

`s[-1]` == `'r'`

`s[-2]` == `'u'`

`s[-4]` == `'t'`

`s[-6]` == `'A'`

S

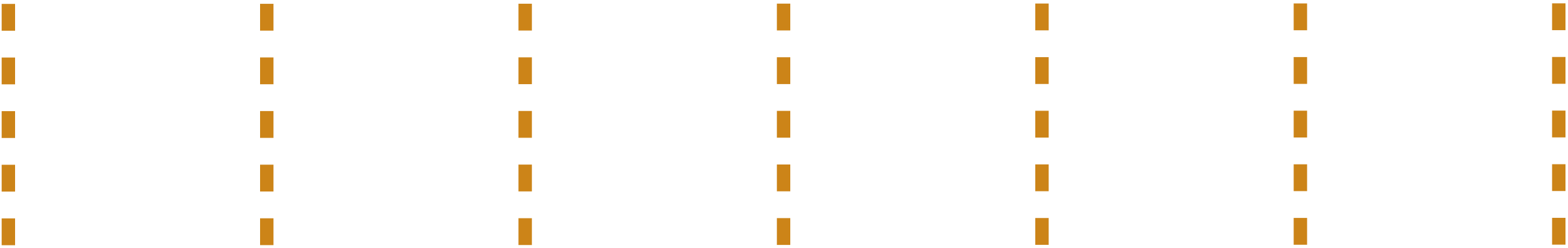
=

=



0 1 2 3 4 5 6





Negative Indexing

`s = 'Arthur'`

0	1	2	3	4	5	6
Arthur						
-6	-5	-4	-3	-2	-1	0

<code>s[-1]</code>	<code>==</code>	<code>'r'</code>
<code>s[-2]</code>	<code>==</code>	<code>'u'</code>
<code>s[-4]</code>	<code>==</code>	<code>'t'</code>
<code>s[-6]</code>	<code>==</code>	<code>'A'</code>

Slicing

`s = 'Arthur'`

The diagram shows the string 'Arthur' with its characters indexed from 0 to 6. Vertical dashed orange lines mark the boundaries between characters: index 0 is at the start of 'A', index 1 is between 'A' and 'r', index 2 is between 'r' and 't', index 3 is between 't' and 'h', index 4 is between 'h' and 'u', index 5 is between 'u' and 'r', and index 6 is at the end of the string. The string is enclosed in single quotes, and the assignment 's =' is shown to the left.

Index	Character
0	A
1	r
2	t
3	h
4	u
5	r
6	