

Практическое занятие № 13

Тема практического занятия: Работа со словарем.

Цель практического занятия: формирование практических навыков работы со словарями, которые могут быть применены в различных областях программирования и анализа данных.

В результате выполнения данной работы обучающийся должен уметь:

1. Добавлять, удалять и обновлять элементы в словаре.
2. Разбираться в основных понятиях словаря, таких как ключ и значение.
3. Использовать циклы и итераторы для обхода элементов словаря
4. Применять полученные знания и навыки для решения конкретных задач, требующих работы со словарями.

знать:

1. Что такое словарь и как он отличается от других структур данных.
2. Как объявить и создать словарь в выбранном языке программирования.
3. Основные методы и операции работы со словарями, такие как получение значения по ключу, проверка наличия ключа, перебор элементов словаря.
4. Принципы хэширования и использования хэш-функций при работе со словарями.

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).
 - Интерактивная панель 86" с OPS ПК
- Программное обеспечение:
- Microsoft Visual Studio.

Общие теоретические сведения:

Словарь - это структура данных, представляющая собой неупорядоченную коллекцию пар ключ-значение.

В словаре ключ - это уникальное значение, которое используется для идентификации и доступа к соответствующему значению данных. Каждое значение данных в словаре связано с определенным ключом. Ключи в словаре должны быть уникальными, то есть не могут повторяться.

Ключи в словаре могут быть любого неизменяемого типа данных, такого как число, строка или кортеж. Обычно ключами словаря являются строки или

числа. Когда используется ключ для доступа к значению данных в словаре, словарь быстро выполняет поиск, чтобы найти связанное с этим ключом значение.

Использование ключей в словаре позволяет легко находить и обрабатывать данные, что делает словарь мощным инструментом в программировании.

Словарь может применяться для следующих целей:

1. Хранение и организация информации: словарь может использоваться для хранения пар ключ-значение, где каждому ключу соответствует определенное значение. Это позволяет легко и быстро найти и получить доступ к нужным данным.

2. Быстрый поиск: словарь предоставляет эффективные алгоритмы поиска по ключу, что позволяет быстро получить доступ к значению, необходимому пользователю. Это особенно полезно при больших объемах данных.

3. Видоизменение данных: словарь может использоваться для модификации или обновления значений, связанных с определенным ключом. Это позволяет легко изменять и обрабатывать информацию в словаре.

4. Установление соответствия: словарь может использоваться для установления соответствия между двумя наборами данных. Например, он может использоваться для привязки идентификаторов пользователя к их именам или адресам.

5. Преобразование данных: словарь может использоваться для преобразования формата данных из одной структуры в другую. Например, словарь может быть использован для преобразования JSON-объекта в объект Python или наоборот.

6. Группировка и классификация данных: словарь может использоваться для группировки и классификации данных по определенным параметрам. Например, словарь может быть использован для классификации студентов по курсам или по результатам экзаменов.

В целом, словарь – это структура данных, которая предоставляет мощные функциональные возможности для хранения, управления и обработки информации. Его применение может быть очень разнообразным в зависимости от конкретной задачи или сценария использования.

Задание:

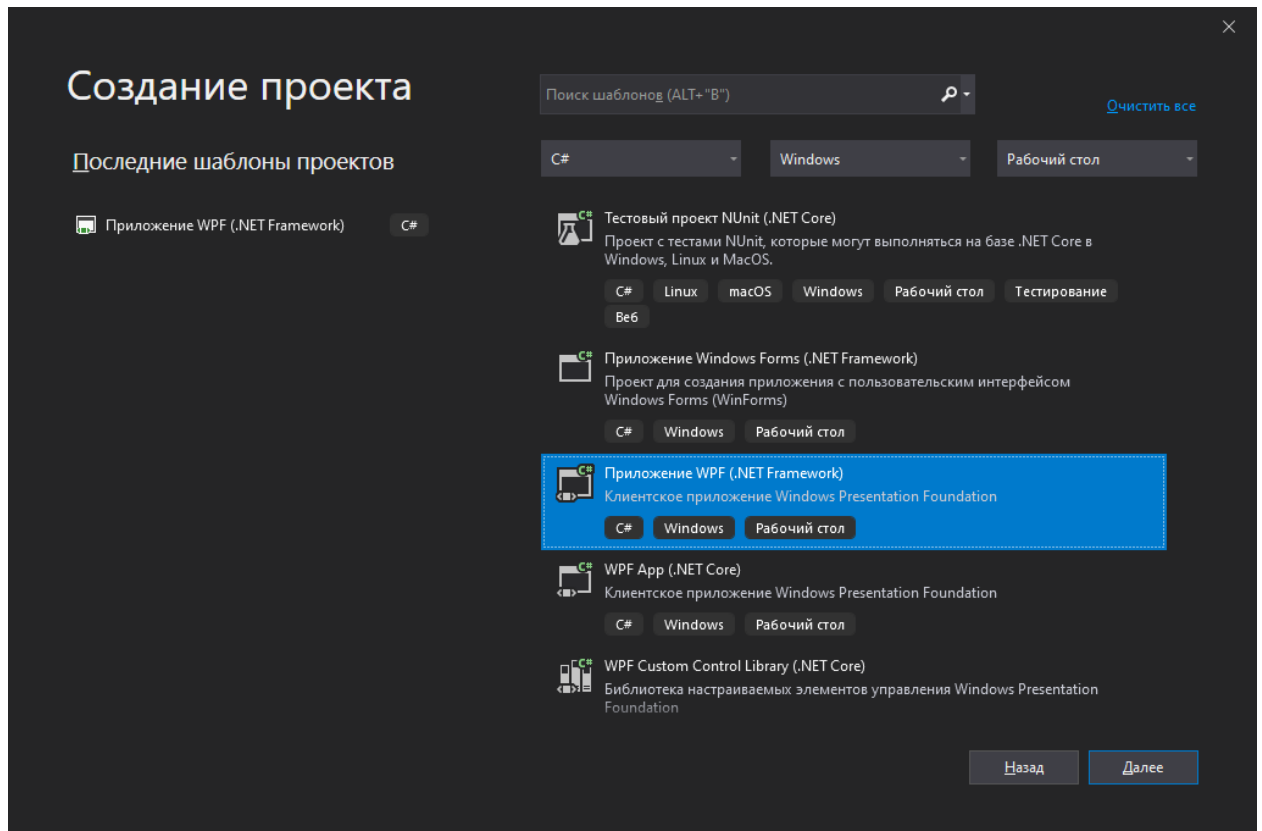
1. Разработайте приложение №1, которое будет решать следующую задачу. Составить частотный словарь для строки, введенной пользователем. Необходимо подсчитать, сколько раз какой символ встречается.
2. Разработайте приложение №2, которое будет решать следующую задачу. Вводится набор оценок студента. Необходимо подсчитать, сколько каких оценок получено студентом.

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 приложения для MS Visual Studio:



2. Разработайте интерфейс приложения.
3. Разработайте обработчик событий, который позволит формировать и выдавать частотный словарь.
4. Проверьте работоспособность приложения.
5. Создайте проект для приложения №2.
6. Разработайте интерфейс приложения.
7. Разработайте обработчик событий, который позволит формировать словарь с оценками и выдавать результат.
8. Проверьте работоспособность приложения.

Пример инициализации словаря:

```
var people = new Dictionary<int, string>()  
{  
    [5] = "Tom",
```

```
[6] = "Sam",  
[7] = "Bob"  
};
```

Пример перебора элементов словаря:

```
var people = new Dictionary<int, string>()  
{  
    [5] = "Tom",  
    [6] = "Sam",  
    [7] = "Bob"  
};  
foreach(var person in people)  
{  
    Console.WriteLine($"key: {person.Key} value: {person.Value}");  
}
```

Пример обращений к элементам словаря:

```
var people = new Dictionary<int, string>()  
{  
    [5] = "Tom",  
    [6] = "Sam",  
    [7] = "Bob",  
};  
// получаем элемент по ключу 6  
string sam = people[6]; // Sam  
Console.WriteLine(sam); // Sam  
// переустанавливаем значение по ключу 6  
people[6] = "Mike";  
Console.WriteLine(people[6]); // Mike
```

// добавляем новый элемент по ключу 22

people[22] = "Eugene";

Console.WriteLine(people[22]); // Eugene

Требование к отчету:

1. Приложение №1 разработано, компилируется, запускается.
2. Задача №1 решена корректно.
3. Приложение №2 разработано, компилируется, запускается.
4. Задача №2 решена корректно.

Контрольные вопросы:

1. Что такое словарь в программировании и зачем он используется?
2. Как создать словарь?
3. Что такое ключ в словаре?
4. Какой тип данных может быть использован в качестве ключа?
5. Как добавить новую пару ключ-значение в словарь?
6. Как получить значение из словаря по его ключу?
7. Как проверить наличие ключа в словаре?
8. Как удалить элемент из словаря?
9. Как обновить значение ключа в словаре?
10. Как получить все ключи из словаря?
11. Как получить все значения из словаря?

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>.