# Cryptography Help in Python3

## **Convert HEX to Raw Bytes**

>>> bytes.fromhex('11aa22bb33cc44dd')

b'\x11\xaa"\xbb3\xccD\xdd'

#### **Convert Raw Bytes to HEX**

>>> b'\x11\xaa"\xbb3\xccD\xdd'.hex()

'11aa22bb33cc44dd'

### **Convert ASCII string to Raw Bytes**

>>> 'This is a typical string'.encode()

b'This is a typical string'

### **Convert ASCII string to HEX**

>>> 'This is a typical ascii string'.encode()

b'This is a typical ascii string'

>>> b'This is a typical ascii string'.hex()

'546869732069732061207479706963616c20617363696920737472696e67'

## **Convert Raw Bytes to ASCII string**

>>> b'This is a typical ascii string'.decode()

'This is a typical ascii string'

### **Convert random Raw Bytes to ASCII string**



**This cannot be done!** Random raw bytes do not correspond to ascii or utf-8 codes.

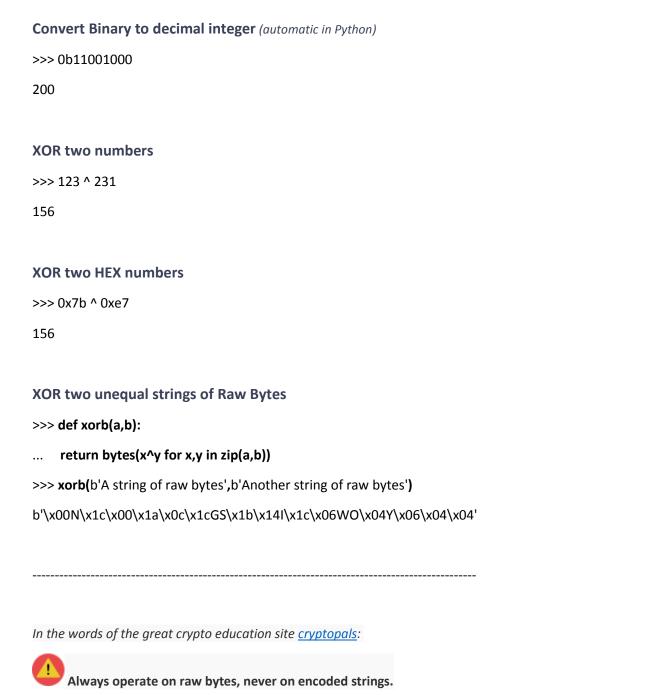
>>> b'\x11\xaa"\xbb3\xccD\xdd'.decode()

Traceback (most recent call last):

File "<stdin>", line 1, in <module>

UnicodeDecodeError: 'utf-8' codec can't decode byte 0xaa in position 1: invalid start byte

```
Find the decimal ASCII code of a character
>>> ord('A')
65
Find the character from an ASCII code
>>> chr(65)
'A'
Convert an integer in range(0, 256) into a Raw Byte
>>> bytes([200])
b'\xc8'
Convert a Raw Byte into an integer
>>> int(b'\xc8'.hex(),16)
200
>>> ord(b'\xc8')
200
Convert decimal integer to HEX
>>> hex(200)
'0xc8'
Convert HEX to decimal integer (automatic in Python)
>>> 0xc8
200
Convert decimal integer to Binary
>>> bin(200)
'0b11001000'
```



**Only use hex and base64 for pretty-printing.** This means convert everything to raw bytes first, do the necessary manipulation to break the code or encrypt the message and only at the end convert back to ascii or hex for display. This prevents a lot of confusion and programming bugs.