Abstract: This simulation illustrates a pitfall of zero slope in the Newton-Raphson method of finding roots of $f(x)=0$.

**INPUTS: Enter the Following**

Function in $f[x]$  

```mathematica
In[527]:= f[x_] := Sin[x]
```

Range of 'x' you want to see the function

```mathematica
In[528]:= x_begin = -10;
x_end = 10;
```

```mathematica
In[530]:= curve = Plot[f[x], {x, x_begin, x_end}, PlotLabel -> 
  "Entered function on given interval", TextStyle -> {FontSize -> 11}];
```

Entered function on given interval

Initial guess
Because this method uses a line tangent to the function at the initial guess, we must calculate the derivative of the function to find the slope of the line at this point. Here we will define the derivative of the function $f(x)$ as $g(x)$.

$$
\text{Iteration 1}
$$

```mathematica
In[533]:= \[ x1 = x0 - f[x0] / g[x0] \]

Out[533]= ComplexInfinity

In[534]:= \[ e_\epsilon = Abs[(x1 - x0) / x1 * 100] \]

Out[534]= Indeterminate

In[535]:= \[ \text{tanline}[x_] := f[x0] + ((0 - f[x0]) / (x1 - x0)) * (x - x0) \]

In[536]:= tline = Plot[tanline[x], {x, x0, x1}];
```
In[537]:= Show[
  Graphics[Line[{{x0, 1}, {x0, -1}}]], curve,
  Graphics[Line[{{x1, 1}, {x1, -1}}]], tline, Axes -> True,
  PlotLabel -> "Entered function on given interval with upper and lower guesses and estimated root", TextStyle -> {FontSize -> 11}];

Graphics::gptn: Coordinate ComplexInfinity in (ComplexInfinity, 1) is not a floating-point number.

Graphics::gptn: Coordinate ComplexInfinity in (ComplexInfinity, -1) is not a floating-point number.

function on given interval with upper and lower guesses and estimated