

# Семинар #10: Библиотека SFML. Домашнее задание.

## Задача 1. 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. Этот момент был более рассмотрен в семинаре #4.
- В папке `../classroom_tasks_solutions/context_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 выделенные 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`.

## Задача 2. Карточки

Создайте 20 прямоугольников случайного цвета, но одинакового размера, так чтобы все они были перетаскиваемыми. Если карточки перекрываются, то перетаскиваться должна верхняя карточка. Используйте класс `Draggable` из файла `draggable.cpp`.

## Задача 3. Ползунок:

Создайте класс `Slider`, который будет описывать элемент интерфейса ползунок. Графическое оформление на ваше усмотрение, в качестве примера:



Ползунок должен работать также как и обычный ползунок в ОС Windows или Linux. При нажатии на сам ползунок и зажатии кнопки, он переходит в состояние перемещение и остаётся в нём до момента отпущения клавиши мыши (даже если курсор вышел далеко за пределы полоски). Другой способ регулирования положения ползунка - это нажатие на саму полосу. В этом случае ползунок сразу перемещается в выбранное место. При изменении положения ползунка должен меняться текст, показывающий числовое значение. Минимальное и максимальное значение ползунка должно задаваться в конструкторе и храниться в приватных переменных.

- Создайте круг и 1 ползунок. При изменении положения ползунка должен меняться радиус этого круга.
- Создайте ещё 3 ползунка. При изменении положения этих ползунков должен меняться цвет круга (RGB компоненты).