**Computer Programming - Math Functions**

There are many built-in Math functions that we can access in Visual Basic. The follow tables listed many of them and gives an example of its implementation:

|  |  |  |  |
| --- | --- | --- | --- |
| **Function** | **Purpose** | **Syntax** | **Example** |
| Math.Round() | Rounds off decimal places | Math.Round(value or expression, # of decimals) | p = Math.Round(3.1415926, 4) (p= 3.1416) |
| Math.PI | Provides value of PI (3.1415926….) | Math.PI | TwoPi = Math.PI \* 2 (TwoPi = 6.28 ish) |
| Math.Pow | Raises a value to a specified power | Math.Pow(value or expression, Power) | x = Math.Pow(4, 3) (x= 64 or 43) |
| Math.Sqrt | Returns the square root of a value | Math.Sqrt(value or expression) | x = Math.sqrt(81) (x = 9) |
| Math.Abs | Returns the absolute value of a value | Math.Abs(value or expression) | x = Math.abs(-7) (x = 7) |
| Math.Truncate | Returns the integer value of the number | Math.Truncation(value or expression) | x = Math.Truncate(3.1415926) (x = 3) |
| Mod | The mod function calculates the remainder between two integers | Num1 MOD Num2,( Num1 and Num2 are integers) | x = 19 Mod 4 (x = 3 )4 goes into 19 four times and has a remainder of 3. |

Math.cos, Math.sin, Math.truncate, Math.acos, Math.asin, Math.atan, etc….

**Computer Programming - Math Functions**

There are many built-in Math functions that we can access in Visual Basic. The follow tables listed many of them and gives an example of its implementation:

|  |  |  |  |
| --- | --- | --- | --- |
| **Function** | **Purpose** | **Syntax** | **Example** |
| Math.Round() | Rounds off decimal places | Math.Round(value or expression, # of decimals) | p = Math.Round(3.1415926, 4) (p= 3.1416) |
| Math.PI | Provides value of PI (3.1415926….) | Math.PI | TwoPi = Math.PI \* 2 (TwoPi = 6.28 ish) |
| Math.Pow | Raises a value to a specified power | Math.Pow(value or expression, Power) | x = Math.Pow(4, 3) (x= 64 or 43) |
| Math.Sqrt | Returns the square root of a value | Math.Sqrt(value or expression) | x = Math.sqrt(81) (x = 9) |
| Math.Abs | Returns the absolute value of a value | Math.Abs(value or expression) | x = Math.abs(-7) (x = 7) |
| Math.Truncate | Returns the integer value of the number | Math.Truncation(value or expression) | x = Math.Truncate(3.1415926) (x = 3) |
| Mod | The mod function calculates the remainder between two integers | Num1 MOD Num2,( Num1 and Num2 are integers) | x = 19 Mod 4 (x = 3 )4 goes into 19 four times and has a remainder of 3. |

Math.cos, Math.sin, Math.truncate, Math.acos, Math.asin, Math.atan, etc….