

FOOFISH ([HTTPS://FOOFISH.NET/](https://foofish.net/))

首页 ([HTTPS://FOOFISH.NET/](https://foofish.net/))

分类 ([HTTPS://FOOFISH.NET/CATEGORIES.HTML](https://foofish.net/categories.html))

标签 ([HTTPS://FOOFISH.NET/TAGS.HTML](https://foofish.net/tags.html))

[FOOFISH.NET/TAGS.HTML](https://foofish.net/tags.html)

DJANGO教程 ([HTTPS://FOOFISH.NET/CATEGORY/DJANGOJIAO-CHENG.HTML](https://foofish.net/category/djangojiao-cheng.html))

小工具 ([HTTPS://FOOFISH.NET/TOOLS/INDEX.HTML](https://foofish.net/tools/index.html))

Base64编码详解

By 刘志军 (<https://foofish.net/author/liu-zhi-jun.html>), 2020-12-21, 分类: [python \(https://foofish.net/category/python.html\)](https://foofish.net/category/python.html)

🔖 [base64 \(https://foofish.net/tag/base64.html\)](https://foofish.net/tag/base64.html)

base64 是什么?

Base64, 顾名思义, 就是包括小写字母a-z、大写字母A-Z、数字0-9、符号"+"、"/"一共64个字符的字符集, (另加一个 "=", 实际是65个字符, 至于为什么还会有一个 "=", 这个后面再说)。任何符号都可以转换成这个字符集中的字符, 这个转换过程就叫做base64编码。

base64怎么转换

首先将字符串（图片等）转换成二进制序列，然后按每6个二进制位为一组，分成若干组，如果不足6位，则低位补0。每6位组成一个新的字节，高位补00，构成一个新的二进制序列，最后根据base64索引表中的值找到对应的字符。

标准 Base64 索引表										
数值	字符		数值	字符		数值	字符		数值	字符
0	A		16	Q		32	g		48	w
1	B		17	R		33	h		49	x
2	C		18	S		34	i		50	y
3	D		19	T		35	j		51	z
4	E		20	U		36	k		52	0
5	F		21	V		37	l		53	1
6	G		22	W		38	m		54	2
7	H		23	X		39	n		55	3
8	I		24	Y		40	o		56	4
9	J		25	Z		41	p		57	5
10	K		26	a		42	q		58	6
11	L		27	b		43	r		59	7
12	M		28	c		44	s		60	8
13	N		29	d		45	t		61	9
14	O		30	e		46	u		62	+
15	P		31	f		47	v		63	/

我们举个例子， 假设有字符串“abc”， 我们要对其进行base64编码，最后结果会是什么呢？

原始字符	a	b	c	
ascii编码	97	98	99	
二进制位	01100001	01100010	01100011	
编码转换	00011000	00010110	00001001	00100011
base64索引值	24	22	9	35
base64字符	Y	W	J	j

字符串abc对应3个字节，一共24位，按6位为一组可分为4组，在每组的高位补上00，经过转换，abc的base64编码是YWJj，由原来的3个字母变成了4个，所以base64会比原字符串更长。

那问题来了，假设原始字符串不够3个字节，只有一个字节或者两个字节怎么办？

以两个字节为例，按照上面的转换逻辑，经过编码转换，第三个字节只有4位，需要在第三组前后都要加两个0，转换后的字符串是YWI。为了凑齐4个字节，还要在末尾补上一个“=”号，最后得到的base64编码就是：“YWI=”

原始字符	a	b		
ascii编码	97	98		
二进制位	01100001	01100010		
编码转换	00011000	00010110	00001000	
base64索引值	24	22	8	
base64字符	Y	W	I	=

如果原始字符只有一个字节，原理是类似的，第二个字节除了前面补两个0，还要在后面补4个0，得到的字符串是YQ，剩下两个字节也用等号“=”来凑。所以a的base64编码就是YQ==

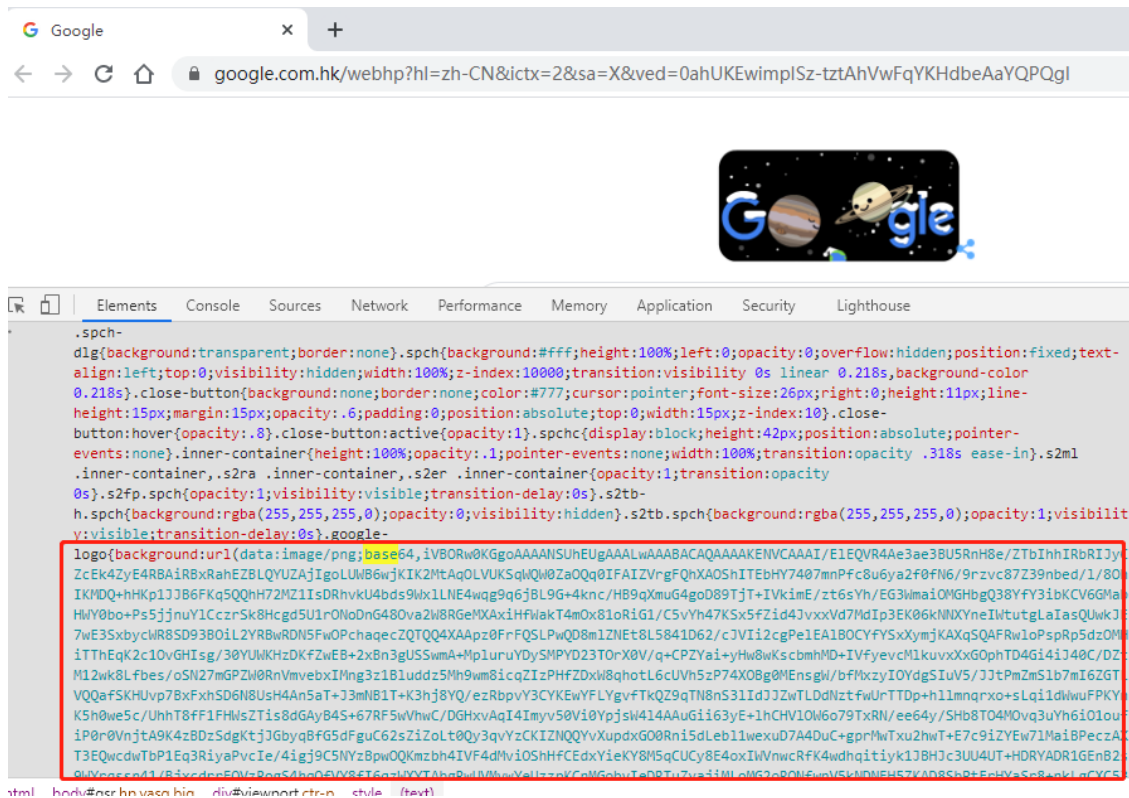
原始字符	a			
ascii编码	97			
二进制位	01100001			
编码转换	00011000	00010000		
base64索引值	24	16		
base64字符	Y	Q	=	=

总结一点就是只要原始字符串的长度不能被3整除，后面的位都会用0来补充。

base64用在哪些场景

1、html中的图片用base64表示

打开google的首页，就能看到某些样式中的图片不是一个资源地址，而是base64编码的字符串，这么做有什么好处呢？当然是减少了一次http的请求，但也并不是什么图片都适合用base64来处理，因为图片越大，转换的base64的字符串就越长，对带宽的要求更高了。



2、邮件传输

早期电子邮件的收发只允许使用ascii字符，这就导致无法发送非ascii字符，图片等二进制文件。所以在MIME对电子邮件进行了扩展，该扩展协议就指定了内容的传输的编码格式，可以是base64，base64编码使得在电子邮件中传输图片成为可能。

当然也可以将base64的内容放在url中传输。

在主流编程语言中，都内置的base64模块，可以直接调用，无需自己重复造轮子

python 示例

```
# 编码
>>> base64.b64encode(b'abc')
b'YWJj'

# 解码
>>> base64.b64decode(b'YWJj')
b'abc'
```

除了基本的`base64`，还有一种`url safe`形式的编码方式，目的是将`+/`替换成`-_`，因为标准的`Base64`并不适合直接放在`URL`里传输，`URL`编码器会把标准`Base64`中的`/`和`+`字符变为形如`%XX`的形式，而这些`%`号在存入数据库时还需要再进行转换。

```
>>> base64.b64encode(b'i\xcf\xbf')
b'ac+/'

# 使用"url safe"的base64编码会把+ / 替换成 - _
>>> base64.urlsafe_b64encode(b'i\xcf\xbf')
b'ac-_'
```

base64 是加密算法吗?

`base64`不是加密算法，他只是一种编码方式，数据从一种形式转换为另一种形式进行传输/存储。

关注公众号「Python之禅」，回复「1024」免费获取Python资源



Please enable JavaScript to view the comments.

SITEMAP

分类 (<https://foofish.net/categories.html>)
标签 (<https://foofish.net/tags.html>)
关于 (<https://foofish.net/pages/about.html>)
(<https://foofish.net/feeds/all.atom.xml>)
RSS (<https://foofish.net/feeds/rss.xml>)

SOCIAL

GitHub (<https://github.com/lzjun567>)
微博 (<http://weibo.com/lzjun567>)
知乎 (<https://www.zhihu.com/people/zhijun-liu>)

LINKS

次幂数据 (<https://www.cimidata.com/>)
腾讯云 (<https://curl.qcloud.com/3AERV1DS>)
Vimiix (<https://www.vimiix.com/>)
红色石头 (<http://redstonewill.com/>)
五分钟学算法 (<https://www.cyxiaowu.com/>)
Python知识圈 (<https://www.pyzhishiquan.com/>)
IT码农 (<https://tanqingbo.cn>)
无涯教程网 (<https://www.learnfk.com>)

© foofish 2016~2023

粤ICP备16102228号-1 (<http://www.beian.miit.gov.cn>)