■ Simple Pendulum - approximate and exact

(Local) In[24]:=
```
Clear["Global`*"];
```

■ Define the constants to some useful values (try changing these) and setup the initial conditions

(Local) In[25]:=
```
val = {q0 Æ Pi / 4, l Æ 1, g Æ 10, m Æ 1};
```

(Local) In[26]:=
```
init1 = {q1[0] ä q0, q1'[0] ä 0};
```

(Local) In[27]:=
```
init2 = {q2[0] ä q0, q2'[0] ä 0};
```

■ Define the approximate (small angle) equation of motion and solve for *θ*[t]

(Local) In[28]:=
```
eq1 = m l^2 q1''[t] ä - m g l q1[t];
```

(Local) In[29]:=
```
eq1 = Append[init1, eq1] /. val;
```

(Local) In[30]:=
```
dsol1 = DSolve[eq1, q1[t], t];
```

(Local) In[31]:=
```
plot1 = ParametricPlot[Evaluate[{t, q1[t]} /. dsol1],
   {t, 0, 8}, PlotStyle Æ RGBColor[0, 0, 1], GridLines Æ Automatic,
   Frame Æ True, FrameLabel Æ {"t", "q"}, RotateLabel Æ False];
```

■ Define the exact equation of motion and solve for *θ*[t]

(Local) In[32]:=
```
eq2 = m l^2 q2''[t] ä - m g l Sin[q2[t]];
```

(Local) In[33]:=
```
eq2 = Append[init2, eq2] /. val;
```

(Local) In[34]:=

```
dsol2 = NDSolve[eq2, q2[t], {t, 0, 8}];
```

(Local) In[35]:=

```
plot2 = ParametricPlot[Evaluate[{t, q2[t]} /. dsol2],
   {t, 0, 8}, PlotStyle Æ RGBColor[1, 0, 0], GridLines Æ Automatic,
   Frame Æ True, FrameLabel Æ {"t", "q"}, RotateLabel Æ False];
```

■ Now plot the two together (blue = SHM, red = exact solution)

(Local) In[36]:=

```
Show[plot1, plot2];
```