Difference Between Fundamental Data Types and Derived Data Types

The difference between fundamental data types and derived data types is based on their usage and functionality. The fundamental data type or primitive data type refers to the set of values that a data item may take. The derived data types are defined by the user. The difference between fundamental data types and derived data types is presented in the table given below.

Fundamental Data Types vs Derived Data Types	
Fundamental Data Types	**Derived Data Types**
They are referred to as primitive types or basic types.	They are referred to as aggregate types as they are formed by the aggregation of basic types.
Fundamental types are int, char, float, etc.	Derived types are array, structure, union, etc.
The integer type is classified as signed int, unsigned int, long int, and short int.	Pointers are used to store the address of variables.
Integers are used to store integer values in the range.	An array data type is used to store the homogenous data items.
The float type stores decimal numbers.	The structure uses primitive data types. It stores heterogeneous data.

What are Fundamental Data Types?

In C programming, the fundamental data type or primitive data type refers to the set of values that a data item may take. Every variable in C is associated with a data type. Each type requires some amount of memory and has operations associated with it.

In C, the fundamental data type is also called basic data type or primitive data type. Each basic data type has a range of valid values and sizes in bytes associated with them. These basic data types are divided into four types, as shown in the below table.

Data Type	Size of Bytes	Range	Use
char	1	–128 to 127	To store characters
int	2	–32768 to 32767	To store integer numbers
float	4	3.4E–38 to 3.4E+38	To store floating-point numbers
double	8	1.7E–308 to 1.7E+308	To store big floating-point numbers

What are Derived Data Types?

In C, the derived data types are defined by the user. This is the basic difference between fundamental data types and derived data types, as fundamental data types are predefined for the language and cannot be modified or redefined by the user or programmer. It is formed by combining one or more basic data types.

By using derived data types, we can develop a variety of new types from the basic types. The derived data types are available to provide convenience to the programmer in writing long source codes. Widely used derived data types in C programming are arrays and structures. Apart from array and structures, we have union and enumeration.

Key Difference between Fundamental Data Types and Derived Data Types

The key difference between Fundamental Data Types and Derived Data Types are given below.

- In fundamental data types, each data type requires different amounts of memory and has some specific operations which can be performed over it. In derived data types, data types are defined by the user itself. It comprises Class, Union, Arrays, Structures, Enumeration, Pointers, etc.
- In fundamental data types, we utilize character for characters. It falls under the char, signed char, and unsigned char categories. However, in Derived Data Types, pointers are used to store variable addresses.
- In fundamental data types, when a return value is not necessary, a void is utilized. A derived data type is similar to a structure, but with each Union member having access to the same memory location.