



АДМИНИСТРАЦИЯ ГОРОДА НИЖНЕГО НОВГОРОДА

**Муниципальное бюджетное общеобразовательное учреждение
«Школа № 106»**

603132 г. Нижний Новгород, бульвар Заречный, д. 16
тел. (831) 251 88 12, тел./факс (831) 240 00 70
e-mail: s106_nn@mail.52gov.ru

Принята на педагогическом совете
Протокол № 294-о от 30.08.2024

УТВЕРЖДАЮ:
Директор МБОУ «Школа № 106»
_____ С.А.Антипова
«30» августа ____ 2024г.

Дополнительная общеобразовательная
программа – дополнительная общеразвивающая программа

«Основы Python»
техническая направленность

Возраст обучающихся: 14-18лет
Срок реализации: 1 год

Составитель:
Глинов М.М , педагог
дополнительного
образования

г. Нижний Новгород, 2024

Оглавление

Пояснительная записка	3
Новизна и отличительные особенности программы.....	3
Актуальность программы	3
Педагогическая целесообразность	4
Цель и задачи программы	4
Задачи программы	4
Условия реализации программы	5
Планируемые результаты обучения	5
Календарно учебный план	6
Учебно-тематический план	6
Условия реализации программы	9
Оценочные и методические материалы	10
Формы аттестации:	11
Список литературы.....	12

Пояснительная записка

На базе МБОУ Школа №106 обеспечивается реализация образовательных программ технологической направленности, разработанных в соответствии с требованиями законодательства в сфере образования и с учётом рекомендаций Федерального оператора учебного предметов «Информатика» и «Технология».

Программа «Основы языка программирования Python» составлена на основе курса Д.П. Кириенко «Основы языка программирования Python», М.: Бином, 2014 г., Примерной программы внеурочной деятельности начального и основного образования, Стандартов второго поколения в форме кружка в 9-11 классах.

Программа «Основы программирования на языке «Python» направлена на подготовку творческой, технически грамотной, гармонично развитой личности, обладающей логическим мышлением, способной анализировать и решать задачи в команде в области информационных технологий, решать ситуационные кейсовые задания, основанные на групповых проектах. Занятия по данному курсу рассчитаны на общенаучную подготовку обучающихся, развитие их мышления, логики, математических способностей, исследовательских навыков. Курс направлен на изучение основ программирования на языке Python. В рамках курса обучающиеся смогут познакомиться с физическими, техническими и математическими понятиями. Приобретённые знания будут применимы в творческих проектах.

Направленность: техническая.

Новизна и отличительные особенности программы.

Программа представляет особый интерес в связи с широким распространением цифровой техники в обществе, в связи с возрастающей потребностью обучающихся в освоении цифровых технологий и повышением их интереса к современным языкам программирования.

В век информационных технологий практически каждый школьник к 7-8 классу уже владеет навыками работы на персональном компьютере и работы в сети Интернет.

Содержание программы предусматривает ознакомление воспитанников с основами работы в интегрированной среде программирования и основными возможностями языка программирования Python. Программа позволяет освоить практически все операторные конструкции и познакомиться с основными функционального и объектного программирования.

Воспитанники, прошедшие обучение по данной программе, получают знания, умения и навыки, необходимые для сознательного выбора в будущем профессии связанной с программированием. Программа позволяет школьникам познакомиться и понять тонкости профессии программист и профессий связанных с разработкой IT-приложений, оценить себя в этих профессиях, выработать профессионально важные качества.

Актуальность программы

Актуальность курса состоит в том, что он направлен на:

1. формирование целостного мировоззрения, соответствующего современному уровню развития науки и общественной практики за счет развития представлений об информации как важнейшем стратегическом ресурсе развития личности, государства, общества; понимания роли информационных процессов в современном мире;
2. совершенствование общеучебных и общекультурных навыков работы с информацией в процессе систематизации и обобщения имеющихся и получения новых знаний, умений и способов деятельности в области информатики и ИКТ; развитию навыков самостоятельной учебной деятельности школьников (учебного проектирования, моделирования, исследовательской деятельности и т.д.);

3. воспитание ответственного и избирательного отношения к информации с учётом правовых и этических аспектов ее распространения, воспитанию стремления к продолжению образования и созидательной деятельности с применением средств ИКТ.

При реализации данной образовательной программы позволяет создать условия:

1. для расширения содержания школьного образования по информатике;
2. для повышения познавательной активности обучающихся в области программирования;
3. для развития личности ребёнка в процессе обучения информатики и технологии, его способностей, формирования и удовлетворения социально значимых интересов и потребностей;
4. для работы с одаренными школьниками, организации их развития в различных областях образовательной, творческой деятельности.

Педагогическая целесообразность

Педагогическая целесообразность программы заключается в привлечении учащихся к занятиям техническим творчеством, что способствует развитию логического мышления, творческих способностей и навыков решения задач программирования.

Программирование мотивирует к занятиям в различных научных областях (физики, информатики, алгебры, геометрии и др.), развивает воображение и способствует ранней профориентации подростков. Для достижения поставленных задач занятия проводятся в формате «от простого к сложному». Учащиеся вспоминают свои знания по основам алгоритмизации и программирования и на их основе, углубляя их, учатся составлять простые и сложные программы.

Практическая значимость

В рамках предлагаемого курса «Программирование на языке Python» изучение основ программирования на языке Python – это не столько средство подготовки к будущей профессиональной деятельности, сколько формирование новых общеинтеллектуальных умений и навыков: разделение задачи на этапы решения, построение алгоритма и др.

Особую роль программирование служит для формирования мыслительных и психических процессов учащихся (внимание, память, логика), освоения приёмов умственных действий, самостоятельного нахождения и составления алгоритмов решения задач, умения строить модели, чётко и лаконично реализовывать этапы решения задач. Использование этих возможностей для формирования предметных и метапредметных результатов учащихся особенно важно, т.к. именно они активизирует процесс индивидуально-личностного становления учащихся.

Цель и задачи программы

Цель изучения курса: формирование интереса обучающихся к изучению профессий, связанных с основами программирования через освоение языка Python.

Задачи программы

обучающие:

1. освоение основных алгоритмических конструкций;
2. обучение основам алгоритмизации и программирования;
3. освоение первоначальных навыков программирования на языке программирования высокого уровня Python;
4. приобщение к проектно-творческой деятельности;

воспитывающие:

1. воспитание интереса к информационной и коммуникационной сфере человеческой деятельности;

2. воспитание потребности соблюдать этические и правовые нормы работы с информацией;
3. воспитание бережного отношения к техническим устройствам;

развивающая:

1. развитие творческого воображения, алгоритмического мышления учащихся;
2. развитие навыков планирования проекта, умения работать в группе;
3. развитие навыков ориентации в информационных потоках окружающего мира и применения точной и понятной инструкции для решения учебных задач и в повседневной жизни.

Условия реализации программы

Программа «Основы Python» состоит из модулей (36 часов в год). Общее количество учебных часов – 36.

Занятия с обучающимися будут проходить в МБОУ Школа №106

Возраст детей, участвующих в реализации данной дополнительной общеобразовательной общеразвивающей программы определён в соответствии с возрастными ограничениями игр, используемыми при реализации программы - 14-18 лет.

Уровень сложности: стартовый.

Форма получения образования – очная, возможно обучение в дистанционном формате с использованием ИКТ на различных платформах.

Форма занятий - групповая, индивидуальная (группа может делиться на подгруппы при проведении практических работ).

Режим занятий: 1 раз в неделю 1 учебный час – 45мин.

Срок реализации программы 1 года.

Пол обучающихся – смешанный.

Принцип набора – свободный, количество человек в группе 10-15 человек.

Виды деятельности по программе: занятия теоретические и практические.

Система отслеживания и оценивания результатов: тестирование, опрос, самостоятельные и контрольные работы, презентация (выставка, состязание, конкурс).

Планируемые результаты обучения

Основным результатом обучения является формирование вектора развития обучающихся с упором на формирование у них алгоритмического мышления.

Личностные результаты:

1. сформированность мировоззрения, соответствующего современному уровню развития науки и техники;
2. осознанный выбор будущей профессии и возможностей реализации собственных жизненных планов;
3. сформированность представлений о мире профессий, связанных с программированием, и требованиях, предъявляемых различными востребованными профессиями, такими как программист, системный администратор
4. навыки сотрудничества в образовательной, учебно-исследовательской, проектной и других видах деятельности
5. навыки взаимо и самооценки, навыки рефлексии.

Метапредметные результаты:

1. владение навыками познавательной, учебно-исследовательской и проектной деятельности, навыками разрешения проблем;
2. способность и готовность к самостоятельному поиску методов решения практических задач, применению различных методов познания.

Предметные результаты:

1. навыки алгоритмического мышления и понимание необходимости формального описания алгоритмов;
2. владение стандартными приёмами написания программы для решения стандартной задачи с использованием основных конструкций программирования и отладки таких программ на языке Python;
3. знание особенностей структуры программы, представленной на языке Python
4. представление о модулях, входящих в состав среды Python
5. возможности и ограничения использования готовых модулей,
6. представление о величине, ее характеристиках
7. знание что такое операция, операнд и их характеристики,
8. знание принципиальные отличия величин, структурированных и не структурированных,
9. знание принципиальные отличия величин, структурированных и не структурированных, представление о составе арифметического выражения;
10. знание математических функций, входящих в Python, представление о логических выражениях и входящих в них операндах, операциях и функциях

Кроме этого, занятия по программе будут способствовать развитию интереса к технике, конструированию, программированию, высоким технологиям, формировать навыки коллективного труда.

Календарно учебный план

Год обучения	Дата начала обучения по программе	Дата окончания обучения	Всего учебных недель	Всего учебных недель	Режим занятий
1 год	01.09.2022	31.05.2023	36	36	В соответствии с расписанием

Учебно-тематический план

№	Раздел программы	Количество часов			Формы контроля
		теория	практика	всего	
1	Знакомство с языком Python	0,5	0,5	1	Обсуждение, Тестирование
2	Переменные и выражения	1	3	4	Обсуждение, Тестирование
3	Условные операторы	1	4	5	Обсуждение, Тестирование
4	Циклы	3	4	7	Обсуждение, Тестирование
5	Функции	3	5	8	Обсуждение, Тестирование
6	Строки	3	5	8	Обсуждение, Тестирование
7	Итоговое тестирование по курсу	0	3	3	Итоговый проект
	ИТОГО	11,5	24,5	36	

Содержание программы

Тема 1. Знакомство с языком Python (1 ч.)

Общие сведения о языке Python. Установка Python на компьютер. Режимы работы Python. Что такое программа. Первая программа. Структура программ на языке Python. Комментарии.

-Практическая работа 1.1. Установка программы Python.

-Практическая работа 1.2. Режимы работы с Python 8

Учащиеся должны знать / понимать: понятие программы; структура программы на Python; режимы работы с Python.

Учащиеся должны уметь: выполнить установку программы; выполнить простейшую программу в интерактивной среде; написать комментарии в программе.

Тема 2. Переменные и выражения (4 ч.)

Типы данных. Преобразование типов. Переменные. Оператор присваивания. Имена переменных и ключевые слова. Выражения. Операции. Порядок выполнения операций. Математические функции. Композиция. Ввод и вывод. Ввод данных с клавиатуры. Вывод данных на экран. Пример скрипта, использующего ввод и вывод данных. Задачи на элементарные действия с числами. Решение задач на элементарные действия с числами.

Практическая работа 2.1. Переменные

Практическая работа 2.2. Выражения

Практическая работа 2.3. Задачи на элементарные действия с числами

Учащиеся должны знать / понимать: общую структуру программы; типы данных; целые, вещественные типы данных и операции над ними; оператор присваивания; операторы ввода-вывода.

Учащиеся должны уметь: пользоваться интерфейсом среды программирования Python; использовать команды редактора; организовывать ввод и вывод данных; записывать арифметические выражения.

Тема 3. Условные операторы (5 ч.)

Логический тип данных. Логические выражения и операторы. Сложные условные выражения (логические операции and, or, not). Условный оператор. Примеры решения задач с условным оператором. Множественное ветвление. Реализация ветвления в языке Python.

Практическая работа 3.1. Логические выражения

Практическая работа 3.2. "Условный оператор"

Практическая работа 3.3. Множественное ветвление

Самостоятельная работа 2. Решение задач по теме "Условные операторы".

Учащиеся должны знать / понимать: назначение условного оператора; способ записи условного оператора; логический тип данных; логические операторы or, and, not;

Учащиеся должны уметь: использовать условный оператор; создавать сложные условия с помощью логических операторов.

Тема 4. Циклы (7 ч.)

Понятие цикла. Тело цикла. Условия выполнения тела цикла. Оператор цикла с условием. Оператор цикла while. Бесконечные циклы. Альтернативная ветка цикла while. Обновление переменной. Краткая форма записи обновления. Примеры использования циклов. Оператор цикла с параметром for. Операторы управления циклом. Пример задачи с использованием цикла for. Вложенные циклы. Циклы в циклах. Случайные числа. Функция randrange. Функция random. Примеры решения задач с циклом.

Практическая работа 4.1. "Числа Фибоначчи"

Практическая работа 4.2. Решение задачи с циклом for.

Практическая работа 4.3. Реализация циклических алгоритмов

Практическая работа 4.4. Случайные числа

Практическая работа 4.5. Решение задач с циклом.

Самостоятельная работа 3 по теме "Циклы"

Учащиеся должны знать / понимать: циклы с условием и их виды; правила записи циклов условием; назначение и особенности использования цикла с параметром; формат записи цикла с параметром; примеры использования циклов различных типов.

Учащиеся должны уметь: определять вид цикла, наиболее удобный для решения поставленной задачи; использовать цикл с условием; определять целесообразность применения и использовать цикл с параметром для решения поставленной задачи;

Тема 5. Функции (8 ч.) Создание функций. Параметры и аргументы. Локальные и глобальные переменные. Поток выполнения. Функции, возвращающие результат. Анонимные функции, инструкция lambda. Примеры решения задач с использованием функций. Рекурсивные функции. Вычисление факториала. Числа Фибоначчи.

Практическая работа 5.1. Создание функций

Практическая работа 5.2. Решение задач с использованием функций

Учащиеся должны знать / понимать:

понятие функции; способы описания функции; принципы структурного программирования; понятие локальных переменных подпрограмм; понятие формальных и фактических параметров подпрограмм; способ передачи параметров.

Учащиеся должны уметь: создавать и использовать функции; использовать механизм параметров для передачи значений.

Тема 6. Строки (8 ч.) Составной тип данных - строка. Доступ по индексу. Длина строки и отрицательные индексы. Преобразование типов. Применение цикла для обхода строки. Срезы строк. Сравнение строк. Оператор in. Модуль string. Операторы для всех типов последовательностей (строки, списки, кортежи). Примеры решения задач со строками.

Практическая работа 6.1. Строки

Практическая работа 6.2. Решение задач со строками.

Учащиеся должны знать / понимать: назначение строкового типа данных; операторы для работы со строками; процедуры и функции для работы со строками; операции со строками.

Учащиеся должны уметь: описывать строки; соединять строки; находить длину строки; вырезать часть строки; находить подстроку в строке; находить количество слов в строке.

Тема 7. Итоговое тестирование по курсу (3 ч.)

Календарно учебный план

№ п/п	Содержание занятия	Планируемая дата	Фактическая дата
1	Общие сведения о языке Python. Практическая работа 1.1. Установка программы Python.	03.09.2022	
2	Переменные. Практическая работа 2.1. Переменные	10.09.2022	
3	Выражения. Ввод и вывод	17.09.2022	
4	Практическая работа 2.2. Выражения	24.09.2022	
5	Практическая работа 2.3. Задачи на элементарные действия с числами.	01.10.2022	
6	Логические выражения и операторы. Практическая работа 3.1. Логические выражения	08.10.2022	
7	Условный оператор. Множественное ветвление.	15.10.2022	
8	Практическая работа 3.2. "Условный оператор"	22.10.2022	
9	Практическая работа 3.3. Множественное ветвление	29.10.2022	

10	Самостоятельная работа 1. Решение задач по теме "Условные операторы".	12.11.2022	
11	Оператор цикла с условием. Практическая работа 4.1. "Числа Фибоначчи"	19.11.2022	
12	Оператор цикла for.	26.11.2022	
13	Практическая работа 4.2. Решение задачи с циклом for.	03.12.2022	
14	Вложенные циклы. Случайные числа. Практическая работа 4.3. Реализация циклических алгоритмов	10.12.2022	
15	Практическая работа 4.4. Случайные числа.	17.12.2022	
16	Практическая работа 4.5. Решение задач с циклом	24.12.2022	
17	Практическая работа 4.5. Решение задач с циклом	14.01.2023	
18	Создание функций. Локальные переменные	21.01.2023	
19	Практическая работа 5.1. Создание функций	28.01.2023	
20	Практическая работа 5.1. Создание функций	04.02.2023	
21	Практическая работа 5.2. Решение задач с использованием функций	11.02.2023	
22	Рекурсивные функции	18.02.2023	
23	Рекурсивные функции	25.02.2023	
24	Практическая работа 5.3. Рекурсивные функции	04.03.2023	
25	Практическая работа 5.3. Рекурсивные функции	11.03.2023	
26	Строки	18.03.2023	
27	Строки	25.03.2023	
28	Практическая работа 6.1. Строки	01.04.2023	
29	Практическая работа 6.1. Строки	08.04.2023	
30	Срезы строк	15.04.2023	
31	Срезы строк	22.04.2023	
32	Практическая работа 6.2. Решение задач со строками.	29.01.2023	
33	Практическая работа 6.2. Решение задач со строками.	06.05.2023	
34	Итоговый тест по курсу «Основы языка программирования Python»	13.05.2023	
35	Итоговый тест по курсу «Основы языка программирования Python»	20.05.2023	
36	Итоговый тест по курсу «Основы языка программирования Python»	27.05.2023	

Условия реализации программы

Материально-техническое обеспечение образовательного процесса:

-12–15 компьютеров (рабочих мест) для обучающихся и один компьютер (рабочего места) для педагога.

-компьютеры объединены в локальную сеть с возможностью выхода в Интернет.

Минимальные требования к техническим характеристикам каждого компьютера следующие:

-процессор – с тактовой частотой 2 ГГц;

-оперативная память – не менее 1 Гб;

-жидкокристаллический монитор с диагональю не менее 15 дюймов;

-жёсткий диск – не менее 250 Гб;

-клавиатура;

-мышь;

-аудио-карта и акустическая система (наушники или колонки).

-Мультимедийный проектор на рабочем месте учителя;

Требования к программному обеспечению компьютеров:
-операционная система Windows
-текстовый редактор (Блокнот) и текстовый процессор (MS Word или OpenOffice Writer);
-табличный процессор (MS Excel или OpenOffice Calc);
-среда программирования Python 3+

Оценочные и методические материалы

Для отслеживания результативности образовательной деятельности по программе проводятся: входной, текущий и промежуточный контроль.

Входная диагностика: проводится в сентябре с целью выявления первоначального уровня знаний, умений и навыков, возможностей детей и определения их природных качеств.

Формы: наблюдение, тестирование;

Диагностика физиологических качеств проводится с каждым обучающимся по решению умственно-логических задач.

Текущий контроль: отслеживается на занятиях в течение всего учебного года для отслеживания уровня освоения учебного материала программы и развития личностных качеств обучающегося.

Формы:

- устный опрос;
- педагогическое наблюдение;
- опрос на выявление умения рассказать правила выполнения работы, теоретические знания для выполнения поставленных задач;
- выполнение тестовых заданий на знание терминологии;
- анализ педагогом и обучающимся качества выполнения упражнений.

Промежуточный контроль: предусмотрен с целью выявления уровня освоения программы обучающимися и корректировки процесса обучения.

Диагностика уровня личностного развития обучающихся проводится по следующим параметрам:

- культура речи,
- умение слушать,
- умение выделить главное,
- умение планировать свои действия,
- умение ставить задачи,
- самоконтроль,
- воля,
- выдержка,
- самооценка,
- мотивация,
- социальная адаптация.

Методические материалы

1. Используемые методики, методы и технологии
2. Дидактические средства
3. Информационные источники

При обучении используются основные методы организации и осуществления учебно-познавательной работы, такие как словесные, наглядные, практические, индуктивные и проблемно-поисковые. Выбор методов (способов) обучения зависит от психофизиологических, возрастных особенностей детей, темы и формы занятий. При этом в процессе обучения все методы реализуются в теснейшей взаимосвязи.

Методика проведения занятий предполагает постоянное создание ситуаций успешности, радости от преодоления трудностей в освоении изучаемого материала. С первых занятий обучающиеся знакомятся с охраной труда на занятиях, противопожарной безопасностью, к правильной организации тренировочного процесса, рациональному использованию рабочего времени, грамотному использованию инвентаря. В основу программы легли определенные педагогические принципы:

-принцип дополнительности. Монолог педагога уступает место смысловому диалогу, взаимодействию, партнерству, ориентация на реальную свободу развивающейся личности.

-принцип открытости учебной и воспитательной информации. Мир знаний "открывается" перед обучающимися благодаря работе его сознания, как главной личной ценности. Педагог не "преподносит" знания в готовом для понимания виде, а придает им контекст открытия.

-принцип уважения к личности ребенка в сочетании с разумной требовательностью к нему предполагает, что требовательность является своеобразной мерой уважения к личности ребенка. Разумная требовательность всегда целесообразна если продиктована потребностями воспитательного процесса и задачами развития личности.

-принцип сознательности и активности обучающихся предполагает создание условий для активного и сознательного отношения к обучению, условий для осознания обучающимися правильности и практической ценности получаемых знаний, умений и навыков.

-принцип дифференцированного и индивидуального подхода в обучении предполагает необходимость учета индивидуальных возможностей и возрастных психофизиологических особенностей каждого обучающегося при выборе темпа, методов и способа обучения.

-принцип преемственности, последовательности и систематичности заключается в такой организации обучающегося процесса, при которой каждое занятие является логическим продолжением ранее проводившейся работы, позволяет закреплять и развивать достигнутое, поднимать обучающегося на более высокий уровень развития.

-принцип доступности и пассивности заключается в применении основного правила дидактики "от простого к сложному, от известного к неизвестному".

Основные технологии используемые при реализации программы:

- здоровье сберегающие;
- игровые технологии;
- технологии коллективного творчества.

Формы аттестации:

Предметом диагностики и контроля в курсе «Основы программирования на языке Python» являются внешние образовательные продукты учащихся (созданные программы), а также их внутренние личностные качества (освоенные способы деятельности, знания, умения), которые относятся к целям и задачам курса.

Практическим результатом обучения воспитанников является также самостоятельно выполненный проект, включающий в себя не только компьютерную программу, но и описание алгоритма её работы, контрольные тесты для её проверки, описание используемых языковых конструкций и оригинальных приёмов программирования, область практического применения и другую документацию, отражающую работу над созданием проекта.

Темы проектов выбираются учащимися как самостоятельно, так и из числа предложенных преподавателем, но всегда обсуждается в группе. Необходимо чтобы программа имела практическое применение и была интересна не только авторам проекта. Воспитанник должен почувствовать важность выбранной им темы в глазах общества, желание показать своё мастерство и уникальность технологического решения созданной им программы.

Обеспечивается возможность презентации проекта на школьной научно-практической конференции, конкурсе или фестивале проектов.

Примерные требования при защите проекта Учащиеся должны (по желанию) продемонстрировать самостоятельно разработанную программу, показать владение техническими, программными средствами разработки проекта, знание основ программирования, уметь ответить на вопросы по содержанию.

По техническим средствам:

- какие технические средства были задействованы при создании проекта;
- объяснить минимальные требования к программному и аппаратному обеспечению для работы программы.

По технологии разработки проект:

- охарактеризовать программные средства, используемые при выполнении проекта;
- объяснить технологические приемы разработки;
- наличие документации; - рекомендации по работе с программой.

По разработке программы:

- алгоритм должен быть оптимальным по скорости выполнения и максимально простым в реализации на языке программирования;
- код должен быть «читаемым» (должны быть соблюдены отступы, обязательное наличие комментариев к коду программы и т. д.);
- программа должна выполнять поставленные задачи, и правильно работать; - наличие собственных конструктивных решений;
- знание основ программирования и знание технология программирования;
- наличие дружественного интерфейса.

По содержанию проекта:

- актуальность работы,
- глубина и логика раскрытия темы,
- форма подачи информации,
- заинтересованность в проведении исследования,
- области применения разработки, - источники информации.

Список литературы

1. Домашняя страница Python www.python.org . Справочные материалы, официальная документация.
2. Сайт проекта Интуит: Национальный открытый университет, курс «Введение в программирование на Python», <http://www.intuit.ru/studies/courses/12179/1172/info> .
3. Сайт проекта Интуит: Национальный открытый университет. Курс «Язык программирования Python» <http://www.intuit.ru/studies/courses/49/49/info> .
4. Python. Подробный справочник Дэвида М. Бизли — книга со справочной информацией о языке Python и модулях стандартной библиотеки