

### МИНОБРНАУКИ РОССИИ

### федеральное государственное бюджетное образовательное учреждение высшего образования

### «Московский государственный технологический университет «СТАНКИН» (ФГБОУ ВО «МГТУ «СТАНКИН»)

### Институт информационных систем и технологий

### Кафедра инженерной графики

**Отчет по выполнению лабораторной работы №1**

## по дисциплине «Геометрическое моделирование и компьютерная графика»

## Направление подготовки: 09.03.03 «Прикладная информатика»

## Профиль: «Управление данными»

## Руководитель: Бейреш А. М.

Оценка Подпись Дата

## Студент:

## Зарубежнов Б.П.

## Группа ИДБ-20-11

## 

Подпись Дата

## Москва, 2023

**СОДЕРЖАНИЕ**

[Задание 3](#_TOC_250002)

[Код 3](#_TOC_250001)

[Результаты работы программы 6](#_TOC_250000)

# Задание

## №1. Сделать изменение фона по нажатию на разные кнопки с использованием библиотек SFML и ImGui.

# Код

#include <iostream>

#include "SFML/Graphics/CircleShape.hpp"

#include "SFML/Graphics/Image.hpp"

#include "SFML/Graphics/RenderTexture.hpp"

#include "SFML/Graphics/RenderWindow.hpp"

#include "SFML/Graphics/Sprite.hpp"

#include "SFML/Graphics/Text.hpp"

#include "SFML/Graphics/Texture.hpp"

#include "SFML/System/Clock.hpp"

#include "SFML/Window/Event.hpp"

#include "imgui-SFML.h"

#include "imgui.h"

// Структура для хранения информации о каждой кнопке

struct ColorButton

{

std::string name;

sf::Color color;

};

// Массив цветов для 10 кнопок

ColorButton colorButtons[] = {{"Red", sf::Color(255, 0, 0)},

{"Green", sf::Color(0, 255, 0)},

{"Blue", sf::Color(0, 0, 255)},

{"Yellow", sf::Color(255, 255, 0)},

{"Purple", sf::Color(255, 0, 255)},

{"Cyan", sf::Color(0, 255, 255)},

{"Dark Red", sf::Color(128, 0, 0)},

{"Dark Green", sf::Color(0, 128, 0)},

{"Dark Blue", sf::Color(0, 0, 128)},

{"Dark Yellow", sf::Color(128, 128, 0)}};

sf::Color windowColor = sf::Color::White;

void HandleUserInput(sf::RenderWindow &window, const sf::Event &event)

{

switch (event.type)

{

case sf::Event::Closed:

window.close();

break;

default:

break;

}

}

void Update(sf::RenderWindow &window, const sf::Time &deltaClock)

{

// Make some time-dependent updates, like: physics, gameplay logic, animations, etc.

}

void Render(sf::RenderWindow &window)

{

// Draw some sfml/opengl items

}

void RenderGui(sf::RenderWindow &window)

{

ImGui::Begin("Default window");

for (const auto &button : colorButtons)

{

if (ImGui::Button(button.name.c\_str()))

{

windowColor = button.color;

}

}

ImGui::End();

}

int main()

{

sf::RenderWindow window(sf::VideoMode(800, 800), "Geometry modeling 1");

window.setFramerateLimit(60);

if (!ImGui::SFML::Init(window))

{

std::cout << "ImGui initialization failed\n";

return -1;

}

sf::Clock deltaClock;

while (window.isOpen())

{

sf::Event event;

while (window.pollEvent(event))

{

ImGui::SFML::ProcessEvent(window, event);

HandleUserInput(window, event);

}

sf::Time deltaTime = deltaClock.restart();

ImGui::SFML::Update(window, deltaTime);

Update(window, deltaTime);

window.clear(windowColor);

RenderGui(window);

Render(window);

ImGui::SFML::Render(window);

window.display();

}

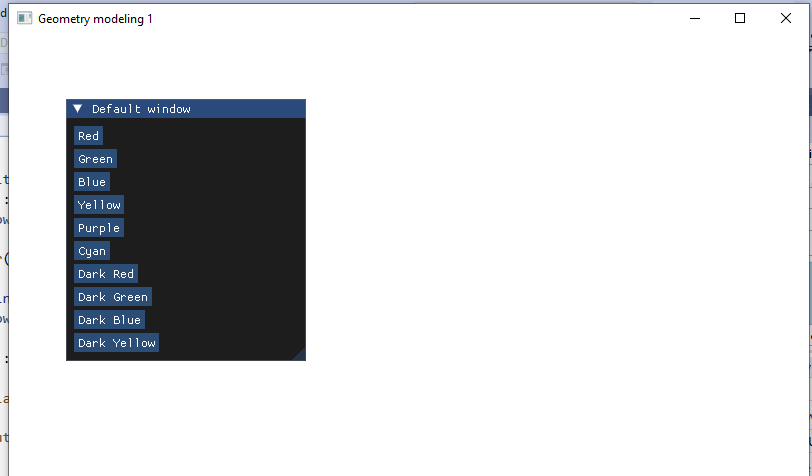
ImGui::SFML::Shutdown();

return 0;

}

# Результаты работы программы

## После запуска программы получим следующее окно:



## Если поочередно нажимать на кнопки, можно наблюдать изменение цвета фона окна:



