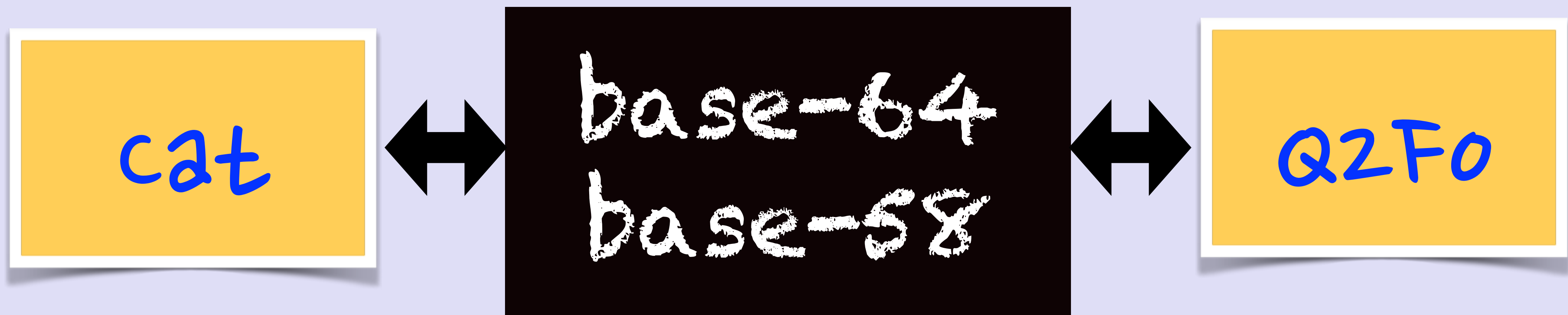


BLOCKCHAIN TUTORIAL 13

Base-64 and base-58 encoding



BLOCKCHAIN TUTORIAL 13

Base-64 and base-58 encoding

BASE-64

- Base-64 is basically a way of encoding arbitrary binary data in ASCII text. Base-64 encoding schemes are commonly used when there is a need to encode binary data (for example images, audio) that needs to be stored and transferred over media that are designed to deal with textual data. This is to ensure that the data remains intact without modification during transport.
- Base-64 encoding schemes uses A–Z, a–z, and 0–9 for the first 62 values, and the symbols + (plus) and / (slash). The = (equal) symbol is used as a padding character.
- Base-64 maps 3 bytes ($8 \times 3 = 24$ bits) in 4 characters that span 6-bits ($6 \times 4 = 24$ bits). When the number of bytes to encode is not divisible by three that is, if there are only one or two bytes of input for the last 24-bit block, then extra bytes with value zero is added so there are always three bytes.

BASE-64

- For example a base-64 encoded image 

```
R0lGODlhZgAZALMAAAAABgYGCEhISkpKSkxQjk5QkZGTnF7i2uEvXOEvXeMvX+UxqSx0MfS5dP
e7Nbn7ywAAAAAZgAZAAAE/hDJSau900vNu/9gKI5kaZ5oOj10xmpK82qza1+11Vvyrre3IMX38g
2BRklSiCK2lojZEsos0aNAjDT7sXgI4LB4TC6HDacpt8vxEQDwuHxOr8cJ6fWRpsfp33aBgndKL
QsyDAiHDomLLA0LWAuLiUWGPQ6QWCycipiaL5QIgaABDJwMAgACpyypACwABacGAAanB3hYMry8
L5iPm5wNWMXAmcKdx8SPnC2kragA0a6wjNcMp6cNuqgJ3pYrT5kMC+HhhenFNZaZ5s9xsXCx8tb
WDgf3+SzdDpGS6diNE7duoLotAAv+e0GqHj0H8SCyGDDPAUVr/QgGzIKuYEFntEhCKnqkQG0xhh
DnOJSY0l7FjOo8biR4DiSxYjhjsFjIEV5FOitdCuUXUyB0mR477jmKoFWlnqMiAm35UOrLo1CNa
h1oBOEMBR0ZWrUXNOjVo1tpAnmXFCr0sAZPjq360yxGrASXJWPk8ZhIYzbRPotFjde0kAyGUnUA
U7CnYZEQJ/DoiNzBS5AdE4jFCpWqzq5U2SVapfQEUnSz9FluvQBA7Bjy55Nu3bsA61z697Nm0Q
EAAA7
```

Source: <http://www.mobilefish.com/services/base64/base64.php>

BASE-64

- For example base-64 encode the word "Cat"

	C	a	t
ASCII code decimal	67	97	116
ASCII code binary *	01000011	01100001	01110100
Join 3 bytes into a 24-bit string	010000110110000101110100		
Create group of 6 bits	010000	110110	000101 110100
Convert binary to decimal	16, 54, 5, 52		
Base-64 encoded	Q2F0		

* Cat is represented as a byte sequence of a 8-bit padded ASCII characters

BASE-64

- On the right is the base-64 symbol chart.
- Lookup the decimal value in the chart and find its corresponding character in the map.
- Decimal values: 16, 54, 5, 52 corresponds to: Q2F0
- Thus, the word “Cat” base-64 encoded: Q2F0

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
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	/

BASE-64

- For example base-64 encode the word “Hi”

	H	i	
ASCII code decimal	72	105	
ASCII code binary *	01001000	01101001	00000000
Join 3 bytes into a 24-bit string	010010000110100100000000		
Create group of 6 bits	010010	000110	100100 000000
Convert binary to decimal		18, 6, 36, 0	
Base-64 encoded	SGk=		

* Add extra byte with value zero so there are always three bytes.

BASE-58

- Base-58 encoding schemes also converts binary to alphanumeric text. It is similar to base-64 encoding without using 0 (zero), O (capital o), I (capital i), l (lowercase L), + (plus) and / (slash) because they look the same in some fonts.
- The base-58 symbol chart on the right is used in Bitcoin and is specific to the Bitcoin project.

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