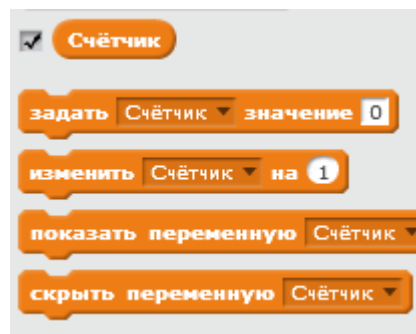


2 уровень (часть II)

Во второй части инструкции по реализации 2 уровня игры предлагается создать счётчик, осуществляющий подсчет монет, набранных Марио, а также таймер с обратным отсчётом, лимитирующий время на сбор монет Марио. Отсчёт ведётся с 20 секунды до 0. Если игрок не набирает 10 монет за 20 секунд, то игра заканчивается.

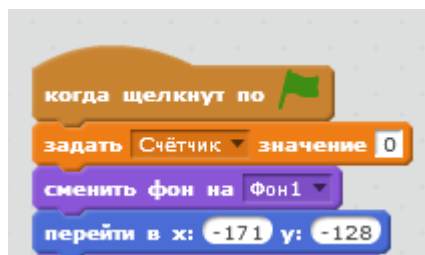
Счётчик

Счётчиком в данном случае будет переменная. Перейдите в блок *Данные* и создайте переменную под названием *Счётчик*. При создании переменной укажите *Для всех спрайтов*.



После того, как переменная создана, задайте её значение, т.е. то число, с которого будет начинаться подсчёт монет во 2 уровне. В данном случае, значение переменной *Счётчик* будет равняться 0.

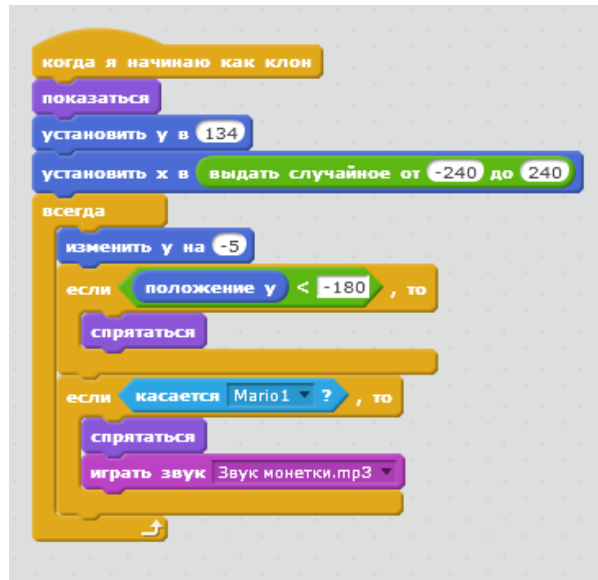
В спрайте *Марио1*, под командой *Когда щелкнут по флажку* поставьте блок *Задать значение* и укажите это значение (введите 0).



На сцене вверху слева появится монитор переменной со значением 0.

Команда *Задать счётчику значение 0* обновляет (обнуляет) переменную *Счётчик*, если нажат .

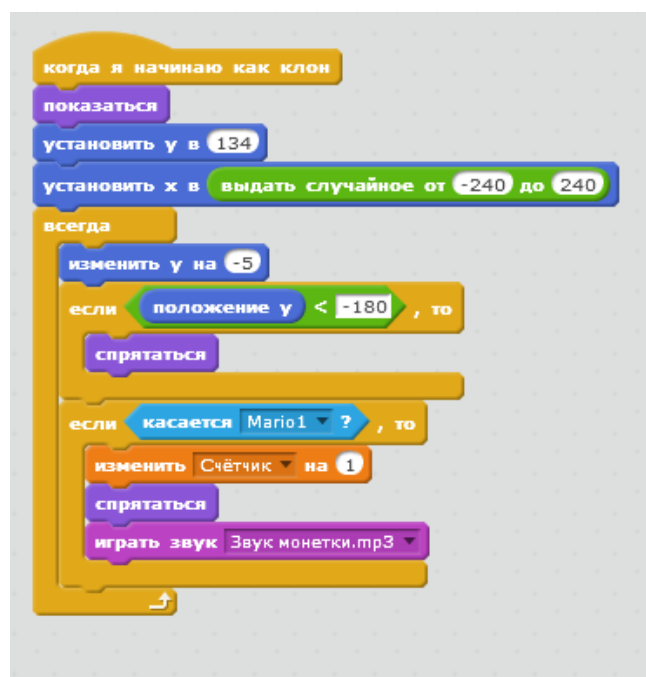
Переменная задана. Теперь необходимо условие, по которому значение переменной будет изменяться. Нам подойдёт условие, когда Марио ловит монетки. Такое условие уже задано – это скрипт спрайта *Монета*.



Когда монета касается Марио, она прячется, и проигрывается звук монетки.

Чтобы переменная *Счётчик* начала изменяться, т.е. считать монеты, дополните условие командой из блока *Данные*.

В условии *Если... то...* вставьте команду *Изменить счётчик на...* Задайте изменение переменной на *1*.

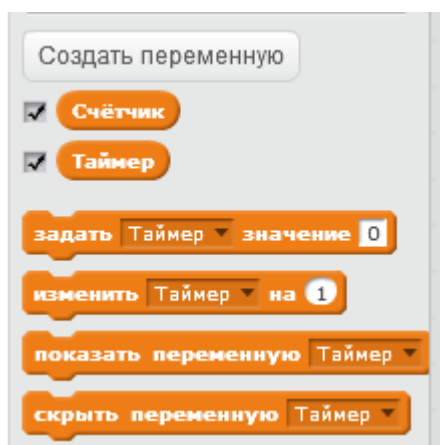


Программа считает пойманные монеты.

Таймер

В группе *Сенсоры* есть блок *Таймер*, однако у пользователей он широкую популярность не приобрёл, т.к. имеет мало команд для работы и не позволяет сделать обратный отсчёт. Однако таймер, отвечающий условиям игры, можно создать из переменной и других блоков.

Для начала создайте переменную и назовите её *Таймер*.



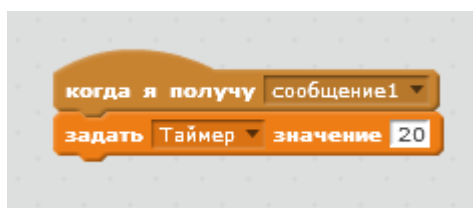
Составим скрипт для таймера. Данный скрипт мы рекомендуем разместить в спрайте *Марио1*.

В нашем случае начало 2 уровня и запуск других скриптов уровня осуществляется при помощи команд *Передать сообщение...* и *Когда я получу сообщение...*

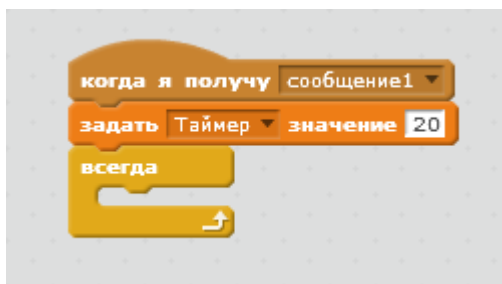
Таймер должен запускаться после перехода Марио на 2 уровень. Выделите спрайт *Марио1* и составьте скрипт для таймера. В начале скрипта используется команда *Когда я получу сообщение1*.



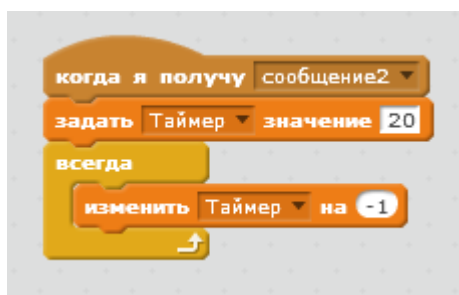
По условиям игры Марио необходимо собрать 10 монет за 20 секунд, поэтому переменной *Таймер* задайте значение *20*.



Таймер должен отсчитывать секунды, пока время не закончится (0). Команда, которая «отсчитывает 1 секунду», должна повторяться. Для повторения используйте цикл *Всегда*.



Для создания таймера с обратным отсчётом используйте команду, которая будет изменять значение переменной *Таймер* на *-1*. В цикл *Всегда* вставьте блок *Изменить таймер на -1*.



Если сейчас запустить игру, то на 2 уровне таймер фактически моментально отсчитает 20 секунд и далее на мониторе переменной будет доступно огромное число с отрицательным значением. Это происходит, потому что для команды *Изменить таймер на -1* не задано время, за которое она должна отнимать *1* от значения переменной, т.е. необходимо вставить команду, которая будет задавать время в цикле перед тем, как отнимется *1* от значения переменной. Вставьте команду *Ждать 1 секунд* перед блоком *Изменить таймер на -1*, иначе при переходе на 2 уровень от переменной сразу отнимется 1, и таймер для игрока начнется не с 20, а с 19.

