

Brian — PS 2 — 2025-01-21 — Solution

EIWL3 Sections 5-8

Exercises from *EIWL3* Section 5

```
(* 5.1 *) Reverse[Range[10]^2]  
(* I could square and reverse or reverse and square. *)
```

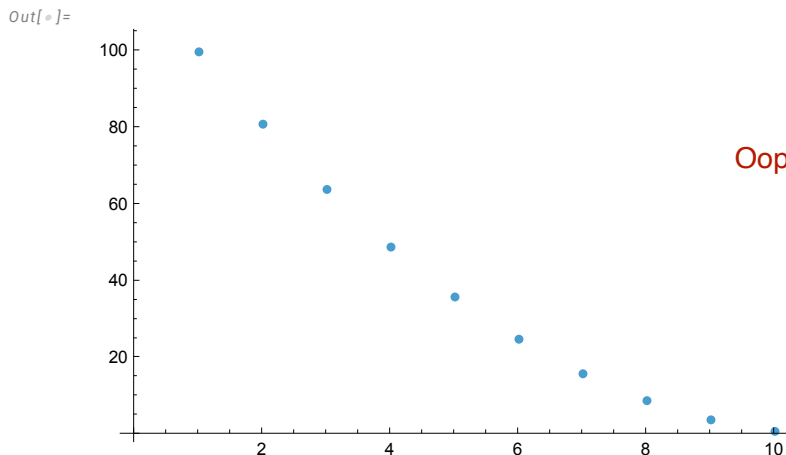
```
Out[8]=  
{100, 81, 64, 49, 36, 25, 16, 9, 4, 1}
```

```
(* 5.2 *) Total[Reverse[Range[3]]^2]
```

```
Out[11]=  
14
```

Oops. I did the first 3 instead of the first 10.

```
(* 5.3 *) ListPlot[Reverse[Range[10]]^2]
```



Oops. I accidentally did descending.

```
(* 5.4 *) Sort[Join[Range[4], Range[4]]]
```

```
Out[10]=  
{1, 1, 2, 2, 3, 3, 4, 4}
```

```
(* 5.5 *) Range[10, 20, 1] (* Range[10, 20, 1] is simpler  
and clearer but it isn't what Wolfram requested us to do. *)
```

```
Out[12]=  
{10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20}
```

```
Range[11] + 9 (* This way uses Plus as Wolfram requested *)
```

```
(* 5.6 *) Sort[Join[Range[5]^2, Range[5]^3]]
```

```
Out[13]=  
{1, 1, 4, 8, 9, 16, 25, 27, 64, 125}
```

```
(* 5.7 *) Length[IntegerDigits[2^128]]
```

```
Out[ ]=
```

```
39
```

```
(* 5.8 *) First[IntegerDigits[2^32]]
```

```
Out[ ]=
```

```
4
```

```
(* 5.9 *) Take[IntegerDigits[2^100], 10]
```

```
Out[ ]=
```

```
{1, 2, 6, 7, 6, 5, 0, 6, 0, 0}
```

```
(* 5.10 *) Max[IntegerDigits[2^20]]
```

```
Out[ ]=
```

```
8
```

```
(* 5.11 *) Count[IntegerDigits[2^1000], 0]
```

```
Out[ ]=
```

```
28
```

```
(* 5.12 *) Sort[IntegerDigits[2^20]][[0]]
```

```
(* I am using a special notation for Part *)
```

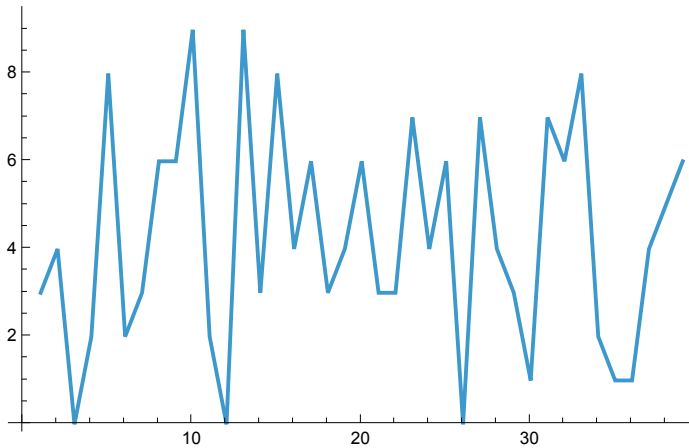
```
Out[ ]=
```

```
List
```

I boo-booed — I should have finished 5.12 with `[[2]]` not `[[0]]`

```
(* 5.13 *) ListLinePlot[IntegerDigits[2^128]]
```

```
Out[ ]=
```



```
(* 5.14 *) Drop[Take[Range[100], 20], 10]
```

```
Out[ ]=
```

```
{11, 12, 13, 14, 15, 16, 17, 18, 19, 20}
```

Exercises from *EIWL3* Section 6

(* 6.1 *) `Table[1000, 5]`

Out[*]=

{1000, 1000, 1000, 1000, 1000}

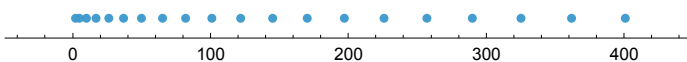
(* 6.2 *) `Table[n3, {n, 10, 20}]`

Out[*]=

{1000, 1331, 1728, 2197, 2744, 3375, 4096, 4913, 5832, 6859, 8000}

In[15]:= (* 6.3 *) `NumberLinePlot[Table[n2, {n, 20}]]`

Out[15]=



(* 6.4 *) `Table[i, {i, 2, 20, 2}]` (* I assume he wants to use Table with steps, but there are lots of other ways of doing this. E.g., *)

In[14]:= `Table[i, {i, 10}] 2`

Out[14]=

{2, 4, 6, 8, 10, 12, 14, 16, 18, 20}

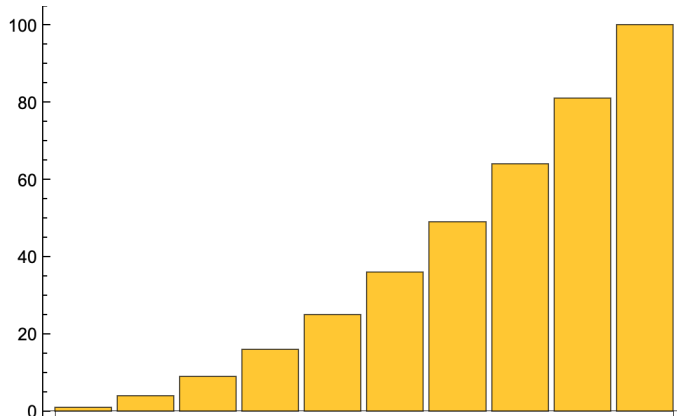
In[13]:= (* 6.5 *) `Table[i, {i, 10}]`

Out[13]=

{1, 2, 3, 4, 5, 6, 7, 8, 9, 10}

In[12]:= (* 6.6 *) `BarChart[Table[i2, {i, 10}]]`

Out[12]=



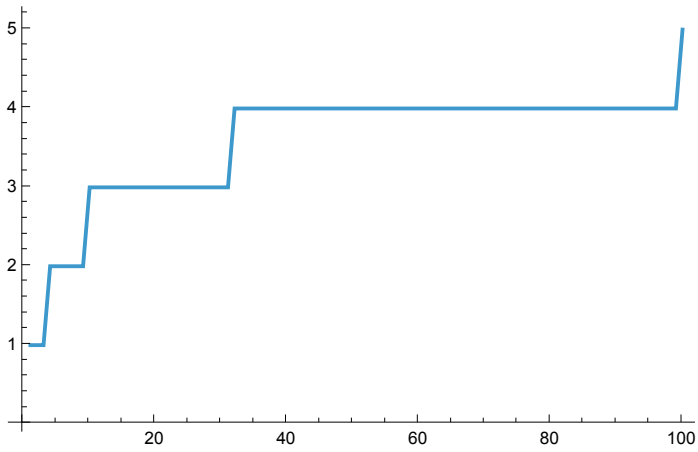
(* 6.7 *) `Table[IntegerDigits[i2], {i, 10}]`

Out[16]=

{{1}, {4}, {9}, {1, 6}, {2, 5}, {3, 6}, {4, 9}, {6, 4}, {8, 1}, {1, 0, 0}}

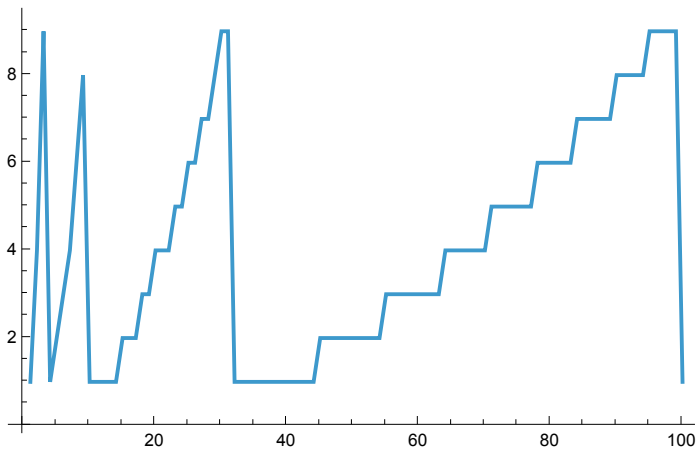
```
In[18]:= (* 6.8 *) ListLinePlot[Table[Length[IntegerDigits[i2]], {i, 100}]]
```

Out[18]=



```
In[17]:= (* 6.8 *) ListLinePlot[Table[First[IntegerDigits[i2]], {i, 100}]]
```

Out[17]=



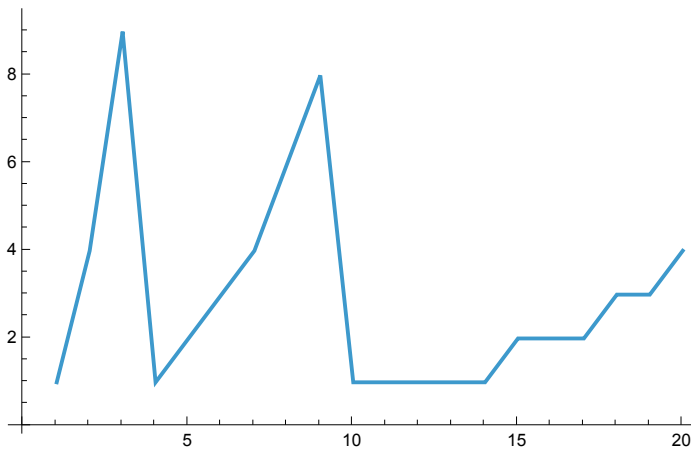
```
In[19]:= (* 6.9 *) Table[First[IntegerDigits[i2]], {i, 20}]
```

Out[19]=

{1, 4, 9, 1, 2, 3, 4, 6, 8, 1, 1, 1, 1, 1, 2, 2, 2, 3, 3, 4}

```
(* 6.10 *) ListLinePlot[Table[First[IntegerDigits[i^2]], {i, 20}]]
```

Out[21]=



Were we supposed to do the first 100?

Exercises from EIWL3 Section 7

```
(* 7.1 *) {Red, Yellow, Green}
```

Out[*]=



```
(* 7.2 *) Column[{Red, Yellow, Green}]
```

Out[*]=



```
(* 7.3 *) ColorNegate[Orange]
```

Out[*]=



```
(* 7.4 *) Table[Hue[i], {i, 0, 1, 0.02}]
```

Out[48]=



```
In[51]:= (* 7.5 *) Table[RGBColor[1.0, green, 1.0], {green, 0, 1, 0.05}]
```

Out[51]=



```
(* 7.6 *) Blend[{Pink, Yellow}]
```

Out[55]=



```
(* 7.7 *) Table[Blend[{Hue[i], Yellow}], {i, 0, 1, 0.05}]
```

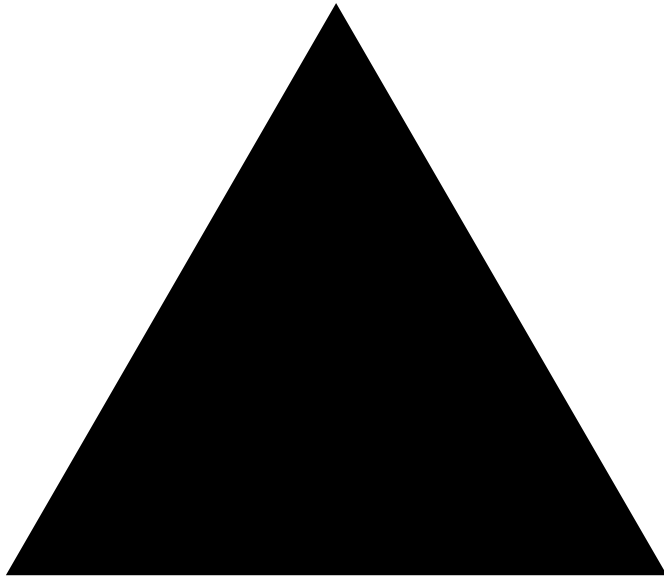
Out[54]=



Exercises from *EIWL3* Section 8

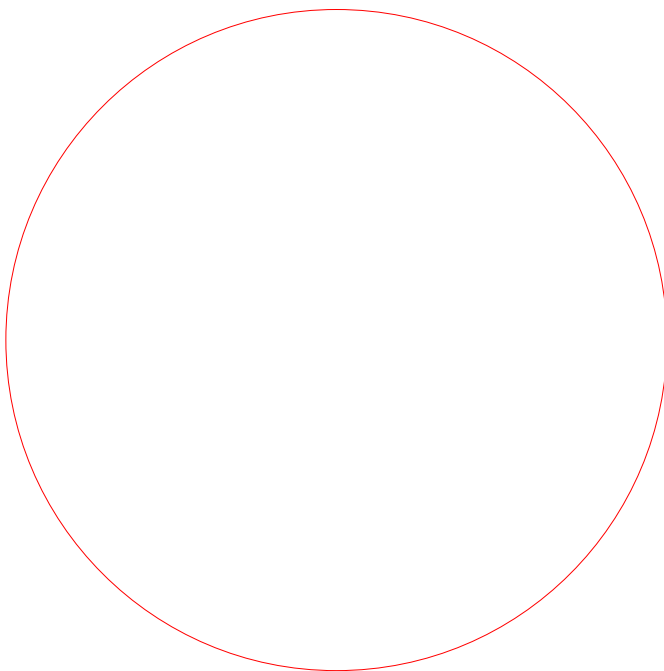
(* 8.1 *) Graphics[RegularPolygon[3]]

Out[]=



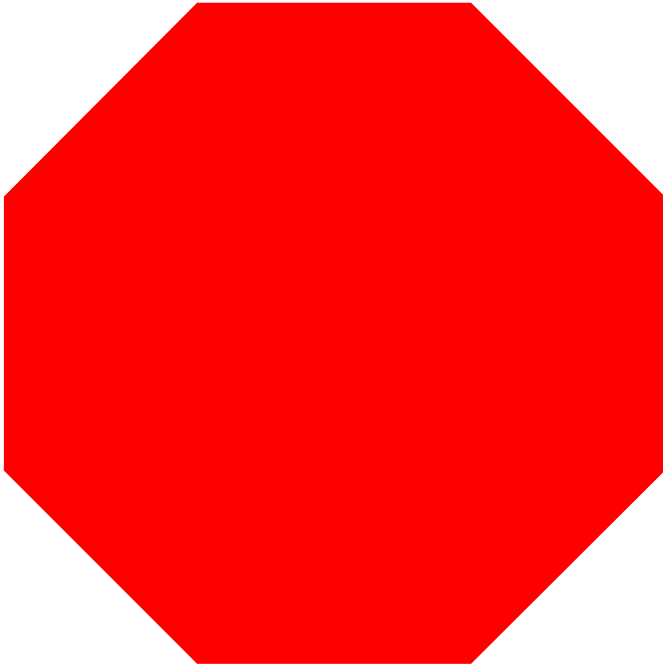
(* 8.2 *) Graphics[Style[Circle[], Red]]

Out[]=



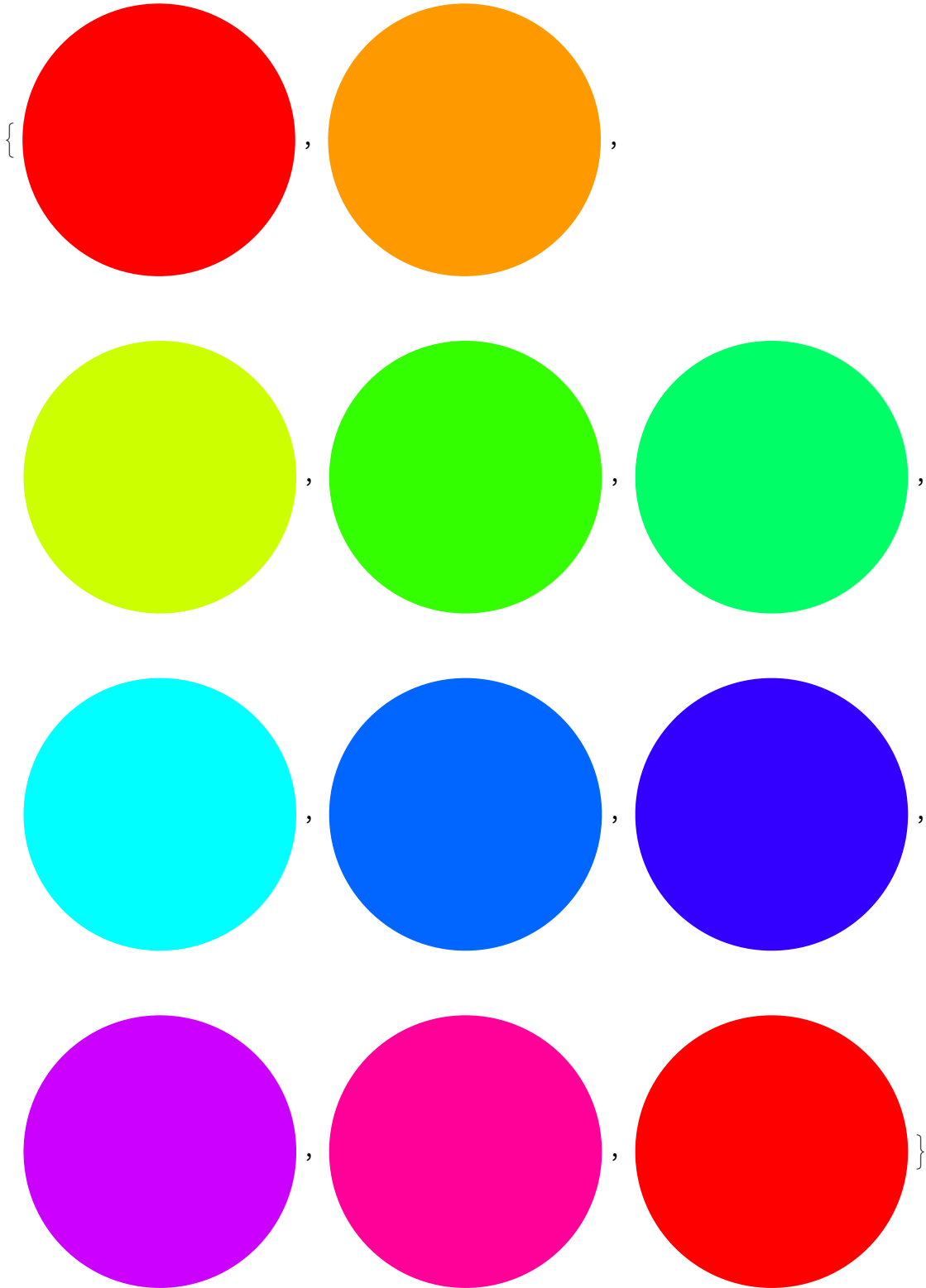
```
(* 8.3 *) Graphics[Style[RegularPolygon[8], Red]]
```

Out[8]=



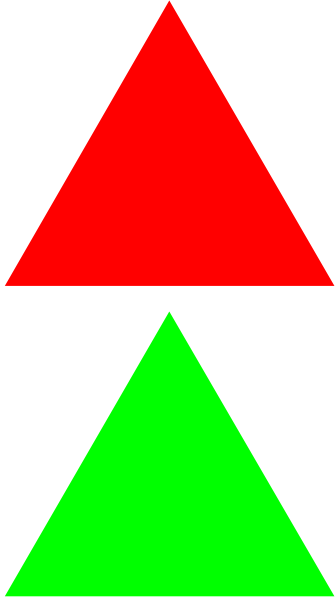

```
(* 8.4 *) Table[Graphics[Style[Disk[], Hue[i]]], {i, 0.0, 1.0, 0.1}]
```

```
Out[8]=
```



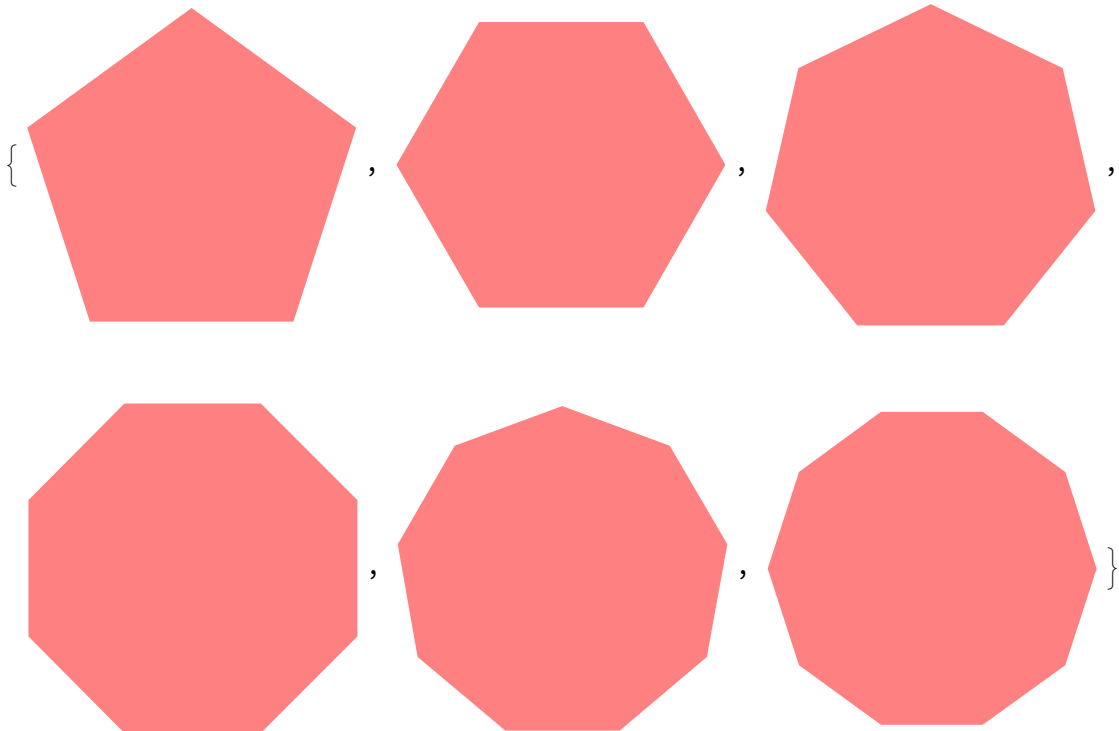
```
(* 8.5 *) Column[{
  Graphics[Style[RegularPolygon[3], Red]],
  Graphics[Style[RegularPolygon[3], Green]]
}] (* The nested brackets and braces got deep
enough that I used indenting to help me get it right. *)
```

Out[8]=



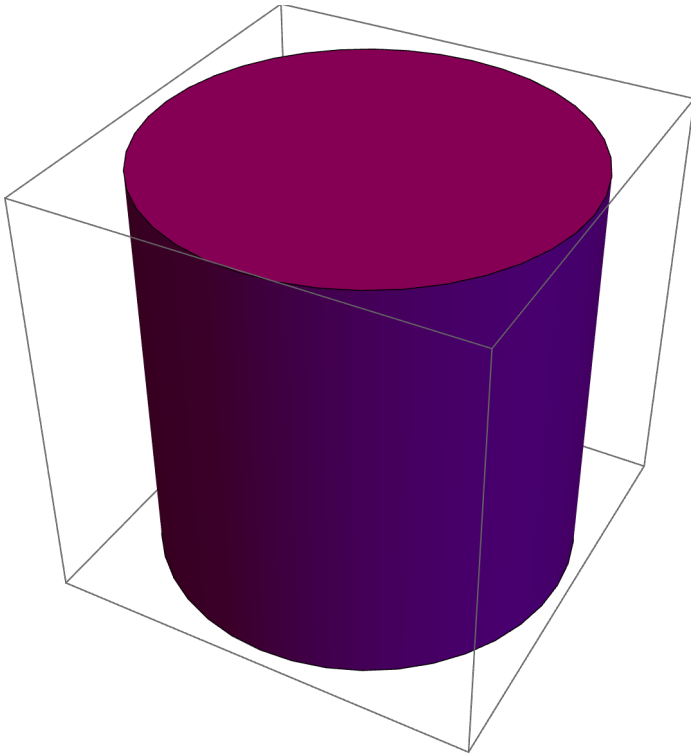
```
(* 8.6 *) Table[Graphics[Style[RegularPolygon[i], Pink]], {i, 5, 10}]
```

Out[9]=



```
(* 8.7 *) Graphics3D[Style[Cylinder[], Purple]]
```

```
Out[8]=
```



```
(* 8.8 *) Graphics[Table[  
  Style[RegularPolygon[i], RandomColor[]],  
  {i, 8, 3, -1}  
]]
```

Out[8]=

