Character Arrays

Definition of Character Array

A character array is a collection of the variables of "char" datatype; it can be a one-dimensional array or a two-dimensional array. It is also called "null terminated string". A Character array is a sequence of the characters that are stored in consecutive memory addresses. In a character array, a particular character can be accessed by its index. A "Null character" terminates the character array".

For example,

char name[ ]={'A', 'j', 'a', 'y', '\0'};

or

char name[ ]="Ajay";

Name[0]	Name[1]	Name[2]	Name[3]	Name[4]
'A'	'j'	'a'	'y'	'\0'

Character Array (char Name[5])

Here, "char" is a character data type, "name" is a variable name of the character array. I had shown two ways to initialize the character array. In the first method, the null is explicitly mentioned and in the second method, the compiler automatically inserts the null.

The end of the string is always a null character; it is the terminating character of the character array. A character array is not a built-in data type; we create character array by declaring it. You can not apply standard operators on a character array.

To operate on character array there are some built-in function such as,(strlen( ), strlwr( ), strupr( ), strcat( )). As the standard operators can not be applied to a character array, they cannot take part in any expression.

The character pointer to a character array can also be created.

For example:

char s1[ ]="Hello";

char s2[ ]="Class";

s1= s1+s2; //error operators can not be applied

s2=s1; //error

Character pointer

char *s= "Morning";

char *p;

p=s; //executes

In above example, we had declared two character array s1, s2 and two character pointers s and p. Character array s1 and s2 are initialized, we can see that neither the addition operator(+) nor the assignment operator works on the character array. But a character pointer can be assigned to another character pointer.

Remember once the character array is initialized it can not be initialized again to another set of character. The access to a character array or a null terminated string is fast as compared to string in C++.

Definition of String

A string is not a built-in data type of C++. It a class object of type "string". As in C++ creating a class is just like creating a "type". The class "string" is a part of C++ library. It holds the set of character or character array as a whole. There are three reasons behind the development of a standard string class.

- First is "consistency", the character arrays are not data types in their own right.
- Second is "convenience", you can not use standard operators on a character array.
- Third is "safety", array boundaries are easily overrun.
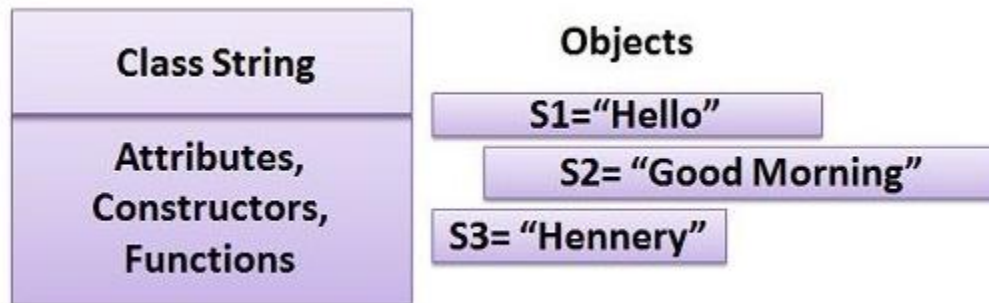
Example:

string s1;

s1= "Hello";

string s2("Good morning");

string s3="Hennery";

string s4;

In the above declaration, four string variable or objects (s1, s2, s3, s4) are declared. Here we have shown three ways of initializing the string. The string s1 is declared and then separately initialized. The string s2 is initialized by the constructor of class "String". The string s3 is initialized at the time of its

declaration as normal data type do. We can apply the standard operator to the string variables.

s4=s1;// assigning one string object to other

s4=s1+s2; // adding two string and storing the result in third string

if(s3 > s2) // comparing two strings

strings s5(s1); creating a new string object using existing string object

In above code, various operators are applied on a string and various operations are performed. The first statement copies one string object to another string object. In the second statement, two string are concatenated and stored in a third string. In the third statement, two string are compared. In the fourth statement, a new string object is created using the already existing string object.

Access to the string is slow as compared to a character array or null terminated string.

Key Differences Between Character Array and String

1. A character array is a collection of variables which are of character datatype. String is a class that is instantiated to declare strings.

2. Using index value you can access a character from a character array. On the other hand, if you want to access a particular character in a string, you can access it by function string's_name.charAt(index).

3. As an array is not a datatype similarly a character also is not a datatype. On the other hand, String being a class act as a reference type hence, it can be said String is a data type.

4. You can not apply any operator on a character array whereas, you can apply operators on String.

5. Being an array character array has a fixed length and its boundaries can be easily overrun. Where String does not have any boundaries.

6. Array elements are stored in a contiguous memory location hence that can be accessed faster than string variable.