```cpp
//四：string对象上的操作
//a)判断是否为空empty()，返回的布尔值
string s1;
if (s1.empty()) //成立
{
    cout << "s1为空" << endl;
}
```

```cpp
//b)size()/length()：返回字节/字符数量 代表该字符串的长度。unsigned int
string s1;
cout << s1.size() << endl;
cout << s1.length() << endl;
string s2 = "我爱中国";
cout << s1.size() << endl;
cout << s1.length() << endl;
```

```cpp
//h)s.c_str():返回一个字符串s中的内容指针。返回一个指向正规的c字符串的指针常量，也就是以\0结尾的。
//这个函数的引入是为了与c语言兼容，在c语言中没有string类型，所以我们得通过string类对象的c_str()成
  //员函数把string对象转换成c语言中的字符串样式；
string s10 = "abC";
const char *p = s10.c_str(); //abC

char str[100];
strcpy_s(str, sizeof(str), p); 已用时间 <= 1ms
cout << str << endl; //"abC"
```

```cpp
//c)s[n]：返回s中的第n个字符（n是个整型值），n代表的是一个位置，位置从0开始，到.size()-1;
//如果用下标n超过这个范围的内容，或者本来人家一个空字符串，你也用s[n]去访问，都会产生不可预测的结果；
string s3 = "I Love China!";
if (s3.size() > 4)
{
    cout << s3[4] << endl;
}
```

```cpp
//j)字面值和string相加 +
string s1 = "abc";
string s2 = "defg";
string s3 = s1 + " and " + s2 + 'e';
cout << s3 << endl;  //abc and defge

//string s5 = "abc" + "def"; //语法上不允许这么加
//string s5 = "abc" + s1 + "def"; //中间夹一个string对象，语法上就允许。
//string s5 = "abc" + "def" + s2;错误，看来两个字符串不能挨着相加，否则就会报语法错
```

s10.c_str()返回的是存储在s10这个string对象中的字符串(char*类型)的地址,而不是s10这个对象的地址(首地址),实际上与首地址差4.

```cpp
//k)范围for针对string的使用：c++11中提供了范围for：能够遍历一个序列中的每一个元素
//string可以看成是 一个字符序列；
string s1 = "I Love China";
//for (auto c : s1) //auto：变量类型自动推断 char ，二章的第三节
//{
//  cout << c << endl; //每次输出一个字符，后边跟一个换行符
//}
for (auto &c : s1)
{
    //toupper()把小字符转成大写，大写字符没变化。
    c = toupper(c); //因为c是一个引用，所以这相当于改变s1中的值；
}
cout << s1 << endl;  //I LOVE CHINA
```