

Лабораторная работа № 11

Тема: Класс HttpClient. Деплой приложений.

Цель: закрепить теоретические знания и применить их на практике.

Задание:

Вариант №1.

Задание 1: Отправка простого GET-запроса.

Создайте простое приложение Windows Forms с одной кнопкой и двумя текстовыми полями. При нажатии на кнопку программа должна прочитать адрес сайта из первого текстового поля (например: <https://github.com/>, <https://metanit.com/>), отправить GET-запрос к веб-сайту и вывести содержимое ответа (Status Code) в текстовое поле.

Требования:

- Использовать метод GetAsync класса HttpClient.
- Вывести результат запроса в TextBox.
- Использовать асинхронные методы (async/await) для выполнения запроса, чтобы не блокировать интерфейс.
- Добавьте обработку возможных исключений и уведомления пользователя о неудачной попытке подключения.

Добавьте на форму еще две кнопки (Headers, Content). Реализуйте соответствующий функционал кнопок.

Задание 2. Загрузка изображения с URL и отображение в PictureBox.

Создайте форму с PictureBox и Button.

По нажатию на кнопку используйте HttpClient для загрузки случайного изображения с URL (например, <https://picsum.photos/500/500>).

Отобразите загруженное изображение в PictureBox.

***Задание 3.** Работа с API.

Создайте приложение Windows Forms с одной кнопкой и текстовым полем, которое получает актуальный курс эфира (ETH) в USD с использованием API cryptocompare.com и библиотеки System.Text.Json.

Задание 4. Деплой приложения.

- Создайте .exe файл со всеми зависимостями.
- создайте установщик с использованием Microsoft Visual Studio Installer Projects

Вариант №2.

Задание 1: Отправка простого GET-запроса.

Создайте простое приложение Windows Forms с одной кнопкой и двумя текстовыми полями. При нажатии на кнопку программа должна прочитать адрес сайта из первого текстового поля (например: <https://yandex.kz/>, <https://stackoverflow.com/>), отправить GET-запрос к веб-сайту и вывести содержимое ответа в текстовое поле.

Требования:

- Использовать метод GetAsync класса HttpClient.
- Вывести результат запроса в TextBox.
- Использовать асинхронные методы (async/await) для выполнения запроса, чтобы не блокировать интерфейс.
- Добавьте обработку возможных исключений и уведомления пользователя о неудачной попытке подключения.

Добавьте на форму еще две кнопки (StatusCode, Version). Реализуйте соответствующий функционал кнопок.

Задание 2. Загрузка изображения с URL и отображение в PictureBox.

Создайте форму с PictureBox и Button.

По нажатию на кнопку используйте HttpClient для загрузки случайного изображения с URL (например, <https://loremflickr.com/200/300/cat>).

Отобразите загруженное изображение в PictureBox.

***Задание 3.** Работа с API.

Создайте приложение Windows Forms с одной кнопкой и текстовым полем, которое получает актуальный курс Ripple (XRP) в USD с использованием API cryptocompare.com и библиотеки System.Text.Json.

Задание 4. Деплой приложения.

- Создайте .exe файл со всеми зависимостями.
- создайте установщик с использованием Microsoft Visual Studio Installer Projects

Отчет должен содержать (см. образец):

- номер и тему лабораторной работы;
- фамилию, номер группы студента и вариант задания;
- скриншоты окна Visual Studio с исходным кодом программы и комментариями;
- скриншоты с результатами выполнения программ;
- пояснения, если необходимо;
- выводы.

Отчеты в формате pdf отправлять на email: colledge20education23@gmail.com