

УДАЯН ДАС, ОБРИ ЛОУСОН  
КРИС МЭЙФИЛД, НАРГЕС НОРУЗИ

начинаем программировать на

**PYTHON**

с примерами и заданиями

Издательство АСТ  
г. Москва

УДК 004.43  
ББК 32.973.2  
Д20

*Содержание книги «Introduction to Python Programming» на английском языке, подготовленной компанией OpenStax, которая является некоммерческой инициативой Университета Райса в области образовательных технологий, распространяется на условиях лицензии*

*Creative Commons Attribution 4.0 International License (CC BY 4.0).*

*Свободный доступ к этой книге в веб-формате или формате PDF*

*можно получить на сайте OpenStax.org по ссылке:*

*<https://openstax.org/details/books/introduction-python-programming>.*

*В книге используется программа OpenStax Python Code Runner. Программа запуска кода разработана компанией Wiley, и все права на нее защищены.*

### **Дас, Удаян.**

Д20 Начинаем программировать на Python. С примерами и заданиями / У. Дас, О. Лоусон, К. Мэйфилд, Н. Норузи. — Москва : Издательство АСТ, 2025. — 416 с. : ил. — (Быстрый старт в программирование).

ISBN 978-5-17-178636-6.

Эта книга — универсальное руководство для начинающих разработчиков программного обеспечения, желающих освоить популярный язык программирования Python с нуля. Здесь представлено все самое необходимое для уверенного старта: от основ синтаксиса, логики кода до работы с данными и создания собственных программ. Благодаря продуманной структуре подачи материала и ясному языку объяснения читатели смогут сразу же с головой погрузиться в программирование, даже не имея технического образования. В издании используется современный подход к обучению: оно не перегружено теорией, в нем делается упор на понимание через действие.

Важной особенностью данного пособия является акцент на практике: каждый раздел сопровождается интерактивными заданиями, обучающими вопросами и контрольными точками, которые помогают закрепить усвоенный материал. После изучения теории вы сразу сможете применить знания «в боевых условиях» благодаря интерактивным ссылкам на интегрированный браузерный компилятор, который позволяет писать, запускать и отлаживать код без необходимости устанавливать дополнительное программное обеспечение.

Книга подойдет старшим школьникам, студентам и всем, кто хочет начать собственный путь в программировании, опираясь на надежную основу. После ее изучения вы не только поймете, как именно работает Python, но и научитесь уверенно использовать его для решения практических задач.

**УДК 004.43**  
**ББК 32.973.202-018.2**

**ISBN 978-5-17-178636-6**

© 2024 Rice University

Перевод на русский язык: ООО «Интеджер».

Издание на русском языке: ООО «Издательство АСТ».

# Содержание

Предисловие.....	7
<b>1. Операторы .....</b>	<b>13</b>
1.1. Подготовка.....	14
1.2. Ввод/вывод .....	18
1.3. Переменные .....	22
1.4. Основы работы со строками.....	25
1.5. Основы работы с числами.....	29
1.6. Сообщения об ошибках .....	33
1.7. Комментарии .....	37
1.8. Почему именно Python? .....	41
1.9. Итоги главы .....	44
<b>2. Выражения .....</b>	<b>47</b>
2.1. Python shell .....	48
2.2. Преобразование типов .....	52
2.3. Смешанные типы данных.....	55
2.4. Ошибки чисел с плавающей точкой.....	59
2.5. Деление целых чисел.....	63
2.6. Модуль math.....	67
2.7. Форматирование кода.....	72
2.8. Карьера в области Python .....	77
2.9. Итоги главы .....	79
<b>3. Объекты.....</b>	<b>82</b>
3.1. Строки: повторение .....	83
3.2. Форматированные строки.....	88
3.3. Переменные: повторение.....	92
3.4. Основы работы со списками .....	96
3.5. Основы работы с кортежами.....	99
3.6. Итоги главы .....	102
<b>4. Решения.....</b>	<b>103</b>
4.1. Логические значения (булевы значения).....	104
4.2. Условные операторы if-else.....	109
4.3. Логические операторы .....	113
4.4. Приоритет операторов .....	117
4.5. Цепочки решений .....	121
4.6. Вложенные решения.....	127
4.7. Условные выражения .....	130
4.8. Итоги главы .....	132
<b>5. Циклы .....</b>	<b>134</b>
5.1. Цикл while .....	135
5.2. Цикл for .....	139

5.3. Вложенные циклы .....	143
5.4. Операторы break и continue.....	148
5.5. Блок else в циклах.....	152
5.6. Итоги главы .....	155
<b>6. Функции .....</b>	<b>157</b>
6.1. Определение функций.....	158
6.2. Поток выполнения.....	162
6.3. Область действия переменных .....	166
6.4. Параметры.....	172
6.5. Возвращаемые значения .....	177
6.6. Именованные аргументы.....	182
6.7. Итоги главы .....	185
<b>7. Модули.....</b>	<b>187</b>
7.1. Основы модулей .....	188
7.2. Имена импортов .....	192
7.3. Код верхнего уровня.....	195
7.4. Функция help().....	199
7.5. Поиск модулей .....	204
7.6. Итоги главы .....	209
<b>8. Строки.....</b>	<b>210</b>
8.1. Операции со строками.....	211
8.2. Срезы строк .....	214
8.3. Поиск и проверка строк .....	218
8.4. Форматирование строк.....	224
8.5. Разделение и объединение строк.....	232
8.6. Итоги главы .....	235
<b>9. Списки.....</b>	<b>236</b>
9.1. Изменение списков. Итерации по списку .....	237
9.2. Сортировка и реверсирование списков.....	240
9.3. Общие операции со списками .....	243
9.4. Вложенные списки.....	246
9.5. Генераторы списков.....	250
9.6. Итоги главы .....	254
<b>10. Словари.....</b>	<b>255</b>
10.1. Основы словарей .....	256
10.2. Создание словарей .....	258
10.3. Операции со словарями.....	261
10.4. Условные операторы и циклы в словарях.....	266
10.5. Вложенные словари и генераторы словарей .....	271
10.6. Итоги главы .....	275
<b>11. Классы.....</b>	<b>278</b>
11.1. Основы объектно-ориентированного программирования.....	279
11.2. Классы и экземпляры .....	282
11.3. Методы экземпляра .....	287
11.4. Перегрузка операторов.....	291

11.5. Использование модулей с классами .....	296
11.6. Итоги главы .....	299
<b>12. Рекурсия .....</b>	<b>301</b>
12.1. Основы рекурсии .....	302
12.2. Простая математическая рекурсия .....	304
12.3. Рекурсия со строками и списками .....	307
12.4. Подробнее про математическую рекурсию .....	310
12.5. Использование рекурсии для решения задач .....	314
12.6. Итоги главы .....	318
<b>13. Наследование .....</b>	<b>319</b>
13.1. Основы наследования .....	320
13.2. Доступ к атрибутам .....	324
13.3. Методы .....	328
13.4. Иерархическое наследование .....	334
13.5. Множественное наследование и mix-in-классы .....	338
13.6. Итоги главы .....	342
<b>14. Файлы .....</b>	<b>344</b>
14.1. Чтение из файлов .....	345
14.2. Запись в файлы .....	349
14.3. Файлы в разных расположениях и работа с CSV-файлами .....	353
14.4. Обработка исключений .....	357
14.5. Вызов исключений .....	362
14.6. Итоги главы .....	365
<b>15. Data Science .....</b>	<b>367</b>
15.1. Введение в data science .....	368
15.2. NumPy .....	371
15.3. Pandas .....	374
15.4. Исследовательский анализ данных .....	380
15.5. Визуализация данных .....	386
15.6. Итоги главы .....	392
<b>Ответы .....</b>	<b>395</b>



# ПРЕДИСЛОВИЕ

## О КНИГЕ

Эта книга предоставляет комплексный материал для изучения понятий и обретения навыков программирования, который по объему соответствует большинству вводных курсов. Широкий спектр сценариев, контекстов и задач отражает применение программирования в различных дисциплинах и профессиях. Данный материал подходит для разной аудитории, включая тех, кто изучает информатику, бизнес, естественные и социальные науки, статистику, науку о данных и смежные области обучения и деятельности.

Данное издание — это интерактивное пособие, которое обучает основным понятиям программирования, навыкам решения задач и с помощью практических заданий помогает изучить один из самых востребованных языков программирования на данный момент — Python. Вам не нужно будет скачивать различные сторонние программы для проверки вашего кода: вы получите доступ к уникальному интегрированному компилятору, благодаря которому сможете сразу применить полученные знания на практике. Ссылки на видеоролики, упражнения для развития критического мышления, предоставляемые в начале каждого раздела, изучение сторонних инструментов и видов деятельности в области программирования — все это способствует осмысленному и продуктивному обучению.

Содержание книги разделено на главы, каждая из которых содержит 6–8 разделов. Каждый раздел построен по следующей схеме:

- Цели обучения
- 1–3 подраздела, содержащих поясняющую информацию
- Практические задания

Первый подраздел предназначен для того, чтобы определить, какие задачи будут решаться в данном разделе. Весь дальнейший материал будет подытожен в одном предложении: «После изучения этого раздела вы должны уметь». Практические задания полностью соответствуют основным целям обучения конкретного раздела и дают возможность применить изученный материал на практике.

## СТРУКТУРА ОБУЧЕНИЯ

### Краткий текст и видеoaнимации

Данная книга разработана для интерактивного обучения. В ней сделан акцент на интерактивности и практике путем предоставления доступа к компилятору кода, видеороликам и ссылкам на внешние источники. Благодаря этому материал часто более лаконичен, чем в классических учебниках по программированию. Содержание каждого раздела разбито на подразделы. Подраздел начинается с небольшого введения в тему с определениями основных терминов и кратким объяснением их актуальности и важности. Затем данная информация подробно объясняется с помощью видеoaнимации и интерактивных обучающих вопросов.

Анимационные видеоролики используют пошаговый подход для демонстрации выполнения кода на языке Python. Пояснения к каждому шагу помогают

понять синтаксис Python. Они показывают, как используются основные понятия программирования, демонстрируют процесс присвоения значений переменным, подчеркивают построчное выполнение кода, применяют решение задач для создания программ и многое другое.

## ЧЕКПОИНТ

### Отображение вывода для пользователя

После прочтения новой информации и просмотра объяснения с помощью видеоанимации в каждом подразделе вы также найдете задания в виде обучающих вопросов. Данные вопросы помогают более детально разобрать и закрепить изученный материал, они непосредственно разбирают основные ошибки, которые обычно допускают начинающие программисты, а также дополнительно объясняют некоторые смежные (схожие) понятия.

В случае неправильного ответа будут даны пояснения, которые укажут, почему данный ответ неверен, а также помогут разобраться в вопросе и наведут читателя на правильный ответ.

Неправильные варианты ответов, как правило, являются следствием некорректного понимания полученной информации или результатом распространенной ошибки. Даже если вы дали правильный ответ, рекомендуется изучить пояснения, чтобы ознакомиться с распространенными заблуждениями и лучше закрепить усвоенный материал.



Доступ к мультимедийному контенту  
(<https://www.openstax.org/r/displaying-output>).

## Практические упражнения по программированию

В конце каждого раздела есть 1–2 практических упражнения. В книгу включены ссылки на интегрированную среду программирования, известную как OpenStax Python Code Runner, которая позволяет писать программы непосредственно в браузере. Компилятор кода требует предварительного ввода данных перед запуском программы. Пример выполнения кода в интегрированной среде OpenStax Python Code Runner:

**TRY IT**

**Favorite song**

The program below asks for your name and displays a friendly greeting. Run the program and see what happens. In the error message, EOF stands for End of File.

- Many of the programs in this chapter expect input from the user. Enter your name in the input box below the code. Run the program again, and see what changes.
- Copy the following lines to the end of the program: `print("What is your favorite song?")`  
`song = input()` `print("Cool! I like", song, "too.")`
- The modified program reads two lines of input: name and song. Add your favorite song to the input box below your name, and run the program again.

The next section of the book will explain how `print()` and `input()` work. Feel free to experiment with this code until you are ready to move on.

```
1 print("What is your name?")
2 name = input()
3 print("Nice to meet you, " + name + ".")
4
5
```

input

Run Reset all

Output

## Условные обозначения, используемые в книге

В книге используются следующие типографские обозначения:

<b>Полужирный</b>	Указывает на словарные слова, впервые встречающиеся в главе.
<i>Курсив</i>	Указывает на подчеркнутый текст, имена файлов и расширения файлов.
Моноширинный шрифт	Используется для списков элементов кода внутри абзацев. Элементы кода включают в себя имена переменных, ключевые слова Python и т. д.
<b>Полужирный моноширинный шрифт</b>	Показывает команды или клавиатурный ввод, которые пользователь должен вводить дословно.
Пр:	Сокращение для «Пример:».

## Об авторах

### Старшие авторы-составители



Старшие авторы-составители (слева направо): Удаян Дас, Обри Лоусон, Крис Мэйфилд и Наргес Норузи.

#### **Удаян Дас, колледж Святой Марии в Калифорнии**

Удаян Дас, доктор философии, является доцентом и директором программы обучения по компьютерной науке в калифорнийском колледже Святой Марии. Получил докторскую степень в области информатики и степень магистра в сфере компьютерной инженерии в Илинойском технологическом институте. В круг его научных интересов входят беспроводные сети, образование в области компьютерных наук и расширение участия людей в программировании, а также языковые модели на основе графов данных для обработки технической документации. Он также является сторонником идеи включения этики в процесс обучения в области компьютерных наук и инженерии. Программа по компьютерным наукам, которую он разработал и запустил в колледже Святой Марии в Калифорнии, акцентирует внимание на этике и социальной справедливости, готовит высококвалифицированных специалистов в области информатики.

**Обри Лоусон, компания Wiley**

Обри Лоусон — разработчик контента по компьютерным наукам в компании zyBooks. Получила степени бакалавра и магистра в области информатики в Университете Клемсона, ее докторская диссертация посвящена образованию в области компьютерных наук.

**Крис Мэйфилд, Университет Джеймса Мэдисона**

Крис Мэйфилд, доктор философии, является профессором компьютерных наук в Университете Джеймса Мэдисона. Его исследования посвящены образованию и профессиональному развитию в области информатики на уровне бакалавриата и средней школы. Получил докторскую степень по компьютерным наукам в Университете Пердью и степень бакалавра по информатике и немецкому языку в Университете Юты.

**Наргес Норузи, Калифорнийский университет в Беркли**

Наргес Норузи получила степень магистра и доктора философии в Университете Торонто, специализируясь на прикладном обучении. Работает над проектами в области прикладного машинного обучения с упором на биологию и образование. Ее исследования в области обучения компьютерным наукам сосредоточены на использовании искусственного интеллекта в классе для устранения разрыва в неравенстве уровня образования и руководстве программами, которые способствуют равному доступу к знаниям для любых студентов.

**Авторы-составители**

Авторы-составители (слева направо): Ямуна Раджасекхар и Рид Канемару.

**Ямуна Раджасекхар, компания Wiley**

Ямуна Раджасекхар, доктор философии, является директором по контенту, авторским разработкам и исследованиям в компании Wiley. Работает в различных дисциплинах над стратегией исследований, авторской педагогикой и обучением, а также разработкой контента для компьютерных наук и ИТ. Получила степень магистра и доктора философии в Университете Северной каролины в Шарлотте, специализируясь на обучении компьютерной инженерии. До прихода в компанию Wiley в качестве автора контента Ямуна была доцентом кафедры компьютерной инженерии в Университете Майами, где ее исследования были сосредоточены на вспомогательных технологиях со встроенными системами, также принимала непосредственное участие в образовании в области компьютерной инженерии.

**Рид Канемару, компания Wiley**

Рид Канемару получил степень бакалавра в области компьютерных наук в Калифорнийском университете в Риверсайде в 2020 году и степень магистра

в области компьютерных наук в Калифорнийском университете в Риверсайде в 2021 году. После окончания университета работает разработчиком контента/программного обеспечения в компании zuBooks.

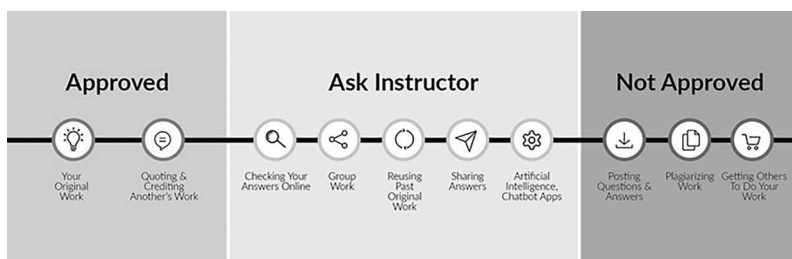
## Рецензенты

Мел Ахимиомона, Общественный колледж округа Балтимор  
 Дойна Бейн, Калифорнийский государственный университет, Фуллертон  
 Филипп Брэдфорд, Университет Коннектикута  
 Джеймс Браман, Общественный колледж округа Балтимор  
 Роберт Барроуз, Колледж ДюПейдж  
 Дина Энгель, Нью-Йоркский университет  
 Габриэль Феррер, колледж Хендрикса  
 Назли Харди, Университет Миллерсвилля  
 Мэтью Херц, Университет Буффало  
 Рания Ходход, Университет штата Колумбус  
 Акира Кавагути, Городской колледж Нью-Йорка  
 Кевин Лин, Университет Вашингтона  
 Матин Пируз, Фресно Стэйт  
 Мухаммед Рахман, Государственный университет Клейтона  
 Джерри Рид, Колледж Валенсии  
 Кэтрин Тамерлано, Общественный колледж Куяхога  
 Линда Тансил

## Академическая честность

Академическая честность способствует доверию, пониманию, справедливости и качественному обучению. Хотя студенты могут столкнуться с серьезными трудностями как на своих занятиях, так и в личной жизни, ответственный и добросовестный подход к выполнению собственной работы, а также сохранение высокого уровня ее подлинности приведет к значимым результатам, которые выйдут далеко за пределы их обучения в колледже. Преподаватели, администраторы, компании-поставщики образовательных ресурсов и студенты должны работать вместе, чтобы получать положительный опыт от взаимодействия, что на выходе будет приносить только качественный результат.

Мы понимаем, что студенты обучаются наиболее продуктивно, когда правила академической честности устанавливаются в самом начале курса. В связи с этим OpenStax создал интерактивный инструмент, который поможет обсудить академическую честность в вашем курсе.



Авторство: Copyright Rice University, OpenStax, по лицензии CC BY 4.0.

Посетите наш слайдер академической честности (<https://www.openstax.org/r/academic-integrity-slider>). Щелкните и перетащите иконки в нужные области слайдера, чтобы привести эти методы в соответствие с политикой вашего учебного заведения и курса. Затем вы можете включить этот график в ваш учебный план, представить его на первом занятии курса или создать памятку для студентов.

В OpenStax мы также разрабатываем ресурсы для поддержки качественного опыта обучения и оценки. Пожалуйста, посетите страницу этого учебника для получения обновлений. Для более глубокого изучения стратегий академической честности мы настоятельно рекомендуем посетить сайт Международного центра академической честности (ICAI) <https://academicintegrity.org>.

## Платформы объединения сообществ

Компания OpenStax сотрудничает с Институтом изучения управления знаниями в образовании (ISKME), чтобы предложить общественные хабы на OER Commons — платформе для преподавателей, позволяющей бесплатно обмениваться созданными сообществом ресурсами, которые поддерживают книги OpenStax. Через наши общественные хабы преподаватели могут загружать свои собственные материалы или скачивать ресурсы для использования в своих курсах, включая дополнительные вспомогательные материалы, учебные пособия, мультимедиа и соответствующий контент курса. Мы настоятельно рекомендуем преподавателям присоединиться к хамам по темам, наиболее актуальным для их преподавания и исследований, чтобы обогатить свои курсы и взаимодействовать с другими преподавателями. Чтобы получить доступ к общественным хамам, посетите сайт [www.oercommons.org/hubs/openstax](http://www.oercommons.org/hubs/openstax) ([https://oercommons.org/groups/openstax-introduction-to-python-programming/14678/?hub\\_id=27](https://oercommons.org/groups/openstax-introduction-to-python-programming/14678/?hub_id=27)).

## Технологические партнеры

Для обеспечения доступности высококачественных учебных материалов наши технологические партнеры предлагают дополнительные недорогие инструменты, интегрированные с книгами OpenStax. Чтобы получить доступ к технологическим опциям для вашего учебника, посетите страницу вашей книги на OpenStax.org.

# 1. ОПЕРАТОРЫ

## Краткое содержание главы

- 1.1. Подготовка
- 1.2. Ввод/вывод
- 1.3. Переменные
- 1.4. Основы работы со строками
- 1.5. Основы работы с числами
- 1.6. Сообщения об ошибках
- 1.7. Комментарии
- 1.8. Почему именно Python?
- 1.9. Итоги главы

## ВВЕДЕНИЕ

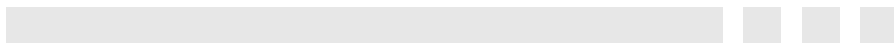
Компьютеры и программы в современном мире встречаются повсюду. Они напрямую влияют на многие аспекты повседневной жизни и общества в целом. Люди используют программы почти во всех сферах жизни: в общении, при совершении покупок, развлечении, мониторинге здоровья и для многих других потребностей. Изучение программирования открывает двери во многие профессии и дает возможность сделать этот мир еще лучше.

Программы состоят из операторов, которые выполняются один за другим.

**Оператор** описывает некоторое действие, которое должно быть выполнено.

Оператор `print("Доброе утро!")` сообщает Python, что необходимо вывести на экран сообщение "Доброе утро!". Оператор `count = 0` присваивает переменной `count` численное значение, равное 0.

В этой главе вы познакомитесь с операторами ввода и вывода, научитесь объявлять и присваивать переменные, а также поработаете с базовой арифметикой в программировании. Ошибки — это естественная часть программирования, поэтому в данной главе содержатся советы, которые помогут разобраться и понять сообщения об ошибках. В конце главы приводится краткая история языка Python, а также объясняется, почему Python является столь популярным языком программирования в наши дни.



# 1.1. Подготовка

## Цели обучения

После изучения этого раздела вы должны уметь:

- Назвать два примера использования компьютерных программ в повседневной жизни.
- Объяснить, почему Python — хороший язык программирования для изучения.

## Компьютерные программы

**Компьютер** — это электронное устройство, которое хранит и обрабатывает информацию. Примерами современных компьютеров являются смартфоны, планшеты, ноутбуки, настольные компьютеры. Технически **программа** — это последовательность инструкций, которые может выполнять компьютер. Программы помогают людям решать повседневные задачи, создавать новые технологии, а также могут просто принести удовольствие.

Цель данной книги — научить базовым основам программирования и решению задач. Написание программ — это творческое, весьма полезное, но самое главное — приносящее удовольствие занятие! Для чтения и понимания данной книги не требуется какой-либо предварительной подготовки в области информатики. Существует огромное количество различных типов программ. В этой книге мы рассмотрим программы, которые обычно выполняются «за кадром».

## ЧЕКПОИНТ

### *Потоковое воспроизведение музыки онлайн*



Доступ к мультимедийному контенту  
(<https://openstax.org/books/introduction-python-programming/pages/1-1-background>)

## МАТЕРИАЛ НА ПРАКТИКЕ

### *Компьютеры и программы*

1. Сколько типов программ было описано в видеоанимации?

- a. 3
- b. 4
- c. 5

2. Написание какого типа программ будет подробно разобрано в данной книге?
- a. инструмент, обобщающий музыкальные предпочтения человека
  - b. мобильное приложение для управления и обмена плейлистами песен
  - c. веб-сайт, на котором представлены лучшие художники за последние пять лет
3. Какое из следующих устройств является примером компьютера?
- a. проводные наушники, подключаемые к смартфону
  - b. пульт дистанционного управления, который приостанавливает или пропускает текущую песню
  - c. Wi-Fi-колонка, передающая музыку с Amazon
4. Чтение данной книги требует серьезных специальных знаний в области математики.
- a. правда
  - b. ложь

## ДАЛЬНЕЙШЕЕ ИЗУЧЕНИЕ

В последующих главах этой книги рассказывается о том, как писать аналитические программы, используя реальные данные. Примеры библиотек, предоставляющих доступ к потоковым онлайн-сервисам, включают Spotify (<https://openstax.org/r/100spotify>), Youtube (<https://openstax.org/r/100pytube>) и Pydora (<https://openstax.org/r/100pydora>). Инструменты, связанные с Python, часто имеют в названии буквы «ру».

## Язык программирования Python

Эта книга познакомит вас с Python (<https://openstax.org/r/100python>), одним из самых востребованных языков программирования на сегодняшний день. Ведущие технологические гиганты, такие как Google, Apple, NASA, Instagram, Pixar и другие, широко используют Python.

Одна из причин популярности Python — наличие огромного количества библиотек для выполнения реальной работы. **Библиотека** — это уже готовый набор кода, который может быть использован в других программах. Python поставляется с широкой стандартной библиотекой (<https://openstax.org/r/100pythlibrary>), которая позволяет решать повседневные вычислительные задачи, такие как извлечение данных из файлов и создание итоговых отчетов. Кроме того, сообщество активно разрабатывает и множество других практических библиотек для Python. Например, Pandas (<https://openstax.org/r/100pandas>) — это широко используемая библиотека для анализа данных.

Еще одна причина популярности Python — краткий и понятный синтаксис. **Синтаксис** языка определяет, каким образом должен быть структурирован код. Правила синтаксиса определяют ключевые слова, символы и форматирование, используемые в написании кода. По сравнению с другими языками программирования Python более лаконичен и прост.

### ПРИМЕР 1.1

#### *Hello World на Python и Java*

По традиции Hello World (<https://openstax.org/r/100helloworld>) — это первая программа, которую принято писать при изучении нового языка программирования.