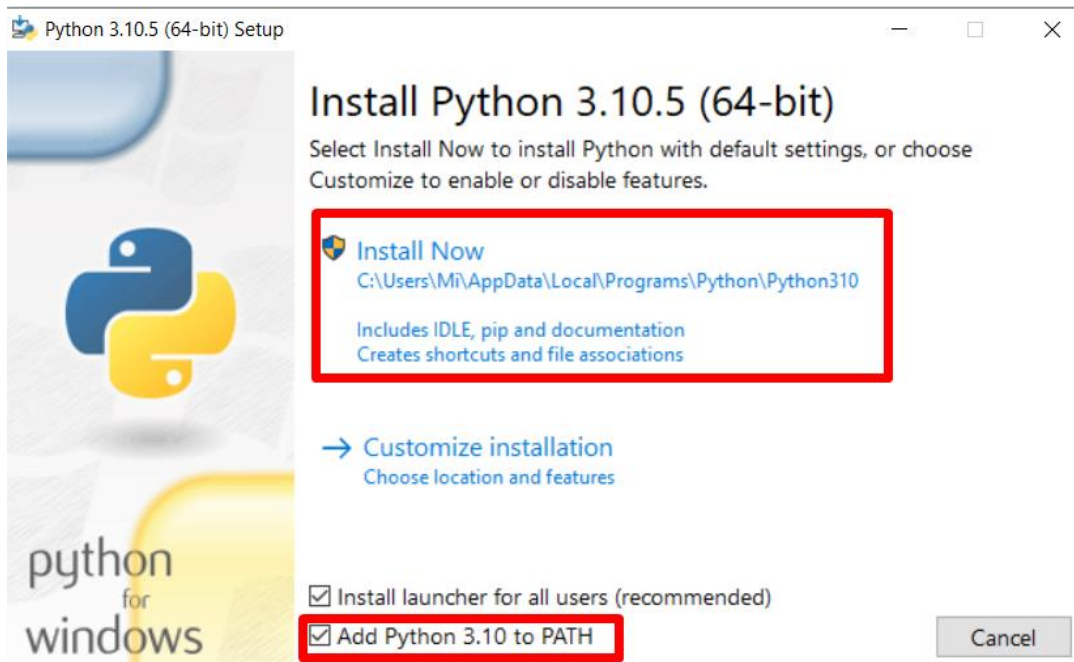
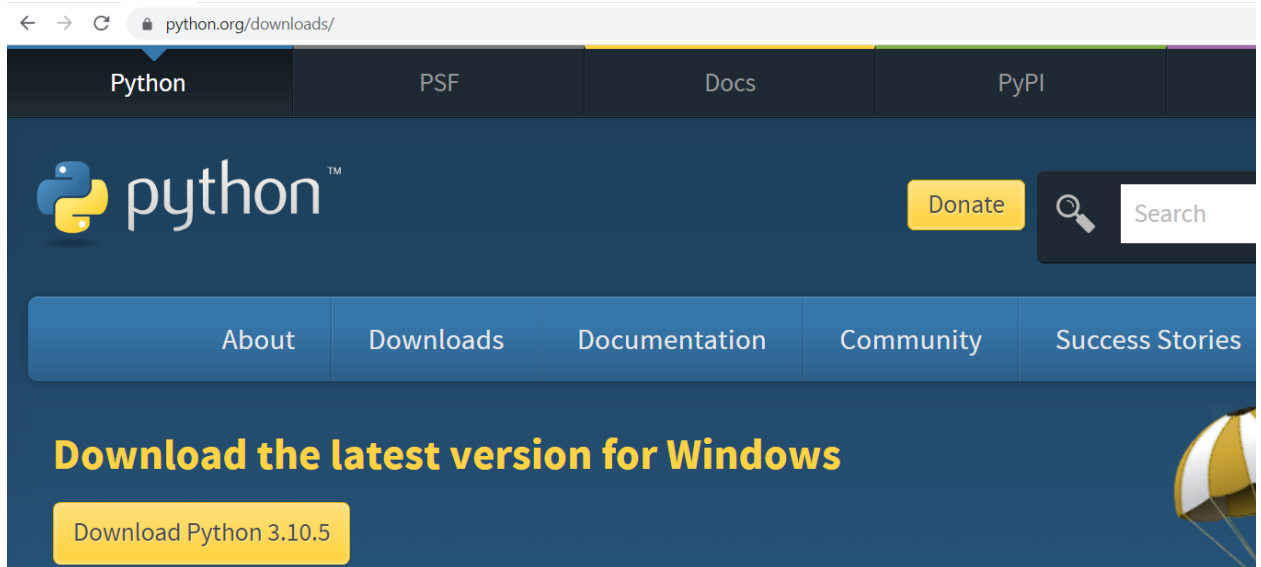


Лабораторная работа 1. Начало работы с Python.

Задание 1. Установка необходимого ПО

1. Установите Python (далее инструкция по установке на Windows)

<https://www.python.org/downloads>



Важно! Убедитесь в том, что вы установили флажок в разделе Add Python 3.x to PATH, чтобы интерпретатор был установлен в вашем пути выполнения.

Далее, просто нажмите на Install Now. Это все, что нужно. Через несколько минут у вас в системе должен быть рабочий Python 3

2. Установите PyCharm

<https://www.jetbrains.com/pycharm/download>

Download PyCharm

Windows macOS Linux

Professional

For both Scientific and Web Python development. With HTML, JS, and SQL support.

Download

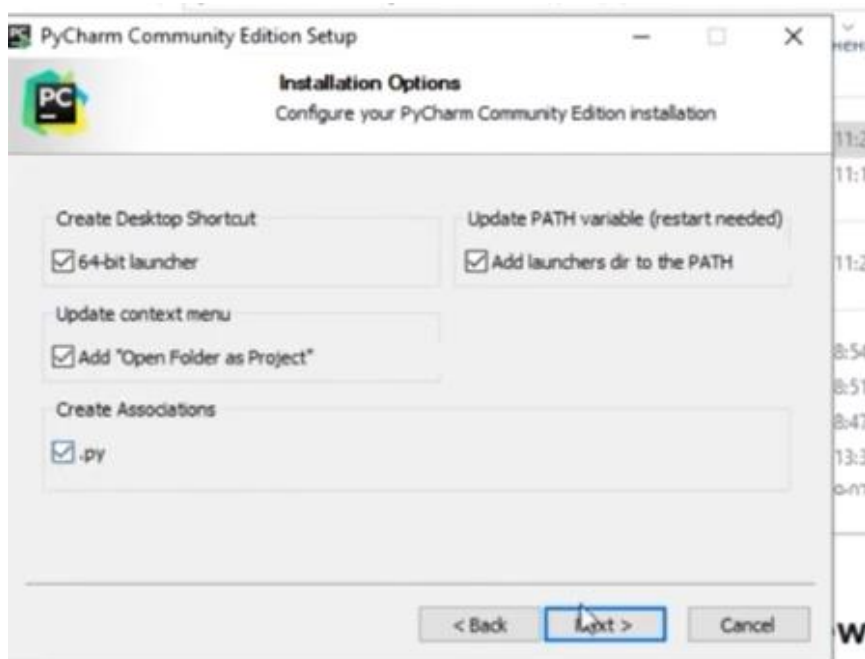
Free 30-day trial available

Community

For pure Python development

Download

Free, built on open-source



Задание 2. Работа в режиме командной строки интерпретатора

1. Откройте командную строку ПК, и посмотрите установлен ли Python его версию:

```
c:\Users\Mi>python --version
Python 3.10.5
```

2. Для перехода в режим интерпретатора дайте команду `python` и нажмите `enter`. Вы попадете в интерактивный режим, в нем выполните для примера сложение и вычитание, выведите на экран текст:

```
C:\Users\Mi>python
Python 3.10.5 (tags/v3.10.5:f377153, Jun
n win32
Type "help", "copyright", "credits" or "
>>> 100 + 200
300
>>> 100 - 200
-100
>>> print ("hello, world!")
Hello, world!
>>> 'Hello!'
'Hello!'
>>>
```

3. Выйдите из режима интерпретатора `python` обратно в командную строку Windows с помощью команд `quit()` или `exit()` или комбинацией `Ctrl-Z` и далее `Enter`. Например:

```
>>> quit()
C:\Users\Mi>_
```

4. Установите `ipython`. Для этого в командной строке Windows дайте команду

pip install ipython

```
C:\Users\Mi>pip install ipython
Collecting ipython
  Downloading ipython-8.4.0-py3-none-any.w
-----
collecting stack-data
  Downloading stack_data-0.3.0-py3-none-an
collecting prompt-toolkit!=3.0.0,!3.0.1,<
```

5. Перейдите в интерактивный режим IPython, с помощью команды `ipython`. Начните писать команду `print` и после первой введенной буквы "p" увидите подсказку:

```
C:\Users\Mi>ipython
Python 3.10.5 (tags/v3.10.5:f377
Type 'copyright', 'credits' or '
IPython 8.4.0 -- An enhanced Int
In [1]: print("Hello, world!")
```

6. Выполните простые арифметические действия:

```
x = 100 + 10
y = 257 - 94
z = x - y
```

7. Выведите результаты `x`, `y`, `z` на экран:

```
print (x, y, z) \
```

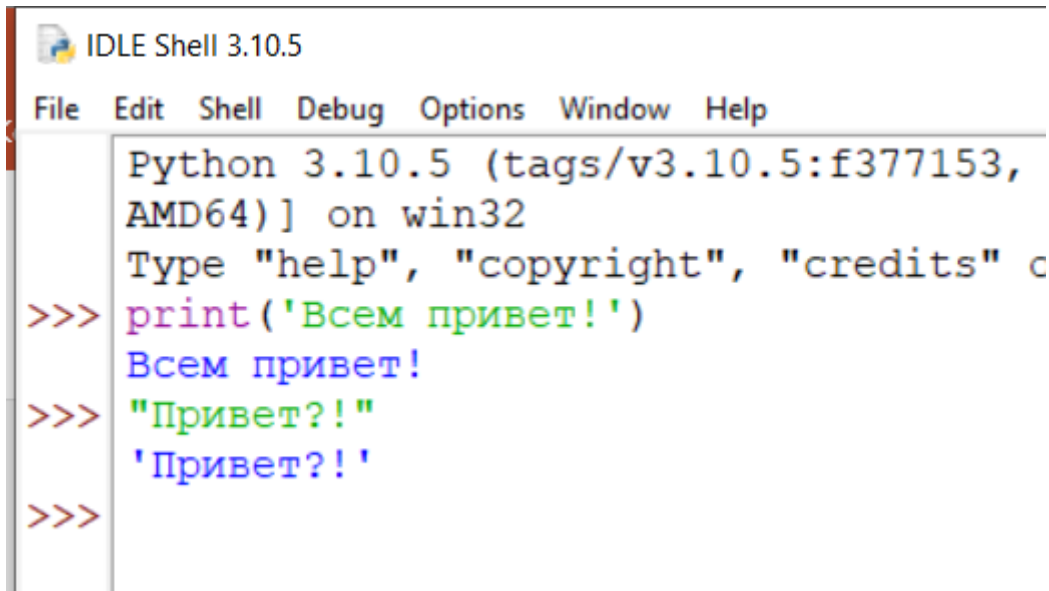
Задание 3. Знакомство с Python IDLE



IDLE (Python 3.10 64-bit)

Приложение

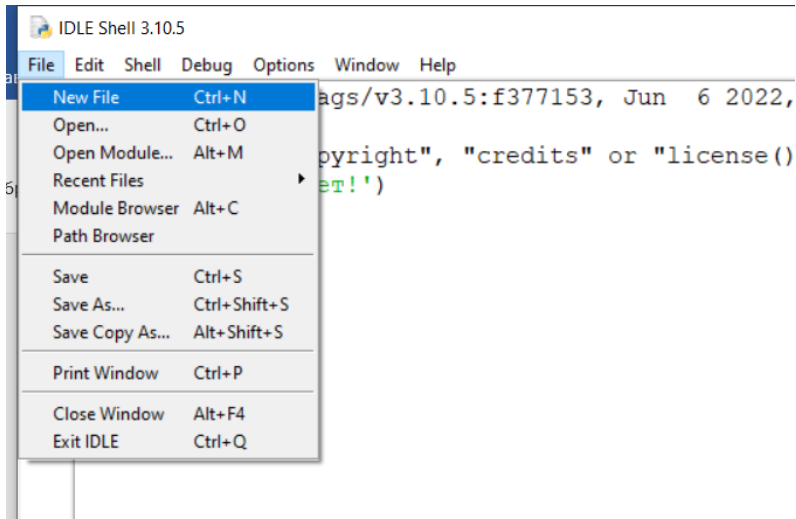
1. Откройте Python IDLE (shell), с помощью команды `print` и без нее выведите на экран приветствие:



The screenshot shows the IDLE Shell 3.10.5 window. The menu bar includes File, Edit, Shell, Debug, Options, Window, and Help. The shell displays the Python version and architecture: Python 3.10.5 (tags/v3.10.5:f377153, AMD64) on win32. It also shows the help text: Type "help", "copyright", "credits" or "license()" for more. The user has entered three lines of code: `>>> print('Всем привет!')`, `>>> "Привет?!"`, and `>>> 'Привет?!'`. The output shows the string "Всем привет!" and the string "Привет?!".

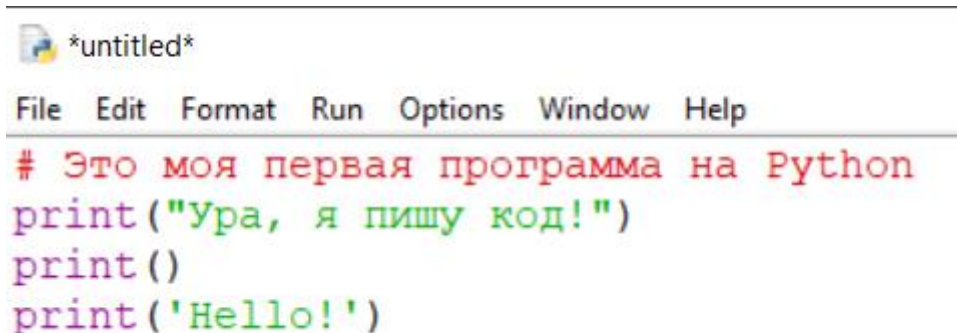
```
IDLE Shell 3.10.5
File Edit Shell Debug Options Window Help
Python 3.10.5 (tags/v3.10.5:f377153,
AMD64) on win32
Type "help", "copyright", "credits" or
"license()" for more
>>> print('Всем привет!')
Всем привет!
>>> "Привет?!"
'Привет?!'
>>>
```

2. В меню File, выберите New File, откроется новое окно, в котором можно создавать файл Python. Напишите в нем следующий код:



The screenshot shows the IDLE Shell 3.10.5 window with the File menu open. The menu options are: New File (Ctrl+N), Open... (Ctrl+O), Open Module... (Alt+M), Recent Files, Module Browser (Alt+C), Path Browser, Save (Ctrl+S), Save As... (Ctrl+Shift+S), Save Copy As... (Alt+Shift+S), Print Window (Ctrl+P), Close Window (Alt+F4), and Exit IDLE (Ctrl+Q). The background shows the same Python code as the previous screenshot.

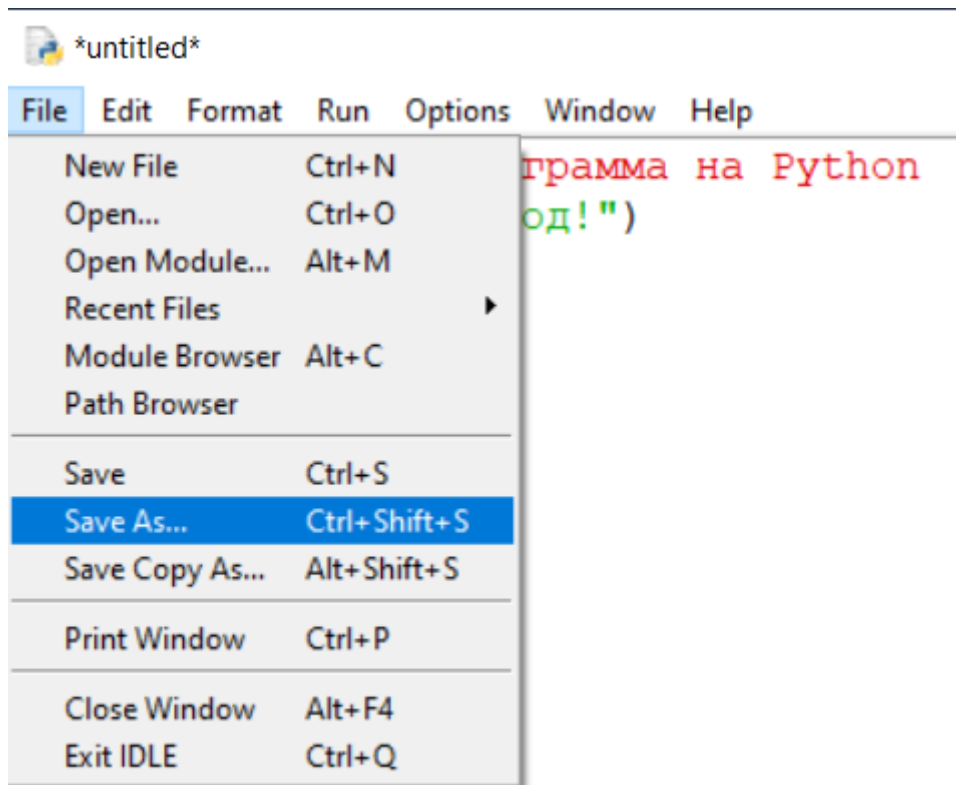
```
IDLE Shell 3.10.5
File Edit Shell Debug Options Window Help
New File Ctrl+N
Open... Ctrl+O
Open Module... Alt+M
Recent Files
Module Browser Alt+C
Path Browser
Save Ctrl+S
Save As... Ctrl+Shift+S
Save Copy As... Alt+Shift+S
Print Window Ctrl+P
Close Window Alt+F4
Exit IDLE Ctrl+Q
Python 3.10.5 (tags/v3.10.5:f377153, Jun 6 2022,
copyright", "credits" or "license()
et!')
```



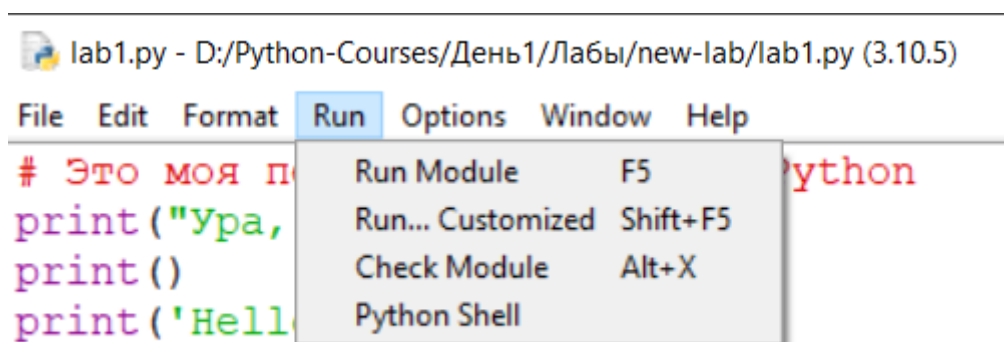
The screenshot shows a new IDLE window titled *untitled*. The menu bar includes File, Edit, Format, Run, Options, Window, and Help. The code entered is: `# Это моя первая программа на Python`, `print("Ура, я пишу код!")`, `print()`, and `print('Hello!')`.

```
*untitled*
File Edit Format Run Options Window Help
# Это моя первая программа на Python
print("Ура, я пишу код!")
print()
print('Hello!')
```

3. Сохраните файл с именем lab1.py через меню или с помощью клавиш Ctrl-S.



4. Далее запустите этот скрипт. Через меню (Run>Run Module) или с помощью F5 клавиши.



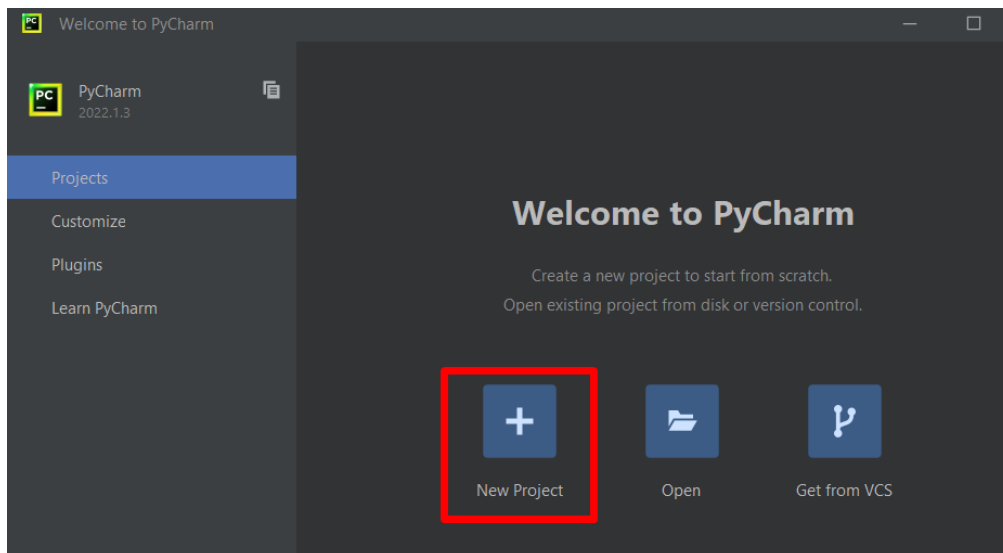
5. Посмотрите на полученный результат.

Ура, я пишу код!

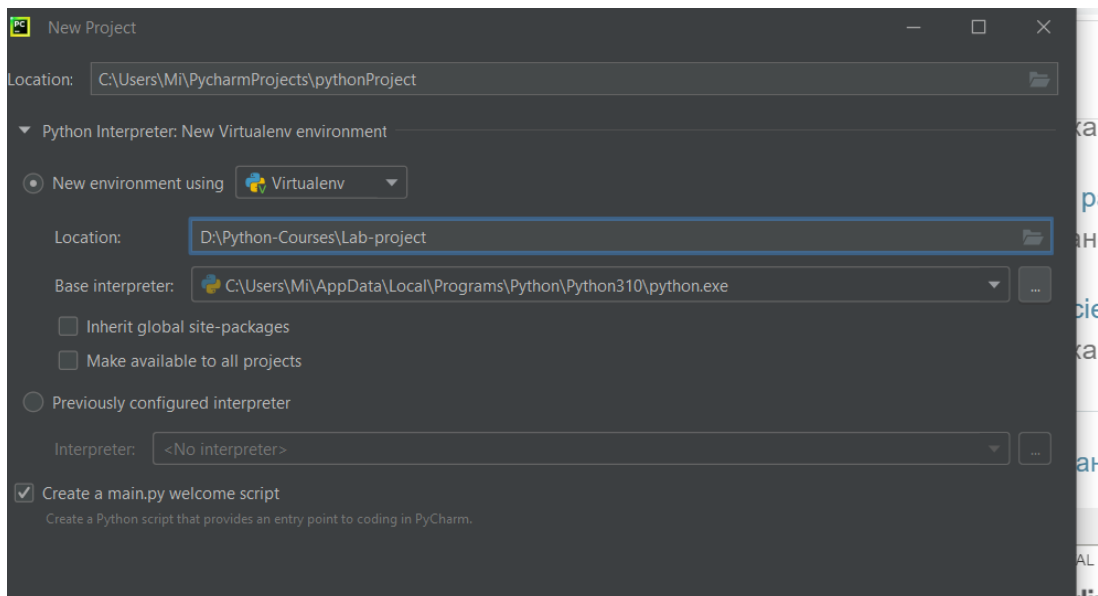
Hello!

Задание 4. Знакомство с PyCharm

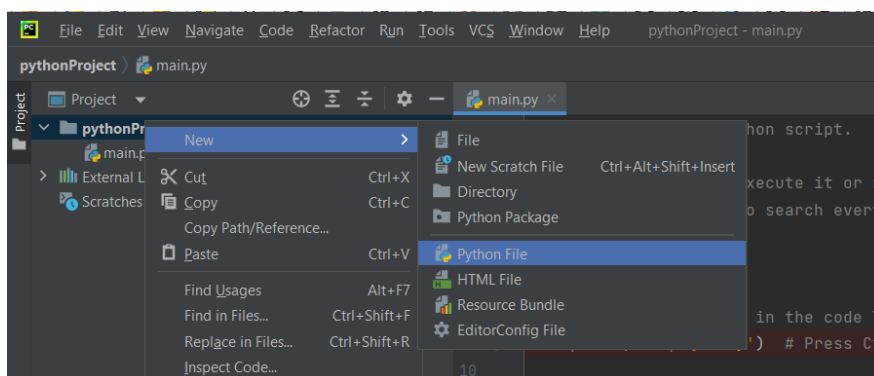
1. Создайте проект

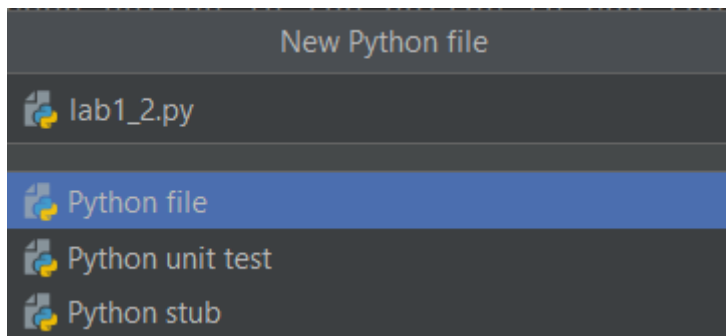


Например, с именем ***Lab-project***:



И в нем файл ***lab1_2.py***

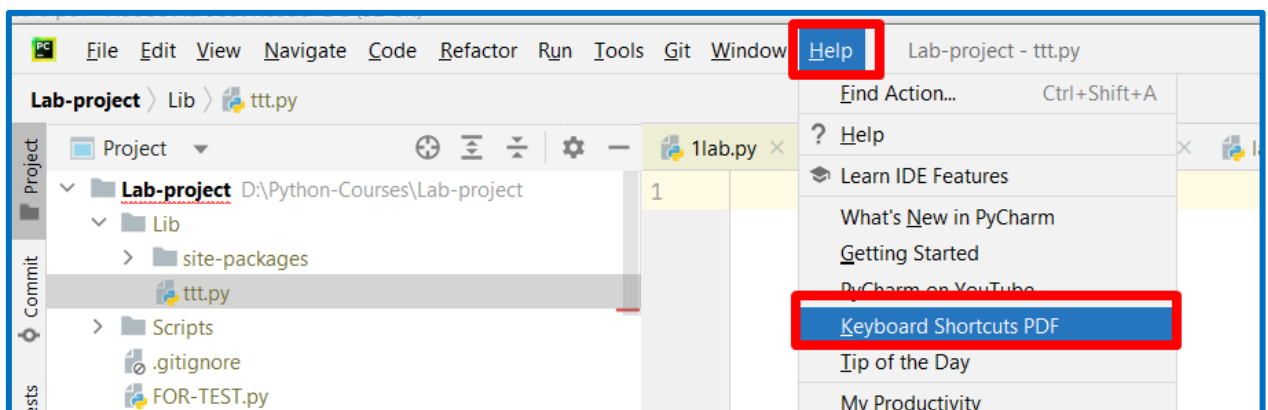




2. Лайфхаки в PyCharm (для Win)

- CTRL-D дублирование строки.
- Ctrl+/ закомментировать строку.
- Ctrl-Alt-L – правильное форматирование по пробелам, отступам и пр.
- Удерживаем Alt и кликаем мышкой по пустым строкам (выделяем их), далее вводим команду – она появится на все выделенных строках.
- Одновременно можно открыть несколько файлов и разделить их отображение по горизонтали или вертикали.
- Если что-то не верно, то подчеркивания появляются- можно посмотреть подсказки ,что не верно.
- через Run запускать дебаг (слева между номером строки и командой можно поставить клик мышкой – красная лампочка – до этого места пойдет отладка).

Можно в формате pdf скачать список «горячих клавиш»:



Find any action inside the IDE Ctrl + Shift + A

CREATE AND EDIT

Show intention actions	Alt + Enter
Basic code completion	Ctrl + Space
Smart code completion	Ctrl + Shift + Space
Type name completion	Ctrl + Alt + Space
Complete statement	Ctrl + Shift + Enter
Parameter information / context info	Ctrl + P / Alt + Q
Quick definition	Ctrl + Shift + I
Quick / external documentation	Ctrl + Q / Shift + F1
Generate code	Alt + Insert
Override / implement members	Ctrl + O / Ctrl + I
Surround with...	Ctrl + Alt + T
Comment with line comment	Ctrl + /
Extend / shrink selection	Ctrl + W / Ctrl + Shift + W
Optimize imports	Ctrl + Alt + O
Auto-indent lines	Ctrl + Alt + I
Cut / Copy / Paste	Ctrl + X / Ctrl + C / Ctrl + V
Copy document path	Ctrl + Shift + C
Paste from clipboard history	Ctrl + Shift + V
Duplicate current line or selection	Ctrl + D
Move line up / down	Ctrl + Shift + Up / Down
Delete line at caret	Ctrl + Y
Join / split line	Ctrl + Shift + J / Ctrl + Enter
Start new line	Shift + Enter
Toggle case	Ctrl + Shift + U
Expand / collapse code block	Ctrl + NumPad + / -
Expand / collapse all	Ctrl + Shift + NumPad + / -
Save all	Ctrl + S

VERSION CONTROL

VCS operations popup...	Alt + `
Commit	Ctrl + K
Update project	Ctrl + T
Recent changes	Alt + Shift + C
Revert	Ctrl + Alt + Z
Push...	Ctrl + Shift + K
Next / previous change	Ctrl + Alt + Shift + Down / Up

MASTER YOUR IDE

Find action...	Ctrl + Shift + A
Open a tool window	Alt + [0-9]
Synchronize	Ctrl + Alt + Y
Quick switch scheme...	Ctrl + `
Settings...	Ctrl + Alt + S
Jump to source / navigation bar	F4 / Alt + Home
Jump to last tool window	F12
Hide active / all tool windows	Shift + Esc / Ctrl + Shift + F12
Go to next / previous editor tab	Alt + Right / Alt + Left
Go to editor (from a tool window)	Esc
Close active tab / window	Ctrl + Shift + F4 / Ctrl + F4

FIND EVERYTHING

Search everywhere	Double Shift
Find / replace	Ctrl + F / R
Find in path / Replace in path	Ctrl + Shift + F / R
Next / previous occurrence	F3 / Shift + F3
Find word at caret	Ctrl + F3
Go to class / file	Ctrl + N / Ctrl + Shift + N
Go to file member	Ctrl + F12
Go to symbol	Ctrl + Alt + Shift + N

NAVIGATE FROM SYMBOLS

Declaration	Ctrl + B
Type declaration (JavaScript only)	Ctrl + Shift + B
Super method	Ctrl + U
Implementation(s)	Ctrl + Alt + B
Find usages / Find usages in file	Alt + F7 / Ctrl + F7
Highlight usages in file	Ctrl + Shift + F7
Show usages	Ctrl + Alt + F7

REFACTOR AND CLEAN UP

Refactor this...	Ctrl + Alt + Shift + T
Copy... / Move...	F5 / F6
Safe delete...	Alt + Delete
Rename...	Shift + F6
Change signature...	Ctrl + F6
Inline...	Ctrl + Alt + N
Extract method	Ctrl + Alt + M
Introduce variable / parameter	Ctrl + Alt + V / P
Introduce field / constant	Ctrl + Alt + F / C
Reformat code	Ctrl + Alt + L

ANALYZE AND EXPLORE

Show error description	Ctrl + F1
Next / previous highlighted error	F2 / Shift + F2
Run inspection by name...	Ctrl + Alt + Shift + I
Type / call hierarchy	Ctrl + H / Ctrl + Alt + H

NAVIGATE IN CONTEXT

Select in...	Alt + F1
Recently viewed / Recent locations	Ctrl + E / Ctrl + Shift + E
Last edit location	Ctrl + Shift + Back
Navigate back / forward	Ctrl + Alt + Left / Right
Go to previous / next method	Alt + Up / Down
Go to line / column...	Ctrl + G
Go to code block end / start	Ctrl +] / [
Add to favorites	Alt + Shift + F
Toggle bookmark	F11
Toggle bookmark with mnemonic	Ctrl + F11
Go to numbered bookmark	Ctrl + [0-9]
Show bookmarks	Shift + F11

BUILD, RUN, AND DEBUG

Run context configuration	Ctrl + Shift + F10
Run / debug selected configuration	Alt + Shift + F10 / F9
Run / debug current configuration	Shift + F10 / F9
Step over / into	F8 / F7
Smart step into	Shift + F7
Step out	Shift + F8
Run to cursor / Force run to cursor	Alt + F9 / Ctrl + Alt + F9
Show execution point	Alt + F10
Evaluate expression...	Alt + F8
Stop	Ctrl + F2
Stop background processes...	Ctrl + Shift + F2
Resume program	F9
Toggle line breakpoint	Ctrl + F8
Toggle temporary line breakpoint	Ctrl + Alt + Shift + F8
Edit / view breakpoint	Ctrl + Shift + F8

jetbrains.com/pycharm
jetbrains.com/help/pycharm
blog.jetbrains.com/pycharm
@pycharm



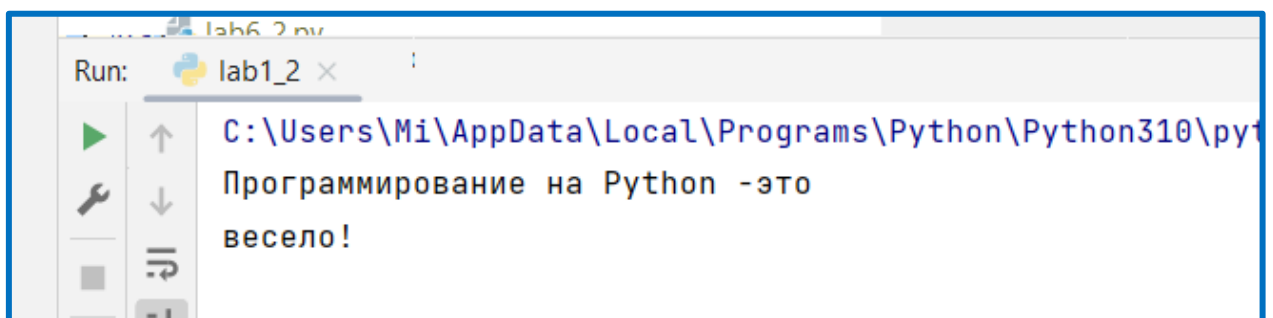
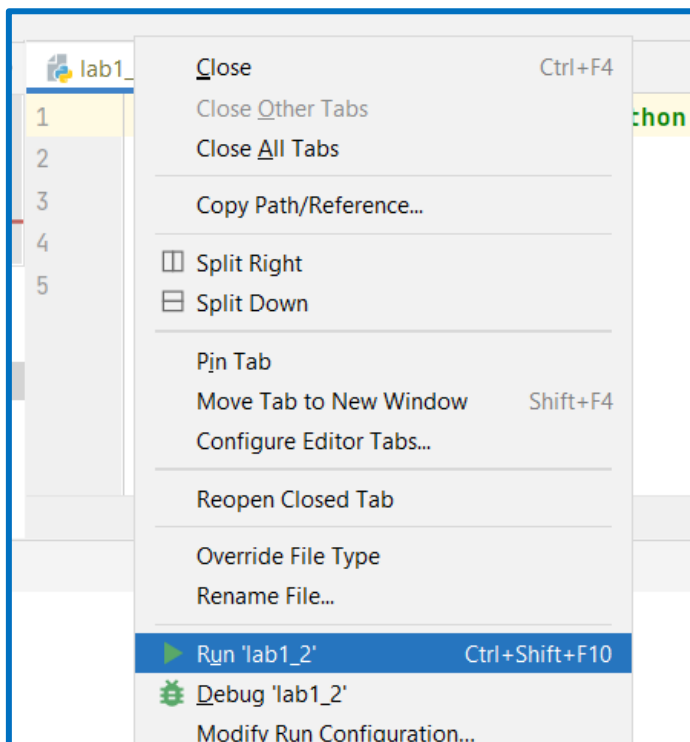
6597

Задание 5. Изучение функции print()

1. Дан код:

```
print('Программирование', 'на Python', '- это')
print('весело!')
```

- Добавьте его в созданный файл lab1_2.py, запустите (в PyCharm навести курсор на название файла и правой кнопкой мыши выбрать Run) и посмотрите результат выполнения (появится в нижней части окна):



2. Отредактируйте код, чтобы получить следующий результат выполнения (должны остаться две команды print, текст в них не меняем, а используем sep и end):

Программирование*на Python***- это...весело!**

Задание 6. Начало работы с input()

1. Напишите программу, которая запрашивает у пользователя 3 ip адреса (поочередно), каждый адрес сохраняет в переменную (ip_1, ip_2, ip_3 соответственно). И выводит результат на экран.
2. Пример, как должна работать программа (здесь пользователь ввел адреса 10.1.1.1, 10.1.1.2, 10.1.1.3):

Введите значение первого ip адреса: 10.1.1.1

Введите значение второго ip адреса: 10.1.1.2

Введите значение третьего ip адреса: 10.1.1.3

Первый адрес: 10.1.1.1

Второй адрес: 10.1.1.2

Третий адрес: 10.1.1.3