Understanding Table and Creating Animations in *Mathematica*

Getting Started

Lists are one of the most important structures in *Mathematica*. Lists are defined with { }.

```
{a, b, c}
```

Table is a useful way to generate a list using a rule, pattern, or function. Table has two (or more) arguments: the first is the element to be used in the pattern and the second argument is a list that tells how to use the first argument to build the list.

Note the differences among the following examples. Look at each carefully and see if you can understand how each output is generated.

```
Table[n, {5}]

Table[n, {n, 5}]

Table[n, {n, 5, 10}]

Table[n, {n, 1, 5, 0.5}]

Table[n, {3}, {5}]

Table[n, {n, 1, 5}, {4}]

Table[n, {4}, {n, 1, 5}]

Table[{n, m}, {n, 5, 10}, {m, -3, 0}]
```

Tables don't have to be just lists of numbers. The more interesting and useful lists involve functions.

```
Table[n^2, {n, 1, 10}]

Table[Cos[n], {n, 0, 2 Pi}]

Table[Cos[n], {n, 0, 2 Pi, Pi / 6}]
```

You can even have a Table be the function within a Table.

```
Table[Table[n, {n, 1, m}], {m, 2, 10}]
```

Graphing Example

Recall how to create a simple Plot.

```
Plot[Sin[x], {x, 0, 2 Pi}, PlotStyle → Hue[1]];

Plot[{Sin[x], x^2}, {x, 0, Pi}, PlotStyle → {Hue[1], Hue[0.65]}];
```

Now use Table to create a list of functions to graph and put that list into a Plot. (Note: Evaluate generates the Table before applying Plot.)

```
Table[Sin[n x], {n, 1, 5}]

Plot[Evaluate[Table[Sin[n x], {n, 1, 5}]], {x, 0, 2 Pi}]

Plot[Evaluate[Table[Sin[n x], {n, 1, 5}]], {x, 0, 2 Pi}, PlotStyle → Table[Hue[n], {n, 0, 1, 0.2}]]
```

Animation Examples

■ Example: Plot

An animation can be created by making a Table of graphs. Each graph becomes a frame in the animation. After evaluating the following code, note how the blue cell brackets on the right-hand side are grouped. Double-click the middle one to close the group of output cells. To run the animation, double-click the image. Note the animation controls in the lower-left corner of the notebook window.

```
Table[Plot[Sin[b x], {x, 0, 2 Pi}], {b, 1, 5}];
```

Add more frames by counting by 0.2 instead of the default value of 1.

```
Table[Plot[Sin[b x], {x, 0, 2 Pi}], {b, 1, 5, 0.2}];
```

Control two graphs with the same Table.

```
Table[
 Plot[{Sin[b x], Sin[(6 - b) x]}, {x, 0, 2 Pi}, PlotStyle → {Hue[1], Hue[0.65]}],
 {b, 1, 5, 0.2}]
```

■ Example: ListPlot

ListPlot is a function that graphs a list of points.

```
ListPlot[{{-2, 3}, {2, -3}, {1 / 2, 1}, {3, 3}}]
```

Add the options as shown to change the appearance.

```
ListPlot[{{-2, 3}, {2, -3}, {1 / 2, 1}, {3, 3}},
 PlotRange → {{-4, 4}, {-4, 4}},
 AspectRatio → Automatic, PlotStyle → {PointSize[0.04], Hue[1]}, GridLines → Automatic]
```

Follow the examples below to build an animation that plots a sequence of points.

```
Table[{n, n^2}, {n, -10, 10}]

ListPlot[Table[{n, n^2}, {n, -10, 10}]];

ListPlot[Table[{n, n^2}, {n, -10, 10}], PlotStyle → {PointSize[0.03], Hue[1]}];

Table[ListPlot[Table[{n, n^2}, {n, -10, m}]], {m, -10, 10}];
```

```
Table[ListPlot[Table[{n, n^2}, {n, -10, m}], PlotRange → {{-10, 10}, {0, 100}}], {m, -10, 10}];
```

```
Table[
  ListPlot[
   Table[{n, n^2}, {n, -10, m}],
   PlotRange → {{-10, 10}, {0, 100}}, PlotStyle → {Hue[1], PointSize[0.03]}],
  {m, -10, 10}];
```

■ Example: Plot (trace the graph)

```
Table[
  Plot[x^2, {x, -1.001, xmax}, PlotRange → {{-1, 1}, {0, 1}}],
  {xmax, -1, 1, 0.1}];
```

Interesting Example

Type in the example code below. Think about what each part means and enjoy the animation you create.

```
Table[Plot[Cos[x - c], {x, 0, 2 Pi}], {c, 0, 2 Pi}];
```

```
Table[
  Plot3D[x Cos[y - c], {x, 0, 1}, {y, 0, 2 Pi}],
  {c, 0, 2 Pi, Pi / 12}];
```

```
Table[
  Plot3D[x Cos[y - c], {x, 0, 1}, {y, 0, 2 Pi},
   Boxed → False, Axes → False, Mesh → False],
  {c, 0, 2 Pi - Pi / 12, Pi / 12}];
```

```
Table[Plot3D[(1 - x^2) Cos[y - c], {x, -1, 1}, {y, 0, 2 Pi}], {c, 0, 2 Pi - Pi / 12, Pi / 12}];
```

Export Animation

Name your Table and reevaluate the code.

```
mygraph = Table[Plot3D[(1 - x^2) Cos[y - c], {x, -1, 1}, {y, 0, 2 Pi}], {c, 0, 2 Pi - Pi / 12, Pi / 12}];
```

To Export as an animated GIF, you will need to choose a location to save the file. You can use the menu Input >> Get File Path . . . to browse and insert a location of your choice more easily.

```
Export["C:\\graphanimation.gif", mygraph]
```

Open the GIF in a web browser to see the animation. To add continuous looping, use the option as shown.

```
Export["C:\\graphanimation.gif", mygraph, ConversionOptions → {"Loop" → True}]
```

Design Your Own Animation: Have Fun!

Abby Brown ~ October 2006 ~ www.abbymath.com