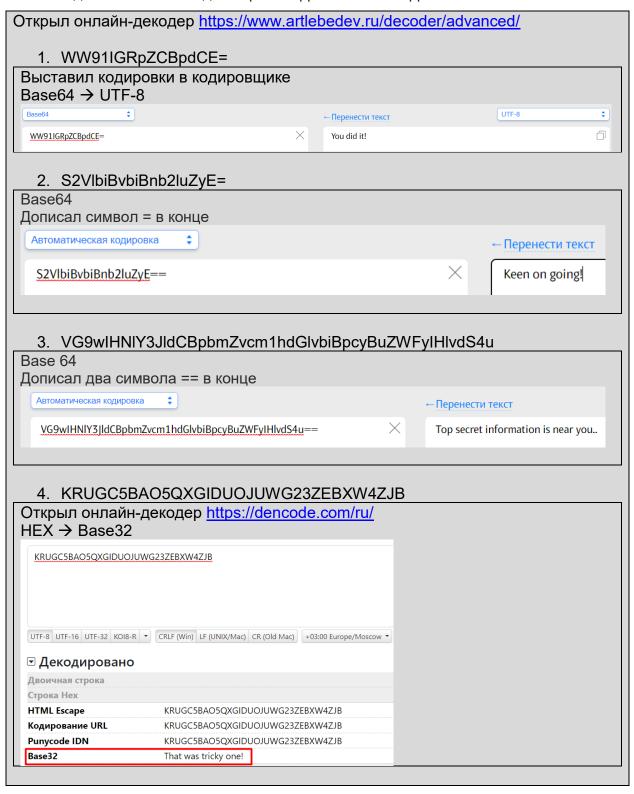
### Задание выполнил студент группы 2345 Романенко Кирилл

#### Задание 1

Одна из кодировок не разбиралась на уроке и чтобы ее раскодировать нужно немного поискать.

В этом задании вам необходимо "расшифровать" 5 "шифротекстов":





В результате вы получите 5 осмысленных фраз на английском языке.

#### Задание 2

"Расшифруйте" следующие "шифротексты":

1. 4c6f72656d20497073756d2069732073696d706c792064756d6d79207465787
4206f6620746865207072696e74696e6720616e64207479706573657474696e
6720696e6475737472792e
HEX
4c6f72656d20497073756d2069732073696d706c792064756d6d792074657874206f6620746865207072696e74696e6720616e64207479706573657474696e6720696e64757
<u>37472792e</u>
UTF-8 UTF-16 UTF-32 KOI8-R  CRLF (Win) LF (UNIX/Mac) CR (Old Mac) +03:00 Europe/Moscow
<b>□</b> Декодировано
Двоичная строка
Строка Hex Lorem Ipsum is simply dummy text of the printing and typesetting industry.
2. 436865636b206f757420746869732074616c6b20696620796f75206861766e27
2. 1000000000000000000000000000000000000
7420646f6e6520736f2068747470733a2f2f7777772e796f75747562652e636f6d
7420646f6e6520736f2068747470733a2f2f7777772e796f75747562652e636f6d 2f77617463683f763d6d4b535136446a427a3377
7420646f6e6520736f2068747470733a2f2f7777772e796f75747562652e636f6d
7420646f6e6520736f2068747470733a2f2f7777772e796f75747562652e636f6d 2f77617463683f763d6d4b535136446a427a3377 HEX  436865636b206f757420746869732074616c6b20696620796f75206861766e277420646f6e6520736f2068747470733a2f2f7777772e796f75747562652e636f6d2f7761746368
7420646f6e6520736f2068747470733a2f2f7777772e796f75747562652e636f6d 2f77617463683f763d6d4b535136446a427a3377
7420646f6e6520736f2068747470733a2f2f7777772e796f75747562652e636f6d 2f77617463683f763d6d4b535136446a427a3377 HEX  436865636b206f757420746869732074616c6b20696620796f75206861766e277420646f6e6520736f2068747470733a2f2f7777772e796f75747562652e636f6d2f7761746368
7420646f6e6520736f2068747470733a2f2f7777772e796f75747562652e636f6d 2f77617463683f763d6d4b535136446a427a3377 HEX  436865636b206f757420746869732074616c6b20696620796f75206861766e277420646f6e6520736f2068747470733a2f2f7777772e796f75747562652e636f6d2f7761746368
7420646f6e6520736f2068747470733a2f2f7777772e796f75747562652e636f6d 2f77617463683f763d6d4b535136446a427a3377 HEX  436865636b206f757420746869732074616c6b20696620796f75206861766e277420646f6e6520736f2068747470733a2f2f7777772e796f75747562652e636f6d2f7761746368
7420646f6e6520736f2068747470733a2f2f7777772e796f75747562652e636f6d 2f77617463683f763d6d4b535136446a427a3377  HEX  436865636b206f757420746869732074616c6b20696620796f75206861766e277420646f6e6520736f2068747470733a2f2f7777772e796f75747562652e636f6d2f7761746368 31763d6d4b535136446a427a3377
7420646f6e6520736f2068747470733a2f2f7777772e796f75747562652e636f6d 2f77617463683f763d6d4b535136446a427a3377  HEX  436865636b206f757420746869732074616c6b20696620796f75206861766e277420646f6e6520736f2068747470733a2f2f7777772e796f75747562652e636f6d2f7761746368 3f763d6d4b535136446a427a3377  UTF-8 UTF-16 UTF-32 KOI8-R * CRLF (Win) LF (UNIX/Mac) CR (Old Mac) +03:00 Europe/Moscow *

В результате вы получите осмысленные фразы на английском языке.

# Задание 3

Во всех последующих заданиях вы будете работать с "сырыми" байтами. Однако у большей части байт нет соответствующего печатного символа (например, 0х00 будет в лучшем случае напечатан как пробел), поэтому данные для заданий будут даваться в кодировках base64 и hex.

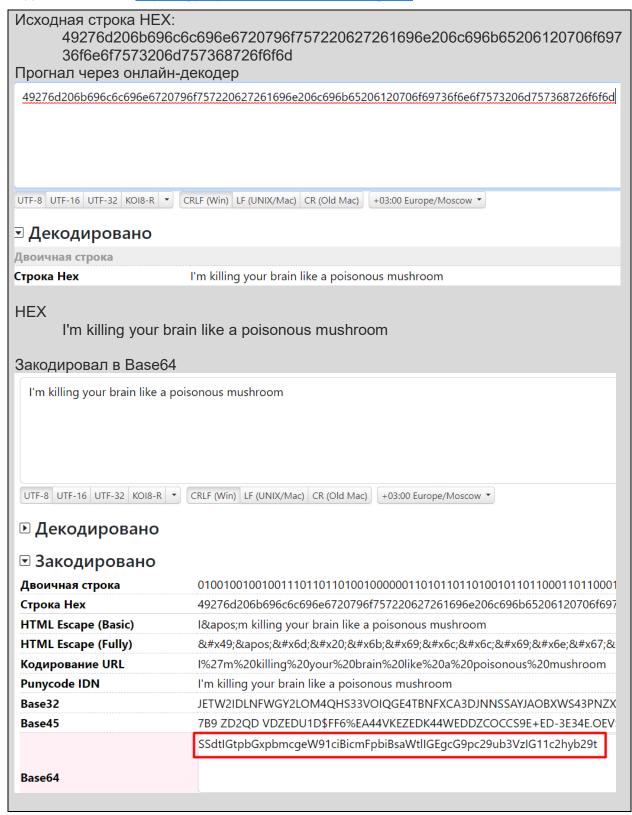
Декодируйте строку:

49276d206b696c6c696e6720796f757220627261696e206c696b65206120706f69736f6e6f7 573206d757368726f6f6d

Результат представьте в base64. На выходе должно получиться:

SSdtIGtpbGxpbmcgeW91ciBicmFpbiBsaWtlIGEgcG9pc29ub3VzIG11c2hyb29t

Задание взято из http://cryptopals.com/sets/1/challenges/1



## Задание 4

Напишите функцию, которая принимает на вход две последовательности байт одинаковой длины и возвращает их побайтовый XOR.

Ваша функция работает правильно, если приняв на вход hex-декодированные 1c0111001f010100061a024b53535009181c и 686974207468652062756c6c277320657965 вернет hex-декодированное значение: 746865206b696420646f6e277420706c6179

В следующем здании мы будем искать уязвимости в нашем свежеиспеченном шифре!

```
def xor(a: str, b: str) -> str:
    if len(a) != ien(b):
        return raise ValueError("Input strings not equal length")

hex(int(a, 16) ^ int(b, 16))

def main():
    hexl = '1c0111001f010100061a024b53535009181c'
    hex2 = '686974207468652062756c6c277320657965'
    ans = '746865206b696420646f6e277420706c6179'

result = xor(hexl, hex2)
    print[f"correct? {ans == result}, result: {result}")

if __name__ == '__main__':
    main()
```