C++ Strings

In C++, string is an object of **std::string** class that represents sequence of characters. We can perform many operations on strings such as concatenation, comparison, conversion etc.

C++ String Example

Let's see the simple example of C++ string.

```
#include <iostream>
using namespace std;
int main( ) {
   string s1 = "Hello";
      char ch[] = { 'C', '+', '+'};
      string s2 = string(ch);
      cout<<s1<<endl;
      cout<<s2<<endl;
}
```

Output:

```
Hello
C++
```

C++ String Compare Example

Let's see the simple example of string comparison using strcmp() function.

```
#include <iostream>
#include <cstring>
using namespace std;
int main ()
{
  char key[] = "mango";
  char buffer[50];
```

```
  do {
     cout<<"What is my favourite fruit? ";
     cin>>buffer;
  } while (strcmp (key,buffer) != 0);
 cout<<"Answer is correct!!"<<endl;
  return 0;
}
```

Output:

```
What is my favourite fruit? apple
What is my favourite fruit? banana
What is my favourite fruit? mango
Answer is correct!!
```

C++ String Concat Example

Let's see the simple example of string concatenation using strcat() function.

```
#include <iostream>
#include <cstring>
using namespace std;
int main()
{
    char key[25], buffer[25];
    cout << "Enter the key string: ";
    cin.getline(key, 25);
    cout << "Enter the buffer string: ";
     cin.getline(buffer, 25);
    strcat(key, buffer);
    cout << "Key = " << key << endl;
    cout << "Buffer = " << buffer<<endl;
    return 0;
}
```

Output:

```
Enter the key string: Welcome to
Enter the buffer string:  C++ Programming.
Key = Welcome to C++ Programming.
Buffer =  C++ Programming.
```


C++ String Copy Example

Let's see the simple example of copy the string using strcpy() function.

```
#include <iostream>
#include <cstring>
using namespace std;
int main()
{
   char key[25], buffer[25];
   cout << "Enter the key string: ";
   cin.getline(key, 25);
   strcpy(buffer, key);
   cout << "Key = "<< key << endl;
   cout << "Buffer = "<< buffer<<endl;
   return 0;
}
```

Output:

```
Enter the key string: C++ Tutorial
Key = C++ Tutorial
Buffer = C++ Tutorial
```

C++ String Length Example

Let's see the simple example of finding the string length using strlen() function.

```
#include <iostream>
#include <cstring>
using namespace std;
int main()
{
   char ary[] = "Welcome to C++ Programming";
   cout << "Length of String = " << strlen(ary)<<endl;
   return 0;
}
```

Output:

```
Length of String = 26
```

C++ String Functions

Function	Description
int compare(const string& str)	It is used to compare two string objects.
int length()	It is used to find the length of the string.
void swap(string& str)	It is used to swap the values of two string objects.
string substr(int pos,int n)	It creates a new string object of n characters.
int size()	It returns the length of the string in terms of bytes.
void resize(int n)	It is used to resize the length of the string up to n characters.
string& replace(int pos,int len,string& str)	It replaces portion of the string that begins at character position pos and spans len characters.
string& append(const string& str)	It adds new characters at the end of another string object.
char& at(int pos)	It is used to access an individual character at specified position pos.
int find(string& str,int pos,int n)	It is used to find the string specified in the parameter.

int find_first_of(string& str,int pos,int n)	It is used to find the first occurrence of the specified sequence.
int find_first_not_of(string& str,int pos,int n)	It is used to search the string for the first character that does not match with any of the characters specified in the string.
int find_last_of(string& str,int pos,int n)	It is used to search the string for the last character of specified sequence.
int find_last_not_of(string& str,int pos)	It searches for the last character that does not match with the specified sequence.
string& insert()	It inserts a new character before the character indicated by the position pos.
int max_size()	It finds the maximum length of the string.
void push_back(char ch)	It adds a new character ch at the end of the string.
void pop_back()	It removes a last character of the string.
string& assign()	It assigns new value to the string.
int copy(string& str)	It copies the contents of string into another.
char& back()	It returns the reference of last character.
Iterator begin()	It returns the reference of first character.
int capacity()	It returns the allocated space for the string.
const_iterator cbegin()	It points to the first element of the string.
const_iterator cend()	It points to the last element of the string.
void clear()	It removes all the elements from the string.
const_reverse_iterator crbegin()	It points to the last character of the string.
const_char* data()	It copies the characters of string into an array.
bool empty()	It checks whether the string is empty or not.
string& erase()	It removes the characters as specified.
char& front()	It returns a reference of the first character.

string& operator+=()	It appends a new character at the end of the string.
string& operator=()	It assigns a new value to the string.
char operator[](pos)	It retrieves a character at specified position pos.
int rfind()	It searches for the last occurrence of the string.
iterator end()	It references the last character of the string.
reverse_iterator rend()	It points to the first character of the string.
void shrink_to_fit()	It reduces the capacity and makes it equal to the size of the string.
char* c_str()	It returns pointer to an array that contains null terminated sequence of characters.
const_reverse_iterator crend()	It references the first character of the string.
reverse_iterator rbegin()	It reference the last character of the string.
void reserve(inr len)	It requests a change in capacity.
allocator_type get_allocator();	It returns the allocated object associated with the string.

← Prev Next →