What are Data Types in C#?

In C#, **data types** define the kind of data that can be stored and manipulated within a program. They specify the type of data a variable can hold, such as integers, floating-point numbers, characters, or strings. C# is a strongly-typed language, which means that you must declare the type of data a variable will store, and type safety is enforced at compile time.

Categories of Data Types in C#

1. **Value Types**: These types store data directly. They include:
    - **Numeric Types**: int, float, double, decimal, byte, short, long, uint, ulong
    - **Boolean Type**: bool
    - **Character Type**: char
    - **Structs**: User-defined structures
2. **Reference Types**: These types store references to the data. They include:
    - **Strings**: string
    - **Arrays**
    - **Classes**: User-defined classes
    - **Delegates**: Types that reference methods
3. **Special Types**:
    - **Object**: The base type for all types in C#

Example of Data Types in C#

Here is an example that demonstrates the use of different data types in C#:

```
using System;

class Program
{
    static void Main(string[] args)
    {
        // Value Types
        int age = 25;               // Integer type
        float height = 5.9f;        // Floating-point type
        double weight = 68.5;       // Double-precision floating-point type
```

```
        decimal salary = 50000.75m; // Decimal type
        bool isEmployed = true;     // Boolean type
        char initial = 'A';         // Character type

        // Reference Types
        string name = "Alice";      // String type
        int[] numbers = { 1, 2, 3 }; // Array type

        // Displaying values
        Console.WriteLine($"Name: {name}");
        Console.WriteLine($"Age: {age}");
        Console.WriteLine($"Height: {height} meters");
        Console.WriteLine($"Weight: {weight} kg");
        Console.WriteLine($"Salary: {salary:C}"); // Formatted as currency
        Console.WriteLine($"Employed: {isEmployed}");
        Console.WriteLine($"Initial: {initial}");
        Console.WriteLine($"Numbers: {string.Join(", ", numbers)}");

        // Object type example
        object obj = "This is an object type";
        Console.WriteLine($"Object: {obj}");
    }
}
```

Explanation:

1. **Value Types**:
    - `int`, `float`, `double`, `decimal`, `bool`, and `char` are examples of value types that hold data directly.
2. **Reference Types**:
    - `string` and arrays are reference types. They hold references to data rather than the data itself.
3. **Object Type**:
    - `object` is the base type from which all other types derive. It can hold any data type, and it provides a way to store values of different types in a single variable.