```cpp
1  #include <iostream>
2  #include <string>
3
4  using namespace std;
5
6  int main() {
7      // String Declaration and Initialization
8      string str1 = "Hello";
9      string str2 = "World";
10     string str = "Hello, World!";
11
12     // length() - Get the length of a string
13     int length = str1.length();  // length is 5
14
15     // append() - Concatenate two strings
16     string combined = str1.append(str2);  // combined is "HelloWorld"
17
18     // find() - Find the first occurrence of a substring
19     size_t found = combined.find("World");  // found is 5
20
21     // substr() - Extract a substring from a string
22     string substring = combined.substr(5, 5);  // substring is "World"
23
24     // compare() - Compare two strings
25     int comparison = str1.compare(str2);  // comparison is positive
26
27     // replace() - Replace a portion of a string with another string
28     string replaced = combined.replace(5, 5, "Everyone");  // replaced is "HelloEveryone"
29
30     // erase() - Remove a portion of a string
31     replaced.erase(5, 9);  // replaced is "Hello"
32
33     // insert() - Insert a string into another string at a specified position
34     replaced.insert(5, " World");  // replaced is "Hello World"
35
36     // empty() - Check if a string is empty
37     bool is_empty = str1.empty();  // is_empty is false
38
39     // clear() - Clear the contents of a string
40     str1.clear();  // str1 is now an empty string
41
42     // c_str() - Get a C-style null-terminated string
43     const char* c_style = combined.c_str();  // c_style is "HelloWorld"
44
45     // size() - Get the size of a string
46     int size = str.size();  // size is 13
47
48     // at() - Access a character at a specified index
49     char ch = str.at(7);  // ch is 'W'
50
51     // front() - Access the first character of a string
52     char first = str.front();  // first is 'H'
53
54     // back() - Access the last character of a string
55     char last = str.back();  // last is '!'
56
```

```cpp
    // substr() - Extract a substring from a string
    string substr = str.substr(7, 5);  // substr is "World"

    // find_first_of() - Find the first occurrence of any character from a set
    size_t found_first = str.find_first_of("o,");  // found_first is 4

    // find_last_of() - Find the last occurrence of any character from a set
    size_t found_last = str.find_last_of("o,");  // found_last is 9

    // replace() - Replace all occurrences of a substring with another string
    string replaced_all = str;
    size_t replace_pos = replaced_all.find("o");
    while (replace_pos != string::npos) {
        replaced_all.replace(replace_pos, 1, "x");
        replace_pos = replaced_all.find("o", replace_pos + 1);
    }
    // replaced_all is "Hellx, Wxrld!"

    // insert() - Insert a string into another string at a specified position
    string inserted = str;
    inserted.insert(7, "Beautiful ");  // inserted is "Hello, Beautiful World!"

    // erase() - Remove a portion of a string
    string erased = str;
    erased.erase(7, 9);  // erased is "Hello, !"

    // compare() - Compare two strings
    string str3 = "Hello";
    string str4 = "World";
    int comparison2 = str3.compare(str4);  // comparison2 is negative

    // find_first_not_of() - Find the first character not present in a given set
    size_t not_found = str.find_first_not_of("Helo, Wrld!");  // not_found is 8

    // find_last_not_of() - Find the last character not present in a given set
    size_t last_not_found = str.find_last_not_of("Helo, Wrld!");  // last_not_found is 12

    // compare() - Compare a substring of two strings
    int comparison3 = str1.compare(0, 4, str2);  // comparison3 is positive

    // find_first_not_of() - Find the first character not present in a given string
    size_t not_found2 = str.find_first_not_of("Hello, World!");  // not_found2 is 12

    // find_last_not_of() - Find the last character not present in a given string
    size_t last_not_found2 = str.find_last_not_of("Hello, World!");  // last_not_found2 is
6

    // to_upper() - Convert string to uppercase
    string uppercase = str;
    for (char& c : uppercase) {
        c = toupper(c);
    }

    // to_lower() - Convert string to lowercase
    string lowercase = str;
    for (char& c : lowercase) {
        c = tolower(c);
    }

    // stoi() - Convert string to integer
```

```cpp
    string num_str = "12345";
    int num = stoi(num_str);  // num is 12345

    // to_string() - Convert integer to string
    int value = 6789;
    string value_str = to_string(value);  // value_str is "6789"

    // getline() - Read a line of input into a string
    string input;
    cout << "Enter a line of text: ";
    getline(cin, input);  // Reads the entire line, including spaces

    // substr() - Extract a substring from a string until a specified character
    string data = "Temperature: 25C";
    string temp_str = data.substr(data.find(":") + 2);  // temp_str is "25C"

    // Output
    cout << "String length: " << length << endl;
    cout << "Combined strings: " << combined << endl;
    cout << "First occurrence of 'World': " << found << endl;
    cout << "Substring from index 5: " << substring << endl;
    cout << "Comparison of 'Hello' and 'World': " << comparison << endl;
    cout << "Replaced string: " << replaced << endl;
    cout << "Erased string: " << erased << endl;
    cout << "Inserted string: " << inserted << endl;
    cout << "Is str1 empty? " << (is_empty ? "Yes" : "No") << endl;
    cout << "C-style string: " << c_style << endl;
    cout << "Size of string: " << size << endl;
    cout << "Character at index 7: " << ch << endl;
    cout << "First character: " << first << endl;
    cout << "Last character: " << last << endl;
    cout << "Substring from index 7: " << substr << endl;
    cout << "First occurrence of 'o' or ',': " << found_first << endl;
    cout << "Last occurrence of 'o' or ',': " << found_last << endl;
    cout << "String with replaced 'o' characters: " << replaced_all << endl;
    cout << "String with inserted 'Beautiful ': " << inserted << endl;
    cout << "String with erased characters: " << erased << endl;
    cout << "Comparison of 'Hello' and 'World': " << comparison2 << endl;
    cout << "First character not in 'Hello, World!': " << not_found << endl;
    cout << "Last character not in 'Hello, World!': " << last_not_found << endl;
    cout << "Comparison result: " << comparison3 << endl;
    cout << "First character not in 'Hello, World!': " << not_found2 << endl;
    cout << "Last character not in 'Hello, World!': " << last_not_found2 << endl;
    cout << "Uppercase: " << uppercase << endl;
    cout << "Lowercase: " << lowercase << endl;
    cout << "String to integer: " << num << endl;
    cout << "Integer to string: " << value_str << endl;
    cout << "Entered line of text: " << input << endl;
    cout << "Substring after ':' character: " << temp_str << endl;

    return 0;
}
```