

	Reference Table	Shifted Table (e.g. if first offset character is <b>B</b> )	Shifted Table (e.g. if first offset character is <b>F</b> )	
0	A	/	+	<p><b>Requirement:</b></p> <p>You are to write an encoder that takes in a plaintext and encode it to another obfuscated string. The logic of the encoding / decoding is given below:</p> <p><b>Logic:</b></p> <p>The message is encoded. The first character of the encoded message will indicate the starting mapping index. Any character not in the reference table will mapped back to the same character.</p> <p>e.g. the message</p> <p>HELLO WORLD</p> <p>Is encoded to:</p> <p><b>BGDKKN VNQKC</b></p> <p>If the first character <b>B</b> is taken as the offset character (Shift table forward by <b>1</b> element).</p> <p><b>FC/GGJ RJMG.</b></p> <p>If the first character <b>F</b> is taken as the offset character (Shift table forward by <b>5</b> elements).</p> <p>To decode it, you need to take the first character for offset and match it backwards to get the original clear text.</p> <p><b>Constraint</b></p> <p>The solution must implement the following 2 methods:</p> <pre>public String encode (String plainText); public String decode (String encodedText);</pre> <p><b>Bonus</b></p> <p>The solution should also demonstrate the concept of <b>OOP</b>.</p>
1	B	A	,	
2	C	B	-	
3	D	C	.	
4	E	D	/	
5	F	E	A	
6	G	F	B	
7	H	G	C	
8	I	H	D	
9	J	I	E	
10	K	J	F	
11	L	K	G	
12	M	L	H	
13	N	M	I	
14	O	N	J	
15	P	O	K	
16	Q	P	L	
17	R	Q	M	
18	S	R	N	
19	T	S	O	
20	U	T	P	
21	V	U	Q	
22	W	V	R	
23	X	W	S	
24	Y	X	T	
25	Z	Y	U	
26	0	Z	V	
27	1	0	W	
28	2	1	X	
29	3	2	Y	
30	4	3	Z	
31	5	4	0	
32	6	5	1	
33	7	6	2	
34	8	7	3	
35	9	8	4	
36	(	9	5	
37	)	(	6	
38	*	)	7	
39	+	*	8	
40	,	+	9	
41	-	,	(	
42	.	-	)	
43	/	.	*	