Computer Programming for Non-majors

Lecture #6 – Numbers, Numbers Everywhere – Using Library Functions For Basic Calculations

Math Library

- The Math library in Python provides programmers with a large range of mathematical functions.

Basic Math Functions

- **`math.ceil(x)`** – returns a float equal to the smallest integer greater than x
- **`math.floor(x)`** - returns a float equal to the largest integer less than x
- **`math.trunc()`** – truncates the float to an integer (similar to floor)
- **`math.fabs(x)`** – absolute value of x
- **`math.factorial(x)`** – the product of every integer from 1 to x (x must be positive)

Basic Math Functions – An Example

```
# We need to import the math library
import math

# Our two sample values
x = 2.718281828
y = - 3.14159

# Let's write the values
print("x = ", x, "\ty = ", y)
```

```
#  Ceiling, floor and truncation
print("ceil(x) = ", math.ceil(x),
          "\tceil(y) = ", math.ceil(y))
print("floor(x) = ", math.floor(x),
          "\tfloor(y) = ", math.floor(y))
print("trunc(x) = ", math.trunc(x),
          "\ttrunc(y) = ", math.trunc(y))

# Absolute value and factorial
print("fabs(x) = ", math.fabs(x),
          "\tfabs(y) = ", math.fabs(y))
print("factorial(trunc(x)) = ",
          math.factorial(math.trunc(x)))
```

Basic Math Functions – The Output

```
>>>
x =  2.718281828  y =  -3.14159
ceil(x) =  3      ceil(y) =  -3
floor(x) =  2     floor(y) =  -4
trunc(x) =  2     trunc(y) =  -3
fabs(x) =  2.718281828  fabs(y) =  3.14159
factorial(trunc(x)) =  2
>>>
```

Power and Logarithm Functions

- **`math.exp(x)`** – e^x – x raised to the e power
- **`math.log(x)`** = $ln\ x$ = $\log_e$ x – natural logarithm of x
- **`math.log(x, base)`** = $log_{base}\ x$ = logarithm of x using the specified base
- **`math.log10(x)`** = $log_{10}\ x$ = common logarithm of x
- **`math.pow(x, y)`** = x^y = x to the y power
- **`math.sqrt(x)`** = $\sqrt{x}$ = square root of x

Exponent and Logarithm Functions An Example

```
# We need to import the math library
import math

# Our two sample values
x = 2
y = 3

# Let's write the values
print("x = ", x, "\ty = ", y)
```

```
#  Exponential and logarithm functions
print("exp(x) = ", math.exp(x),
            "\texp(y) = ", math.exp(y))
print("log(x) = ", math.log(x),
            "\tlog(y) = ", math.log(y))
print("log(x, y) = ", math.log(x, y))
print("log(x, 10) = ", math.log(x, 10),
            "\tlog(y, 10) = ", math.log(y, 10))
print("log10(x) = ", math.log10(x),
            "\tlog10(y) = ", math.log10(y))
```

```
# power and square root functions
print("pow(x, y) = ", math.pow(x, y),
           "\tfabs(y, x) = ", math.pow(y, x))
print("sqrt(x) = ", math.sqrt(x),
           "\tsqrt(y) = ", math.sqrt(y))
```

Exponents and Logarithms – Output

Trigonometric Functions

- **`math.sin(x)`** – sine of *x*, measured in radians
- **`math.cos(x)`** – cosine of *x*, measured in radians
- **`math.tan(x)`** – tangent of *x*, measured in radians

Trigonometric Functions – An Example

```
# We need to import the math library
import math

# Our two sample values
x = 3.14159/2
y = 3.14159/3

# Let's write the values
print("x = ", x, "\ty = ", y)
```

```
#  Trigonometric functions
print("sin(x) = ", math.sin(x),
                "\n\tsin(y) = ", math.sin(y))
print("cos(x) = ", math.cos(x),
                "\n\tcos(y) = ", math.cos(y))
print("tan(x) = ", math.tan(x),
                "\n\ttan(y) = ", math.tan(y))
```

Trigonometric Functions - Output

```
>>>
x =  1.570795     y =  1.0471966666666666
sin(x) =  0.9999999999991198
      sin(y) =  0.8660249615191342
cos(x) =  1.3267948966775328e-06
      cos(y) =  0.5000007660251953
tan(x) =  753695.9951408089
      tan(y) =  1.732047269454573
>>>
```

Inverse Trig Functions – An Example

```
# We need to import the math library
import math

# Our two sample values
x = 0
y = 1

# Let's write the values
print("x = ", x, "\ty = ", y)
```

```
#  Inverse Trigonometric functions
print("asin(x) = ", math.asin(x),
                    "\n\tasin(y) = ", math.asin(y))
print("acos(x) = ", math.acos(x),
                    "\n\tacos(y) = ", math.acos(y))
print("atan(x) = ", math.atan(x),
                    "\n\tatan(y) = ", math.atan(y))
```

Inverse Trig Functions - Output

```
>>>
x =  0      y =  1
asin(x) =  0.0
      asin(y) =  1.5707963267948966
acos(x) =  1.5707963267948966
      acos(y) =  0.0
atan(x) =  0.0
      atan(y) =  0.7853981633974483
>>>
```

Constants and Angle Conversions

- math.pi = 3.14159265358979…
- math.e = 2.718281728…
- math.degrees(x) – converts x degrees to radians
- math.radians(x) – converts x radians to degrees

Angle Conversion – An Example

```
# We need to import the math library
import math

# Our two sample values
x1 = 45
x2 = 60
y1 = math.pi/3
y2 = math.pi/2

#  Let's write the constant values
print("pi = ", math.pi, "\n\t\te = ", math.e)
```

Constants and Angle Conversion – An Example

```
# Let's write the values
print("x1 = ", x1, "\n\t\tx2 = ", x2)
print("y1= ", y1, "\n\t\ty2 = ", y2)


#  Inverse Trigonometric functions
print("radians(x1) = ", math.radians(x1),
           "\n\t\tradians(x2) = ",
math.radians(x2))
print("degrees(y1) = ", math.degrees(y1),
        "\n\t\tdegrees(y2) = ", math.degrees(y2))
```

Constants and Angle Conversions - Output

```
>>>
pi =  3.141592653589793
            e =  2.718281828459045
x1 =  45
            x2 =  60
y1=  1.0471975511965976
            y2 =  1.5707963267948966
radians(x1) =  0.7853981633974483
            radians(x2) =  1.0471975511965976
degrees(y1) =  59.99999999999999
            degrees(y2) =  90.0
>>>
```

TrigTable.py

```
# We'll need the math library
import math

#  x will go from 0 degrees to 90 degrees
for x in range(0, 91) :
    #convert degrees to radians
    radians = x*math.pi/180.0

    #  calculate sine, cosine and tangent
    #  for x
    sinx  = math.sin(radians)
    cosx = math.cos(radians)
```

```
    #  tan 90 degrees is infinite
    #  so just write "infinity"
    #  Inboth cases print the values
    if x != 90 :
      tanx = math.tan(radians)
      print("   %2d  %7.5f\t%7.5f\t%9.5f"
              %(x, sinx, cosx, tanx))
    else :
      tanx = "infinity"
      print("   %2d  %7.5f\t%7.5f  %s"
              %(x, sinx, cosx, tanx))
```

CalcE.py

```
# We'll need the math library
import math

sum = 0
for i in range(20) :
    term = 1.0 / math.factorial(i)
    sum = sum + term

print (sum)
print (math.e)
```

CalcE2.py

```
# We'll need the math library
import math

term = 1
sum = 1
for i in range(1, 19) :
    term = term/i
    sum = sum + term

print (sum)
print (math.e)
```

EvalEqn.py

```
#We will need the math library
import math

#  Evaluate the expression (ax^4-b)/pi *sin(pi x)
a = float(input("Enter the value of a ?"))
b = float(input("Enter the value of b ?"))
x = float(input("Enter the value of x ?"))

result = ((a * math.pow(x, 4) - b) / math.pi) *
math.sin(math.pi * x)
print("The answer is ", result)
```