

## Задания

1. Данна строка `['Adam', 'Bob', 'Cindy']`. Получить из нее массив. Вывести на экран элементы массива последовательно.
2. Дан объект `{name: "Adam", age: 25}`. Преобразовать этот объект в JSON, предварительно изменив в каждом поле имя свойства на значение, а значение – на имя свойства. Вывести этот строковый объект на экран.
3. Дан объект `{three: 3, two: 2, five: 5, four: 4}`. Отсортировать его по возрастанию значений и преобразовать в JSON с отступом от левого края, равным 2.
4. Создать функцию, которая принимает строку в формате '`{ключ1: значение1, ключ2: значение2, ... ключN: значениеN}`' и возвращает JSON, содержащий ключ и соответствующий ему угол сектора на круговой диаграмме (pie-chart).

Пример работы:

```
func('{"A": 4, "B" : 5, "C" : 6, "D" : 1, "E" : 4}') //  
{"A":72,"B":90,"C":108,"D":18,"E":72}
```

5. Используя **один** метод, добавить в пустой объект 3 свойства, значения которых нельзя перезаписать. Все 3 свойства должны быть доступны в циклах, но не удаляемы.
6. Реализовать функцию-конструктор **User()**, возвращающий объекты с неперезаписываемым и неудаляемым свойством **name**, значением которого является ваше имя. Свойство **name** должно быть доступно для перебора в цикле.
7. Создать функцию, которая принимает в качестве параметров массивы **A** и **B** с элементами-строками и возвращает массив элементов **A**, которые являются подстроками элементов **B**. Возвращаемый массив отсортировать в лексикографическом порядке. При решении использовать коллекцию **Set**.

Пример работы:

```
A = ['some', 'less', 'hold', 'absence'];  
B = ['stronghold', 'wireless', 'seamless', 'lonesome', 'troublesome',  
'script', 'algorithm'];  
func(A, B) // ['hold', 'less', 'some']
```