

Создаем информационный виджет

Создаем элемент “Информационный виджет”, выбираем тип HTML

Присваиваем название “Часы и статусы”, вставляем код

```

<style>
.clockwidget {
width: 202px;
height: 202px;
border-radius: 50%;
color: #fff;
text-align: center;
padding:15px 10px 15px 10px;
background-color: rgba(0, 0, 0, .4);
}
</style>
<div class="clockwidget"><br/><br/>
<div style="font-size:25px; color:#FFFFFF">%ThisComputer.Day%</div> \\\день
<div style="font-size:15px; color:#FFFFFF">%ThisComputer.Date%</div> \\\дата
<div style="font-size:25px; color:#FFFFFF">%ThisComputer.timeNow%</div>\\время сейчас
<div style="font-size:15px; color:#FFFFFF">Восход:&nbsp;,%ThisComputer.SunRiseTime%&nbsp;
Закат:&nbsp;,%ThisComputer.SunSetTime%</div> \\\Восход и закат

%System.stateDetails% </img>

%Communication.stateDetails%</img>
</div>

```

Указываем метод WidgetSwitch объекта SceneMain. Когда кликаете на метод, то информация на виджете меняется с 1 до 4-9 (любой цифры как настроите).

Как создали первое состояние, делаем клон

Нажимаем клонировать

Состояния



- Часы и статусы
- Активность
- Кто-то дома
- Входная дверь

Добавить новое состояние

Создать копию (клонировать)

State Название

Часы и статусы

Удалить

Код

```
1 <style>
2 .clockwidget {
3   width: 202px;
4   height: 202px;
5   border-radius: 50%;
6   color: #fff;
7   text-align: center;
8   padding:15px 10px 15px 10px;
9   background-color: rgba(0, 0, 0, .4);
10 }
11 </style>
12 <div class="clockwidget"><br/><br/>
13 <div style="font-size:25px; color:#FFFF
14 <div style="font-size:15px; color:#FFFF
```

После клона вы должны увидеть название сору. Переименовываем в нужное название, например Активность.

Если вдруг ошиблись и нужно удалить состояние, то поставьте галочку на “Удалить” и после удалите “Сохранить”, таким образом удалите “Состояние”.

Создаем состояние “Активность” путем клонирования “Часы и статусы”, там заменяем только код

```
<style>
.clockwidget {
  width: 202px;
  height: 202px;
  border-radius: 50%;
  color: #fff;
  text-align: center;
  padding:10px 10px 10px 10px;
  background-color: rgba(0, 0, 0, .4);
}
</style>
<div class="clockwidget"><br/><br/><br/><br/><br/>
<div style="font-size:25px; color:#ADFF2F">Активность</div>
<div style="font-size:15px; color:#FFFFFF">%ThisComputer.somebodyHomeText%</div>
</div>
```

Создаем состояние “Кто-то дома” путем клонирования “Активность”, там заменяем только код

```
<style>
.clockwidget {
  width: 202px;
  height: 202px;
  border-radius: 50%;
  color: #fff;
  text-align: center;
  padding:10px 10px 10px 10px;
  background-color: rgba(0, 0, 0, .4);
}
</style>
<div class="clockwidget"><br/>
<div style="font-size:25px; color:#ADFF2F">Кто-то дома</div>
<div style="font-size:15px; color:#FFFFFF">%admin.atHome%</div>
</div>
```

Значение взял здесь, у меня bluetooth метка если сработала, то передает значение пользователю 1, если человек ушел, то будет 0. В классе **Users** создаете своих домочадцев и привязываете к ним метку.

Создаем состояние “Входная дверь” путем клонирования “Активность”, заменяем только код

```
<style>
.clockwidget {
  width: 202px;
  height: 202px;
  border-radius: 50%;
  color: #fff;
  text-align: center;
  padding:15px 10px 15px 10px;
  background-color: rgba(0, 0, 0, .4);
}
</style>
<div class="clockwidget"><br/><br/><br/><br/><br/>
<div style="font-size:25px; color:#ADFF2F">Входная дверь</div>
<div style="font-size:20px; color:#FFFFFF">%XiaomiSmartDoor01.TextRus%</div>
</div>
```

XiaomiSmartDoor01.TextRus взял из класса SOpenable.

Создаем сцену или импортируем ее из архива, ссылка в шапке

Создаем свойства объекта, где

WidgetMode - отвечает за возврат информации через 10 секунда на информацию по умолчанию “У меня это Часы и дата”

WidgetSwitch - отвечает за смену информации при клике

TimeDay - отвечает за смену дня день\ночь

В родительском методе создаем методы **WidgetMode** и **WidgetSwitch**

 [Панель управления / classes](#)

[Начало](#) / [Сцены](#)

[Основное](#)

[Свойства](#)

[Методы](#)

[Объекты](#)

[Шаблон отображения](#)

[+](#) [Добавить новый метод](#)

Название	Описание
WidgetMode	
WidgetSwitch	

В родительском свойстве создаем **WidgetMode** и привязываем метод к свойству, это нужно для того, чтобы при изменении значения срабатывал метод.

Привязываем созданный метод **WidgetMode** к свойству **WidgetMode**

На выходе у вас получится так

В метод **WidgetMode** вставляем код

```
if (getGlobal('SceneMain.WidgetMode') >= 2) {
    clearTimeout('BackMainScene');
    $timerCode=<<<EOT
    setGlobal('SceneMain.WidgetMode', '1');
EOT;
    setTimeout('BackMainScene', $timerCode, (int)('10'));
}
```

В виде блокнот

В метод **WidgetSwitch** вставляем код

```
if (getGlobal('SceneMain.WidgetSwitch') == 1) {
    setGlobal('SceneMain.WidgetSwitch', '2');
    setGlobal('SceneMain.WidgetMode', '2');
} else if (getGlobal('SceneMain.WidgetSwitch') == 2) {
    setGlobal('SceneMain.WidgetSwitch', '3');
    setGlobal('SceneMain.WidgetMode', '3');
} else if (getGlobal('SceneMain.WidgetSwitch') == 3) {
    setGlobal('SceneMain.WidgetSwitch', '4');
    setGlobal('SceneMain.WidgetMode', '4');
} else if (getGlobal('SceneMain.WidgetSwitch') == 4) {
    setGlobal('SceneMain.WidgetSwitch', '1');
    setGlobal('SceneMain.WidgetMode', '1');
}
```

В виде блокли