|  |  |
| --- | --- |
| 1. map的下标操作 2. 使用下标操作的返回值 | map<int, int> map1{ { 1,1 },{ 2,2 },{ 3,3 },{ 4,4 },{ 5,5 } };  int a1 = map1.at(2);  map1[6] = 6; |
| 1. map的下标操作 2. map和unordered\_map容器提供了下标运算符和at函数 3. set不支持下标，因为set么有关键字相连的值，没有意义 4. 不能对multimap与unorderred\_multimap进行下标操作，因为这些容器中有可能有多个值与一个关键字相关联 5. Map的下标运算符，如果关键字不存在map，会创建一个元素并插入到map中，关联值将进行初始化。   步骤：  1.搜索Anna元素，未找到  2.将新的关键字-值对插入到word\_count中，关键字是一个conststring，保存anna，值进行值初始化，为0  3.提取新插入的元素，并将值1赋予它  5) 只能非const的map和unordered\_map   1. 使用下标操作的返回值 2. map的下标运算(返回mapped\_type)与解引用一个迭代器返回的类型不同（返回value\_type） 3. 正常的下标运算与解引用迭代器得到同样的类型 4. 相同的是返回的是左值，可以修改 | |

