

## Практическое занятие № 12

**Тема практического занятия:** Работа со списками.

**Цель практического занятия:** формирование практических навыков работы со списками и их применения в решении задач программирования.

**В результате выполнения данной работы обучающийся должен уметь:**

1. Добавлять, удалять и изменять элементы в списке
2. Использовать циклы и итераторы для обхода элементов списка.
3. Разбираться в основных типах данных, которые могут быть хранены в списке (числа, строки, объекты и т.д.).
4. Применять полученные знания и навыки для решения конкретных задач, требующих работы со списками.

**знать:**

1. Что такое список и чем он отличается от других структур данных
2. Как объявлять и создавать списки в выбранном языке программирования.
3. Основные методы и операции работы со списками.
4. Различные способы и стратегии использования списков в задачах программирования и анализа данных.

**Перечень оборудования, необходимого для выполнения задания:**

- Автоматизированные рабочие места по количеству обучающихся (процессор Intel Core i7 или аналогичный, БП 700 Вт, 32 Гб ОЗУ, SSD 512 Гб, HDD 4 ТБ SATA 7200 rpm, RTX 3060 12GB);
  - Автоматизированное рабочее место преподавателя (процессор Intel Core i7 или аналогичный, БП 700 Вт, 32 Гб ОЗУ, SSD 512 Гб, HDD 4 ТБ SATA 7200 rpm, RTX 3060 12GB);
  - Монитор 34", изогнутый, 3440x1440, 6 ms, 178°/178°, 300 cd/m2, 20M:1, HDMI, DP, USB, регулировка по высоте (1 монитор на одно рабочее место).
  - Сервер (2 x Intel Xeon Gold 5218R или аналогичный, 256 Гб, DDR4-2933 ECC Registered DIMM, 2U, 800W(1+1), 4 x GLAN. IPMI 2.0, RAID 1GB, 2 x HDD 1.2 ТБ SAS 12 Gbit/s 10000 rpm, 2 x SSD 960 Гб SATA 2.5", Hot-Swap HDD Bay, Rail kit) или выделение аналогичного по характеристикам виртуального сервера.
  - Интерактивная панель 86" с OPS ПК
- Программное обеспечение:
- Microsoft Visual Studio.

**Общие теоретические сведения:**

Список - это структура данных, представляющая собой упорядоченную коллекцию элементов. Элементы могут быть любого типа данных: числа, строки, объекты и другие списки.

Особенность списка заключается в том, что элементы в нём располагаются в определенном порядке и имеют свои позиции, называемые индексами. Благодаря индексам можно легко получать доступ к элементам списка и работать с ними.

Списки обладают множеством полезных свойств и методов, которые позволяют добавлять, удалять, изменять и искать элементы, сортировать элементы по определенным критериям, фильтровать список по заданному условию, а также выполнять другие операции. Списки являются одним из основных инструментов для работы с коллекциями данных в программировании и анализе данных.

В целом, списки являются часто используемой структурой данных, которая предоставляет гибкость и эффективность при работе с коллекциями данных. Их использование рекомендуется в широком спектре задач, где необходима упорядоченная коллекция элементов.

### **Задание:**

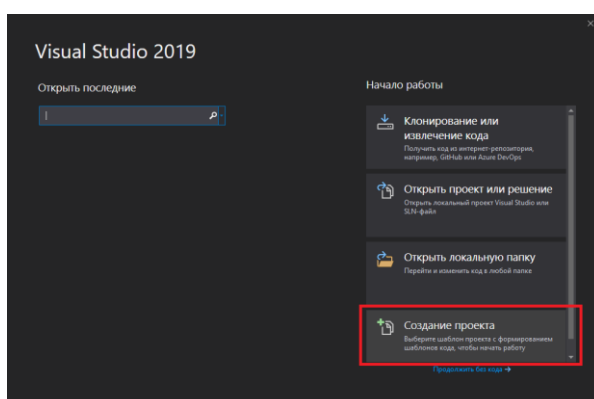
1. Разработайте приложение №1, которое позволит решить следующую задачу. Пользователем формируется список из фамилий студентов. Необходимо выдать список из  $n$  неповторяющихся фамилий (число  $n$  задается пользователем).
2. Разработайте приложение №2, которое позволит решить следующую задачу. Описать класс «Задача проекта» с полями и соответствующими им свойствами: наименования задачи, срок выполнения задачи. У пользователя должна быть возможность создать список задач. Требуется вывести их в порядке сроков выполнения: от самого раннего к самому позднему.

### **Указания по технике безопасности:**

Инструкция по технике безопасности при работе в лаборатории, оборудованной компьютерной техникой.

### **Технология выполнения работы (этапы, последовательность действий):**

1. Создайте проект для приложения №1. Пример создания проекта в среде MS Visual Studio



2. Разработайте интерфейс приложения.
3. Разработайте обработчик событий, который позволит добавлять элементы в список студентов.
4. Разработайте обработчик событий, который позволит выдать нужное число фамилий. Помните, что для того, чтобы фамилии не повторялись, уже выбранную фамилию следует из списка удалить.
5. Проверьте работоспособность приложения.
6. Создайте проект для приложения №2.
7. Разработайте интерфейс приложения.
8. Опишите класс, соответствующий задаче проекта.
9. Разработайте обработчик событий, который позволит добавлять элементы в список задач.
10. Разработайте обработчик событий, который позволит вывести все задачи в нужном порядке.
11. Проверьте работоспособность приложения.

Пример кода для обработки списка приведен ниже.

```
static void Main(string[] args)
{
    List<int> list = new List<int>();
    Random rnd = new Random();
    bool answer = false;
    for (int i = 0; i < 10000; i++)
        list.Add(rnd.Next(-100000, 100001));
    for (int i = 0; i < 10000; i++)
        for (int j = 0; j < 10000; j++)
            if (list[i] == list[j] && i != j)
                answer = true;
    Console.WriteLine(answer ? "Есть" : "Нет");
    Console.ReadKey();
}
```

Данный код генерирует список, содержащий случайные числа. Затем в коде определяется, есть ли в списке повторяющиеся элементы. Обратите внимание, что добавление элементов в список производится с помощью метода Add.

### Требование к отчету:

1. Приложение №1 разработано, компилируется, запускается.
2. Задача №1 решена корректно: список студентов формируется, выдается нужное количество случайных фамилий, в выданных элементах нет повторяющихся.
3. Приложение №2 разработано, компилируется, запускается.
4. Для приложения №2 разработан класс, соответствующий задаче проекта.

5. Задача №2 решена корректно: список задач формируется, выводится в правильной последовательности.

### **Контрольные вопросы:**

1. Как объявить и создать список в выбранном языке программирования?
2. Как добавить элемент в конец списка?
3. Как вставить элемент в определенную позицию в списке?
4. Как удалить элемент из списка?
5. Как получить индекс элемента в списке?
6. Как получить количество элементов с определенным значением в списке?
7. Как отсортировать список по возрастанию или убыванию?
8. Как перевернуть порядок элементов в списке?
9. Как объединить два списка в один?
10. Как удалить все элементы из списка?
11. Как скопировать список?
12. Как использовать цикл или итератор для обхода элементов списка?
13. Как применить фильтрацию к списку с использованием определенного условия?

### **Основные и дополнительные источники, электронные ресурсы:**

1. Подбельский, В. В. Язык C#. Базовый курс: учебное пособие / В. В. Подбельский. - 2-е изд., перераб. и доп. - Москва: Финансы и статистика, 2022. - 408 с. - ISBN 978-5-00184-079-4. - Текст: электронный. - URL: <https://znanium.com/catalog/product/1913989>
2. Гуриков, С. Р. Основы алгоритмизации и программирования на Python: учебное пособие / С.Р. Гуриков. — Москва: ИНФРА-М, 2023. — 343 с. — (Среднее профессиональное образование). - ISBN 978-5-16-016906-4. - Текст: электронный. - URL: <https://znanium.com/catalog/product/1927269>.