

UDMS Library Documentation

Nik and Thn

05-10-2020

Room (functionsRoom.lua)

All the examples assume that the 'Room' variable contains a LuaRoom. This is true by default inside all the individual, group and settings scripts.

getRoomName(room)

Returns the name of the room.

```
local roomName = require('functionsRoom').getRoomName(Room)
```

getScriptPath(room)

Returns the containing folder of the room's settings.lua.

```
local scriptPath = require('functionsRoom').getScriptPath(Room)
```

getSceneName(room)

Returns the name of the unity scene the room is using.

```
local sceneName = require('functionsRoom').getSceneName(Room)
```

useCameraScript(room)

Use the camera.lua script in the room/scenario. The camera.lua script must exist in the room's path.

```
require('functionsRoom').useCameraScript(Room)
```

addCamera(room)

Alias for useCameraScript(room)

```
require('functionsRoom').addCamera(Room)
```

getGroups(room)

Returns a dictionary of the groups in the room. Format: key: groupName, value: group

```
local groups = require('functionsRoom').getGroups(Room)
```

getGroupNames(room)

Returns a list with the names of all the groups in the room.

```
local groupNames = require('functionsRoom').getGroupNames(Room)
```

getGroup(room, groupName)

Returns a room's group from its name.

```
local groupName = 'dancers'  
local group = require('functionsRoom').getGroup(Room, 'dancers')
```

TODO: etc

Group (functionsGRP.lua)

The ‘LFG’ variable in all the examples below is assumed to be defined like, where the ‘Group’ variable contains a LuaGroup. The ‘Group’ variable exists by default inside each group script. It can also be set on any script using the room’s getGroup function.

Also, all the group member ids that appear are assumed to be in range.

```
local LFG = require('functionsGRP')(Group)
```

getMembers()

Returns a list of the group’s members.

```
local members = LFG.getMembers()
```

dirToAgent(i, j)

Returns the direction (Vector3) of the agent with id i to the agent with id j.

```
local dir_0_to_2 = LFG.dirToAgent(0, 2)
```

TODO: etc

Objects (functionsOBJ.lua)

setParent(go, parentGo)

Sets the go as a child of parentGo.

```
-- Assuming go and parentGo are gameObjects or MonoBehaviours  
require(functionsOBJ').setParent(go, parentGo)
```

TODO: etc

Lights

lgtGetColor(go)

Sets the color of the light component of the go.

```
-- Assuming go is a gameObject/MonoBehaviour on a gameObject with a light component  
require(functionsOBJ').lgtGetColor(go)
```

TODO: etc

Trails

trailGetTime(go)

Returns the alive time of the trail particles.

```
-- Assuming go is a gameObject/MonoBehaviour on a gameObject with a trail component IN CHILDREN  
local time = require(functionsOBJ').trailGetTime(go)
```

TODO: etc

Camera (functionsCAM.lua)

`stateUpdate(TIME, state)`

Updates the camera based on the TIME and state parameters. If the state isn't given, it uses the camera's current state.

```
-- Updates the camera using the update3 function.  
require('functionsCAM')(self).stateUpdate(TIME, 3)
```

TODO: etc

Effects (effects.lua)

`setProperty(effectProperty, value)`

Sets the effectProperty to the value specified.

```
require('effects').setProperty(lutEffect.Saturation, 0.5)
```

TODO: etc

Utilities (utils.lua)

TODO: etc

Logic (logic.lua)

Animations List (animations.lua)

Look Up Textures List (luts.lua)

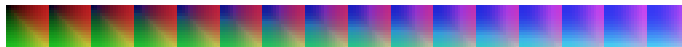


Figure 1: Action3D16