B64 Base64 Encoding Reference

toolpilot.dev/cheatsheets/base64-cheat-sheet/

Base64 alphabet, padding rules, URL-safe variant, encoding/decoding examples in multiple languages.

What Is Base64?

- Encodes binary data as ASCII text using 64 printable characters
- 3 bytes of binary → 4 Base64 characters (33% size increase)
- Used in: email attachments (MIME), data URLs, JWT tokens, HTTP Basic Auth
- Not encryption — easily decoded, no security value
- Standard alphabet: A-Z, a-z, 0-9, +, / with = padding

Alphabet (64 characters + padding)

Value	Char	Value	Char	Value	Char	Value	Char
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
10	K	26	a	42	q	58	6
63	/	pad	=				

Padding Rules

Input bytes	Output chars	Padding
1 byte	2 chars + ==	e.g. "A" → "QQ=="
2 bytes	3 chars + =	e.g. "AB" → "QUI="
3 bytes	4 chars	e.g. "ABC" → "QUJD"

Standard vs URL-Safe

Variant	Extra chars	Padding	Use case
Standard (RFC 4648)	+ /	= required	MIME, general
URL-Safe (RFC 4648 §5)	- _	= omitted	URLs, JWT, filenames

Code Examples

```
# Python
import base64
encoded = base64.b64encode(b'hello').decode()
decoded = base64.b64decode(encoded)

# URL-safe
base64.urlsafe_b64encode(b'hello')

# JavaScript
btoa('hello')          // encode
atob('aGVsbG8=')       // decode

// URL-safe (manual)
btoa(str).replace(/\+/g,'-').replace(/\//g,'_')
```