

Глава 1

НАЧАЛА ПРОГРАММИРОВАНИЯ

§ 1.1

Общие сведения о языке программирования Python

Ключевые слова:

- язык программирования
- программа
- алфавит
- * служебные слова
- типы данных
- структура программы
- оператор присваивания

Языки программирования — это формальные языки, предназначенные для записи алгоритмов, исполнителем которых является компьютер. Алгоритмы, записанные на языках программирования, называют **программами**.

Существует несколько тысяч языков программирования. Один из самых популярных современных языков программирования называется Python (произносится «пайтон», хотя в России многие называют язык просто «питон»). Его разработал в 1991 году нидерландский программист Гвидо ван Россум. Язык Python непрерывно совершенствуется, и сейчас большинство программистов используют его третью версию — Python 3. Именно с этой версией будем работать и мы.

Python — язык программирования высокого уровня, предназначенный для решения самого широкого круга задач. С его помощью можно обрабатывать различные данные, проводить математические вычисления, создавать изображения, работать с базами данных, разрабатывать веб-сайты.



Для того, чтобы разрабатывать программы на языке Python, нужно установить на компьютер интерпретатор Python. Во многих операционных системах, например в macOS и Linux, этот интерпретатор входит в стандартную поставку и устанавливается вместе с операционной системой. Чтобы установить Python в операционной системе Microsoft Windows, скачайте последнюю версию программы-установщика Python 3 для Windows с официального сайта <http://www.python.org/> (для этого зайдите в меню **Downloads** и выберите **Windows**), запустите загрузившийся файл и следуйте указаниям установщика. Обратите внимание: в установщик Python для Windows встроена интегрированная среда разработки IDLE (произносится «айдл»), предназначенная для ввода, просмотра, редактирования, запуска или отладки программы на языке Python.

1.1.1. Алфавит и словарь языка

Основой языка программирования Python, как и любого другого языка, является **алфавит** — набор допустимых символов, которые можно использовать для записи программы. Это:

- латинские прописные и строчные буквы (A, B, C, ..., X, Y, Z, a, b, c, ..., x, y, z);
- арабские цифры (0, 1, 2, ..., 7, 8, 9);
- специальные символы (знак подчёркивания; круглые, квадратные скобки; знаки арифметических операций; # — знак начала однострочного комментария и др.).

В качестве неделимых элементов (составных символов) рассматриваются следующие последовательности символов:

`>=` и `<=` (знаки \geq и \leq);

`!=` (знак \neq);

`"""` или `'''` (утроенные двойные или одинарные кавычки, ставящиеся в начале и в конце многострочного комментария).

В языке также существует некоторое количество различных цепочек символов, рассматриваемых как единые смысловые элементы с фиксированным значением. Такие цепочки символов называются **служебными словами**. В таблице 1.1 приведены основные служебные слова, которые мы будем использовать при записи программ на языке Python.

Таблица 1.1

Служебное слово языка Python	Значение служебного слова
and	и
break	прервать
elif	иначе если
else	иначе
False	ложь
float	вещественный тип данных (с плавающей точкой)
for	для
if	если
input	ввод
integer	целый
list	список
not	не
or	или
print	печать
string	строковый тип данных (цепочка символов)
True	истина
while	пока

Для обозначения переменных, программ и других объектов используются **имена (идентификаторы)** — любые отличные от служебных слов последовательности букв, цифр и символа подчёркивания, начинающиеся с буквы или символа подчёркивания.

Прописные и строчные буквы в именах различаются, например, `f` и `F` — две разные переменные.

Длина имени может быть любой. Для удобства рекомендуется использовать имена, передающие смысл объектов, с длиной не более 15 символов.



В программах на языке Python (начиная с версии 3) есть возможность использовать в именах буквы национальных алфавитов (от русских до китайских иероглифов). Но это считается очень плохим стилем, так делать не рекомендуется. Подумайте почему.

1.1.2. Типы данных, используемые в языке Python

В языке Python используются различные типы данных (табл. 1.2).

Таблица 1.2

Название	Обозначение	Допустимые значения
Целочисленный	<code>int (integer)</code>	Сколько угодно большие, размер ограничен оперативной памятью
Вещественный	<code>float</code>	Любые числа с дробной частью
Строковый	<code>str (string)</code>	Любые последовательности символов из таблицы Unicode
Логический	<code>bool (boolean)</code>	False и True

В вещественном числе целая часть от дробной отделяется точкой, при этом перед точкой и после неё должно быть, по крайней мере, по одной цифре. Пробелы внутри числа недопустимы.

Произвольный набор символов, заключённый в одинарные или двойные кавычки, считается строковой величиной (строкой). Строка может содержать любые символы, набираемые на клавиатуре, в том числе буквы национальных алфавитов.

В отличие от многих других языков программирования переменные в языке Python не нужно объявлять. *Тип переменной определяется автоматически в тот момент, когда ей присваивается новое значение.*

Тип каждой переменной может динамически изменяться по ходу выполнения программы. Определить, какой тип имеет переменная в текущий момент, можно с помощью функции (команды) `type()`.

1.1.3. Режимы работы интерпретатора Python

Интерпретатор Python может работать в двух режимах:

- через командную строку (в командном, или интерактивном режиме), когда каждая введённая команда сразу выполняется;

- в программном режиме, когда программа сначала записывается в файл (обычно имеющий расширение *.py*) и при запуске выполняется целиком.

Изучение языков программирования принято начинать с программы, выводящей на экран надпись: «Привет, мир!». На Python соответствующая программа будет иметь вид:

```
print("Привет, мир!")
```

Для вывода на экран последовательности символов (текста, надписи) используется встроенная команда `print`. Последовательность символов, которые должны быть выведены на экран, заключается в двойные кавычки и записывается в круглых скобках. Вместо двойных кавычек можно использовать одинарные кавычки.

В начале строки (левее команды `print()`) не должно быть пробелов — таково требование языка Python.

Для запуска программы выбираем в меню **Пуск** → **Программы** → **Python 3.7** → **IDLE**. В результате откроется окно *Python Shell*, в котором символы `>>>` означают приглашение ввести команду. После ввода строки нажимаем клавишу *Enter*. На следующей строке сразу отобразится результат, а далее — приглашение для ввода новой команды (рис. 1.1).

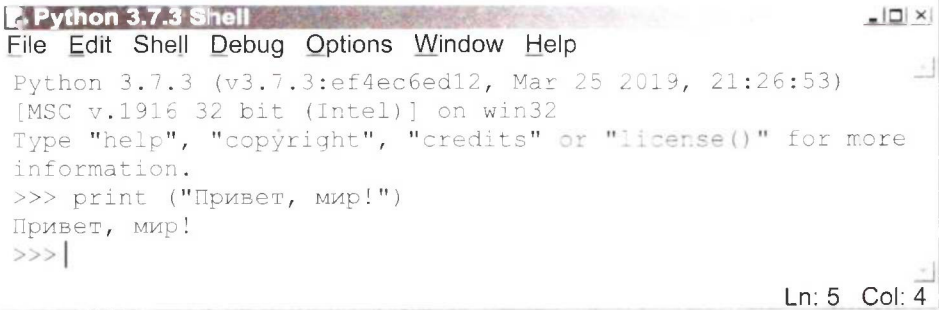


Рис. 1.1. Работа в командном режиме

Для создания файла с программой в меню **File** выбираем пункт **New File**. В открывшемся окне набираем текст программы, а затем сохраняем его под каким-нибудь именем (например, *test.py*), выбрав пункт меню **File** → **Save As**. Запустить программу на выполнение можно, выбрав пункт меню **Run** → **Run Module** или нажав клавишу *F5*.