```
> restart;with(DEtools):with(plots):
> # Oscillator (for PHY322 - Salgado)

> m:=(1/2);k:=1; Force(t):=0;
  ode := diff(y(t),t,t) = -(k/m)*y(t) + Force(t)/m;
> #?DEplot
```

DEtools[DEplot] - plot solutions to a system of DEs

Calling Sequences

DEplot(**deqns**, **vars**, **trange**, **options**)
DEplot(**deqns**, **vars**, **trange**, **inits**, **options**)
DEplot(**deqns**, **vars**, **trange**, **xrange**, **yrange**, **options**)
DEplot(**dproc**, **vars**, **trange**, **number**, **xrange**, **yrange**, **options**)
DEplot(**deqns**, **vars**, **trange**, **inits**, **yrange**, **xrange**, **options**)

Parameters

`deqns` - list or set of first order ordinary differential equations, or a single differential equation of any order

`dproc` - function representation for first order ordinary differential equations, or a single differential equation of any order

`vars` - dependent variable, or list or set of dependent variables

`trange` - range of the independent variable

`number` - equation of the form **'number'=integer** indicating the number of differential equations when **deqns** is given as a function (dproc) instead of expressions

`inits` - set or list of lists; initial conditions for solution curves

`xrange` - range of the first dependent variable

`yrange` - range of the second dependent variable

`options` - (optional) equations of the form **keyword=value**

```
> ode1 := diff(y(t),t) = v(t);
  ode2 := diff(v(t),t) = subs(diff(y(t),t)=v(t),rhs(ode));
  odesystem:=[ode1,ode2];
> diff(ode1,t);ode_original:=subs( ode2, %);
> DEplot ( odesystem , [y(t),v(t)], t=0..12,
  v=-4..4,y=-3..3,arrows=SLIM);
> ICsystem:=[[y(0)=2,v(0)=0]];
  DEplot ( odesystem , [y(t),v(t)],  t=0..12, ICsystem,
  v=-4..4,y=-3..3,arrows=SLIM,linecolor=blue,thickness=5);
  #default stepsize may have poor resolution
> DEplot ( odesystem , [y(t),v(t)],  t=0..12, ICsystem,
  stepsize=0.01,
  v=-4..4,y=-3..3,arrows=SLIM,linecolor=blue,thickness=5);
```

```
> ### carefully copy from the previous command...
  ### note the use of [ ] and T
  animate(DEplot,
  [
  odesystem , [y(t),v(t)],  t=0..T, ICsystem, stepsize=0.01,
  v=-4..4,y=-3..3,arrows=SLIM,linecolor=blue,thickness=5
  ] ,
  T=0.01..12);
>
> p:=dsolve(ode);
  odeplot(p);
> ICsystem;
  IC:=op(op(1,subs(v=D(y),ICsystem)));
  ### my Maple method to convert from ICsystem

  p:=dsolve({ode, IC },numeric);
  odeplot(p,color=blue,thickness=5);
> p:=dsolve({ode, IC },numeric, range=0..10);
  odeplot(p,color=blue,thickness=5);
> ### ATTEMPT SYMBOLIC SOLUTION
> ode, IC;
  solution:=dsolve({ode, IC} );
> pos(t):=rhs(solution);
> plot(pos(t),t=0..10,y=-3..3,color=blue,thickness=5);
> vel:=diff(pos(t),t);
  plot(vel(t),t=0..10,color=green,thickness=5);
> KineticEnergy:=(1/2)*m*vel(t)^2;
  PotentialEnergy:=(1/2)*k*pos(t)^2;
> plot([ KineticEnergy , PotentialEnergy ,
  KineticEnergy+PotentialEnergy ],
  t=-0..10,color=[blue,green,red],thickness=[5,5,8]);
```