

```

(*by Nasser M. Abbasi 5/6/14*)
Manipulate[
  tick;

  Module[{r, alpha, len, z, eq, f, x, y, E0},

    If[state == "init" || state == "reset" || state == "paused",
      {a, b, c, p, ellipse} = makeEllipse[e];
      n = Sqrt[ $\frac{\mu}{a^3}$ ];
      x = a;
      y = 0
    ];

    If[state == "running" || state == "step",
      currentTime = Mod[currentTime + stepSize, 2 Pi Sqrt[ $\frac{a^3}{\mu}$ ]];

      sol = E0 /. FindRoot[currentTime == Sqrt[ $\frac{a^3}{\mu}$ ] (E0 - e Sin[E0]), {E0,  $\frac{\text{currentTime}}{1 - e}$ ]];

      E0 = sol;
      r = a * (1 - e * Cos[E0]);
      x = a Cos[E0];
      y = a Sqrt[1 - e^2] Sin[E0];
      If[state == "running",
        tick = Not[tick]
      ]
    ];

    Graphics[
      {
        First@ellipse,
        {LightGray, Line[{{c, 0}, {c, p}}]},
        {Green, Disk[{c, 0}, 0.05]},
        {Gray, Disk[{-c, 0}, 0.05]},
        {Gray, Dashed, Line[{{0, -b}, {0, b}}]},
        {Gray, Dashed, Line[{{-a, 0}, {a, 0}}]},
        {Blue, Disk[{x + c, y}, 0.02]},
        Arrow[{{c, 0}, {x, y}}]
      }, PlotRange -> {{-1.1 a, 1.1 a}, {-1.1 a, 1.1 a}}, ImagePadding -> 30, ImageSize -> 300
    ]
  ],
  Grid[{
    {
      Grid[{
        {
          Button[Text[Style["run", 12]],
            state = "running"; tick = Not[tick], ImageSize -> {80, 35}],
        }
      ]
    }
  ]
}

```

```

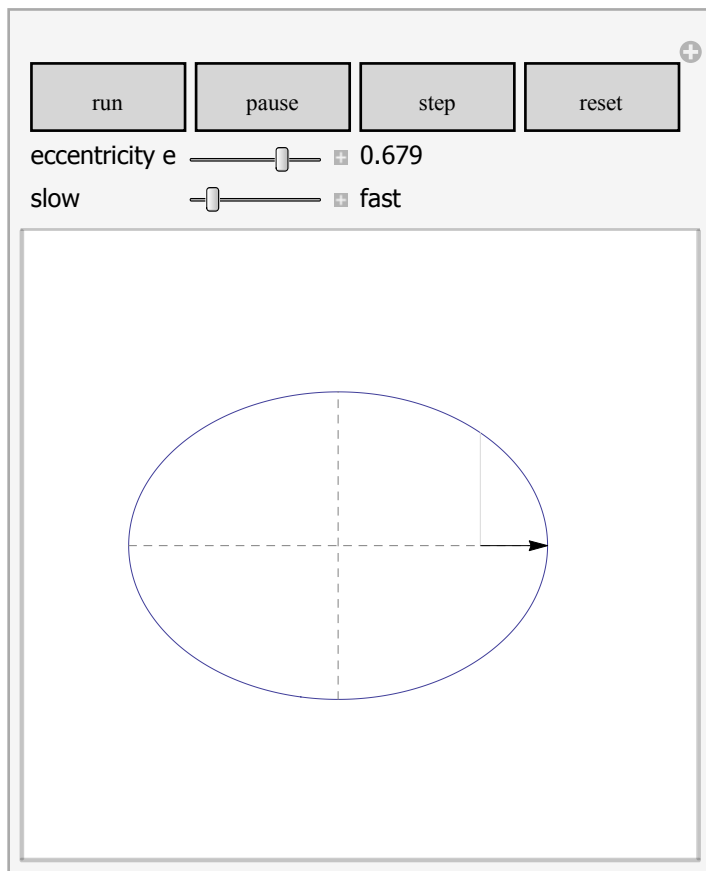
    Button[Text[Style["pause", 12]], state = "paused";
      tick = Not[tick], ImageSize → {80, 35}],
    Button[Text[Style["step", 12]], state = "step"; tick = Not[tick],
      ImageSize → {80, 35}],
    Button[Text[Style["reset", 12]], state = "reset"; currentTime = 0;
      tick = Not[tick], ImageSize → {80, 35}]
  }
}, Spacings → {0.4, .2}, Alignment → Center
], SpanFromLeft
},
{
  "eccentricity e",
  Manipulator[
    Dynamic[e, {e = #; {a, b, c, p, ellipse} = makeEllipse[e]; tick = Not[tick]} &],
    {0, 0.9, 0.001}, ImageSize → Tiny, ContinuousAction → True],
  Dynamic[padIt2[e, {4, 3}]]
},
{
  "slow",
  Manipulator[Dynamic[stepSize, {stepSize = #} &],
    {1, 60, 1}, ImageSize → Tiny, ContinuousAction → False],
  "fast"
}
}, Spacings → {0.4, .2}, Alignment → Left],
{{tick, False}, None},
{{state, "init"}, None},
{{e, .9}, None},
{{stepSize, 60}, None},
{{currentTime, 0}, None},
{{currentE, 0}, None},
{{ellipse, 0}, None},
{{a, 0}, None},
{{c, 0}, None},
{{b, 0}, None},
{{p, 0}, None},
{{n, 0}, None},
TrackedSymbols => {tick},
Initialization =>
(
   $\mu = 3.986 * 10^5$ ;
  (*definitions used for parameter checking*)
  integerStrictPositive = (IntegerQ[#] && # > 0 &);
  integerPositive = (IntegerQ[#] && # ≥ 0 &);
  numericStrictPositive = (Element[#, Reals] && # > 0 &);
  numericPositive = (Element[#, Reals] && # ≥ 0 &);
  numericStrictNegative = (Element[#, Reals] && # < 0 &);
  numericNegative = (Element[#, Reals] && # ≤ 0 &);
  bool = (Element[#, Booleans] &);
  numeric = (Element[#, Reals] &);
  integer = (Element[#, Integers] &);
  (*-----*)
  padIt1[v_?numeric, f_List] := AccountingForm[v,

```

```

    f, NumberSigns → {"-", "+"}, NumberPadding → {"0", "0"}, SignPadding → True];
(*-----*)
padIt1[v_?numeric, f_Integer] := AccountingForm[Chop[v],
    f, NumberSigns → {"-", "+"}, NumberPadding → {"0", "0"}, SignPadding → True];
(*-----*)
padIt2[v_?numeric, f_List] := AccountingForm[v,
    f, NumberSigns → {"", ""}, NumberPadding → {"0", "0"}, SignPadding → True];
(*-----*)
padIt2[v_?numeric, f_Integer] := AccountingForm[Chop[v],
    f, NumberSigns → {"", ""}, NumberPadding → {"0", "0"}, SignPadding → True];
(*-----*)
makeEllipse[e_] := Module[{a = (6378 + 300), c, b, x, y, p},
    c = e a;
    b = Sqrt[a^2 - c^2];
    p = a (1 - e^2);
    {a, b, c, p, ContourPlot[(x/a)^2 + (y/b)^2 == 1, {x, -1.1 a, 1.1 a},
        {y, -1.1 a, 1.1 a}, PlotPoints → 100, AspectRatio → Automatic]}
]
)
]

```



Out[1112]=