МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ

“БРЕСТСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ”

КАФЕДРА ИИТ

ОТЧЁТ

по лабораторной работе №7

**«Создание интерфейса простых приложений»**

Выполнил:

студент 2 курса

группы ПО-9

Мисиюк Алексей Сергеевич

Проверил:

Козик И. Д.

Задание.

1. Создать интерфейс приложения задачи предложенного варианта. Используя лейауты, удобно расположить элементы приложения. Представить решение задачи и готовый исполняемый файл приложения.

2. Добавить созданному приложению новую функциональность на основе событий. Создать события обработки «горячих клавиш» дублирующих функции кнопок, а при наведении курсора мыши на ячейку таблицы в компоненте ToolTip должно отображаться ее текущее значение.

**Вариант 3 (15 - 12)**

Напишите программу, способную определять следующие параметры массива 7х7: максимальное и минимальное значение, их местоположение, сумму значений элементов массива и среднее арифметическое значение. Создайте интерфейс программы: в таблице 7х7 числа получить случайным образом по сигналу от предусмотренной кнопки; кнопки «Max» и «Min» должны выполнять поиск соответствующих значений и их местоположение, которое должно быть подсвечено цветом в таблице; кнопки «Сумма» и «СА» должны высчитывать указанные значения и выводить результат в поля надписи.

**Код программы**

**main.cpp**

#include "mainwindow.h"

#include <QApplication>

int main(int argc, char \*argv[])

{

QApplication a(argc, argv);

MainWindow w;

w.show();

return a.exec();

}

**mainwindow.h**

#include <QMainWindow>

#include <QTableWidgetItem>

QT\_BEGIN\_NAMESPACE

namespace Ui { class MainWindow; }

QT\_END\_NAMESPACE

class MainWindow : public QMainWindow

{

Q\_OBJECT

public:

MainWindow(QWidget \*parent = nullptr);

~MainWindow();

private slots:

void on\_randomButton\_clicked();

void on\_maxButton\_clicked();

void on\_minButton\_clicked();

void on\_sumButton\_clicked();

void on\_averageButton\_clicked();

void on\_tableWidget\_cellEntered(int row, int column);

private:

Ui::MainWindow \*ui;

void highlightCell(int row, int column);

};

**mainwindow.cpp**

#include "mainwindow.h"

#include "./ui\_mainwindow.h"

#include <QRandomGenerator>

#include <QToolTip>

MainWindow::MainWindow(QWidget \*parent)

: QMainWindow(parent)

, ui(new Ui::MainWindow)

{

ui->setupUi(this);

connect(ui->tableWidget,&QTableWidget::cellEntered, this, &MainWindow::on\_tableWidget\_cellEntered);

ui->randomButton->click();

}

MainWindow::~MainWindow()

{

delete ui;

}

void MainWindow::on\_randomButton\_clicked()

{

for (int row = 0; row < ui->tableWidget->rowCount(); ++row) {

for (int col = 0; col < ui->tableWidget->columnCount(); ++col) {

int randomNumber = QRandomGenerator::global()->bounded(100);

QTableWidgetItem \*item = new QTableWidgetItem(QString::number(randomNumber));

ui->tableWidget->setItem(row, col, item);

}

}

ui->resultLabel->clear();

}

void MainWindow::on\_maxButton\_clicked()

{

int max = 0;

int maxRow = -1;

int maxCol = -1;

for (int row = 0; row < ui->tableWidget->rowCount(); ++row) {

for (int col = 0; col < ui->tableWidget->columnCount(); ++col) {

QTableWidgetItem \*item = ui->tableWidget->item(row, col);

int value = item->text().toInt();

if (value > max) {

max = value;

maxRow = row;

maxCol = col;

}

}

}

highlightCell(maxRow, maxCol);

ui->resultLabel->setText(QString("Max: %1").arg(max));

}

void MainWindow::on\_minButton\_clicked()

{

int min = 100;

int minRow = -1;

int minCol = -1;

for (int row = 0; row < ui->tableWidget->rowCount(); ++row) {

for (int col = 0; col < ui->tableWidget->columnCount(); ++col) {

QTableWidgetItem \*item = ui->tableWidget->item(row, col);

int value = item->text().toInt();

if (value < min) {

min = value;

minRow = row;

minCol = col;

}

}

}

highlightCell(minRow, minCol);

ui->resultLabel->setText(QString("Min: %1").arg(min));

}

void MainWindow::on\_sumButton\_clicked()

{

int sum = 0;

for (int row = 0; row < ui->tableWidget->rowCount(); ++row) {

for (int col = 0; col < ui->tableWidget->columnCount(); ++col) {

QTableWidgetItem \*item = ui->tableWidget->item(row, col);

int value = item->text().toInt();

sum += value;

}

}

ui->resultLabel->setText(QString("Sum: %1").arg(sum));

}

void MainWindow::on\_averageButton\_clicked()

{

int sum = 0;

int count = ui->tableWidget->rowCount() \* ui->tableWidget->columnCount();

for (int row = 0; row < ui->tableWidget->rowCount(); ++row) {

for (int col = 0; col < ui->tableWidget->columnCount(); ++col) {

QTableWidgetItem \*item = ui->tableWidget->item(row, col);

int value = item->text().toInt();

sum += value;

}

}

double average = static\_cast<double>(sum) / count;

ui->resultLabel->setText(QString("Average: %1").arg(average));

}

void MainWindow::on\_tableWidget\_cellEntered(int row, int column)

{

QTableWidgetItem \*item = ui->tableWidget->item(row, column);

if (item) {

QToolTip::showText(ui->tableWidget->mapToGlobal(QPoint()), item->text());

}

}

void MainWindow::highlightCell(int row, int column)

{

for (int i = 0; i < ui->tableWidget->rowCount(); ++i) {

for (int j = 0; j < ui->tableWidget->columnCount(); ++j) {

QTableWidgetItem \*item = ui->tableWidget->item(i, j);

if (item) {

item->setBackground(Qt::white);

item->setForeground(Qt::black);

}

}

}

if (row >= 0 && column >= 0) {

QTableWidgetItem \*item = ui->tableWidget->item(row, column);

if (item) {

item->setBackground(Qt::blue);

item->setForeground(Qt::white);

}

}

}

**Результат**