

Семинар #1: События библиотеки SFML. Домашнее задание.

Все задачи нужно решить обязательно с использованием только одной сторонней библиотеки – SFML.

Задача 1. Нажатие на клавишу

Напишите программу, которая рисует на экране круг зелёного цвета. При нажатии клавиши пробел цвет круга должен меняться на красный. Круг должен оставаться красным, пока клавиша пробел зажата. После отпускания клавиши пробел круг должен снова становиться зелёным. Решите эту задачу двумя способами:

1. Используя статические методы класса `sf::Keyboard`.
2. Используя события `sf::Event::KeyPressed` и `sf::Event::KeyReleased`.

Задача 2. Столкновение курсора с прямоугольником

Напишите программу, которая бы рисовала прямоугольник, цвет которого изменялся бы при наведении на него мышью. Если курсор мыши находится вне прямоугольника, то он должен рисоваться зелёным цветом. Если курсор мыши находится внутри прямоугольника, то красным. Решите эту задачу двумя способами:

1. Используя статические методы класса `sf::Mouse`.
2. Используя событие `sf::Event::MouseMoved`.

Программа должна работать корректно при изменении размеров окна. Для обнаружения столкновения фигуры прямоугольника с курсором используйте метод `getGlobalBounds`, чтобы получить объект типа `sf::FloatRect`, а затем метод `contains`.

Задача 3. Создание кругов

Напишите программу, в которой можно создавать круги по нажатию левой кнопки мыши. При нажатии ЛКМ в определённом месте окна должен появляться круг белого цвета, причём центр круга должен совпадать с положением курсора мыши. При нажатии клавиши пробел цвет всех кругов должен изменяться на случайный. Программа должна корректно работать при изменении размеров окна.

Для получения случайного цвета можно использовать следующую функцию:

```
#include <random>
sf::Color getRandomColor()
{
    static std::random_device rd;
    static std::mt19937 gen(rd());
    std::uniform_int_distribution<uint8_t> d(0, 255);
    return sf::Color{d(gen), d(gen), d(gen)};
}
```

Задача 4. Тумблер

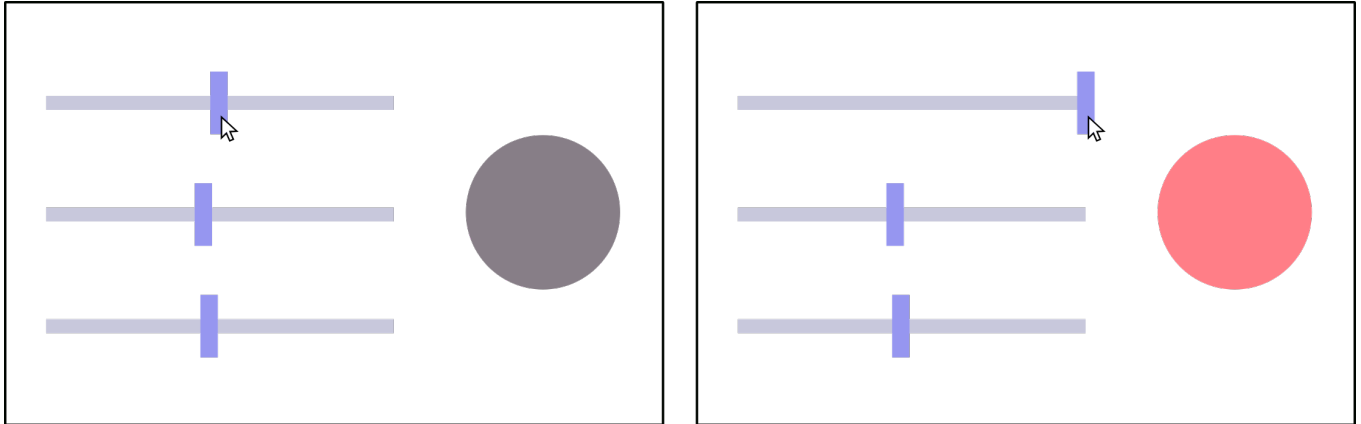
Создайте класс `Toggle`, который бы реализовывал элемент интерфейса тумблер (переключатель).



После того, как создадите класс, напишите программу, в которой бы создавалось 10 тумблеров. Все тумблеры должны работать. Программа должна корректно работать при изменении размеров окна.

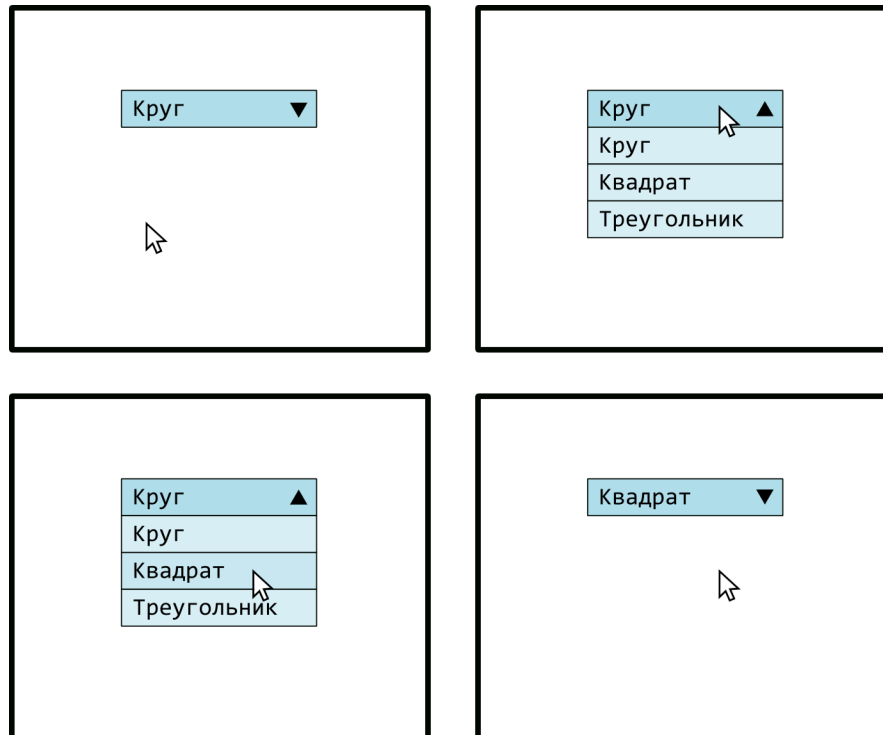
Задача 5. Три слайдера

Напишите программу, которая создаёт круг и три слайдера. Цвет круга должен меняться при изменении положения ползунков слайдеров. Каждый слайдер должен отвечать за свою компоненту цвета круга (красную, зелёную или синюю). Используйте класс `Slider` из `classroom_tasks/code/5gui/2slider`.



Задача 6. Выпадающий список

Создайте класс `DropList`, который бы реализовывал элемент интерфейса выпадающий список без полосы прокрутки.



После создания класса напишите программу, которая реализует выпадающий список с тремя вариантами: `Круг`, `Квадрат` и `Треугольник`. В зависимости от выбранного в списке варианта, на экране должна отображаться соответствующая фигура: круг, квадрат или треугольник. Нарисовать треугольник можно, используя класс круга:

```
sf::CircleShape triangle{50, 3};
```

Тут мы используем второй параметр конструктора `sf::CircleShape`, который задаёт количество сторон рисуемого правильного многоугольника. По умолчанию значение этого параметра равно 30, поэтому при рисовании кругов в SFML на самом деле рисуются правильные 30-ти угольники.

Необязательные задачи

Задача 1. Столкновение курсора с кругом

Напишите программу, которая бы рисовала круг, цвет которого изменялся бы при наведении на него мышью. Если курсор мыши находится вне круга, то круг должен рисоваться зелёным цветом. Если курсор мыши находится внутри круга, то круг должен рисоваться красным цветом.

Задача 2. Select Move Delete

В папке `select_move_delete/` содержится заготовка исходного кода для этого задания. В этой программе есть несколько объектов(кругов), которые можно выделять. Выделение происходит по нажатию левой клавиши мыши. Зажав клавишу `Ctrl` можно выделить несколько объектов. Также в программе реализован прямоугольник выделения (но он пока не выбирает объекты).левой кнопкой мыши с зажатой клавишей левый `Alt` можно создать круг случайного размера. Добавьте следующие возможности в программу:

- Задание случайного цвета. При нажатии клавиши пробел цвет всех выделенных шаров должен меняться на случайный. Для этого понадобится добавить поле `color` в класс `Ball`.
- Выделение объектов с помощью прямоугольника выделения. Прямоугольник выделения должен рисоваться только если нажатие мыши произошло вне кругов. Все объекты, полностью находящиеся внутри прямоугольника выделения, на момент отпускания левой кнопки мыши должны выделяться. Объекты должны выделяться во время изменения прямоугольника, а не только при отпускании кнопки мыши.
- Перемещение всех выделенных объектов при зажатии левой клавиши мыши и её движении. Перемещаться должны все выделенные объекты параллельно (также как перемещаются несколько выделенных значков на рабочем столе). Прямоугольник выделения при этом рисоваться не должен.
- При нажатии клавиши `Delete`, все выделенные объекты должны удаляться. Чтобы удалить элемент из `std::list` используйте итераторы и метод `erase`. При удалении элемента `std::list` нужно внимательно следить за тем, чтобы не испортить итераторы.
- В папке `code/5gui/5context_menu` содержится реализация контекстного меню – класс `ContextMenu`. Используйте этот класс чтобы добавить контекстное меню в программу. Оно должно открываться при нажатии правой кнопки мыши. Добавьте следующие варианты в меню:
 - `Delete` - при нажатии на выделенный объект правой кнопкой мыши и выборе этого варианта все выделенные объекты должны удаляться.
 - `Create` - при нажатии на любое место и выборе этого варианта должен создаваться новый случайный круг.
 - `Random Color` – все выделенные объекты должны окрашиваться в случайный цвет.
 - `Increase` – все выделенные объекты должны увеличиваться на 25 процентов.
 - `Decrease` – все выделенные объекты должны уменьшаться на 25 процентов.
- **Copy Paste Cut** Добавьте возможность копирования, вставки и вырезания объектов. Вызов этих операций должен происходить в контекстном меню или с помощью комбинаций клавиш `Ctrl-C`, `Ctrl-V` и `Ctrl-X`.