

# Ravel



# Append

<p><code>,y</code> gives a list of the atoms of <code>y</code> in “normal” order: the result is ordered by items, by items within items, etc. The result shape is <code>1\$*/\$ y</code>. Thus:</p> <pre> y=: 2 4 \$ 'abcdefgh'  y abcd efgh  ,y abcdefgh         </pre>	<p><code>x,y</code> appends items of <code>y</code> to items of <code>x</code> after:</p> <ol style="list-style-type: none"> <li>1) Reshaping an atomic argument to the shape of the items of the other,</li> <li>2) Bringing the arguments to a common rank (of at least 1) by repeatedly itemizing (<code>, :</code>) any of lower rank, and</li> <li>3) Bringing them to a common shape by padding with fill elements in the manner described in Section II B.</li> </ol> <p>The fit conjunction (<code>, ! . f</code>) provides fill specified by the items of <code>f</code>.</p>
--	--

```

]a=: i. 2 3 3
0 1 2
3 4 5
6 7 8

9 10 11
12 13 14
15 16 17

,a
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17

,"2 a
0 1 2 3 4 5 6 7 8
9 10 11 12 13 14 15 16 17
        
```

The following examples illustrate the dyadic case:

```

('abc', 'de'); ('abc', "0/'de'); (5 6 7, i.2 3); (7, i.2 3)
+-----+
| abcde | ad | 5 6 7 | 7 7 7 |
|        | ae | 0 1 2 | 0 1 2 |
|        |    | 3 4 5 | 3 4 5 |
|        | bd |       |       |
|        | be |       |       |
|        | cd |       |       |
|        | ce |       |       |
+-----+
        
```