Convenience Functions

Janko Thyson

Thursday, October 29, 2014

Reactive objects (focus *visible* part)

Example objects

```r
setReactiveS3(id = "x_1", value = 10)
setReactiveS3(id = "x_2", value = function() "object-ref: {id: x_1}")
```

Compute object UID

```r
computeObjectUid(id = "x_1")
```

```
## [1] "2fc2e352f72008b90a112f096cd2d029"
```

```r
computeObjectUid(id = "x_2")
```

```
## [1] "ab22808532ff42c87198461640612405"
```

Copy reactive objects

```r
copyReactive(id_from = "x_1", id_to = "x_1_copy")
x_1_copy
```

```
## [1] 10
```

Get reactive

```r
getReactive(id = "x_1")
```

```
## [1] 10
```

```r
getReactive(id = "x_1", hidden = TRUE)
```

```
## <environment: 0x0000000009a91330>
## attr(,"class")
## [1] "ReactiveObject.S3" "environment"
```

```r
getReactive(id = "x_2")
```

```
## [1] 10
```

Is reactive

```r
isReactive(id = "x_1")
```

```
## [1] TRUE
```

```r
isReactive(id = "x_2")
```

```
## [1] TRUE
```

```r
x_3 <- "hello world!"
isReactive(id = "x_3")
```

```
## [1] FALSE
```

Unset reactive

```r
unsetReactive(id = "x_1")
```

```
## [1] TRUE
```

```r
isReactive(id = "x_1")
```

```
## [1] FALSE
```

Removing reactive

```r
rmReactive(id = "x_1")
```

```
## [1] TRUE
```

```r
exists("x_1")
```

```
## [1] FALSE
```

```r
## --> already unset
rmReactive(id = "x_2")
```

```
## [1] TRUE
```

```r
exists("x_2")
```

```
## [1] FALSE
```

Reactive objects (focus *invisible* part)

Example objects

```r
setReactiveS3(id = "x_1", value = 10)
setReactiveS3(id = "x_2", value = function() {
  "object-ref: {id: x_1}"
  x_1 * 2
})
```

Get binding function

```r
getBinding(id = "x_1")
```

```
## NULL
```

```r
getBinding(id = "x_2")
```

```
## function ()
## {
##     x_1 <- get(x = "x_1", envir = <environment>, inherits = FALSE)
##     x_1 * 2
## }
```

Get checksum of visible value

```r
getChecksum(id = "x_1")
```

```
## [1] "2522027d230e3dfe02d8b6eba1fd73e1"
```

```r
getChecksum(id = "x_2")
```

```
## [1] "c72d0198505540505acfb4b2b49911e6"
```

Get reactive

```r
getReactive(id = "x_1", hidden = TRUE)
```

```
## <environment: 0x0000000009bd5160>
## attr(,"class")
## [1] "ReactiveObject.S3" "environment"
```

```r
getReactive(id = "x_2", hidden = TRUE)
```

```
## <environment: 0x0000000009a01490>
## attr(,"class")
## [1] "ReactiveObject.S3" "environment"
```

Show pull references

```r
showPullRefs(id = "x_1")
```

```
## character(0)
```

```r
showPullRefs(id = "x_2")
```

```
## [1] "2fc2e352f72008b90a112f096cd2d029"
```

Show push references

```r
showPushRefs(id = "x_1")
```

```
## character(0)
```

```r
showPushRefs(id = "x_2")
```

```
## character(0)
```

```r
setReactiveS3(id = "x_2", value = function() {
  "object-ref: {id: x_1}"
  x_1 * 2
}, push = TRUE)
showPushRefs(id = "x_2")
```

```
## [1] "2fc2e352f72008b90a112f096cd2d029"
```

Clean up

```r
rmReactive("x_1")
```

```
## [1] TRUE
```

```r
rmReactive("x_2")
```

```
## [1] TRUE
```

Registry

Example objects

```r
setReactiveS3(id = "x_1", value = 10)
setReactiveS3(id = "x_2", value = function() "object-ref: {id: x_1}")
```

Get registry

```r
reg <- getRegistry()
ls(reg)
```

```
## [1] "2fc2e352f72008b90a112f096cd2d029" "ab22808532ff42c87198461640612405"
## [3] "ea8c305f523234921d3ff8b7c8bab390"
```

Show registry

```r
showRegistry()
```

```
## [1] "2fc2e352f72008b90a112f096cd2d029" "ab22808532ff42c87198461640612405"
## [3] "ea8c305f523234921d3ff8b7c8bab390"
```

Get from registry

```r
getFromRegistry(id = "x_1")
```

```
## <environment: 0x0000000009b59918>
## attr(,"class")
## [1] "ReactiveObject.S3" "environment"
```

```r
getFromRegistry(id = "x_2")
```

```
## <environment: 0x000000000988dcd0>
## attr(,"class")
## [1] "ReactiveObject.S3" "environment"
```

```r
getFromRegistryByUid(computeObjectUid("x_1"))
```

```
## <environment: 0x0000000009b59918>
## attr(,"class")
## [1] "ReactiveObject.S3" "environment"
```

```r
getFromRegistryByUid(computeObjectUid("x_2"))
```

```
## <environment: 0x000000000988dcd0>
## attr(,"class")
## [1] "ReactiveObject.S3" "environment"
```

Remove from registry

```r
computeObjectUid("x_1") %in% showRegistry()
```

```
## [1] TRUE
```

```r
rmFromRegistry("x_1")
```

```
## [1] TRUE
```

```r
computeObjectUid("x_1") %in% showRegistry()
```

```
## [1] FALSE
```

```r
computeObjectUid("x_2") %in% showRegistry()
```

```
## [1] TRUE
```

```r
rmFromRegistryByUid(computeObjectUid("x_2"))
```

```
## [1] TRUE
```

```r
computeObjectUid("x_2") %in% showRegistry()
```

```
## [1] FALSE
```

Reset registry

```r
## Ensure content that might previously have been removed //
setReactiveS3(id = "x_1", value = 10)
setReactiveS3(id = "x_2", value = function() "object-ref: {id: x_1}")

## Reset //
showRegistry()
```

```
## [1] "2fc2e352f72008b90a112f096cd2d029" "ab22808532ff42c87198461640612405"
## [3] "ea8c305f523234921d3ff8b7c8bab390"
```

```r
resetRegistry()
```

```
## [1] TRUE
```

```r
showRegistry()
```

```
## character(0)
```