

Создание меню

Часто приходится создавать меню для работы с программой. Создадим такое меню для группы. Список студентов в начале сделаем пустым, в пункте меню присвоим значение None.

In [5]:

```
# Menu
"""
Перменная group - это список группы. Переменная choice
указывает пункт меню, который будет выбираться пользователем. Она
инициализируется значением None
Цикл while будет выполняться пока пользователь не введет значение 0
Обычно в меню он стоит последним
При выборе 1 компьютер выведет список студентов или текст
"Список пустой", если он еще не сохдан
При выборе 2 мы добавляем имя студента в конец списка.
При выборе 3 компьютер запрашивает имя удаляемого студента. Далее проверятся, есть ли в список
group такое имя.
Если есть, то удаляется это имя. Первое, когда их несколько Одинаковых.
Если имя не найдено, то об этом сообщается
Метод remove удаляет по значению, а del - по индексу
"""

group = []
choice = None
while choice != "0":
    print(
        """
        1 - Список студентов
        2 - Добавить студента
        3 - удалить студента

        0 - выход
        """
    )
    choice = input("Выбор: ")
    if choice == "0":
        print("Exit")
    elif choice == "1":
        if group == []:
            print("Список пустой")
        else:
            for name in group:
                print(name)
    elif choice == "2":
        name = input("Добавить имя студента: ")
        group.append(name)
    elif choice == "3":
        name = input("Имя удаляемого студента: ")
        if name in group:
            group.remove(name)
        else:
            print("В группе нет студента с таким именем:", name)
    input("\nНажминте Enter")
```

1 - Список студентов
2 - Добавить студента
3 - удалить студента

0 - выход

Выбор: 2
Добавить имя студента: Vitya

1 - Список студентов
2 - Добавить студента
3 - удалить студента

0 - выход

Выбор: 1
Vitya

- 1 - Список студентов
- 2 - Добавить студента
- 3 - удалить студента

0 - выход

Выбор: 3

Имя удаляемого студента: 132

В группе нет студента с таким именем: 132

- 1 - Список студентов
- 2 - Добавить студента
- 3 - удалить студента

0 - выход

Выбор: 3

Имя удаляемого студента: Vitya

- 1 - Список студентов
- 2 - Добавить студента
- 3 - удалить студента

0 - выход

Выбор: 1

Список пустой

- 1 - Список студентов
- 2 - Добавить студента
- 3 - удалить студента

0 - выход

Выбор: 0

Exit

Нажминте Enter

Out[5]:

"

Многоступенчатые циклы

Метод Горнера

In [9]:

```
n = int(input("input exponent of polynom: "))
a = [0] * (n + 1)
m = int(input("input number of points: "))
x = [0] * m
y = [0] * m

# ввод по одному числу в строке
print("input ", n + 1, " of coefficient of the polynom: ")
for i in range(n+1):
    a[i] = float(input())
print(a)
print("input meaning of the ", m, " points")
for i in range(m):
    x[i] = float(input())
print(x)

for k in range(m):
    y[k] = a[n-1]
    for i in range(n-2, -1, -1):
        y[k] = y[k] * x[k] + a[i]
print("argument function")
for k in range(m):
    print('{:6.2f} {:13.5f}'.format(x[k], y[k]))
```

input exponent of polynom: 3

input number of points: 1
input 4 of coefficient of the polynom:
5
4
3
2
[5.0, 4.0, 3.0, 2.0]
input meaning of the 1 points
1
[1.0]
argument function
1.00 12.00000

Сортировка вставками

In [14]:

```
n = int(input("input size of list: "))
x = [0] * n
print("input list one number in string")
x = [int(input()) for i in range(n)]
print(x)
for i in range(n-1):
    for j in range(n-2, i-1, -1):
        if x[j] > x[j+1]:
            x[j], x[j+1] = x[j+1], x[j]
print(x)
```

input size of list: 5
input list one number in string
1
2
3
4
5
[1, 2, 3, 4, 5]
[1, 2, 3, 4, 5]
[1, 2, 3, 4, 5]
[1, 2, 3, 4, 5]
[1, 2, 3, 4, 5]
[1, 2, 3, 4, 5]

In [16]:

```
print("input list in string")
x = list(map(int, input().split()))
print(x)
n = len(x)
for i in range(n):
    for j in range(n-i-1):
        if x[j] > x[j+1]:
            x[j], x[j+1] = x[j+1], x[j]
print(x)
```

input list in string
5 4 3 2 1
[5, 4, 3, 2, 1]
[1, 2, 3, 4, 5]

In [17]:

```
print("input list in string")
x = list(map(int, input().split()))
print(x)
n = len(x)
for i in range(n-1):
    nmin = i
    for j in range(i+1, n):
        if x[j] < x[nmin]:
            nmin = j
    x[i], x[nmin] = x[nmin], x[i]
print(x)
```

input list in string
1 2 5 4 3
[1, 2, 5, 4, 3]

[1, 2, 3, 4, 5]

In [20]:

```
a = list(map(int, input("введи массив a: ").split()))
b = list(map(int, input("input list b: ").split()))
print(a, " ", b)
n = len(a)
m = len(b)
c = [0] * (n+m)
i = 0; j = 0
while i <= (n-1) and j <= (m-1):
    if a[i] <= b[j]:
        c[i+j] = a[i]
        i += 1
    else:
        c[i+j] = b[j]
        j += 1
while i <= (n-1):
    c[m+i] = a[i]
    i += 1
    j += 1
while j <= (m-1):
    c[n+j] = b[j]
    j += 1
print(c)
```

```
введи массив a: 1 8 9 10
input list b: 1 9 10 20
[1, 8, 9, 10]   [1, 9, 10, 20]
[1, 1, 8, 9, 9, 10, 10, 20]
```

In [23]:

```
a = list(map(int, input("введи массив a: ").split()))
b = list(map(int, input("input list b: ").split()))
print(a, " ", b)
n = len(a)
m = len(b)
c = [0] * (n+m)
if a[n-1] < b[m-1]:
    a.append(b[m-1]+1)
else:
    b.append(b[n-1]+1)
i = 0; j = 0
for k in range(n + m):
    if a[i] <= b[j]:
        c[k] = a[i]
        i += 1
    else:
        c[k] = b[j]
        j += 1
print(c)
```

```
введи массив a: 1 2 3 4 5
input list b: 1 2 3 4 5
[1, 2, 3, 4, 5]   [1, 2, 3, 4, 5]
```

```
-----
IndexError                                Traceback (most recent call last)
<ipython-input-23-1ab5f84f5063> in <module>()
      7 i = 0; j = 0
      8 for k in range(n + m):
----> 9     if a[i] <= b[j]:
     10         c[k] = a[i]
     11         i += 1
```

IndexError: list index out of range