Creating objects for Varwin in Unity

[Переключить на русский](https://docs.google.com/document/d/1iFTH4kVb7cpsRCIlG6E4HXQKONMxZiTd8wug5TFAtm4)

Contents

[Creating objects](#_nfp9hphjbxbo)

[Example](#_30j0zll)

[Working with attributes](#_qwayougn0to3)

[Logic](#_xu5d9u2wkexi)

[Actions](#_5bx3ycihio6g)

[Events](#_fwd4nseio9vq)

[Variables](#_302sdr37njhf)

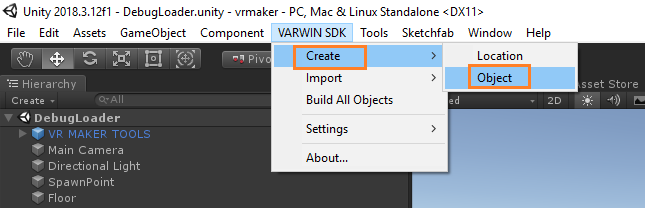
[Creating drop-down lists](#_qr5in1iawttz)

[Objects versioning](#_80el0tu7svds)

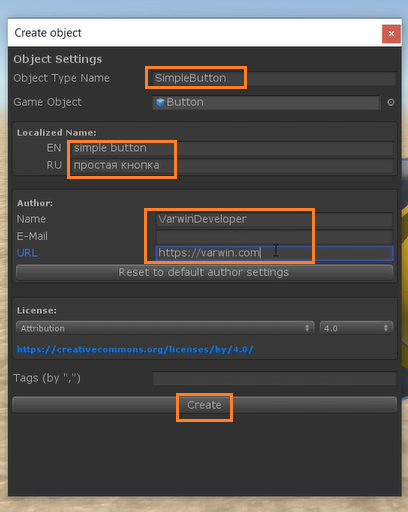
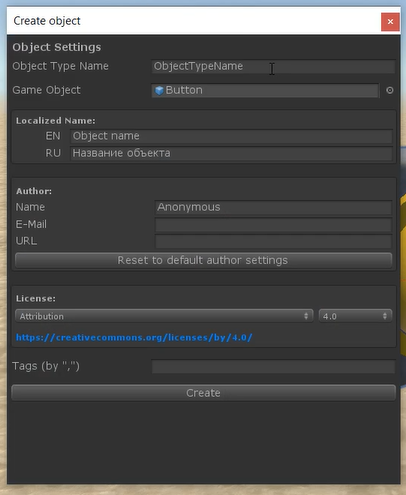
[Additional materials](#_qp3e5u42knt1)

# Creating objects

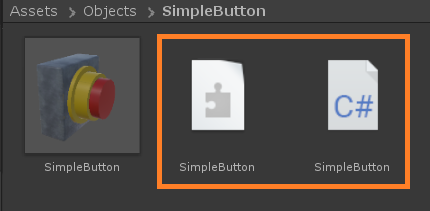
* Create or import a game object and save it as a prefab.
* Select the prefab in its folder.
* Open VARWIN SDK -> Create -> Object



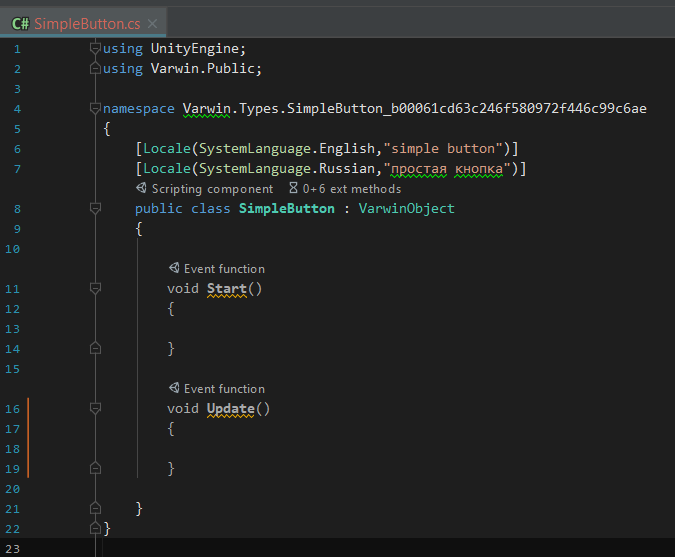
* Select an object, name it. Add tags for fast search. Fill in the info about the object’s author.
* Click Create. You’ll be asked to wait several moments



* Your object has been created. It consists of:
  1. object prefab,
  2. .asmdef file (for object code compilation),
  3. object class stored in a unique namespace. Should the need arise to add more classes, they will have to be stored in the same directory and the same namespace.



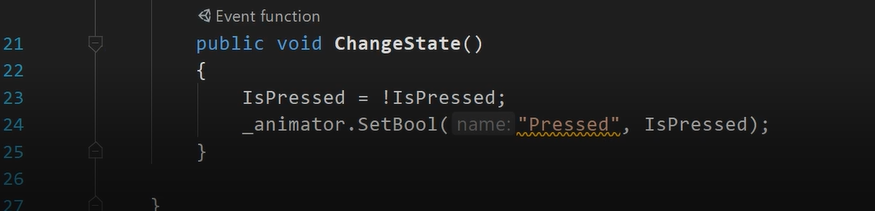
* SimpleButton class has been created for the object. It inherits VarwinObject This is the main object class, which connects the object with the platform.



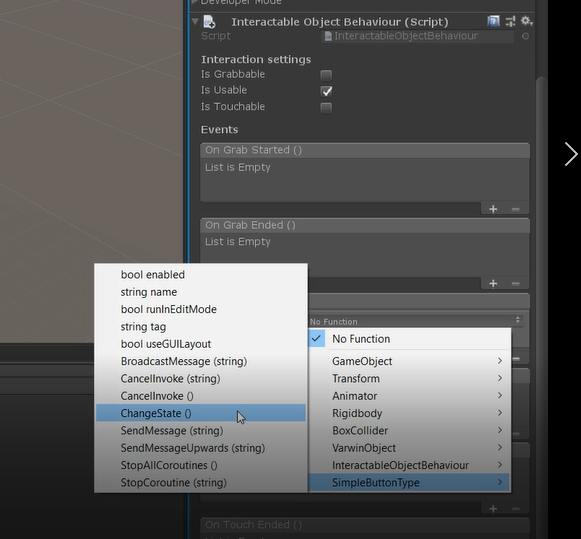
Everything is ready for writing object code.

# Example

The object has a method “ChangeState” in its main class. The method changes bool variable IsPressed.



* To connect this method with pulling the trigger on a controller (“Use”): select method “ChangeState” in Interactable Object Behavior



* You will have to create a function for Blockly, write an attribute for it (in this case, Checker) and name it



* Should this object have several methods, Blockly will group them within one block within this name.
  + E.g., if we create another state of the button, “is released”,

[Checker("pressed")]

[Locale(SystemLanguage.English, "is released")]

[Locale(SystemLanguage.Russian, "отпущена")]

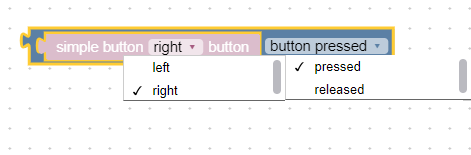
public bool IsReleased()

{

return !IsPressed;

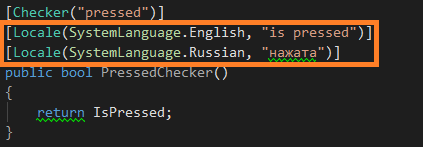
}

* + Blockly will group these methods this way:

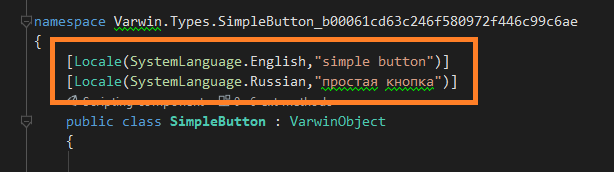


*Blocks grouping example*

* Localization

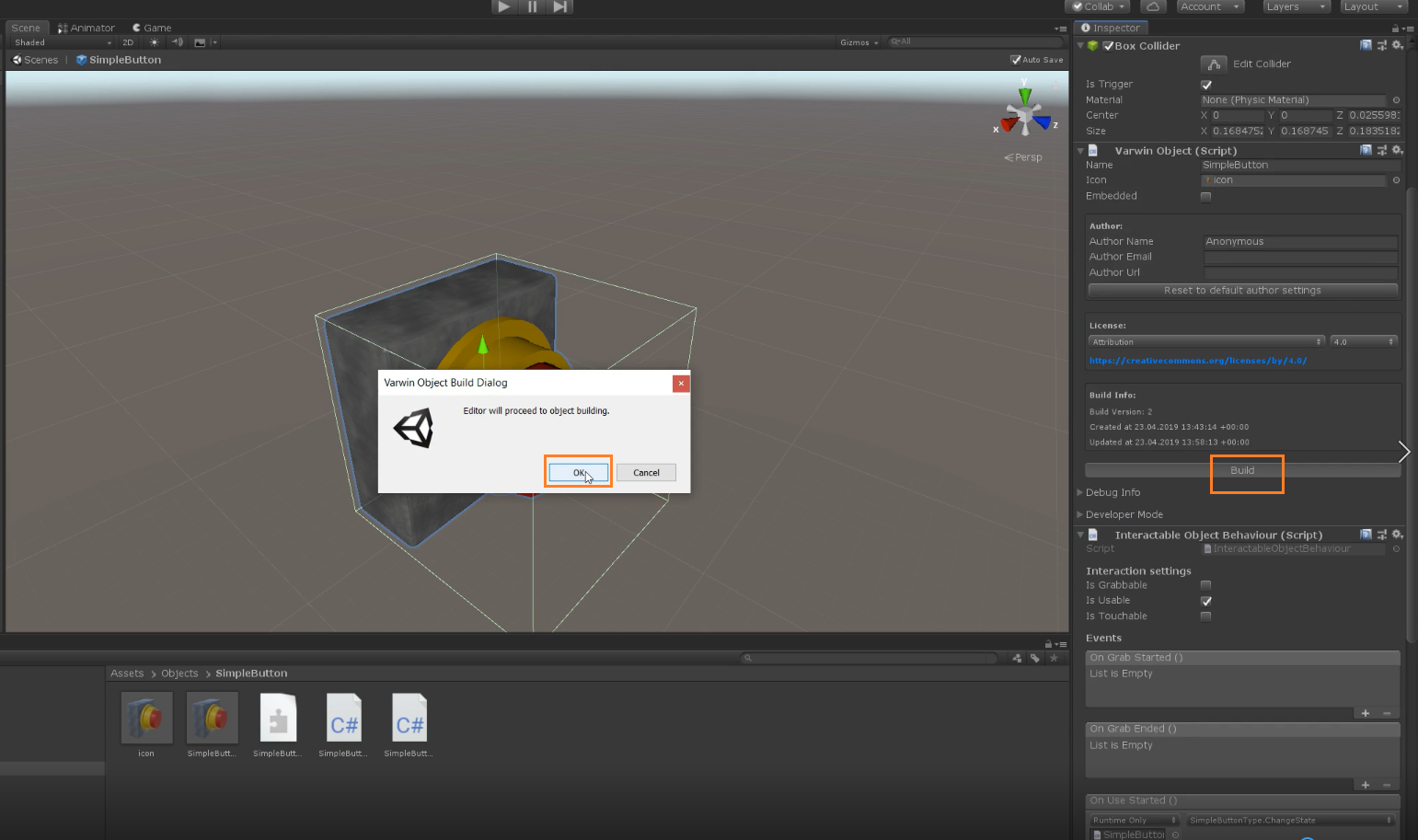


*Command localization*

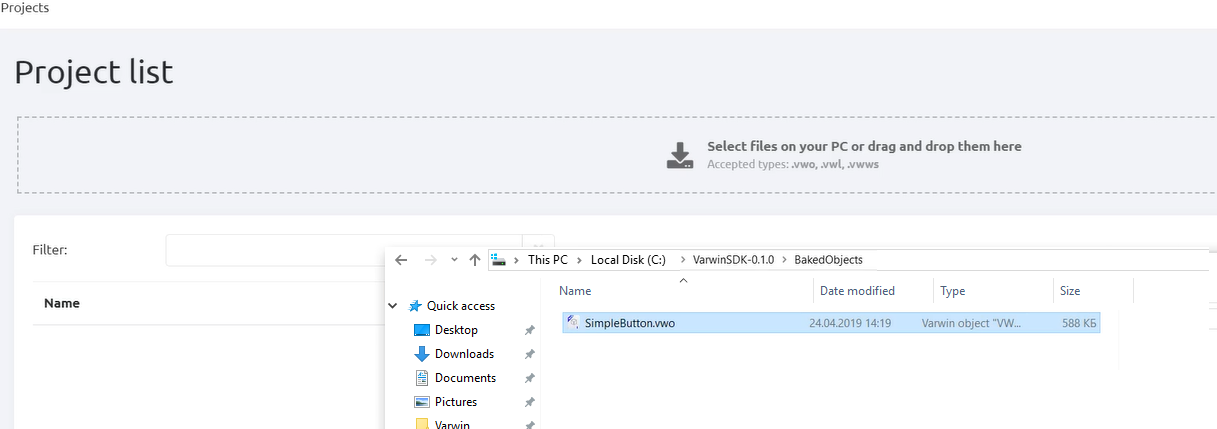


*Object name localization*

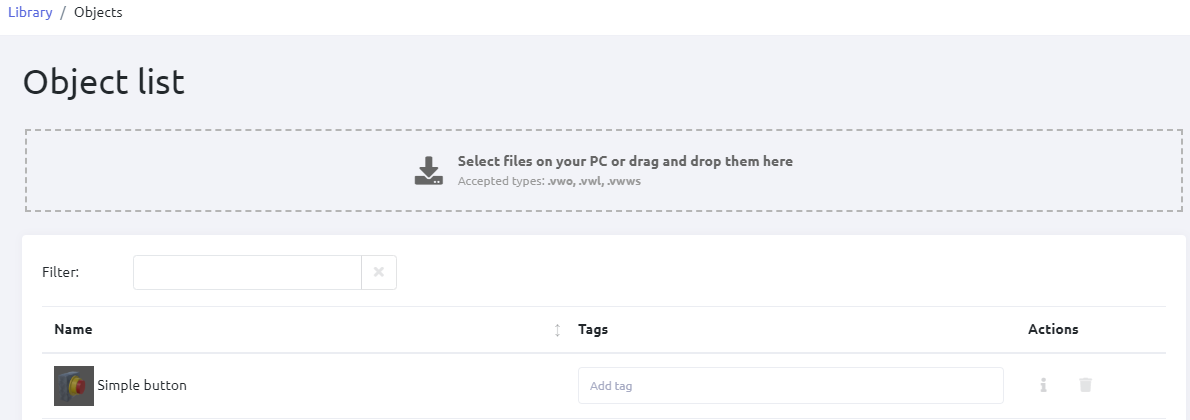
* Everything is ready for object building. Click Build.

**

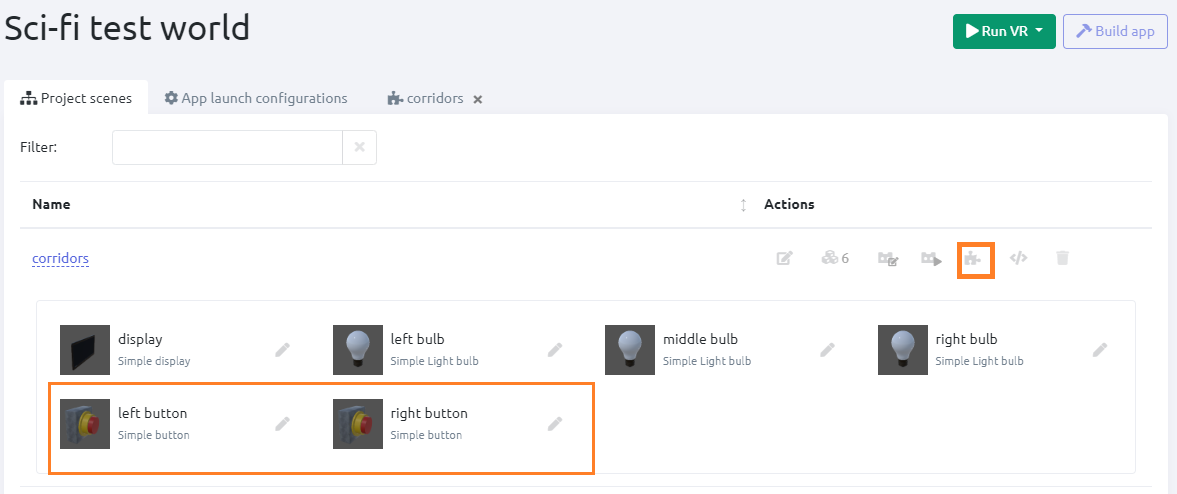
* When the object is ready, the folder containing it will open. Now you can upload the object into Varwin library.

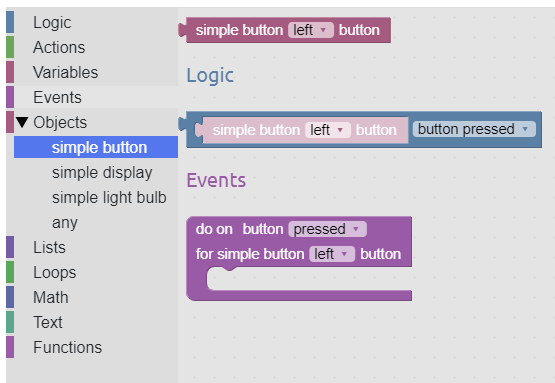


* The object appears in the library



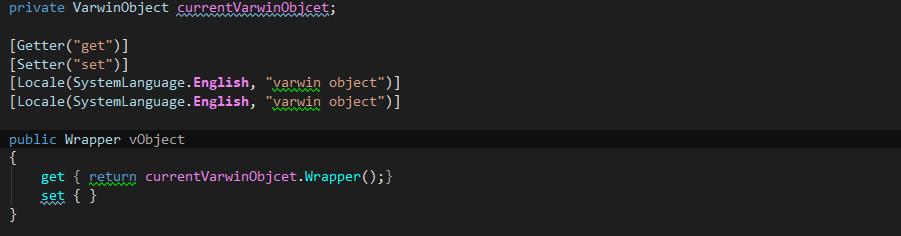
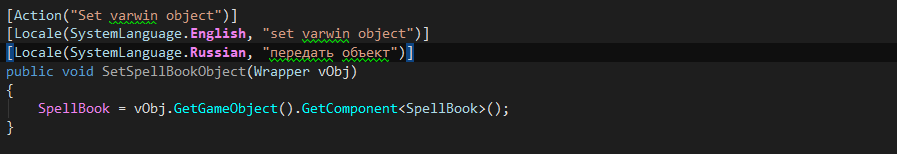
* Now you can place the object into a scene in VR. Click Save.
* Now the object appears in the scene. Open Blockly. You’ll see logic blocks that have been created for the object.





# Working with attributes

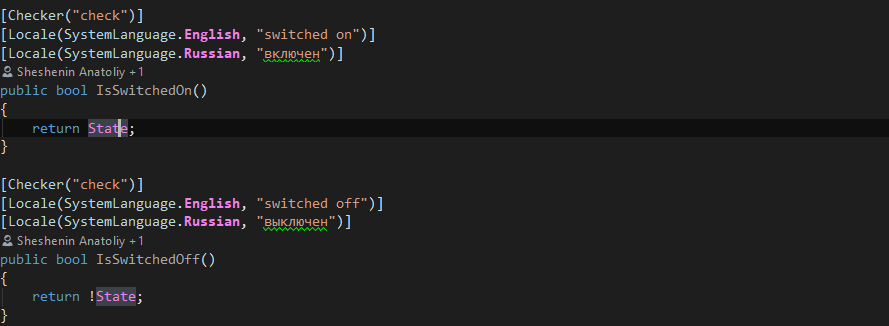
**Wrapper class** is necessary for transferring VarwinObject to Blockly. In order to obtain an object of this type for a specific VarwinObject use GetWrapper() method.

Examples

## Logic

**checker** - logic attribute. Checks true\false.

**Example**

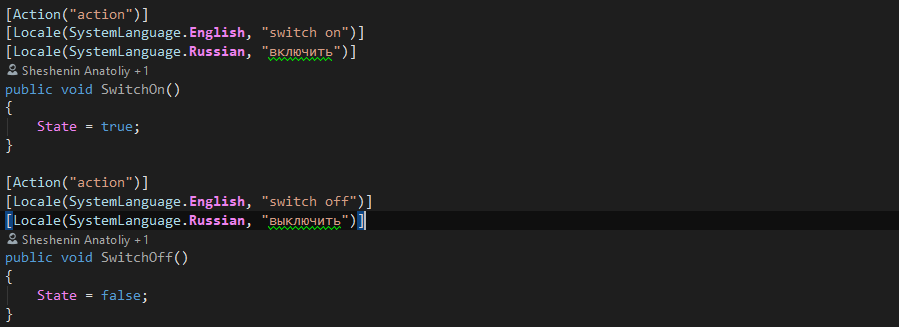


## Actions

**action** with objects. These are common methods which can receive or return the following types:

* simple types float, int, bool, string
* objects of the Wrapper type
* all the above in bulk

**Example**



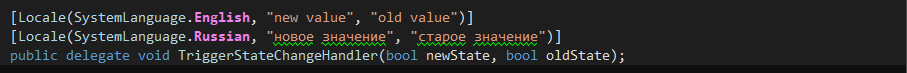
## Events

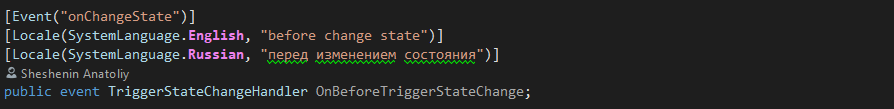
**event**.

The following types can be transferred:

* simple types float, int, bool, string
* objects of the Wrapper type
* all the above in bulk

**Example**





## Variables

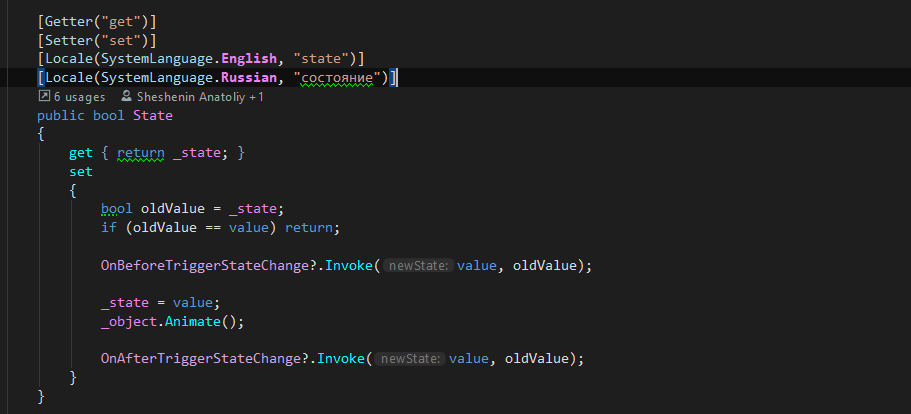
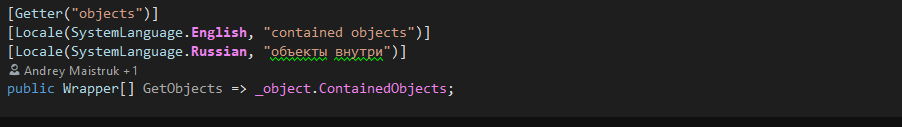
Variables can belong to the following types:

* simple types float, int, bool, string
* objects of the Wrapper type
* all the above in bulk

get - receive a state

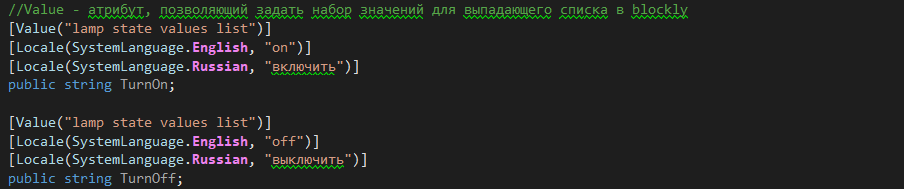
set - appoint a state

**Example**

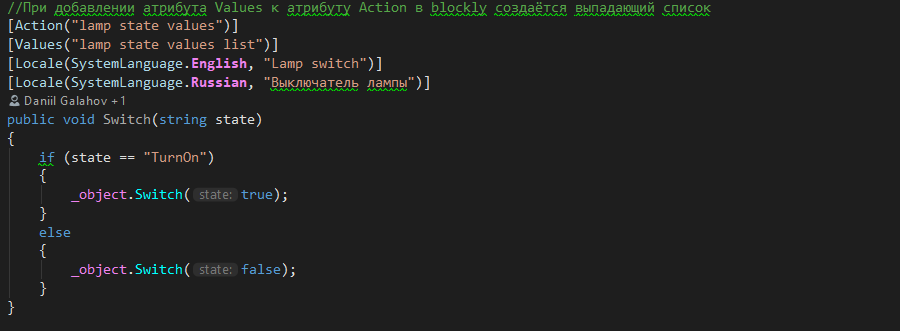


## Creating drop-down lists

Creating values for the list



Using the list



# Objects versioning

Create new versions of existing objects.

* [Objects, scene templates versioning](https://docs.google.com/document/d/1173FW8w7Yf8jQLwJXJqffxpf_xFNhDjAVosemPjr3Bo): instruction

# Additional materials

Video guides: [1](https://www.youtube.com/watch?v=-pbF88u06pI&list=PLz2Z4tRuWcpWOhy-z19iBZurvaW95FXcD&index=1), [2](https://www.youtube.com/watch?v=pTngHii9jqY&list=PLz2Z4tRuWcpWOhy-z19iBZurvaW95FXcD&index=2), [3](https://www.youtube.com/watch?v=YtiJ_O792dU&list=PLz2Z4tRuWcpWOhy-z19iBZurvaW95FXcD&index=3)