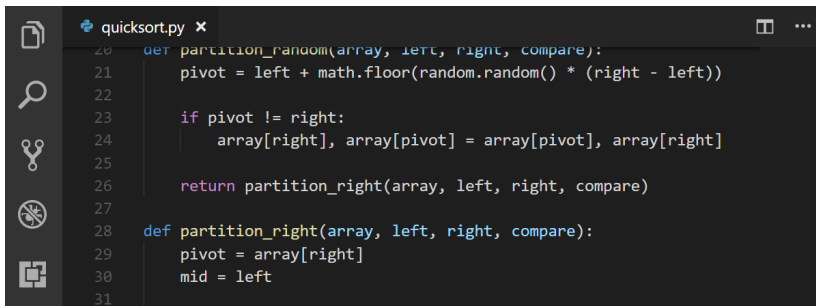


# Занятие 1:

## Введение в язык программирования Python



**Python** - высокоуровневый язык программирования общего назначения с акцентом на производительность разработчика и читаемость кода.

A screenshot of the Visual Studio Code editor interface. The top bar shows the file name 'quicksort.py' with a close button. The left sidebar contains icons for Explorer, Search, Source Control, Run and Debug, and Extensions. The main editor area displays Python code for a quicksort algorithm. The code is as follows:

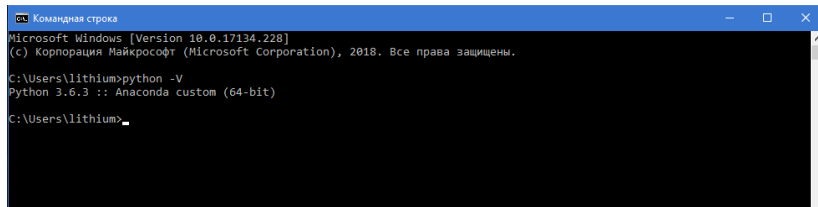
```
20 def partition_random(array, left, right, compare):
21     pivot = left + math.floor(random.random() * (right - left))
22
23     if pivot != right:
24         array[right], array[pivot] = array[pivot], array[right]
25
26     return partition_right(array, left, right, compare)
27
28 def partition_right(array, left, right, compare):
29     pivot = array[right]
30     mid = left
31
```

Рис. 1: Пример типового листинга на языке Python в Visual Studio Code

Данный язык программирования поддерживает несколько парадигм программирования, в том числе структурное, объектно-ориентированное, функциональное и т.д. Основные черты - динамическая типизация, автоматическое управление памятью, поддержка многопоточных вычислений и многое другое. Важной особенностью является то, что не следует смешивать в листингах с кодом табуляцию и отступы с помощью пробелов, иначе это может привести к непредвиденным ошибкам.

# Установка Python

Загрузить установочный файл *Python* необходимой версии можно на официальном ресурсе <https://www.python.org/downloads/>. Существует ряд библиотек, которые работают с *Python 2.7*, не работают с *Python 3* и наоборот. Это связано с тем, что данные версии имеют между собой значительные различия, связанные не только с различным способом вызова *print*, но и прочими, более глубокими структурными изменениями. В ходе установки крайне рекомендуется поставить флаг `ADD TO PATH`, что добавит переменную в переменные среды и предоставит доступ к *Python* из терминала. Чтобы проверить, установлен ли корректно *Python* на ПК и узнать его версию, открываем терминал в Windows и вводим следующую команду:



```
cmd Командная строка
Microsoft Windows [Version 10.0.17134.228]
(c) Корпорация Майкрософт (Microsoft Corporation), 2018. Все права защищены.

C:\Users\lithium>python -V
Python 3.6.3 :: Anaconda custom (64-bit)

C:\Users\lithium>
```

В случае успешной установки будет выведена вся необходимая информация. Если на компьютере установлено несколько версий *Python*, то в команде может быть *python3* или что-то аналогичное.



# Установка Anaconda

Для того, чтобы использовать Python эффективнее и иметь множество заранее установленных базовых библиотек и сопутствующее им ПО, стоит установить бесплатный пакет Anaconda. Актуальную версию можно загрузить по официальному адресу <https://www.anaconda.com/download/> для Python 2.7 или Python 3 соответственно. Также необходимо в ходе установки выбрать параметр Add to PATH.



# Синтаксис Python

```
main.py x
1 import math          #Импорт библиотеки с математическими функциями
2 import collections   #Импорт структур данных
3 import random        #Импорт библиотеки для генерации случайных чисел
4 import greatFunctions #Импорт функций, находящихся в данной папке в другом питон файле
5
6 #функция
7 def getTrueRandomNumber():
8     k = 6;
9     l = 7;
10    k += 1
11    return 4
12
13 #Комментарий
14
15 myString1 = 'Hello'      #Создание строки
16 myString2 = '%Username!%' #Создание другой строки
17
18 print("getTrueRandomNumber returns ", getTrueRandomNumber()) #Вызов print в Python 3 и функции в нём
19
20 myStrings = [myString1, myString2]; #Создание массива строк
21
22 print(myString1, myString2[9])      #Печать двух строк, причём из второй берется лишь один символ
23
24 a = 2;                               #Создание целочисленной переменной
25 c = int('3');                       #Создание целочисленной переменной из строки
26
27 #Использование условного оператора if
28 if a > 1:
29     print('c is', c, ' '.join(myStrings));
30 elif a == 1:
31     print("res2");
32 else:
33     print("res3");
```



```
main.py
35
36 #Использование foreach
37 for country in ["Russia", "Poland", "New Zealand"]:
38     print(country)
39
40 #Ввод с консоли, находящийся в многострочном комментарии
41 ...
42 s = input("Enter an integer: ")
43 try:
44     i = int(s)
45     print("Entered integer is ", i)
46 except ValueError:
47     print("Not a number")
48 ...
49
50 print(math.pi, ascii("12345"));          #Использование констант из подключенных библиотек
51
52 #Кортежи (неизменяемые, несортируемые)
53 cor = (1,2,3,4,5)
54 #Именованные кортежи
55 Human = collections.namedtuple("Human", "humanId Weight FootSize MaritalStatus");
56 humans = [];
57 humans.append(Human(44, 56, 39, 'Single'));
58
59 #Списки (изменяемые, сортируемые)
60 L = [1,6,3]
61
62 #random
63 rand = random.randint(1,100)
64 print(rand)
65
66 #Множества (изменяемые, без повторов)
67 Mn = {4,5,2,1}
```

```
strings.txt — Блокнот
Файл  Правка  Формат  Вид  Справка
summer
day
morning
```

```
greatFunctions.py x
1  #Функции в отдельном файле. Это комментарий
2  def getRadiusOfEarth():
3      return 6400
4
5  def getNamedHello(name):
6      return 'Hello, ' + name + '!'
7
```

```
main.py x
69 #Словари
70 myDictionary = dict(id=1980, name='Anna', MaritalStatus='Single')
71
72 for item in myDictionary.items():
73     print(item[0], item[1])
74
75 #Файлы
76 fin = open('strings.txt', encoding="utf8")
77 dataFromFile = fin.readlines()
78 for item in dataFromFile:
79     print(item)
80
81 print(greatFunctions.getNamedHello('Anna'))
82
```

```
ПРОБЛЕМЫ    ВЫВОД    КОНСОЛЬ ОТЛАДКИ    ТЕРМИНАЛ
getTrueRandomNumber returns 4
Hello !
c is 3 Hello %Username!%
Russia
Poland
New Zeland
3.141592653589793 '12345'
59
id 1980
name Anna
MaritalStatus Single
summer

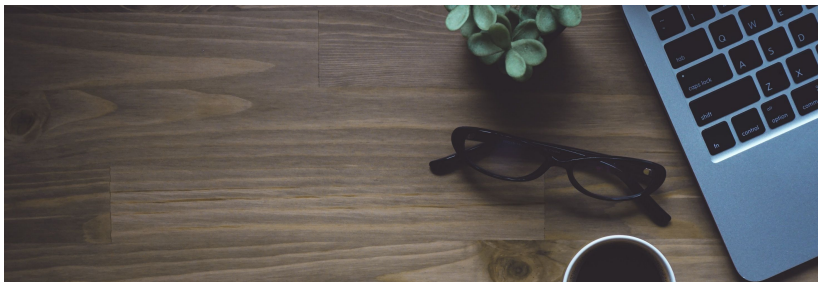
day

morning
Hello, Anna!
```





# Задания для практической работы



1. Написать на Python рекурсивную функцию для получения 17 элемента ряда Фибоначчи (каждое последующее число равно сумме двух предыдущих).
2. Написать на Python рекурсивную функцию получения факториала числа 8.
3. Разделить полученное в пункте 2 на полученное в пункте 1 и умножить на значение числа Пи, взятое в пакете *math*.



# Решение задач

```
task1.py x
1  import math
2
3  def fib(n):
4      if n==1 or n==2:
5          return 1
6      return fib(n-1) + fib(n-2)
7
8  def fac(n):
9      if n == 0:
10         return 1
11     return fac(n-1) * n
12
13  var1 = fib(17)
14  var2 = fac(8)
15  print(var1)
16  print(var2)
17
18  var3 = (var2 / var1) * math.pi
19
20  print(var3)
21
```

ПРОБЛЕМЫ    ВЫВОД    КОНСОЛЬ ОТЛАДКИ    ТЕРМИНАЛ

```
1597
40320
79.31685397165964
```

