

# Реализовать модуль дешифрования данных клиента

Обратите внимание!

```
1  const customers = [
2    {
3      payload: {
4        name: 'Pitter Black',
5        email: '70626c61636b40656d61696c2e636f6d',
6        password: '70626c61636b5f313233'
7      },
8      meta: {
9        algorithm: 'hex'
10     }
11   }
12 ];
13
14
15 const ui = new Ui(customers);
16 const decryptor = new Decryptor();
17 const manager = new AccountManager();
18
19 ui.pipe(decryptor).pipe(manager);
```

1. Реализовать класс **Ui** который будет имплементировать **Readable** интерфейс и будет служить поставщиком данных. Он не будет производить никакого шифрования.
2. Реализовать **Decryptor** который будет реализовывать интерфейс **Transform**.
3. **Decryptor** на вход будет принимать объект с зашифрованными полями **name** и **password** и выполнять их обратное преобразование в строку согласно алгоритма определённого в **meta.algorithm**.

```
1  // Было
2  {
3    payload: {
4      name: 'Pitter Black',
5      email: '70626c61636b40656d61696c2e636f6d',
6      password: '70626c61636b5f313233'
7    },
8    meta: {
9      algorithm: 'hex'
10   }
11 }
12
```

```
13 // Стало
14 {
15     name: 'Pitter Black',
16     email: 'pblack@email.com',
17     password: 'pblack_123'
18 }
```

Обратите внимание!

1. Алгоритм может быть одним из двух **hex** или **base64**. Генерировать ошибку если был передан другой алгоритм.
2. Генерировать ошибку если структура объекта не соответствует.
3. Генерировать ошибку если какое-то из полей объекта содержит пустую строку или любой другой тип отличный от строки.