

# Введение в Python

SkillFactory

Ракова Валерия

# План вебинара

1. Ввод/ вывод
2. Основные типы данных
3. Условия и циклы
4. Функциональное программирование. Рекурсия
5. Декораторы

# Ввод информации

```
a=input()
```

```
print('Введите данные')
```

```
b=input()
```

```
b=input("Введите данные")
```

```
a=input()
```

```
b=input()
```

```
print(a+b)
```

```
a=int(input())
```

```
b=int(input())
```

```
print(a+b)
```

```
a, b, c = [int(input()) for _ in range(3)]
```

```
a, b, c = map(int,input().split())
```

# Вывод информации

```
print('Hello', end = " | ")
print('world',end = " | ")
print('!')
#Hello | world | !
```

```
print('23','01','2021', sep = " - ")
#23 - 01 - 2021
```

```
name='Lera'
age=25
```

```
#Обычная строка:
```

```
print("Меня зовут " + name + ". Мне " + str(age) + " лет.")
```

```
#Строка с форматированием %:
```

```
print("Меня зовут %(name)s. Мне %(age)d лет." % {"name": name, "age": age})
```

```
# Использование f-строк первой версии
```

```
print('Меня зовут {}. Мне {} лет.'.format(name,age))
```

```
#Использование f-строк последней версии:
```

```
print(f"Меня зовут {name} Мне {age} лет.")
```

# Вывод информации. Литералы строк

```
print('Hello \n world!')
```

```
#Hello
```

```
# world!
```

```
print('Hello \t world!')
```

```
#Hello    world!
```

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

```
#Hello " world!
```

```
print('Hello \' world!')
```

```
#Hello ' world!
```

```
print('''Hello  
    world ''')
```

```
#Hello
```

```
#        world
```

## Типы данных. Числа

Задача 1. Электронные часы показывают время в формате `hh:mm:ss`, то есть сначала записывается количество часов, потом обязательно двузначное количество минут, затем обязательно двузначное количество секунд.

Количество минут и секунд при необходимости дополняются до двузначного числа нулями.

С начала суток прошло `n` секунд. Выведите, что покажут часы.

# Типы данных. Числа. Решение задачи 1

```
n = 10745
s = n % 60
n = n // 60
m = n % 60
h = n // 60

print(f"{h:02d}:{m:02d}:{s:02d}")
```

# Типы данных. Строки. Срезы

```
s="Утро, день, ночь, вечер".split(',')
#['Утро', ' день', ' ночь', ' вечер']
txt = "....apple...."
x = txt.strip('.')
print("of all fruits", x, "is my favorite")
#of all fruits apple is my favorite
```

```
txt='Hello world!'
txt[0] #первый элемент
```

```
#срез - [START:STOP:STEP]
txt[:] #вся строка
txt[2:] #?
txt[:2] #'Hlowrd'
txt[-1] #последний элемент
txt[:-2] #?
txt[::-1] #?
```



# Типы данных. Словари

```
passwords = {  
    "user" : "12345",  
    "root" : "10 101",  
    "admin": "aaa"  
}
```

```
passwords.keys()  
passwords.values()  
passwords.get("user")
```

#перебор элементов словаря

```
for name, password in passwords.items():  
    print(name +password)
```

#Задача 2. Как проверить пароль при входе в систему?

## Типы данных. Словари. Решение задачи 2

```
nick = input("Введите ваше имя: ")
pwd = input("Введите ваш пароль: ")
if nick in passwords:
    if passwords[nick] == pwd:
        print("Вы вошли в систему!")
    else:
        print("Неверный пароль!")
else:
    print("Такого пользователя нет!")
```

# Типы данных. Множества

#Задача 3. Есть список выигрышных номеров в лотерее.  
#Как проверить, что билет содержит все нужные номера?

```
positions=[[0,1,2],[0,4,8],[2,4,6]]
ticket1=[0,1,3,5,5,6]
#ticket1=[0,2,4,5,5,6] #?
for p in positions:
    if len(set(ticket1).intersection(set(p)))==3: #set(ticket1)&set(p)
        print(f'Ticket {ticket1} is win!')
```

```
a=set([1,2,3,4])
b=set([3,4,5,6])
print(a&b) #?
print(a|b) #?
print(a.union(b)) #?
print(a.difference(b)) # {1,2}
print(b.difference(a)) # ?
```

# Условия

```
# Задача 4. Вводятся два целых числа. Проверить делится ли первое на второе.  
# Вывести остаток, если не делится  
a = int(input())  
b = int(input())  
if a % b == 0:  
    print("%d делится на %d" % (a, b))  
else:  
    print("%d не делится на %d" % (a, b))  
    print("Остаток: %d" % (a % b))  
# elif ??
```

# Циклы

```
# Задача 5. Вывести на экран столько элементов ряда Фибоначчи,  
# сколько указал пользователь.  
n = int(input())  
f1 = 1  
f2 = 2  
print(f1, f2, end=" ")  
for i in range(3, n+1):  
    print(f1+f2, end=" ")  
    b = f1  
    f1 = f2  
    f2 = b+f1  
print()
```

# Циклы

```
#Задача 6. Посчитать, сколько раз встречается определенная  
#цифра в введенной последовательности чисел.
```

```
n = 10
```

```
d = 5
```

```
count = 0
```

```
for i in range(1,n+1):
```

```
    m = int(input("Число " + str(i) + ": "))
```

```
    while m > 0:
```

```
        if m%10 == d:
```

```
            count += 1
```

```
        m = m // 10
```

```
print("Было введено %d цифр %d" % (count, d))
```

# Циклы

```
#Задача 7. Человек берет кредит в банке на сумму `М` рублей.  
#Процентная ставка `r%`. Сумма, которую человек ежегодно выплачивает,  
# `K` рублей. Через сколько лет человек погасит кредит?  
M = 1000  
K = 70  
r = 6  
  
count = 0  
while M > 0:  
    count += 1  
    M += M * (r/100)  
    M -= K  
  
print(count)
```

# Функции

```
def f(a, b=2):  
    return a + b
```

```
print(f(3,4))  
print(f(3))
```

```
def f2(*args):  
    res=0  
    for i in args:  
        res+=i  
    return res
```

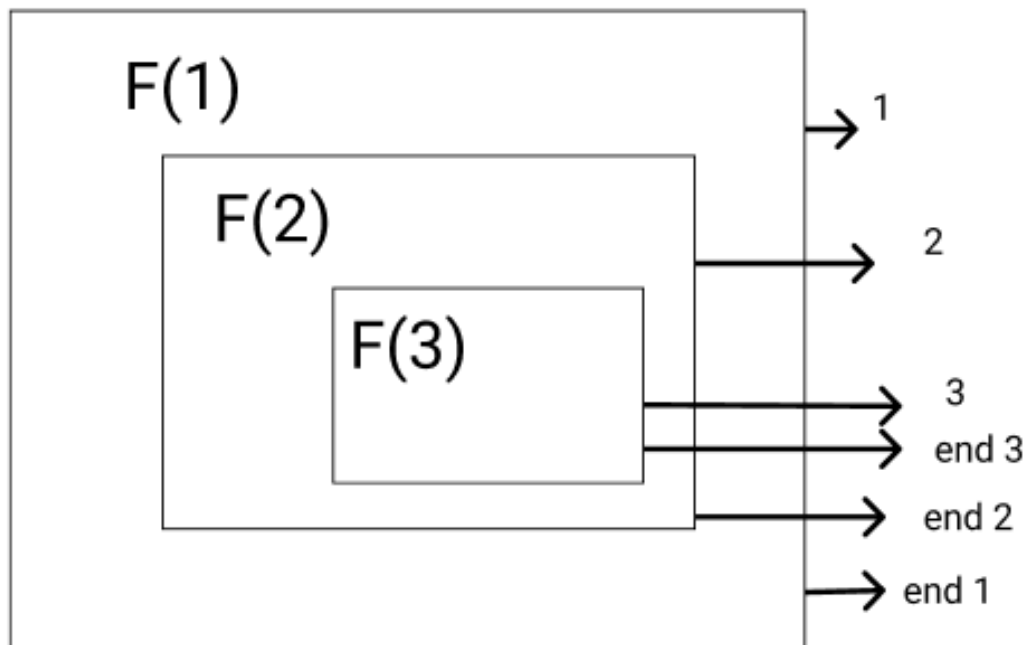
```
print(f2(3,4))  
print(f2(3))  
print(f2(3,4,5,6,7,8))
```



# Рекурсия

```
def F(n):  
    print(n)  
    if n<3:  
        F(n+1)  
    print(f'end {n}')
```

F(1)



# Рекурсия

Задача 8. Дано натуральное число `N`. Выведите слово `YES`, если число `N` является точной степенью двойки, или слово `NO` в противном случае.

```
N = 128
```

```
def check(a):  
    if a % 2 == 0:  
        return check(a // 2)  
    else:  
        if a == 1:  
            return True  
        else:  
            return False
```

```
check(N)
```

# Декораторы

# Задача 9. Напишите декоратор, который печатает имя функции при ее вызове

```
def my_decorator(fn):  
    def wrapper():  
        print(fn.__name__)  
        return fn()  
    return wrapper
```

```
@my_decorator  
def myf():  
    return 1
```

```
print(myf())
```

```
def myf2():  
    return 2
```

```
myf2=my_decorator(myf2)  
print(myf2())
```

**Спасибо за внимание**