

Челлендж «Программирование графических объектов на языке Питон. Корабли нашей мечты»

Знакомимся с Питоном

В последние годы язык программирования *Питон* (английский вариант произношения *Пайтон*) – *Python* – становится всё более популярным, несмотря на то, что это довольно старый язык программирования 😊, ведь Питон появился в 1991 году. К змее-питону название языка не имеет ни малейшего отношения, потому что автор языка Гвидо ван Россум назвал его в честь популярного британского комедийного телешоу 1970-х годов «Летающий цирк Монти Пайтона».

Питон часто предлагают изучать тем, кто никогда ранее не программировал. Одним из преимуществ Питона является универсальность, что позволяет использовать язык для решения самых разных задач. Также специалисты отмечают понятность и лаконичность кода на данном языке, что также делает его привлекательным для начинающих программистов.

В Питоне существует большое количество библиотек, представляющих собой готовые решения для тех или иных задач. Существуют библиотеки для работы с нейросетями, для машинного обучения, анализа данных, математических, научных и инженерных вычислений, визуализации данных, распознавания голоса и лиц, обработки естественного человеческого языка.

Кроме того, если у вас возникнет какая-то проблема, то, скорее всего, вы сможете ее решить простым поиском – наверняка кто-то уже решал (и успешно решил) эту проблему. Для решения задачи с помощью языка программирования создается программа, которая пишется в соответствии с правилами, набор таких правил принято называть синтаксисом языка программирования. Программа представляет собой специальным образом по правилам синтаксиса языка программирования организованный текст, текст пишется в специальной среде, которую необходимо установить на компьютере, ее называют средой разработки. Среда разработки позволяет набирать тексты программ, проверять правильность синтаксиса, преобразовывать текст в программу. Преобразование текста в программу происходит с помощью специальной программы – транслятора.

О трансляторах и компиляторах

Трансляторы бывают компиляторами или интерпретаторами.

Компилятор получает текст программы, проверяет синтаксис, и, если нет ошибок, то создает машинный код. Для запуска программы не нужен больше ни исходный текст, ни среда разработки, т. к. создается исполняемый файл. Работа

компилятора похожа на работу переводчика книги на иностранном языке – пока не будет переведена вся книга, издательство не будет печатать перевод.

Интерпретатор же проверяет синтаксис каждой строки текста и тут же выполняет эту строку. Чтобы запустить программу снова, этот процесс придется повторить. Без среды разработки сделать это не получится. Если вы сделаете синтаксическую ошибку в последней строке, то узнаете вы об этом только после того, как почти вся программа выполнится, и прервется на этой ошибочной строке. Работа интерпретатора похожа на работу синхронного переводчика – каждая сказанная фраза тут же переводится и озвучивается.

Преимущества и недостатки Питона

Одно из самых важных достоинств Питона – простота синтаксиса. Именно поэтому Питон очень хорош в качестве первого языка при обучении технологии программирования. В олимпиадах для школьников и ЕГЭ очень удобно использовать язык Питон.

Но у него есть и недостатки. Один из недостатков – Питон прощает ошибки. Например, при выборе типов данных – для начинающих это и хорошо, и плохо – с одной стороны, вы не задумываетесь о выборе типа, но с другой стороны, вы не сможете научиться дисциплинированно и вдумчиво выбирать типы данных.

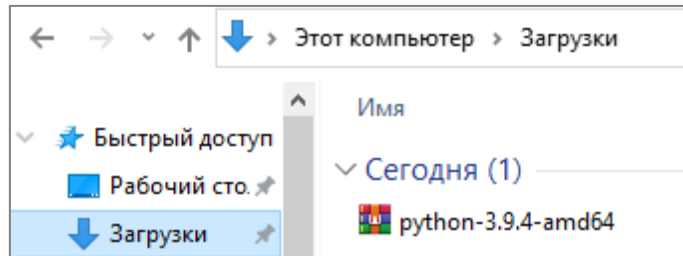
Другой существенный недостаток – скорость. В этом виноват подход к выбору типов данных, так называемая динамическая типизация, которая позволяет прощать небрежный выбор типов данных. И, кроме того, Питон является интерпретируемым языком, а не компилируемым.

Прежде чем научиться программировать на языке Питон, сделаем первые шаги к его изучению – установи Питон на свой компьютер, познакомимся с его интерфейсом, создадим простую программу и сохраним ее, поэкспериментируем, создав несколько простых программ и проверив, как они работают.

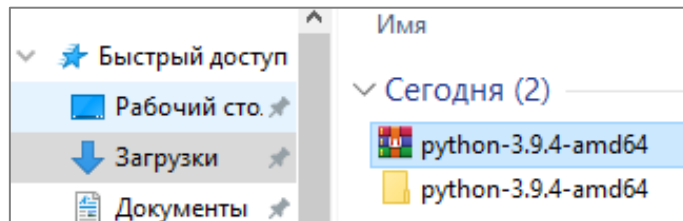
Установка Питона

Для работы нам потребуется установить среду разработки для Питона. Для этого скачайте [файл с архивом](#). Для работы выбрана версия Python-3.9.4, которая используется в пунктах приема экзаменов итоговой аттестации в 2022 году в Воронежской области.

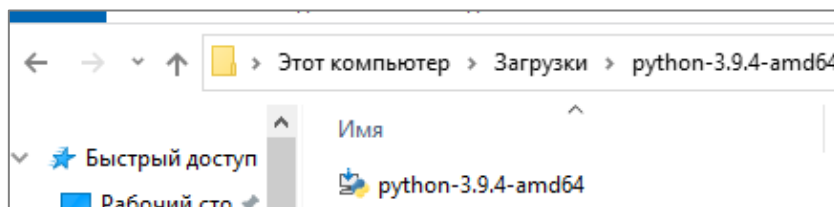
Он появится в вашей папке загрузок.



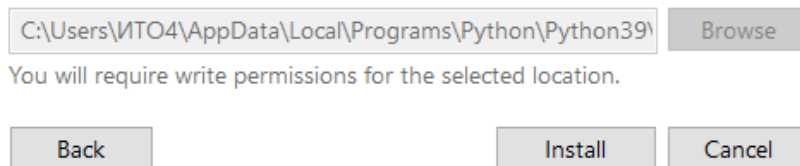
Щелкните правой кнопкой мыши по этому файлу, и в появившемся контекстном меню выберите пункт *Распаковать (извлечь)* в *python-3.9.4-amd64*.



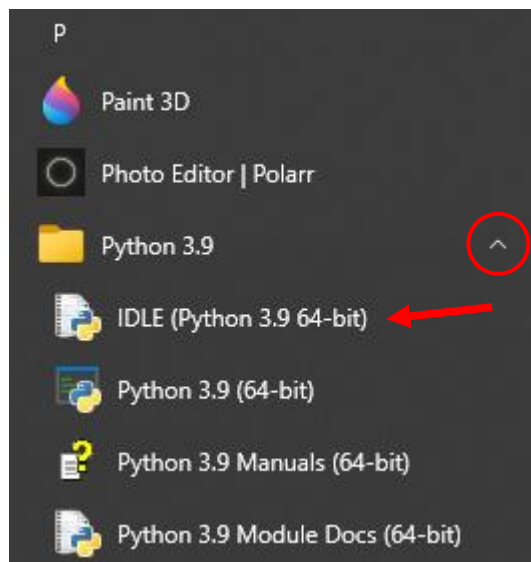
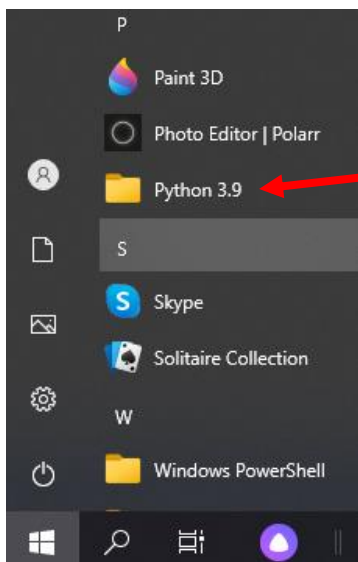
Затем войдите в папку с этим именем и двойным щелчком запустите находящийся там файл.



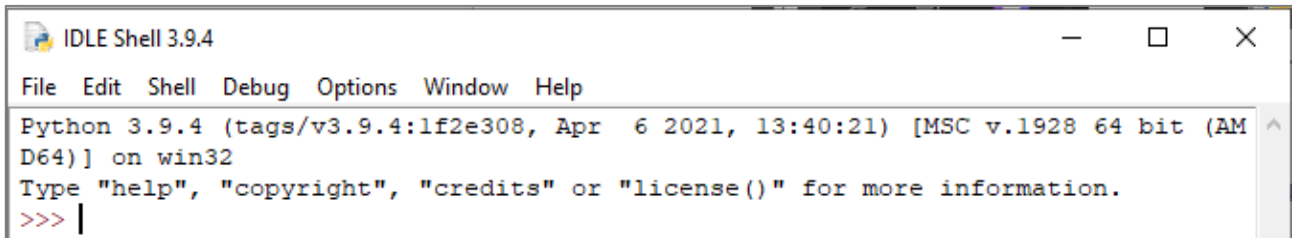
Далее следуйте указаниям на экране. Будет указан каталог на диске C, в который будут размещены файлы Питона и надо запустить команду Install для установки программы.



После установки программы перейдите в главное меню, найдите там пакет Python 3.9, раскройте его нажатием стрелочки справа от названия и запустите среду IDLE, в которой мы будем работать.



После запуска среды открывается следующее окно:



```
Python 3.9.4 (tags/v3.9.4:1f2e308, Apr 6 2021, 13:40:21) [MSC v.1928 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> |
```

В Питоне используется два разных окна – окно консоли и окно программы. Открывшееся окно называется **окном консоли**.

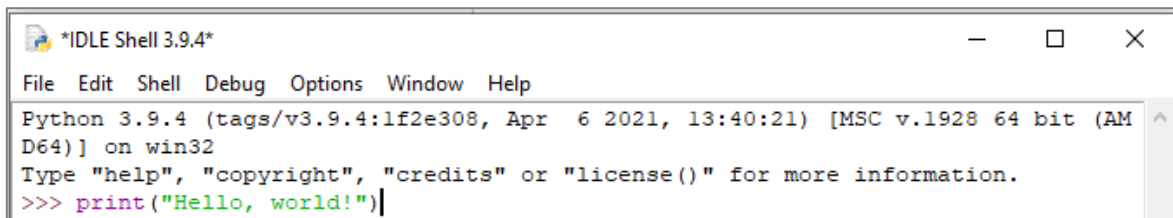
Написание первой программы

Давайте создадим первую программу – по традиции это программа, печатающая на экране *Hello, world!* В среде IDLE, которую мы уже запустили, необходимо набрать соответствующий текст. Набираем его в окне консоли после символа `>>>`. В окне консоли отображается то, что программа печатает, и ошибки.

Среда запустилась в интерактивном режиме и ждет ваших команд. Наберите в строке текст. Обратите внимание на кавычки, они должны быть одинаковыми (двойными или одинарными). Будет напечатано то, что в кавычках.

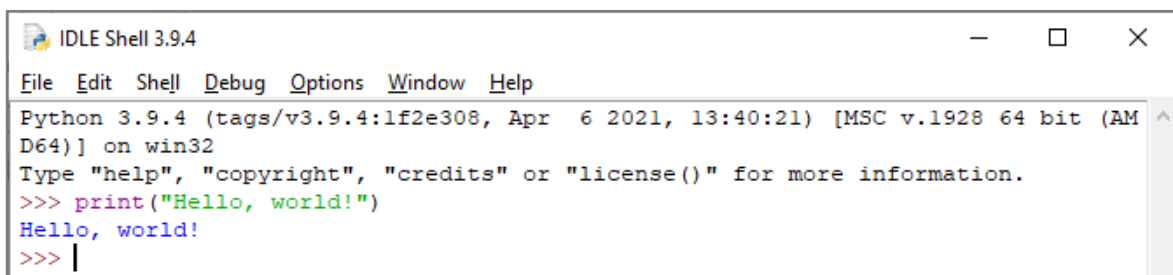
```
print("Hello, world!")
```

Команда *print* означает печать. Конструкция языка, описывающая одно действие, называется **оператор**.



```
*IDLE Shell 3.9.4*
Python 3.9.4 (tags/v3.9.4:1f2e308, Apr 6 2021, 13:40:21) [MSC v.1928 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> print("Hello, world!")
```

Для выполнения программы нажмите клавишу `<Enter>`, в результате напечатается текст *Hello, world!* Текст печатается без кавычек.



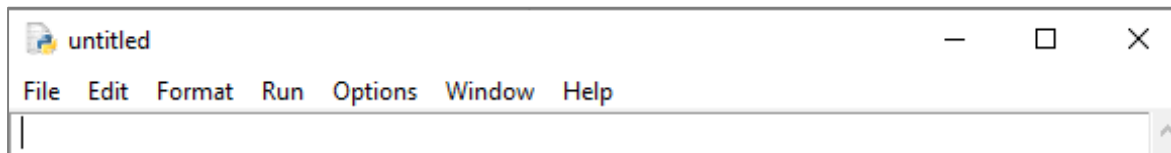
```
IDLE Shell 3.9.4
Python 3.9.4 (tags/v3.9.4:1f2e308, Apr 6 2021, 13:40:21) [MSC v.1928 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> print("Hello, world!")
Hello, world!
>>> |
```

Поздравляем – вы написали свою первую программу! 🍏

Создание файла программы

В интерактивном режиме писать и сразу выполнять по одной команде в каждой строке не очень удобно, поэтому лучше записывать текст программы в отдельном файле и сохранять его, а потом запускать сразу все команды этого файла.

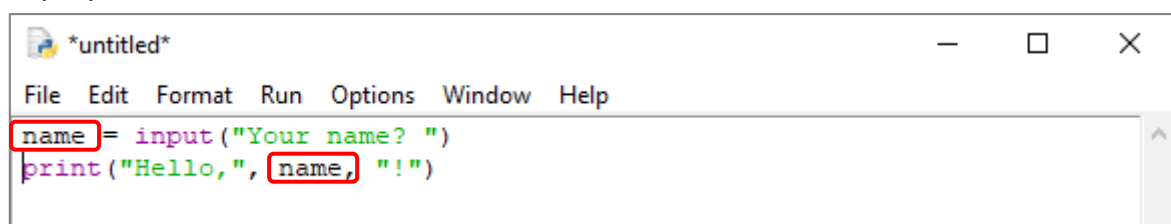
Давайте создадим новый файл, для этого выберите пункт меню *File—New file* (или нажмите *<Ctrl–N>*). Откроется окно, которое называется **окном программы**:



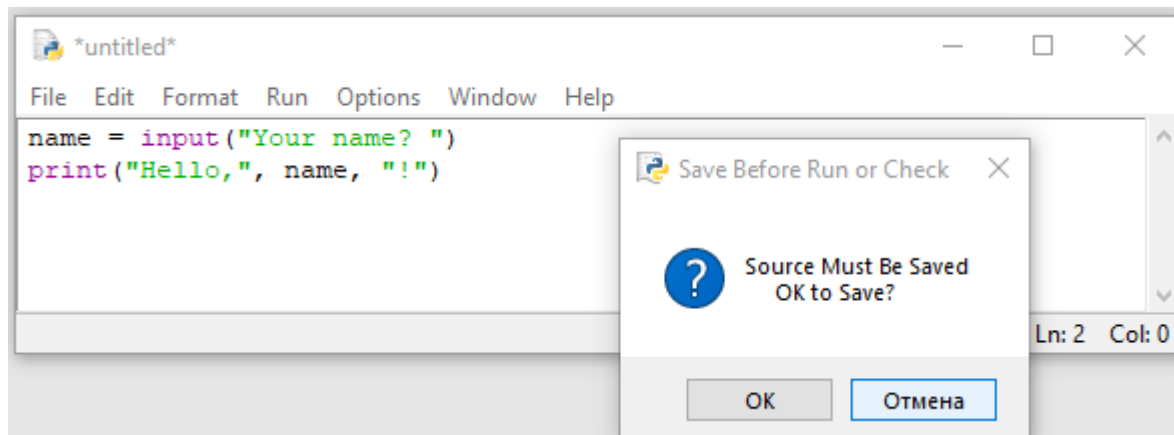
В открывшемся окне программы введите текст:

```
name = input("Your name? ")
print("Hello,", name, "!")
```

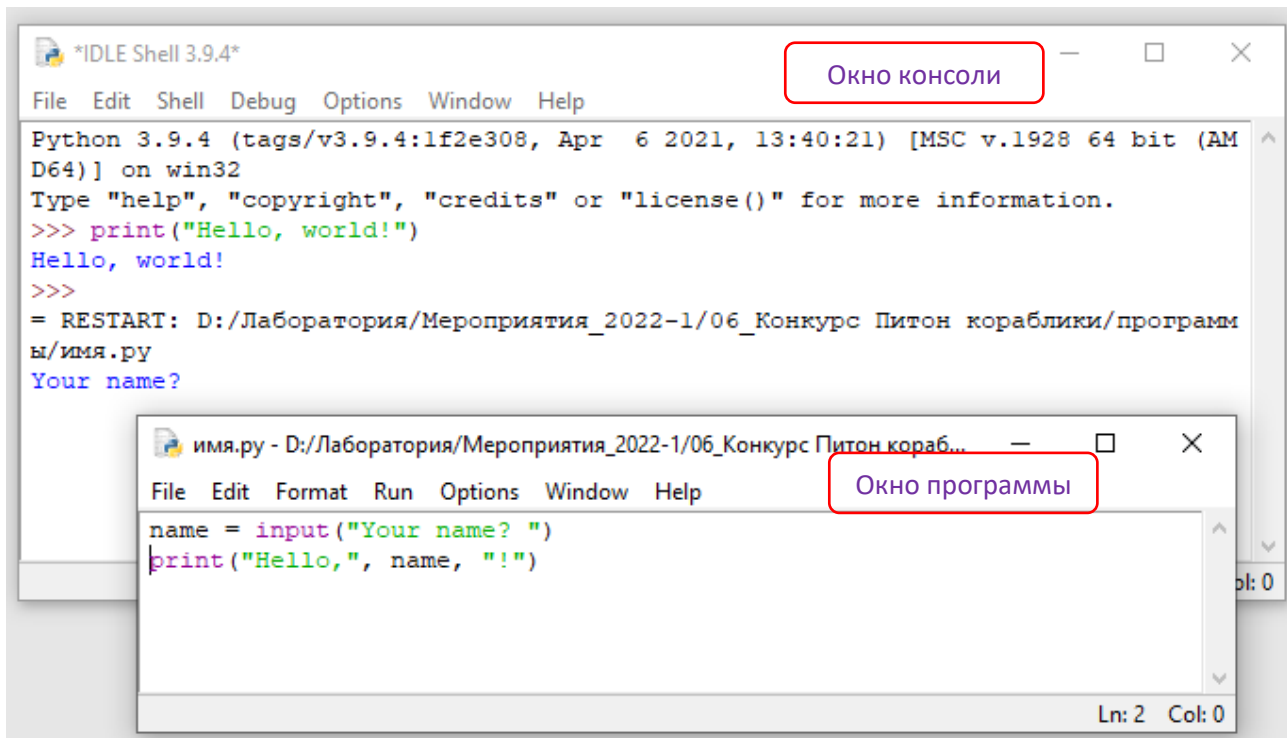
Здесь появилась новая команда *input*, она означает ввод с клавиатуры, т.е. нам будет предложено что-то (например, свое имя) ввести с клавиатуры. Также в этом операторе появилось слово *name*, это **переменная**, которая далее используется в операторе *print*.



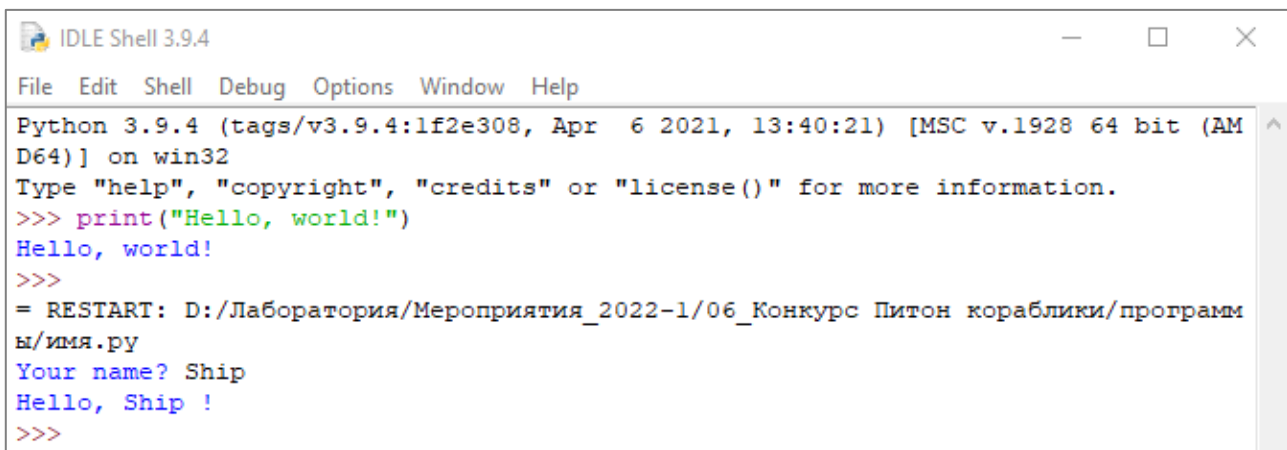
Теперь давайте выполним эту программу. Для этого нажмите *<F5>* (или выберите пункт меню *Run—Run Module*).



Среда предложит вам сохранить файл. Сохраните его в папку для ваших будущих программ. Программа запустится, и вы увидите приглашение ввести ваше имя. Введите имя (мы назвали файл имя), нажмите *<Enter>*, и на экране появится



После вопроса Your name? введите имя, мы ввели Ship, и нажмите на Enter. В результате работы программы будет напечатано **Hello, ship!**



Теперь вы научились вводить данные и выводить их на экран! Можно печатать текст как в латинском, так и в русском регистре.

Запомните простое правило:

Написать программу ⇒ Сохранить программу ⇒ Запустить программу

Ошибки программы

Бывает, что программа может не запуститься. Питон выдает сообщение об ошибке. Например, мы набрали неправильно команду print, например, prant. После запуска программы выполнится команда input, в которой нет ошибок, мы введем имя и после нажатия на Enter возникнет ошибка (помните, Питон – интерпретатор? Он каждый раз анализирует строчки одну за другой). Выдано сообщение, что 'prant' не определено.

```

IDLE Shell 3.9.4
File Edit Shell Debug Options Window Help
= RESTART: D:/Лаборатория/Мероприятия_2022-1/06_Конкурс Питон кораблики/программы/имя.py
Your name? Аврора
Hello, Аврора !
>>>
= RESTART: D:/Лаборатория/Мероприятия_2022-1/06_Конкурс Питон кораблики/программы/имя.py
Your name? Flint
Traceback (most recent call last):
  File "D:/Лаборатория/Мероприятия_2022-1/06_Конкурс Питон кораблики/программы/имя.py", line 2, in <module>
    prant("Hello,", name, "!")
NameError: name 'prant' is not defined
>>>

```



```

имя.py - D:/Лаборатория/Меропр...
File Edit Format Run Options Window Help
name = input('Your name? ')
prant("Hello,", name, "!")
|
Ln: 3 Col: 0

```

Сообщения об ошибках выдаются красным цветом, надо с ними разбираться, искать проблемы, исправлять их и после этого снова запускать программу.

Наиболее часто встречаются следующие ошибки:

- Использование в операторах заглавных букв. В Питоне важен регистр букв, команда `print` будет воспринята, а команда `Print` будет ошибочной.
- Использование одинарных и двойных кавычек. Кавычки используются парой, в которой обе кавычки должны быть одинаковыми – « » или ' '. Открывающей кавычке должна соответствовать такая же закрывающая.
- Необходимо различать дефис - и подчеркивание _.
- Использование скобок. В разных ситуациях будут использоваться различные скобки – (), [], {}. Всегда открывающей скобке должна соответствовать такая же закрывающая.

Задание для самопроверки

Давайте создадим программу, которая будет выводить на экран «Вас приветствует команда №№ n-ской школы!».

Удачной работы!