

Варианты выполнения кода

На прошлом занятии мы с Вами произвели установку Python, а на этом напомним нашу первую программу. Для этого нам понадобится редактор кода или IDE.

ТЕОРИЯ

Что такое IDE и редакторы кода?

Для любого разработчика, вне зависимости от его уровня знаний и языка программирования, главным инструментом создания кода является IDE(Integrated Development Environment, «интегрированная среда разработки») или редактор кода.

IDE — это программное обеспечение, которое состоит из общих инструментов разработчика в едином удобном графическом пользовательском интерфейсе GUI(Graphical User interface).

IDE объединяет несколько инструментов, специально предназначенных для разработки. Эти инструменты обычно включают в себя:

- редактор кода с функциями подсветки синтаксиса и автодополнения;
- инструменты сборки, выполнения и отладки;
- поддержку систем управления версиями и т.д.

Самый популярным IDE для python является PyCharm и мы обязательно познакомимся с ним в этом курсе, но для образовательных целей начнем писать в редакторе кода.

Редакторы кода также являются программным обеспечением; это похоже на текстовый редактор с некоторыми дополнительными функциями. Это не IDE, так как IDE имеет множество инструментов для разработчиков. Есть редакторы кода, которые представляют собой текстовый редактор с подсветкой синтаксиса и возможностями форматирования кода. Большинство хороших редакторов кода могут выполнять код и использовать отладчик, а лучшие даже могут взаимодействовать с системами управления версиями. По сравнению с IDE, хороший редактор кода, как правило, легче и быстрее, но зачастую ценой меньшей функциональности.

Мы начнем писать код в IDLE — это редактор, который поставляется вместе с Python. Это базовый, упрощенный режим программирования на Python. Тем не менее, IDLE хороший редактор для начала программирования и понимания основ языка. В нем есть все, что нам нужно для старта кодинга в мире python.

Варианты выполнения кода

Мы разобрались, где будем писать код, теперь осталось понять какие варианты существуют его выполнения. А их в python всего два:

1. Интерактивный режим (его еще называют консоль Python/Shell)
2. Файловый режим(стандартный вариант) - это более привычный вариант, это когда Вы весь код пишете в файле и запускаете его целиком на исполнение.

Интерактивный вариант

Рассмотрим интерактивный вариант. Находим в списке установленных программ IDLE и запускаем её.

IDLE представляет собой графическую среду разработки, которая устанавливается вместе с Python. После запуска IDLE вы можете вводить по одной команде и результат будет виден тут же, причем каждую инструкцию вы указываете после знаков ">>>", которые сигнализируют о приглашении ввода.

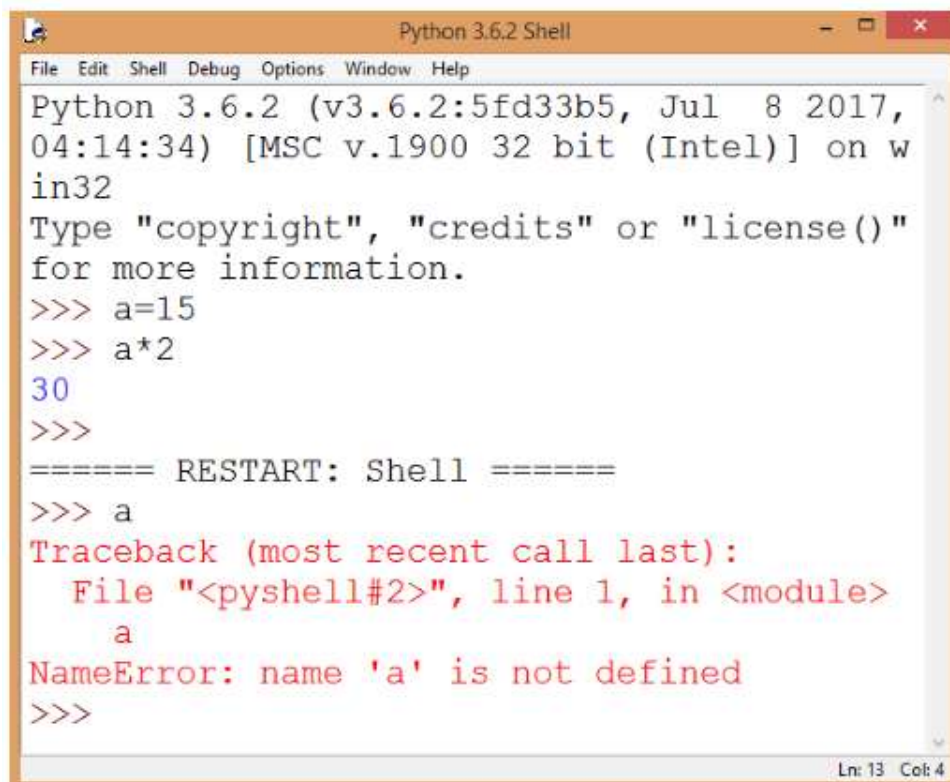


```
Python 3.10.2 (tags/v3.10.2:a58ebcc, Jan 17 2022, 14:12:15) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> 3 +
```

Интерактивный режим используется для:

1. Обучения
2. Экспериментирования
3. Тестирования

Большим минусом интерактивного режима является то, что строки кода и переменные, созданные в процессе работы с консолью Python, не сохраняются. То есть при каждом новом запуске или при перезапуске консоли вы начинаете работать с чистого листа.



```
Python 3.6.2 Shell
File Edit Shell Debug Options Window Help
Python 3.6.2 (v3.6.2:5fd33b5, Jul 8 2017, 04:14:34) [MSC v.1900 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> a=15
>>> a*2
30
>>>
===== RESTART: Shell =====
>>> a
Traceback (most recent call last):
  File "<pyshell#2>", line 1, in <module>
    a
NameError: name 'a' is not defined
>>>
```

К плюсам консоли можно отнести также то, что для вывода результата не нужно писать команду `print()`

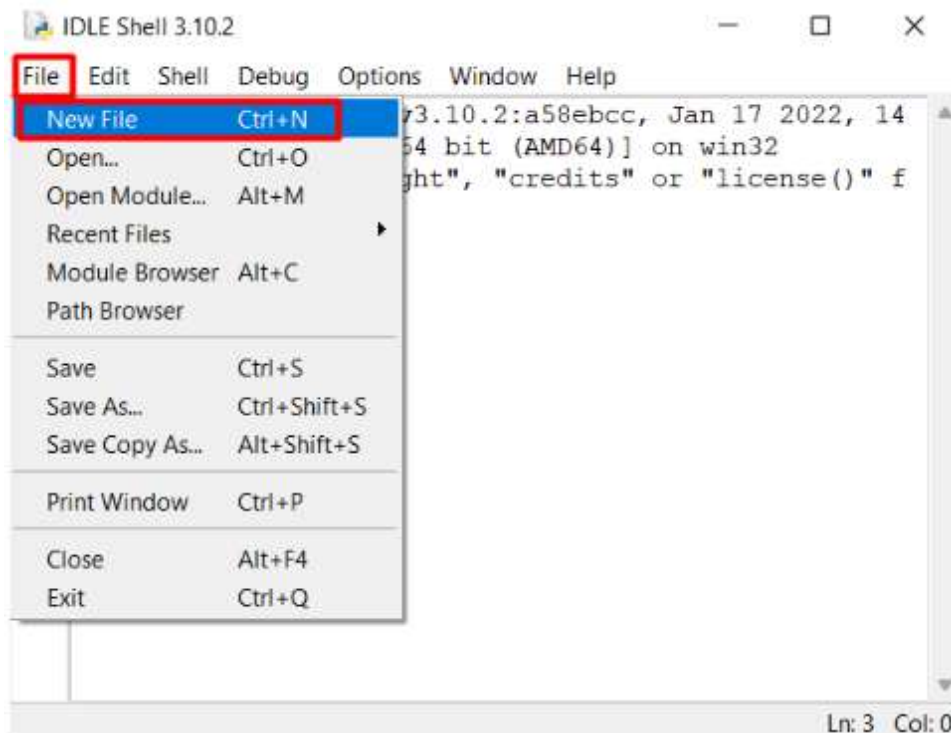


```
Python 3.10.2 (tags/v3.10.2:a58ebcc, Jan 17 2022, 14
:12:15) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" f
or more information.
>>>
>>> 3
3
>>> print(3)
3
>>> 'hello'
'hello'
>>> print('hello')
hello
>>> |
```

Файловый режим

Файловый режим представляет собой стандартный вариант написания и исполнения программы. Вы создаете новый файл, в котором пишете несколько инструкций, и затем целиком все команды идут на исполнение.

Для того, чтобы начать работать в файловом режиме вам нужно находясь в IDLE выбрать меню "File" -> "New file" или сочетание клавиш **ctrl+N**.



Во вновь открывшемся файле вы пишете инструкции, причем каждая новая инструкция пишется с новой строки.

Теперь команды будут выполняться если мы запустим файл:

«Run» → «Run Module» или по команде **f5**

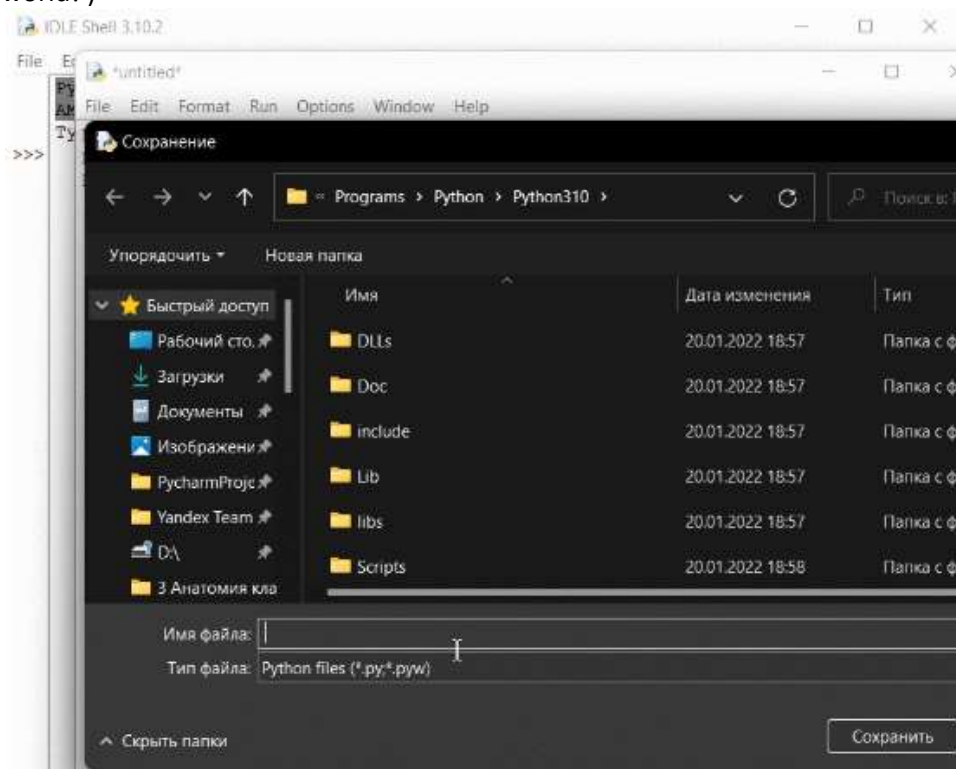
Далее нам предложат сохранить файл . Выберите удобное место и название(например «example»)

В файловом режиме для того, чтобы вывести результат выполнения программы на экран уже потребуется использовать команду `print()`

В `print()` можно вводить различные значения или различные математические выражения через запятую:

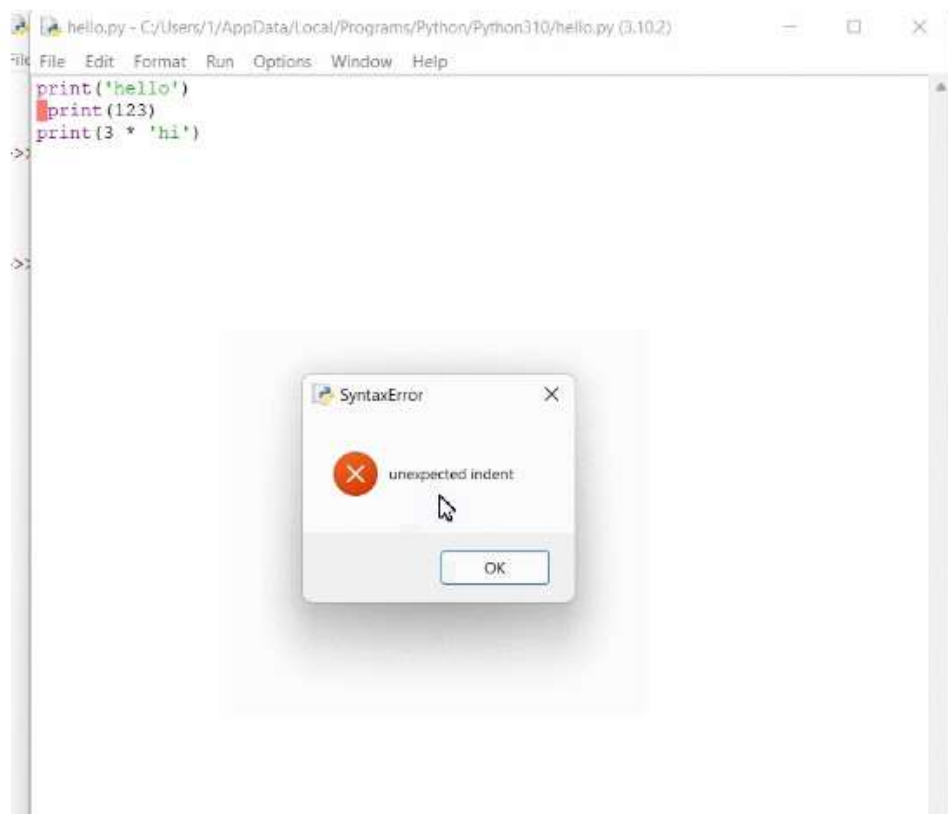
```
print(4, 5, 5 + 6)
```

Если мы хотим вывести текст то нам надо использовать одинарные или двойные кавычки 'hi', "hi" и ввести в них текст. Обычно все начинают с «hello world!»:
`print('hello world!')`



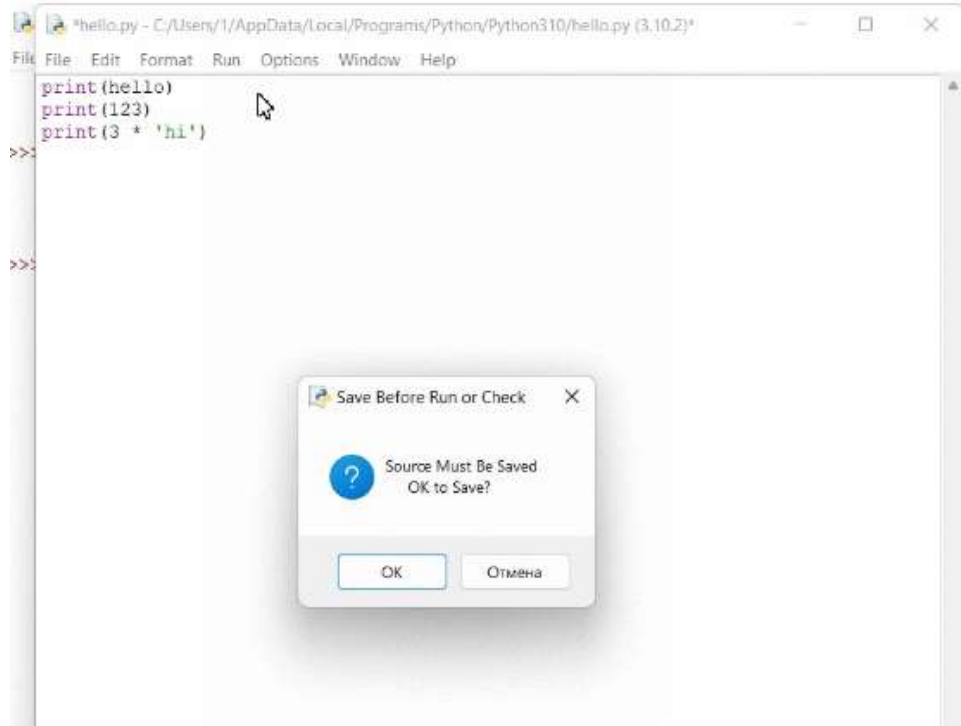
//видос выше

Использовать отступы и пробелы перед командами нельзя, так как это вызовет ошибку `unexpected indent`



//тоже видос выше

Если не поставить кавычки, то произойдёт ошибка, так как Python будет искать переменные с таким названием. Темы переменных мы еще не проходили - их мы будем изучать в дальнейшем



// видик

Преимущество файлового режима перед интерактивным является то, что все инструкции сохраняются в виде файла.