



Методические рекомендации по теме «Создание словаря, работа с его элементами»

Цель:

- дать представление об использовании словарей в языке Python.

Задачи:

- практика применения словарей в Python;
- знакомство с понятие «ключ» применительно к словарям в Python;
- анализ программного кода с целью определения, что выведет программа при конкретных исходных данных;
- исправление ошибок и дописывание программного кода;
- написание программного кода.

Планируемые результаты

Личностные: обучающиеся получают навыки активной коммуникации в группе, осознанной ориентировки в мире ИТ профессий, постановки собственных образовательных целей и задач, владение первичными навыками анализа и критичной оценки получаемой информации.

Предметные: обучающиеся получают представления об использовании словарей в Python.

Метапредметные: обучающиеся получают возможность владения общепредметными понятиями «словарь», «структура», «ключ»; информационно-логическими умениями; умениями самостоятельно планировать пути достижения целей; владения умениями принятия решений и осуществления осознанного выбора; повышения уровня ИКТ – компетентности и расширение кругозора в области информатики и программирования; знакомство с профессиональной деятельностью

программиста в рамках ранней профориентации; развитие интеллектуальных способностей, а также логического и критического мышления.

Материалы к занятию

Приложение 1: Сценарный план видеоролика

Приложение 2: Домашние задание и практика

Приложение 3: Краткие организационно-методические рекомендации по организации работы на занятии

Ход проведения урока

1. Организационный момент.

Мотивация на учебную деятельность.

Приветствие учащихся, сообщение темы и целей занятия.

2. Вводный блок.

Тема.

Преподаватель при необходимости останавливая трансляцию, комментируя дополнительно тему занятия.

Проблемная дискуссия по вопросам:

- Как бы вы объяснили понятие «словарь»?
- Как используются словари в реальной жизни?
- Как мы ищем информацию в словарях?
- Как вы думаете, для чего могут использоваться структуры с названием «словарь» в программировании?

Итоги дискуссии (обобщаются преподавателем и фиксируются ответы учеников на доске, чтобы вернуться к ним и оценить правильность предположений учеников на этапе рефлексии):

- Словарь еще одна структура в Python, которая применяется для хранения больших объемов информации
- Работа словаря в программировании, как и словаря в реальной жизни организуется по принципу ключевых слов

**см. сцены 1 – 2 (здесь и далее приводится Таблица «Содержание видеоролика». Приложение 1).*

3. Блок повторения.

Блиц-опрос.

Преподаватель предлагает ученикам ответить на **5 вопросов** по предыдущей теме; задания выполняются в сопровождении видеоролика с использованием таймера; ученики выполняют задания, голосуют, обсуждают результаты. Процедура голосования определяется инструкцией **в сцене 3**; учитель должен убедиться, что всем понятна процедура голосования. *Преподаватель может поставить ролик на паузу и обсудить результаты голосования; объяснить правильный ответ руководствуясь материалами предыдущего занятия*

**см. сцены 3 – 7*

4. Теоретический блок.

Словари: основные понятие, создание.

Новый материал излагается в сопровождении видеоролика, рекомендуется разместить на доске или флип-чарте изображения объектов, сопровождающих материалы по теме.

Обсуждением вопросов по просмотренным материалам:

- Почему словарь является неупорядоченной структурой?
- Что такое ключ и почему он важен для работы словарей?
- В чем отличие словарей от списков и множеств?
- В чем сходство множеств и словарей?

При необходимости преподаватель может поставить ролик на паузу и дать дополнительные пояснения по материалу; если ответы на вопросы вызывают у учеников затруднения, преподаватель может вывести нужную сцену ролика на экран для помощи ученикам.

**см. сцена 8 – 10, 15, 20, 23*

5. Блок заданий.

Проекты: «Мой словарь», «Добавить, удалить, очистить».

К началу демонстрации блока заданий ученики должны занять рабочие места и запустить Python (терминал IDLE) на своих компьютерах.

«Мой словарь»: включает *практическое задание 1* которое состоит в том, чтобы создать словарь, в котором будут храниться названия команд и их игроки.

После выполнения задания ученики получают работающий программный продукт – программа собственного словаря для хранения информации

«Добавить, удалить, очистить»: включает *практическое задание 2*, которое создает команды работы со словарем программными методами, добавлять и удалять элементы словаря, а также удалять ячейки словаря или весь словарь целиком.

После выполнения задания ученики получают работающий программный продукт – программа, включающая основные инструменты работы с данными собственного словаря.

На сцене разбора задания преподаватель ставит ролик на паузу и вместе с учениками проводит разбор задания.

**см. сцены 11 – 23 (кроме сцен по теории).*

6. Рефлексия. Сообщение домашнего задания.

Завершаем демонстрацией ролика и кратким обобщением материалов занятия. Преподаватель возвращается к зафиксированным в ходе дискуссии в начале урока предположениям учеников и обсуждает насколько их предположения были правильными, делаются выводы.

Преподаватель дает ученикам домашнее задание к следующему занятию (*Приложение 2*).

**см. сцена 24*

Приложение 1

Сценарный план видеоролика

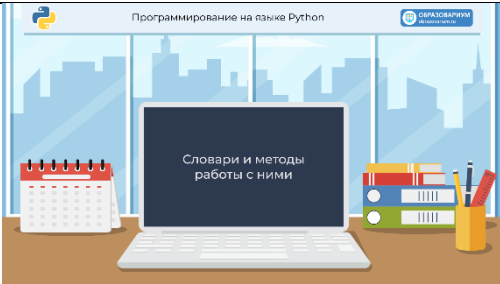
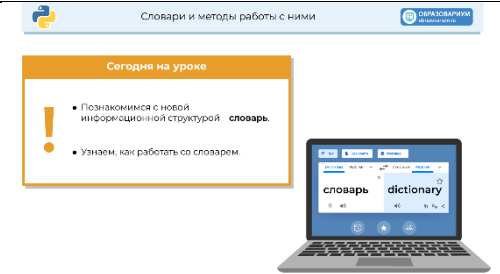
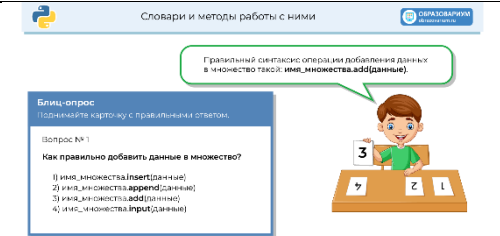
В таблице «Содержание видеоролика» представлены:

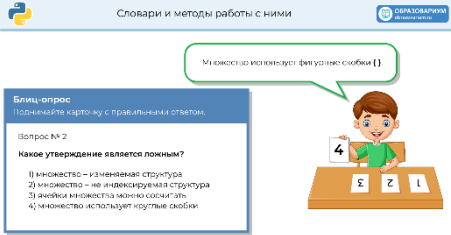
- название блоков видеоролика (тайминг);
- краткое описание содержания в каждом блоке;
- фрагменты из видеоролика, относящиеся к соответствующему блоку;
- номера сцен в каждом блоке.

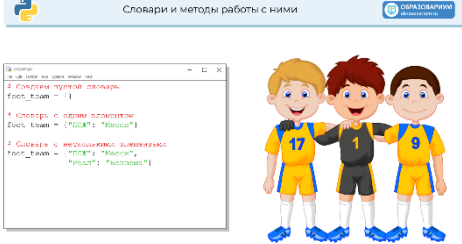
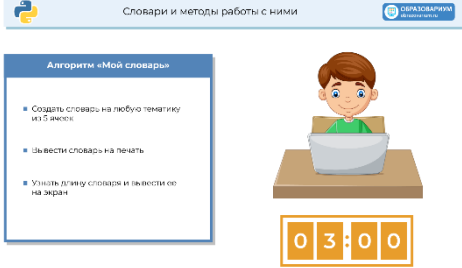
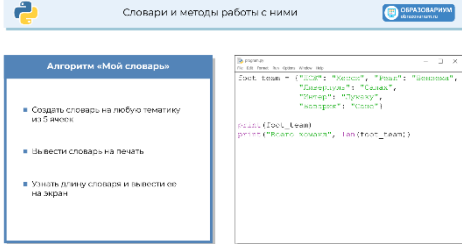
Учитель при подготовке к уроку может ознакомиться с содержанием видеоролика в текстовом формате, при необходимости распечатать фрагменты текста или примеры заданий и задач для использования в работе с учениками. Распечатанные тексты и задания из таблицы также можно применять в качестве раздаточного материала как на уроке, так и для домашних заданий.

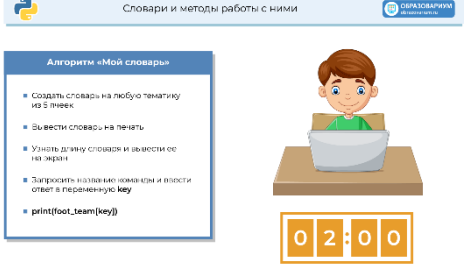
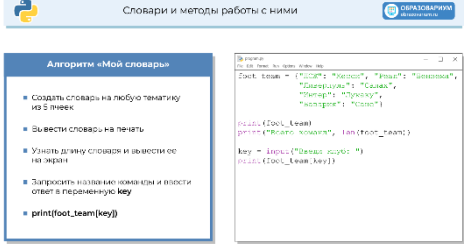
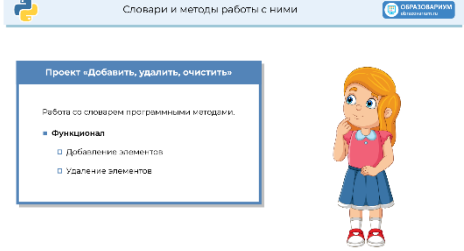
Таблица. Содержание видеоролика

Название блока	Содержание блока и комментарии	Фрагменты из видеоролика	№ сцен
----------------	--------------------------------	--------------------------	--------

<p>Вводный блок. Мы узнаем</p>	<p>Обозначаем ученикам тему и цели урока.</p> <p>Создание словаря, работа с его элементами</p>	 <p>Сцена 1</p>	<p>1 2</p>
	<p>Продолжаем изучать информационные структуры и определимся с понятием словарь.</p> <p>Это слово всем хорошо знакомо, а особенно тем, кто изучает иностранные языки. Именно в словарях можно найти нужную информацию, зная только ключевое слово. Словари в программирование выполняют схожую функцию, давайте рассмотрим: как они создаются и работают.</p>	 <p>Сцена 2</p>	
<p>Блок повторения. Блиц-опрос</p>	<p>Повторение материала предыдущего урока; на столе имеются пронумерованные карточки; после каждого вопроса выбираем ту, номер которой, совпадает с правильным ответом.</p> <p>Первый вопрос. Как правильно добавить данные в множество?</p> <ol style="list-style-type: none"> 1) имя_множества.insert(данные) 2) имя_множества.append(данные) 3) имя_множества.add(данные) 4) имя_множества.input(данные) <p>Ответ 3. Правильный синтаксис операции добавления данных в множество такой: имя_множества.add(данные)</p>	 <p>Сцена 3</p>	<p>3 4 5 6 7</p>

	<p>Второй вопрос. Какое утверждение является ложным?</p> <ol style="list-style-type: none"> 1) множество – изменяемая структура 2) множество – не индексируемая структура 3) ячейки множества можно сосчитать 4) множество использует круглые скобки <p><i>Ответ 4. Множество использует фигурные скобки { }.</i></p>	 <p>Словари и методы работы с ними</p> <p>Множество использует фигурные скобки { }</p> <p>Блиц-опрос Поднимайте карточку с правильными ответом.</p> <p>Вопрос № 2 Какое утверждение является ложным?</p> <ol style="list-style-type: none"> 1) множество – изменяемая структура 2) множество – не индексируемая структура 3) ячейки множества можно сосчитать 4) множество использует круглые скобки 	
	<p>Третий вопрос. Какие индексы у числа 5?</p> <p><code>[[1, 2, 3], [], [4, 5, 6, 7, 8]]</code></p> <p>Поднимите карточки с соответствующими числами.</p> <p><i>Ответ 2 и 1. Индексы числа 5 – [2][1]</i></p> <p><code>[[1, 2, 3], [], [4, 5, 6, 7, 8]]</code></p> <p>0 1 2 3 4</p> <p>0 1 2</p>	<p>Словари и методы работы с ними</p> <p>Индексы числа 5 – [2][1]</p> <p>Блиц-опрос Поднимайте карточку с правильными ответом.</p> <p>Вопрос № 3 Какие индексы у числа 5?</p> <p><code>[[1, 2, 3], [], [4, 5, 6, 7, 8]]</code></p> <p>Поднимите карточки с соответствующими числами.</p>	<p>Сцена 4</p> <p>Сцена 5</p>
	<p>Четвертый вопрос. Какая команда считает длину объекта?</p> <ol style="list-style-type: none"> 1) input 2) len 3) pop 4) append <p><i>Ответ 2. Команда len подсчитывает длину объекта.</i></p>	<p>Словари и методы работы с ними</p> <p>Команда len подсчитывает длину объекта.</p> <p>Блиц-опрос Поднимайте карточку с правильными ответом.</p> <p>Вопрос № 4 Какая команда считает длину объекта?</p> <ol style="list-style-type: none"> 1) input 2) len 3) pop 4) append 	<p>Сцена 6</p>

	<p>Давайте создадим свой первый словарь, в котором будут храниться названия команд и их игроки. Нам необходимо имя для словаря.</p> <p>Для создания словаря мы используем фигурные скобки. Заполним данный словарь; первым элементом укажем данные, которые будут ключом - название команды; через двоеточие – данные, которые будут значением (фамилия игрока). Двоеточие является важным символом в словаре – оно отделяет ключ от значения.</p> <p>Если хотим добавить вторую команду, то между ячейками ставится запятая.</p>	 <p>Словари и методы работы с ними</p> <p>Сцена 10</p>	
<p>Блок заданий.</p> <p>Практические задания:</p> <p>Задание 1</p>	<p>После окончания дикторского текста запускается таймер на 3 мин.</p> <p>Задание 1. Алгоритм «Мой словарь»</p> <ul style="list-style-type: none"> ■ Создать словарь на любую тематику из 5 ячеек ■ Вывести его на печать ■ Узнать длину словаря и вывести ее на экран. 	 <p>Словари и методы работы с ними</p> <p>Сцена 11</p>	<p>11 12 13 14</p>
	<p>Разбор задания 1. Ваш код может выглядеть так.</p> <pre>foot_team = {"ПСЖ": "Месси", "Реал": "Бензема", "Ливерпуль": "Салах", "Интер": "Лукаку", "Бавария": "Сане"}</pre> <pre>print(foot_team) print("Всего команд", len(foot_team))</pre>	 <p>Словари и методы работы с ними</p> <p>Сцена 12</p>	

	<p>После окончания дикторского текста запускается таймер на 2 мин.</p> <p>Задание 1. Алгоритм «Мой словарь». Продолжение ... <i>дописать к существующим шагам алгоритма</i></p> <ul style="list-style-type: none"> ■ Запросить название команды и ввести ответ в переменную key ■ Напечатать (foot_team[key]) 	 <p>Сцена 13</p>	
	<p>Разбор задания 1. Продолжение Ваш код может выглядеть так. ... <i>дописывается к существующему коду</i></p> <pre>key = input("Введи клуб: ") print(foot_team[key])</pre>	 <p>Сцена 14</p>	
<p>Теоретический блок</p>	<p>А теперь давайте попробуем поработать со словарем программными методами. Создадим цикл, внутри которого будем добавлять в словарь ячейки, считывать их значения и удалять.</p> <p>Проект «Добавить, удалить, очистить» Работа со словарем программными методами.</p> <p>Функционал</p> <ul style="list-style-type: none"> • Добавление элементов • Удаление элементов. 	 <p>Сцена 15</p>	<p>15</p>

Блок заданий.
Практические задания:
Задание 2

После окончания дикторского текста запускается таймер на 3 мин.

- Задание 2. Алгоритм «Добавить, удалить, очистить»**
- Создать пустой словарь с любым именем
 - **num = None**
 - Цикл работать пока **num != 0**
 - Вывести текст «0 – выход, 1 – добавить»
 - Запросить число и присвоить его переменной **num**

Словари и методы работы с ними

Алгоритм «Добавить, удалить, очистить»

- Создать пустой словарь с любым именем
- **num = None**
- до тех пор пока num не равно 0
 - Вывести текст «0 - выход, 1 - добавить»
 - Запросить число и присвоить его переменной **num**

Сцена 16

Разбор задание 2. Код программы будет выглядеть так:

```
my_dict = {}          # создаем пустой словарь
num = None
while num != 0:
    num = int(input("0 - выход, 1 - добавить: "))
```

Словари и методы работы с ними

Алгоритм «Добавить, удалить, очистить»

- Создать пустой словарь с любым именем
- **num = None**
- до тех пор пока num не равно 0
 - Вывести текст «0 - выход, 1 - добавить»
 - Запросить число и присвоить его переменной **num**

Сцена 17

После окончания дикторского текста запускается таймер на 3 мин.

- Задание 2. Алгоритм «Добавить, удалить, очистить». Продолжение**
(Код внутри цикла)
- Если введено 1, то
 - Запросить ключ и занести данные в переменную **key**
 - Запросить информацию и занести ее в переменную **data**
 - **my_dict[key] = data**

Словари и методы работы с ними

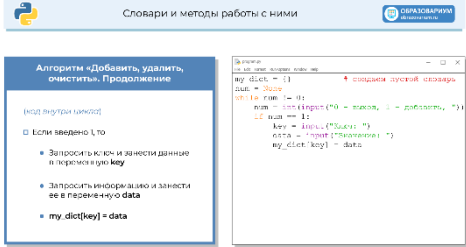
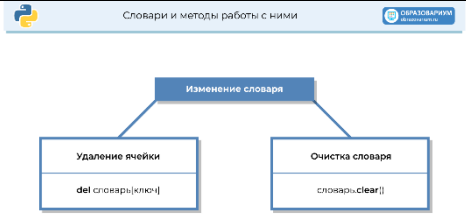
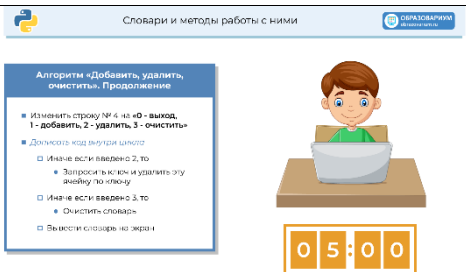
Алгоритм «Добавить, удалить, очистить». Продолжение

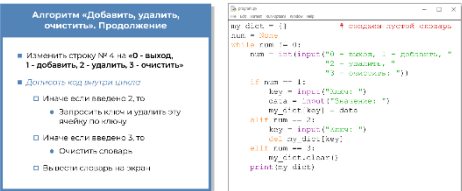
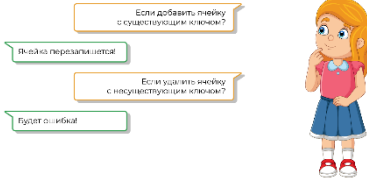
(код внутри цикла)


- Если введено 1, то
 - Запросить ключ и занести данные в переменную **key**
 - Запросить информацию и занести ее в переменную **data**
 - **my_dict[key] = data**

Сцена 18

16
17
18
19

	<p>Разбор задание 2. Продолжение Код программы будет выглядеть так: ... <i>дописывается к существующему коду</i></p> <pre> if num == 1: key = input("Ключ: ") data = input("Значение: ") my_dict[key] = data </pre>	 <p>Сцена 19</p>	
Теоретический блок.	<p>Теперь давайте рассмотрим, как можно удалять ячейки из словаря. Для этого нам надо или знать ключ удаляемой ячейки или можно полностью очистить словарь</p> <p>В этом нам помогут команды del или clear</p> <p>И снова обращаем внимание, что если речь идет о ячейке – то использоваться будут квадратные скобки</p>	 <p>Сцена 20</p>	20
Блок заданий. Практические задания: Задание 2	<p><i>После окончания дикторского текста запускается таймер на 5 мин.</i></p> <p>Задание 2. Алгоритм «Добавить, удалить, очистить». Окончание</p> <ul style="list-style-type: none"> Изменить строку № 4 на «0 - Выход, 1 - Добавить, 2 - Удалить, 3 – Очистить» (Дописать код внутри цикла) <ul style="list-style-type: none"> Иначе если введено 2, то <ul style="list-style-type: none"> Запросить ключ и удалить эту ячейку по ключу Иначе если введено 3, то <ul style="list-style-type: none"> Очистить словарь Вывести словарь на экран 	 <p>Сцена 21</p>	21 22

	<p>Разбор задание 2. Окончание Код программы будет выглядеть так: ... <i>дописывается к существующему коду</i></p> <pre> my_dict = {} # создаем пустой словарь num = None while num != 0: num = int(input("0 - выход, 1 – добавить, " "2 - удалить, 3 - очистить: ")) if num == 1: key = input("Ключ: ") data = input("Значение: ") my_dict[key] = data elif num == 2: key = input("Ключ: ") del my_dict[key] elif num == 3: my_dict.clear() print(my_dict) </pre>	 <p>Сцена 22</p>	
<p>Теоретический блок</p>	<p>Остались нераскрыты два вопроса. Что будет если ввести уже добавить ячейку с уже имеющимся ключом? Очень просто: она перезапишется</p> <p>А если попытаться удалить ячейку с несуществующим ключом? Будет ошибка!</p>	 <p>Сцена 23</p>	<p>23</p>

<p>Блок завершения занятия. Рефлексия. Сообщение домашнего задания</p>	<p><i>Завершаем демонстрацией ролика и кратким обобщением материалов занятия.</i></p> <p>Подведем итоги.</p> <p>Плюсы словарей:</p> <ul style="list-style-type: none"> отсутствуют специальные команды для добавления и замены ячеек; быстрый поиск. <p>Минусы словарей:</p> <ul style="list-style-type: none"> отсутствие индексов; принцип работы через ключи требует особого навыка. <p><i>Преподаватель дает ученикам домашнее задание к следующему занятию (Приложение 2).</i></p>	 <p>Сцена 24</p>	<p>24</p>
----------------------------------------------------------------------------------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------	-----------

Приложение 2

Домашнее задание

Перечислите несколько структур из реальной жизни, которые работают по принципу словарей (данные расположены по принципу «ключ-значение»).

Задание можно выполнить на компьютере и представить результат и код в виде файла или снимка экрана, или распечатки.

Практика

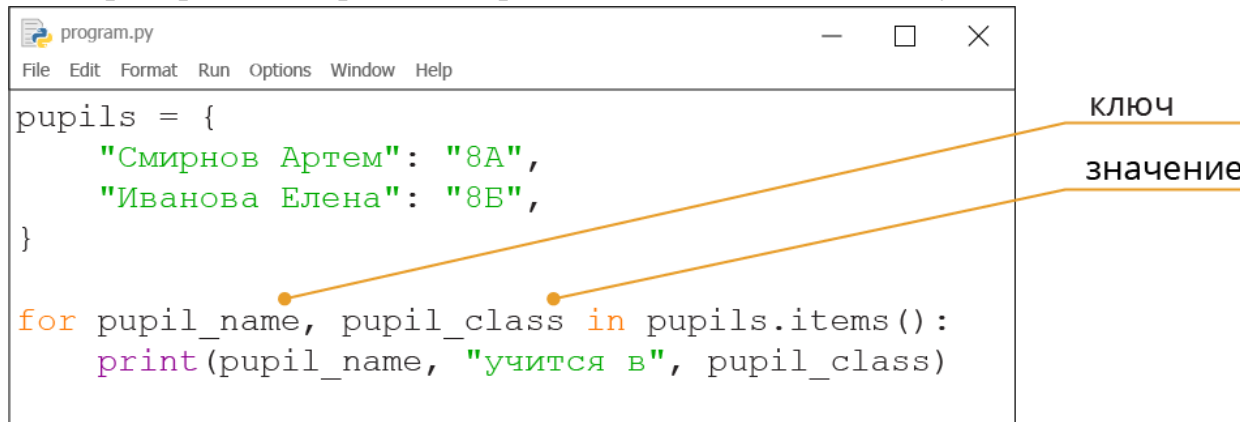
Проект «База данных»

Запросите у пользователя целое число. Создайте словарь, у которого

- ключами будут числа от 2 до введенного числа
- значениями будут списки из двух значений: ключ в квадрате и квадратный корень из ключа.

Выведите значения словаря на экран.

Для перебора в словаре всех пар «ключ-значение» используют конечный цикл и метод **items()** для словаря.



```
program.py
File Edit Format Run Options Window Help

pupils = {
    "Смирнов Артем": "8А",
    "Иванова Елена": "8Б",
}

for pupil_name, pupil_class in pupils.items():
    print(pupil_name, "учится в", pupil_class)
```

ключ

значение

Такой код выведет:

Смирнов Артем учится в 8А

Иванова Елена учится в 8Б

Проект «Переводчик»

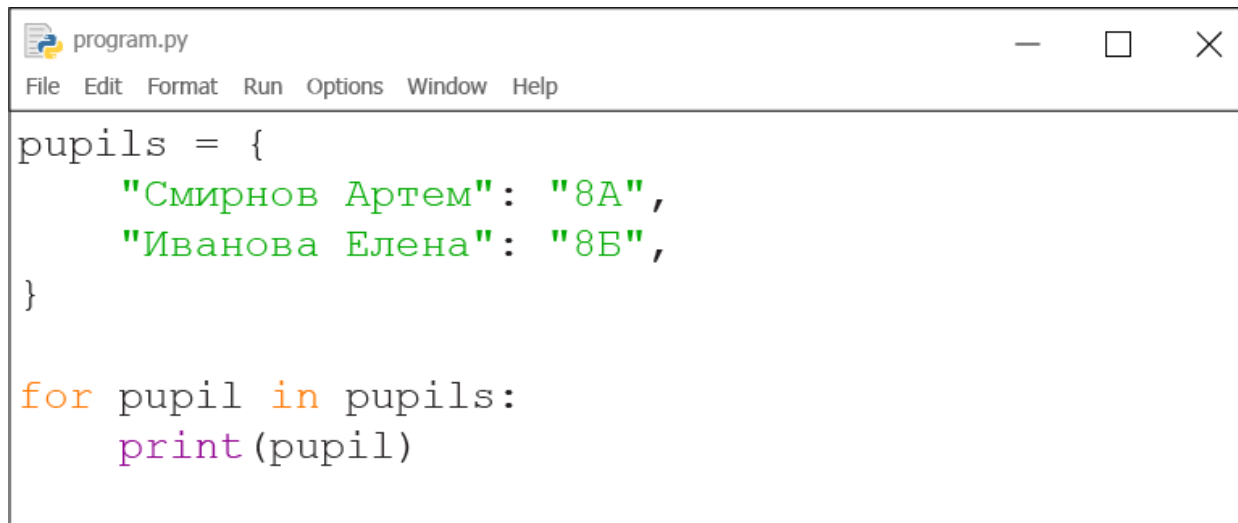
Создайте словарь, ключами которого будут русские слова, а значениями – их перевод на английский язык.

Выведите на экран список доступных для перевода слов.

В бесконечном цикле спрашивайте у пользователя, какое слово он хочет перевести и выводите перевод этого слова.

Внутри цикла спросите пользователя, хочет ли он продолжить работу и в зависимости от ответа остановите цикл или продолжите работу.

Перебор всех ключей в словаре можно организовать так:



```
program.py
File Edit Format Run Options Window Help

pupils = {
    "Смирнов Артем": "8А",
    "Иванова Елена": "8Б",
}

for pupil in pupils:
    print(pupil)
```

Такой код выведет:

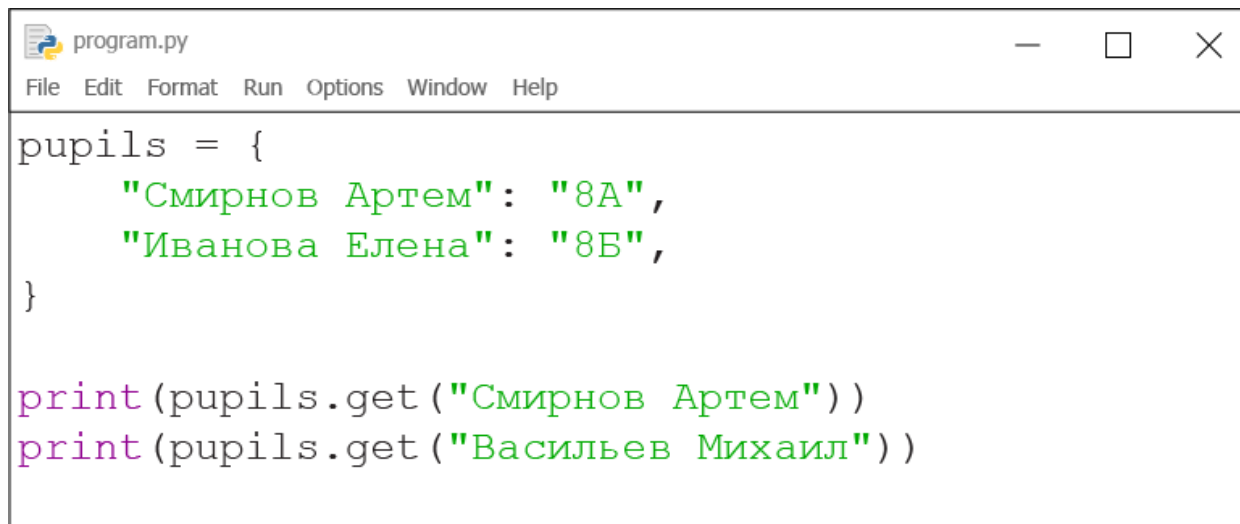
Смирнов Артем

Иванова Елена

Чтобы программа не прерывалась с ошибкой при попытке получить значение из словаря по несуществующему ключу, можно использовать метод **get()**.

С помощью этого метода мы получим

- значение, если ключ в словаре есть,
- None (что означает «ничего»), если ключа в словаре нет.



```
program.py
File Edit Format Run Options Window Help

pupils = {
    "Смирнов Артем": "8А",
    "Иванова Елена": "8Б",
}

print(pupils.get("Смирнов Артем"))
print(pupils.get("Васильев Михаил"))
```

Первый **print** выведет 8А, а второй – None.

Приложение 3

Краткие организационно-методические рекомендации по организации работы на занятии

«Создание словаря, работа с его элементами».

В начале занятия желательно повторить материал по теме «Множества» и «Списки». Для наглядности можно использовать доску, на которой в виде таблицы отмечать общие и различные методы свойства. Перед просмотром ролика можно поинтересоваться, какие ассоциации вызывает у ребят термин «словарь». Какие у него отличия от обыкновенной книги?

Перед просмотром блока повторения из ролика необходимо раздать дидактический материал для выполнения заданий из блока повторение (по 4 пронумерованных карточки)

Во время голосований карточками можно останавливать ролик и вести учет правильных ответов. По окончании блока – отметить тех, у кого наилучший результат. Далее карточки необходимо собрать.

Во время просмотра, ролик можно ставить на паузу, чтобы дописать на доске теоретический материал по данной теме. Особенно в сравнении с уже известными структурами. Можно лишний раз повторить правила создания имен списков, множеств, переменных. Особое внимание уделите фигурным скобкам при создании и выводе на экран и квадратным скобкам – при работе с ячейками.

Перед началом первого самостоятельного задания можете обговорить общую тему словаря для всей группы или дать возможность индивидуального творчества. Во втором задании не рассматривается способ удаления словаря целиком, ввиду отсутствия целесообразности – код станет нерабочим. Но вы можете упомянуть о такой возможности.