

## 23. Codificación de caracteres

Tuesday, November 3, 2020 4:38 PM

La codificación de es el método que permite convertir un carácter de un lenguaje natural (alfabeto) en un símbolo de otro sistema de representación, como un número o una secuencia de pulsos eléctricos en un sistema electrónico, aplicando normas o reglas de codificación.

Ejemplos de codificaciones de caracteres:

- ASCII - Código estándar Estadounidense para el Intercambio de Información
- UNICODE - Estándar de codificación de caracteres diseñado para facilitar el tratamiento informático, transmisión y visualización de textos de numerosos idiomas y disciplinas técnicas, además de textos clásicos de lenguas muertas.
- Shift JIS (para japonés)

### ASCII a decimal

```
import binascii

def to_hex(t, nbytes):
    """Format text t as a sequence of nbytes long values
    separated by spaces.
    """
    chars_per_item = nbytes * 2
    hex_version = binascii.hexlify(t)
    return b' '.join(
        hex_version[start:start + chars_per_item]
        for start in range(0, len(hex_version), chars_per_item)
    )

if __name__ == '__main__':
    print(to_hex(b'abcdef', 1))
    print(to_hex(b'abcdef', 2))
    print(to_hex(b'abcdef', 3))
```

### UNICODE y UTF

```
import unicodedata
from codecs_to_hex import to_hex

text = 'français'

print('Raw : {!r}'.format(text))
for c in text:
    print('  {!r}: {}'.format(c, unicodedata.name(c)))
print('UTF-8 : {!r}'.format(to_hex(text.encode('utf-8'), 1)))
print('UTF-16: {!r}'.format(to_hex(text.encode('utf-16'), 2)))
```

### Base64

Es un sistema de codificación usada para correos electrónicos y otras aplicaciones.

```
import base64
```

```
message = "Python is fun"
message_bytes = message.encode('ascii')
print(message_bytes)
base64_bytes = base64.b64encode(message_bytes)
print(base64_bytes)
base64_message = base64_bytes.decode('ascii')

print(base64_message)
```