# Module 4 part 2 : Data Types and Expressions ("float" and "double")

## "float" and "double" Data Types

Kind Of Data:

- "float" and "double" store Real Numbers (including fractionals)

Inner Representation of Data:

- Double: Each data uses 8 Bytes (64 bits)
- Float: Each data uses 4 Bytes (32 bits)

"Double is DOUBLE the size of a float"

- Inside Memory numbers are represented in IEEE-754
- In a double, 4 Bytes are used for the integer and 4 Bytes for the fractional
- In both a "double" and "float" the decimal point moves or floats in side a bunch of bits depending on its ~~location~~ representation

C++ Literals:

For "double": 3.4, -8.795, 6.0...

- if we add 6.0 and .0 the compiler will treat both as a double

For "floats": 3.4f, -8.795f...

- Prefix of "f" at the end of a number
- Without the "f" it is treated as a "double"

Arithmetic Operators: +, -, *, /, =, ...

- "/" does actual division in this case

"Float" and "double" Data Types

Type Casting

If x1 is an "int" and x2 "int"
And y1 is a "double" and y2 is a "double"
- Can we assign an int to "x1" and a double to "y1"? Yes.

- Can we assign an int to "y2" and a double "x2"?
  - Formally this is illegal. You can't assign a double to int and Vice Versa
    - They are not the same TYPE!

- However, the compiler won't say anything

- What we can do is use "Type Casting" syntax to "cast" one type of data to another.

- Casting: Converting the representation of a data from one type to another type

- Be mindful of what you are casting, if you cast a "double" data value to an "int", it will remove the fractional part of the value.

# Expression With Mixed Types

- What would happen if we try dividing a "double" by an "int" (Example: 5.0 / 2)

- We aren't sure if it will be treated as "div" or "division"

- The compiler will try to resolve the mixing types by casting them to be of the same type.

- Implicit Cast : Casting where the data does not lose accuracy

- Converting an "int" to a "double" does not lose accuracy
- However casting a "double" to an "int" does lose accuracy (Fractional Part)

- The compiler will then change the "int" to "double" and "/" will mean real division.