SM.EXEC - Lua wrapper

```
sm = require("sm")
```

SMEvent

```
e = sm.new_event(id, data_size)
e:set(str)
str = e:get()
e:setid(id)
id = e:getid()
e1..e2
e3 = e1:next()
-e1
__len
__tostring
__gc
```

SMQueue

```
q = sm.new_queue(qsize, plsize, sync)
e = q:top()
q.enqueue(e)
e = q.dequeue()
__len
__tostring
__gc
```

SMQueue2

```
q = sm.new_queue2()
e = q:get()
e = q:gethigh()
q:enqueue(e)
q:lockenqueue(e)
q:enqueuehigh(e)
q:lockenqueuehigh(e)
e = q.dequeue()
e = q.lockdequeue()
__len
__tostring
__gc
```

SMApp

```
app = sm.lookup(handle, name)
app(e)
__tostring
```

SMAppTable

```
handler = sm:loadlib(file)
at = sm.new_apptab()
at:set(app, name)
at:get(name)
at:remove(name)
__tostring
__gc
__len
```

SMFSM

```
fsm = sm.new_fsm(fsm_json, at, type)
__tostring
__gc
```

SMState

```
s = sm.new_state(fsm, plsize)
s:traceadd(e)
e = s:traceget()
s:setkey(str)
str = s:getkey()
s:set(str)
str = s:get()
s:setid(id)
id = s:getid()
s:purge()
s:apply(e)
__tostring
__gc
```

SMArray

```
a = sm.new_array(stsize, plsize)
s = a:find(str)
s = a:get(str)
a:release(s)
__tostring
__gc
__len
```

10-01-2018 00:12PM