

Base 64 python

¿Qué es base 64?

Base 64 es un [sistema de numeración posicional](#) que usa 64 como base. Es la mayor potencia que puede ser representada usando únicamente los caracteres imprimibles de [ASCII](#). Esto ha propiciado su uso para codificación de [correos electrónicos](#), [PGP](#) y otras aplicaciones

Ejemplo:

Una cita de [Thomas Hobbes](#) perteneciente a la obra [Leviathan](#):

Man is distinguished, not only by his reason, but by this singular passion from other animals, which is a lust of the mind, that by a perseverance of delight in the continued and indefatigable generation of knowledge, exceeds the short vehemence of any carnal pleasure.

Se codifica en *base64* como sigue:

TWTFuIG1zIGRpc3Rpbmd1aXNoZWQsIG5vdCBvbmx5IGJ5IGhpcyByZWZzb24sIGJldCBieSB0aGlz
IHNpbmd1bGFyIHBhc3Npb24gZnJvbSBvdGhlciBhbmltYWxzLCB3aGljaCBpcyBhIGxlc3Qgb2Yg
dGhlIG1pbmQsIHRoYXQgYnkgYSBwZXJzZXZlcmFuY2Ugb2YgZGVsaWdodCBpb1B0aGUgY29udGlu
dWVkiGFuZCBpbmRlZmF0aWdhYm91IGdlbmV5YXRpb24gb2Yga25vd2x1ZGdlLCBlc3Qgb2YgZGVsaWdodCBpb1B0aGUgY29udGlu
ZSBzaG9ydCB2ZWhlbWVuY2Ugb2YgYW55IGNhcm5hbCBwbGVhc3VyZS4=

En la cita de arriba el valor codificado de *Man* es TWFu. Codificadas en ASCII, las letras: M, a y n son almacenadas como los [bytes](#) 77, 97 y 110, es decir, 01001101, 01100001, 01101110 en base 2.

Ahora esos tres [bytes](#) se unen y tenemos el búfer de 24 [bits](#), que será 010011010110000101101110. Este número se convertirá a su valor Base 64, que puede hacerse tomando bloques de 6 [bits](#) a la vez (6 [bits](#) forman como máximo 64 valores diferentes en binario: 2^6). A continuación, cogiendo cada vez 6 [bits](#) del búfer, tenemos 4 números ($24 = 6 \times 4$), que entonces son convertidos a su correspondiente valor en Base 64.

Texto de entrada	M								a								n							
ASCII	77								97								110							
Bits	0	1	0	0	1	1	0	1	0	1	1	0	0	0	0	1	0	1	1	0	1	1	1	0
Índice	19								22								5							
Resultado en Base64	T								W								F							

Como Python opera con Base64:

Base64 es muy importante para la representación de datos binarios, tanto que permite que los datos binarios puedan ser representados de manera que parezcan y actúen como texto sin formato. Esto hace que sea más seguro almacenarlos en bases de datos, enviarlos a través de correo electrónico o utilizarlos en formatos basados en texto como XML. Base64 se utiliza básicamente para representar datos en formato ASCII.

Como hemos dicho en la introducción de este artículo, sin Base64 los datos no podrían ser leídos en absoluto.

Lo primero que tenemos que hacer con el fin de utilizar Base64 en Python es importar el módulo de Base64:

```
import base64
```

Con el fin de codificar la imagen, simplemente vamos a utilizar la función `base64.encodestring(s)`. Podemos hacer lo siguiente con el fin de codificar en Base64 nuestra imagen:

```
import base64
image = open('deer.gif', 'rb') #open binary file in read mode
image_read = image.read()
image_64_encode = base64.encodestring(image_read)
```

Si deseas ver la salida del proceso de codificación, escribe lo siguiente:

Realizado por Rodrigo Novas con fines de práctica.

```
print image_64_encode
```

Decodificando una imagen

Para decodificar una imagen utilizando Python, simplemente usamos la función `base64.decodestring(s)`. Por lo tanto, con el fin de decodificar la imagen que codifican en la sección anterior, hacemos lo siguiente:

```
base64.decodestring(image_64_encode)
```