

```
}, this, { Type: IR.ADVANCED_NUMBER, Name: "Temperature", Min: 0, Max: 100 });
</syntaxhighlight> |attentions= Learn the way to a channel, a feedback channel or a token with the
help of Relations in the editor. }}
```

SubDevice.AddCondition (works without a function)

```
{{API Function/ru |description=Add a condition. A condition is not an obligatory part of rules, but it
can be used when a big number of comparisons have to be made before an action is performed. This
condition works without a function. |constructor= SubDevice.AddCondition(Name, Token,
ScriptCall, Compare, [Default], [AdvancedField]) |input =
```

Name	"Temperature drops below..."	type: String Condition name
Path	"Drivers.Nest.CurrentTemperature"	type: String Full path to a channel, a driver feedback channel, a token Subscription to keeping track of changes.
ScriptCall	false	type: Boolean Feature of working via a function. Set false in this realization
Compare		type: String Comparison operator as a string
Default	"23"	type: String Default value
AdvancedField		type: Object, Array It is used to work with complicated conditions. If AdvancedField is used set null in Default parameter. One object or an object array can be sent in AdvancedField. See types of complicated conditions here: ССЫЛКА
output=		
Object		type: Object Condition object

```
]JsSample=
```

```
var l_oDevice = module.GetDevice("MyDevice"); //Getting a device
var l_oSubDevice = module.AddSubDevice("SubDevice 1", l_oDevice); //creating
a subdevice
```