Anglia Ruskin University

The Standard Library and User-Defined Methods

Primitive types - int, float, etc

Complex types - classes

A **class** is a block of code containing data and methods

The .NET Standard Library contains 1000's of classes that can be accessed via *namespaces* to help program development.

Anglia Ruskin University

A **namespace** is an area of memory

using System;

– access all classes (eg Console) and other resources in this namespace area of memory called System

Some Standard Library Namespaces

namespace	*description*
System	General support
System.Collections.Generic	List storage
System.Linq	Query language
System.Text	Character processing
System.Threading.Tasks	Concurrency support

Anglia Ruskin University

Example: $x = y^9$

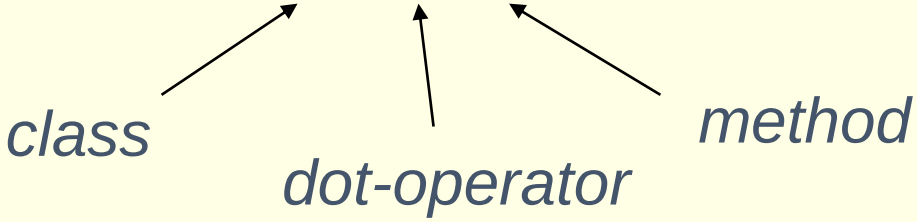
e.g:

```
int x, y = 4;
x = y * y * y * y * y * y * y * y * y;
```

Another way – use the **Pow()** method of the Standard Library **Math** class:

```
using System;

double x, y = 4.0;
x = Math.Pow( y, 9.0 );
```



Anglia Ruskin University

Method	Return value
Math.Sqrt(*x*)	Square root of *x*
Math.Log(*x*)	Natural log of *x*
Math.Max(*x, y*)	Larger of *x* and *y*

Using a class method>

returnvalue = Classname.methodname(arguments);

optional *optional*

When classname is used the method is said to be ***static***

Anglia Ruskin University

User-defined *static methods*

Eg: a program that adds two numbers together and prints result to screen

```
class CalcSum
{
    static void Main( string[ ] args )
    {
        double a = 2.0, b = 3.0, answer;
        answer = a + b;
        Console.WriteLine("Sum is " + answer );
    }
}
```

Anglia Ruskin University

Now devolve the addition to a method called sum

```
class CalcSum
{
    static void Main(string[] args)
    {
        double a = 2.0, b = 3.0, answer;
        answer = CalcSum.sum( a, b );  // Using the method
        Console.WriteLine("Sum is " + answer );
    }

    // Declaring or implementing the method
    static double sum( double x, double y )
    {
        double result;
        result = x + y;
        return result;
    }
}
```

Anglia Ruskin University

Declaring the method >

'class' method

return type

*passed arguments, i.e. they are **copies** of a and b*

```
static double sum( double x,  double y )
{
   double result;
   result = x + y;
   return result;
}
```

Locally declared variables, only 'visible between the '{ }'

*returns the result to **answer** in Main()*

Anglia Ruskin University

General way of declaring a method is:

'static' or left out

'double', 'int', etc, or if nothing returned then 'void'

[static] ***returntype*** **methodname(** ***arguments*** **)**
{
…content of method...
}

List 'type-variable' pairs, or leave empty

Anglia Ruskin University

Conclusions

- The *Standard Library* contains lots of .NET C# classes grouped into **namespaces**.
- Classes in namespaces are accessed using keyword **using**.
- We can *implement/declare* (write) our own methods
- Static (class) methods accessed with **Classname.methodname()**