except `".ee*"`

### Usage

```
.ee.clear()
```

### Author(s)

Hiroshi C. Ito

### Examples

```
.ee.set()  
.ee.clear()
```

`.ee.e2list`                    *convert list of environments to a normal list*

---

### **Description**

This function converts a list of environments to a normal list

### **Usage**

```
.ee.e2list(elist)
```

### **Arguments**

`elist`                    list of environment

### **Author(s)**

Hiroshi C. Ito

### **Examples**

```
testfunc <- function(x){  
  .ee.append("testfunc",environment())  
  return(x+1)}  
.ee.set()  
testfunc(3)  
.ee.e2list(.ee)
```

---

`.ee.get`                    *converts to list*

---

### **Description**

This function converts from ".ee" to a normal list

### **Usage**

```
.ee.get(n = 1)
```

### **Arguments**

`n`                    integer : if not specified, then only the last environment is returned. If 0, all environments are returned. If positive, a list from  $[\text{length}(\text{.ee})-n + 1]$  to  $\text{length}(\text{.ee})$  is returned. If negative, a list from 1 to  $-n$  is returned.

**Author(s)**

Hiroshi C. Ito

**Examples**

```
testfunc <- function(x){
  .ee.append("testfunc",environment())
  return(x+1)}
.ee.set()
testfunc(3)

.ee.get()
.ee.get(0)
.ee.get(2)
.ee.get(-2)
```

---

<code>.ee.l</code>	<i>load objects from a stocked environment</i>
--------------------	--

---

**Description**

This function loads objects from a stocked environment specified by "ee". If "ee" is not specified, then the last stocked environment is used.

**Usage**

```
.ee.l(ee = .ee[[length(.ee)]], vartop = "")
```

**Arguments**

- ee environment from which objects are loaded.
- vartop integer(0) or character : if not specified, then only the arguments of the function are loaded. If 0 then all objects are loaded. If "varname", objects upto "varname" are loaded. When "varname" is the first argument of the function, all arguments are loaded.

**Details**

By adding ".ee.append("function\_name",environment())" at the first line of functions called in your R script, and add ".ee.set()" in your R script, the execution environments of those functions are stocked in a list named ".ee". Each environment can be loaded afterward in .GlobalEnv by ".ee.l" function, where objects in .Globalenv are copied to an environment named ".ee.b", and ".ee.b" becomes the parent environment of .GlobalEnv. The original environment is recovered by ".ee.recover()".

**Author(s)**

Hiroshi C. Ito

**Examples**

```

fact <- function(nn,i,flag_ppp=FALSE){
  .ee.append("fact",environment());
  cat("i =",i," nn =",nn,"\n");
  if(flag_ppp){
    if(nn==2)print(ppp);
  }
  y=nn;
  if(y<=1){
    x=1;
  }else{
    x=y*fact(y-1,i+1,flag_ppp=flag_ppp);
  }
  return(x);
}

.ee.set();
fact(3,1,flag_ppp=FALSE);
.ee.subt_time();

ls.str()
ls.str(all.names=TRUE)

.ee.(0)

.ee.l()
ls.str()

.ee.l(vartop=0)
ls.str()

.ee.l(.ee$fact.2)
ls.str()

.ee.recover()
ls.str()

as.list(.ee$fact.2)

.ee.set()
fact(3,1,flag_ppp=TRUE) ##This causes error because ppp does not exist
.ee.subt_time()

.ee.(0)

.ee.l()
ls.str()

.ee.l(vartop=0)
ls.str()
.ee.recover()

```

---

.ee.look                      *load objects from environment without taking backup*

---

**Description**

This function load object from a stocked environment, without calling ".ee.backup()".

**Usage**

```
.ee.look(ee = .ee[[length(.ee)]], vartop = "")
```

**Arguments**

ee	environment from which objects are loaded.
vartop	integer(0) or character : if not specified, then only the arguments of the function are loaded. If 0 then all objects are loaded. If "varname", objects upto "varname" are loaded. When "varname" is the first argument of the function, all arguments are loaded.

**Author(s)**

Hiroshi C. Ito

**Examples**

```
testfunc <- function(x){  
  .ee.append("testfunc",environment())  
  return(x+1)}  
.ee.set()  
testfunc(3)  
  
.ee.look()
```

---

.ee.ls.str                      *show structure of stocked environments by "ls.str"*

---

**Description**

This function shows the structure of ".ee.get()" by "ls.str" function instead of "ls".

**Usage**

```
.ee.ls.str(n = 1)
```

**Arguments**

`n` integer : if not specified, then only the last environment is returned. If 0, all environments are returned. If positive, a list from `[length(.ee)-n + 1]` to `length(.ee)` is returned. If negative, a list from 1 to `-n` is returned.

**Author(s)**

Hiroshi C. Ito

**Examples**

```
testfunc <- function(x){
  .ee.append("testfunc",environment())
  return(x+1)}
.ee.set()
testfunc(3)
.ee.ls.str()
.ee.ls.str(0)
```

---

`.ee.r` *a short-name version of ".ee.recover()".*

---

**Description**

This function is a short-name version of ".ee.recover()".

**Usage**

```
.ee.r()
```

---

`.ee.recover` *recover original .GlobalEnv*

---

**Description**

This function recovers the original `.GlobalEnv` by copying objects from `".ee.b"`. When the current `.GlobalEnv` is the original one, `".ee.recovered"` is `"TRUE"`.

**Usage**

```
.ee.recover()
```

**Author(s)**

Hiroshi C. Ito



### Examples

```
.ee.set()  
.ee.backup()  
.ee.recover()
```

---

.ee.set	<i>initialize a environment stock list</i>
---------	--

---

### Description

This function generates a list of environment named ".ee".

### Usage

```
.ee.set()
```

### Author(s)

Hiroshi C. Ito

### Examples

```
.ee.set()
```

---

.ee.str	<i>show structure of stocked environments</i>
---------	---

---

### Description

This function shows the structure of ".ee.get()"

### Usage

```
.ee.str(n = 1)
```

### Arguments

n	integer : if not specified, then only the last environment is returned. If 0, all environments are returned. If positive, a list from [length(.ee)-n + 1] to length(.ee) is returned. If negative, a list from 1 to -n is returned.
---	---

### Author(s)

Hiroshi C. Ito

**Examples**

```
testfunc <- function(x){  
  .ee.append("testfunc",environment())  
  return(x+1)}  
.ee.set()  
testfunc(3)  
  
.ee.str()  
.ee.str(0)
```

---

.ee.subt_time	<i>calculate ellapse time</i>
---------------	-------------------------------

---

**Description**

This function calculate elapsed time and put it to ".ee\$time"

**Usage**

```
.ee.subt_time()
```

**Author(s)**

Hiroshi C. Ito

**Examples**

```
.ee.set()  
.ee.subt_time()
```

# Index

.ee., 2  
.ee.a, 2  
.ee.append, 2  
.ee.backup, 3  
.ee.clear, 3  
.ee.e2list, 4  
.ee.get, 4  
.ee.l, 5  
.ee.look, 7  
.ee.ls.str, 7  
.ee.r, 8  
.ee.recover, 8  
.ee.set, 9  
.ee.str, 9  
.ee.subt\_time, 10