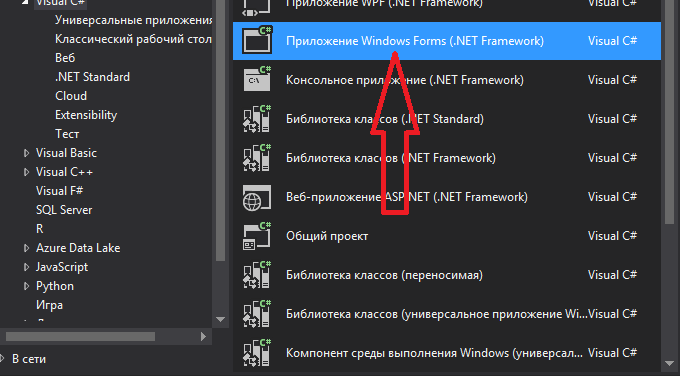
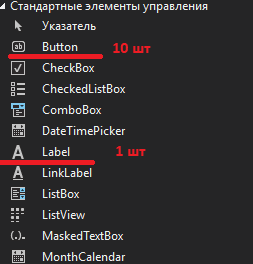
Крестики-нолики

1) Создаем Windows Form

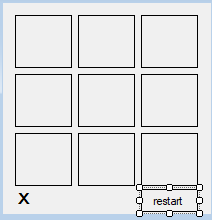


2) Создаем 10 кнопок и 1 элемент label.



3) 10 кнопка будет отвечать за перезапуск игры. В свойствах кнопки изменяем значение text с button10 на restart. У остальных кнопок поля свойства text установить пустыми. Label будет выводить чей сейчас ход крестиков или ноликов.

4) Расположить элементы примерно так:



5) Создать событие для первой кнопки

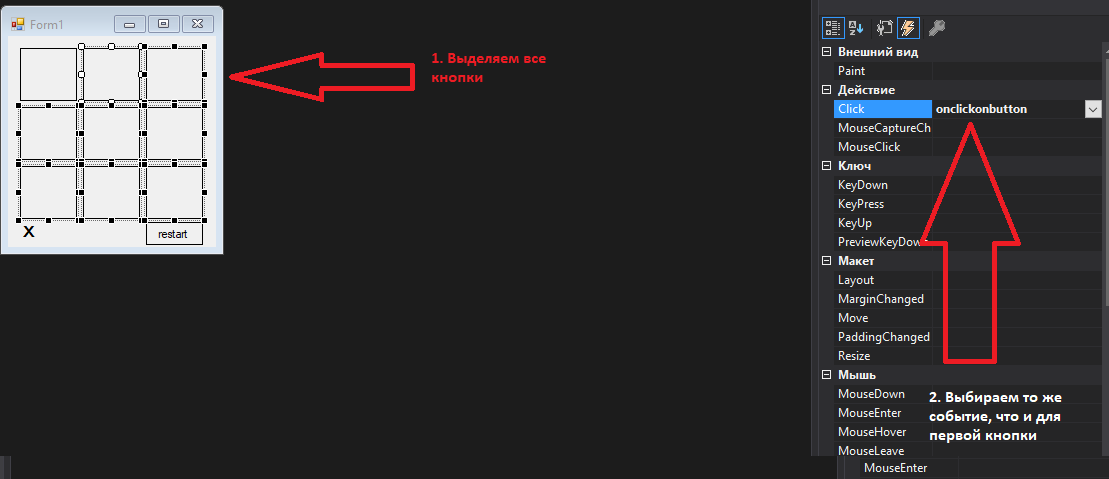


Для всех кнопок можно использовать одно событие.

В будущем при написании кода можно через условие проверять какая кнопка была нажата через параметр метода sender.



6) Выделим через shift все кнопки КРОМЕ кнопки restart.

Зайдем в события и выберем то же событие, что и для первой кнопки. 

Теперь при нажатии любой кнопки будет вызываться один и тот же метод.

7) Начнем писать код

Создадим переменную в классе turn типа **string.** В ней будет хранится значение, отвечающее за ход. Если переменная будет равна “крестики”, то будут ходить крестики, а если “нолики”, то соответственно нолики.

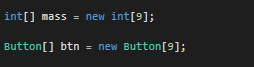


Изначально мы укажем пустую строку т.к. при начале игры будет случайным образом определяться кто ходит первым.

Создадим класс Random, с помощью него мы будем определять, кто ходит первым.

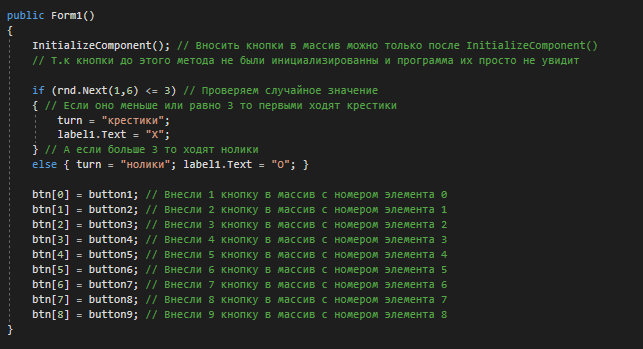


Создадим два массива в классе. Первый массив для переменной. Он будет отвечать за значение на каждой кнопке. 1 – на кнопке крестик, 2 – на кнопке нолик. Второй массив будет хранить в себе все кнопки, чтобы потом можно было сослаться к ним.



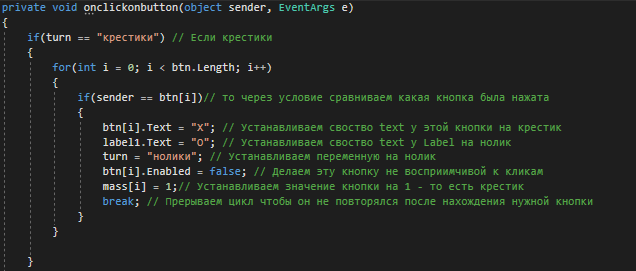
В методе события используется параметр **sender**. В нем указывается объект, по которому мы нажали. Чтобы не создавать много условий по типу if(sender == button1) и т.д. мы создали массив btn с объектами и через цикл будем проверять все кнопки сразу.

Теперь при инициализации формы мы внесем в массив btn все кнопки и укажем кто будет ходить первым.

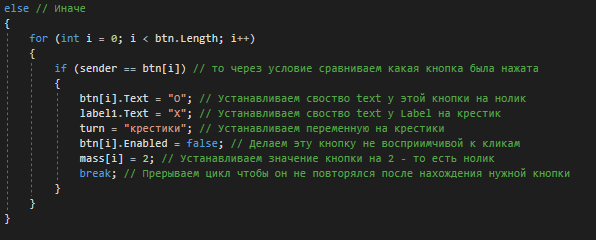


Теперь, когда мы написали код в инициализации формы переходим к методу onclickonbutton (object sender, EventArgs e);

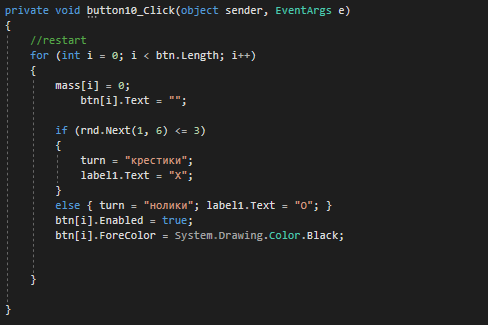
Создаем условие если ходят крестики, то при нажатии кнопки проверяется какая кнопка нажата и в кнопке появляется крестик



Так же делаем с условием else. Только для ноликов



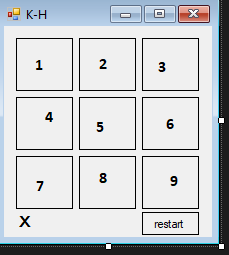
Создадим отдельное событие клика для кнопки рестарта.



Впишем туда этот код. Он сбрасывает через цикл переменную mass отвечающую за значение в каждой кнопке. Случайно перетасовывает следующего игрока, который начнет первый ход. Сделает доступными для клика все кнопки и установит цвет рамки на черный.

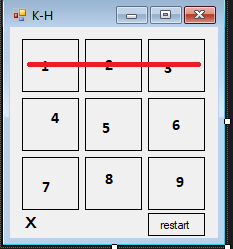
Последнее что осталось сделать это проверить через условия. Совпадают ли значения кнопок, например, по горизонтали в 1 строке. И если совпадают закончить игру.

Кнопки в массиве расположены по порядку.

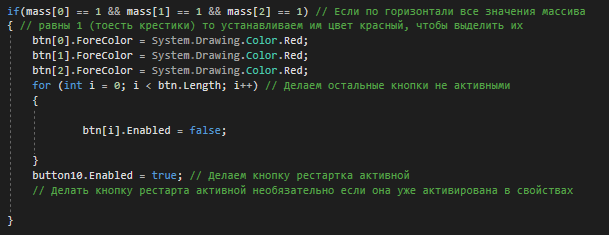


**1 кнопка – это нулевой элемент в массиве btn и mass, 2 кнопка это 1 элемент и т.д.**

Проверяем через условие

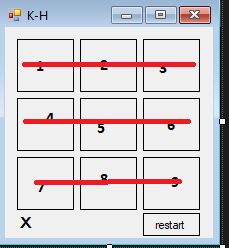


По горизонтали 1 строку

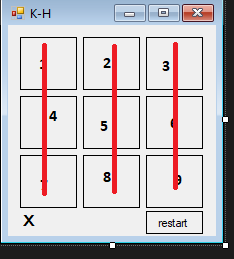


И так же со всеми возможными вариантами кол-во вариантов (всего их 8 – по 8 условий на крестик и на нолик):

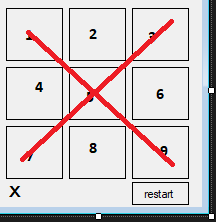
3 по горизонтали – 1 строка 2 строка и 3 строка



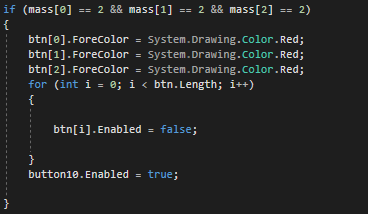
3 по вертикали – 1 строка 2 строка и 3 строка



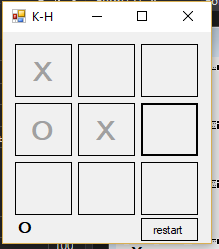
2 по диагонали 1 кнопка 5 кнопка и 6 кнопка и 7 кнопка 5 кнопка и 3 кнопка

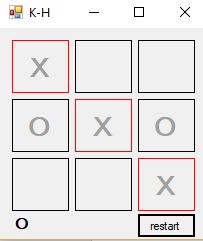


Пример кода с проверкой ноликов по горизонтали 1 строка.



После написания всех условий должно получиться примерно так:





**УЖАС**

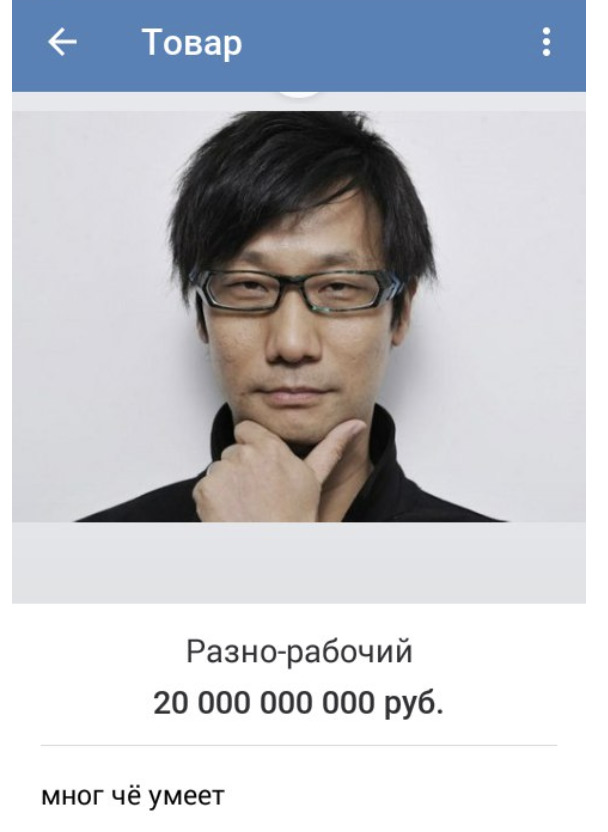
Омг

**АГА! 42**

**Глубина одобряет**

**Опять пасхалки свои делает конченые**

**Made by** [**Grecha**](https://vk.com/gdemoyagrecha2)

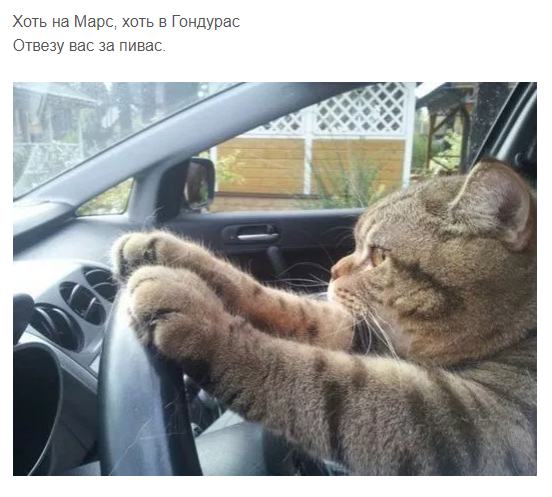






**Внимание!**

**Спасибо за внимание!**



**О это что какой то музон с опенингов поехал?**

**[](https://www.youtube.com/watch?v=huVWysfdj6c)**

**[](https://www.youtube.com/watch?v=VRojmdPSfIc)**











**Все а теперь иди работать**

