This Mathematica notebook came about as a result of two things, the first, not suprisingly, was the fact tha
MAT1750 and the second was because one of the classes I TA for was covering a somewhat computatio
Newton's method for approximating roots of functions – by computation heavy I mean it required dusting o

Of couse, all of these computations can be done manually (i.e., with any standard calculator), but why not

---

My attempt at defining a function to carry out newtons method:

```
NR[f_, start_, n_, p_: 7] := Module[{df, xn, i},
  (*
  f_  is a previously defined function,
  start_  is the x value at which to start the process,
  n_  is the term in the sequence you want,
  p_  is an optional percision value with default = 7
  *)
  If[n == 1, start,
   i = 2;
   xn = start;
   df[x_] = D[f[x], x];
   While[i ≤ n, xn = xn - f[xn]/df[xn]; i++];
   N[xn, p]
  ]
 ]
```

Let’s test it out - we’ll need some functions.

```
F[x_] := x^3 - 2;        (* This one to approximate ∛2 *)
G[x_] := Sin[x]^2;       (* This one for π *)
H[x_] := 1 - Log[x]      (* And this one for ⅇ *)
```

How do we do approximating $\sqrt[3]{2}$ starting at x = 3 ...

```
Grid[{{"n^th term:", 3, 5, 7, 9}, Prepend[Table[N[∛2, 15], {i, 4}], "∛2 :"],
  Prepend[Table[NR[F, 3, i, 15], {i, 3, 9, 2}], "Our Approx:"]},
 Alignment → {{Right}, None},
 Background → {{Cyan}, None},
 Frame → {None, None, {{1, 1} → True, {2, 1} → True, {3, 1} → True}}]
```

n^{th} term:	3	5	7	9
$\sqrt[3]{2}$:	1.25992104989487	1.25992104989487	1.25992104989487	1.25992104989487
Our Approx:	1.53769053917863	1.26160180095504	1.25992104989885	1.25992104989487

Not too bad, Let’s try for π, we’ll start at x = 3.5.

```
Grid[{{"n^th term:", 3, 5, 7, 9}, Prepend[ Table[N[π, 15], {i, 4}], "π:"],
  Prepend[Table[NR[G, 3, i, 15], {i, 3, 9, 2}], "Our Approx:"]},
 Alignment → {{Right}, None},
 Background → {{Cyan}, None},
 Frame → {None, None, {{1, 1} → True, {2, 1} → True, {3, 1} → True}}]
```

n^{th} term:	3	5	7	9
π:	3.14159265358979	3.14159265358979	3.14159265358979	3.14159265358979
Our Approx:	3.10649103014786	3.13282175366864	3.13939999889503	3.14104449101420

Unfortunately, even the ninth term is only good to 3 decimal places, lets try the seventeenth term.

```
Grid[{{N[π, 15]}, {NR[G, 3, 17, 15]}}]
```

3.14159265358979
3.14159051233002

Only FIVE decimal places, maybe this wasn’t the best choice of function for approximating π. Let’s move on to *e*, we’ll start at x = 2.

```
Grid[{{"n^th term:", 3, 5, 7, 9}, Prepend[ Table[N[E, 15], {i, 4}], "e:"],
  Prepend[Table[NR[H, 2, i, 15], {i, 3, 9, 2}], "Our Approx:"]},
 Alignment → {{Right}, None},
 Background → {{Cyan}, None},
 Frame → {None, None, {{1, 1} → True, {2, 1} → True, {3, 1} → True}}]
```

n^{th} term:	3	5	7	9
e:	2.71828182845905	2.71828182845905	2.71828182845905	2.71828182845905
Our Approx:	2.71624392635579	2.71828182845894	2.71828182845905	2.71828182845905

This one turned out pretty well - accurate to 15 decimal places by the 7th term in the sequence!

Let’s see how the varying the starting point changes things. We’ll look at the 5th term in the sequence, starting at x = 3.5, 3.4, 3.3 and 3.2.

```
Grid[{{"Starting pt:", 3.5, 3.4, 3.3, 3.2},
  Prepend[Table[N[π, 15], {i, 4}], "π:"],
  Prepend[Table[NR[G, i, 7, 15], {i, 3.5, 3.2, -0.1}], "Our Approx:"]},
 Alignment → {{Right}, None},
 Background → {{Cyan}, None},
 Frame → {None, None, {{1, 1} → True, {2, 1} → True, {3, 1} → True}}]
```

Starting pt:	3.5	3.4	3.3	3.2
π:	3.14159265358979	3.14159265358979	3.14159265358979	3.14159265358979
Our Approx:	3.14687	3.14551	3.14404	3.1425

Aside from our approximation getting better as we move closer to the root (as expected), for some reason

```
NR[F, 3.1, 7, 15]
```

1.25992

```
NR[F, 3, 7, 15]
```

```
1.25992104989885
```

```
NR[F, π, 7, 15]
```

```
1.25992104992725
```

```
NR[F, 4.512345, 7, 15]
```

```
1.25992
```

I haven’t been able to sort this out. Of course, one can always make use of the built in NSolve function.

```
NSolve[1 - Log[x] == 0, x]
```

```
{{x → 2.71828}}
```

Or even just the solve function to get a more precise solution

```
Solve[1 - Log[x] == 0, x]
```

```
{{x → ⅇ}}
```