

## Работа с двоичными данными в Python

Данные Python → массив байт

```
import struct  
  
binary_data = struct.pack("4sfi", "INFO", 3.14, 42)
```

*Символы форматирования:*

| Символ | Тип    | Размер (байт) | Комментарий  |
|--------|--------|---------------|--|
| x      | -      | 1             | Ничего не значащий байт, служит для выравнивания   |
| c      | bytes  | 1             | 1 символ   |
| b      | int    | 1             | Целое однобайтовое знаковое  |
| B      | int    | 1             | Целое однобайтовое беззнаковое   |
| ?      | bool   | 1             | Логическое значение (True или False)   |
| h      | int    | 2             | Целое двухбайтовое знаковое  |
| H      | int    | 2             | Целое двухбайтовое беззнаковое   |
| i      | int    | 4             | Целое четырехбайтовое знаковое   |
| I      | int    | 4             | Целое четырехбайтовое беззнаковое  |
| l      | int    | 4             | Целое четырехбайтовое знаковое   |
| L      | int    | 4             | Целое четырехбайтовое беззнаковое  |
| q      | int    | 8             | Целое восьмибайтовое знаковое  |
| Q      | int    | 8             | Целое восьмибайтовое беззнаковое   |
| f      | float  | 4             | Вещественное одинарной точности  |
| d      | double | 8             | Вещественное двойной точности  |
| s      | bytes  |               | Строка фиксированного размера; размер указывается пользователем перед символом s, например, 10s значит строку из 10 символов |
| p      | bytes  | 1...256       | Строка изменяемого размера, хранит свою длину в первом байте   |

Массив байт → данные Python

```
import struct  
  
name, fdata, idata = struct.unpack("4sfi", binary_data)
```