**The C++ Standard Library provides a rich set of functions and classes for handling strings through the <string> library.**

**1. Creation and Initialization**

* std::string str;: Declares an empty string.
* std::string str = "Hello, World!";: Initializes a string with a literal.
* std::string str2(str);: Initializes a string as a copy of another string.

**2. Concatenation**

* str += " more text";: Appends text to the string.
* std::string result = str1 + str2;: Concatenates two strings.

**3. Access and Modification**

* char ch = str[i];: Accesses the character at position i.
* str[0] = 'h';: Modifies the character at position 0.
* str.at(i): Accesses the character at position i with bounds checking.
* str.front(): Accesses the first character.
* str.back(): Accesses the last character.

**4. Size and Capacity**

* str.size(): Returns the number of characters in the string.
* str.length(): Synonym for size().
* str.empty(): Checks if the string is empty.
* str.capacity(): Returns the size of the storage space currently allocated.
* str.reserve(n): Requests a change in capacity to at least n characters.

**5. Substrings**

* std::string substr = str.substr(pos, len);: Returns a substring starting at position(pos) upto string length(len).

**6. Comparison**

* if (str1 == str2): Checks if two strings are equal.
* if (str1 != str2): Checks if two strings are not equal.
* if (str1 < str2): Lexicographically compares two strings.

**7. Searching**

* size\_t pos = str.find("text");: Finds the first occurrence of the substring "text".
* size\_t pos = str.rfind("text");: Finds the last occurrence of the substring "text".
* size\_t pos = str.find\_first\_of("aeiou");: Finds the first occurrence of any character in the string "aeiou".
* size\_t pos = str.find\_last\_of("aeiou");: Finds the last occurrence of any character in the string "aeiou".

**8. Insertion and Deletion**

* str.insert(pos, "text");: Inserts "text" at position pos.
* str.erase(pos, len);: Erases len characters starting from pos.
* str.clear(): Erases all characters, making the string empty.

**9. Transformation**

* std::transform(str.begin(), str.end(), str.begin(), ::toupper);: Converts the string to uppercase (requires <algorithm> and <cctype>).
* std::transform(str.begin(), str.end(), str.begin(), ::tolower);: Converts the string to lowercase (requires <algorithm> and <cctype>).

**10. Conversion**

* std::to\_string(val): Converts a number to a string.
* std::stoi(str): Converts a string to an integer.
* std::stof(str): Converts a string to a float.
* std::stod(str): Converts a string to a double.

Note:

1. **size\_t**: This is the type of the variable pos. size\_t is an unsigned integer type that is used to represent the size of objects in bytes