|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 2018\_2\_C++ \_13 | 학번 : | 20157135 | 이름 : | 이상우 |

* STL(Standard Template Library 활용 - map 컨테이너

|  |
| --- |
| #include <iostream>  #include <string>  #include <map>  using namespace std;  int main() {  map<string, string> dic; // 맵 컨테이너 생성. 키는 영어 단어, 값은 한글 단어  // 단어 3개를 map에 저장  dic.insert(make\_pair("love", "사랑")); // ("love", "사랑") 저장  dic.insert(make\_pair("apple", "사과")); // ("apple", "사과") 저장  dic["cherry"] = "체리"; // ("cherry", "체리") 저장  cout << "저장된 단어 개수 " << dic.size() << endl;  string eng;  while (true) {  cout << "찾고 싶은 단어>> ";  getline(cin, eng); // 사용자로부터 키 입력  if (eng == "exit")  break; // "exit"이 입력되면 종료  if(dic.find(eng) == dic.end()) // eng '키'를 끝까지 찾았는데 없음  cout << "없음" << endl;  else  cout << dic[eng] << endl; // dic에서 eng의 값을 찾아 출력  }  cout << "종료합니다..." << endl;  } |
| **[실행 결과]** |

* **auto를 이용한 변수 선언**

|  |
| --- |
| #include <iostream>  #include <vector>  using namespace std;  int square(int x) { return x\*x; }  int main() {  // 기본 타입 선언에 auto 활용  auto c = 'a'; // c는 char 타입으로 결정  auto pi = 3.14; // pi은 double 타입으로 결정  auto ten = 10; // ten은 int 타입으로 결정  auto \*p = &ten; // 변수 p는 int\* 타입으로 결정  cout << c << " " << pi << " " << ten << " " << \*p << endl;  // 함수의 리턴 타입으로 추론  auto ret = square(3); // square() 함수의 리턴 타입이 int 이므로 ret는 int로 결정  cout << \*p << " " << ret << endl;  vector<int> v = { 1,2,3,4, 5 }; //벡터 v에 5개의 원소, 1,2,3,4,5 삽입  vector<int>::iterator it;  for (it = v.begin(); it != v.end(); it++)  cout << \*it << " "; // 1 2 3 4 5 출력  cout << endl;  // 템플릿에 auto를 사용하여 복잡한 선언의 간소화  for (auto it = v.begin(); it != v.end(); it++)  cout << \*it << " "; // 1 2 3 4 5 출력  } |
| **[실행 결과]** |

* 람다식 활용 - 캡쳐리스트에 참조 활용

|  |
| --- |
| #include <iostream>  using namespace std;  int main() {  int sum = 0; // 지역 변수  [&sum](int x, int y) { sum = x + y; } (2, 3); // 합 5를 지역변수 sum에 저장  cout << "합은 " << sum;  } |
| **[실행 결과]** |

* STL 템플릿에 람다식 활용

|  |
| --- |
| #include <iostream>  #include <vector>  #include <algorithm> // for\_each() 알고리즘 함수를 사용하기 위함  using namespace std;  void print(int n) {  cout << n << " ";  }  int main() {  vector<int> v = { 1, 2, 3, 4, 5 };  // for\_each()는 벡터 v의 첫번째 원소부터 끝까지 검색하면서,  // 각 원소에 대해 print(int n) 호출. 매개 변수 n에 각 원소 값 전달  for\_each(v.begin(), v.end(), print);  } |
| **[실행 결과]** |

* 프로그램 과제

1. 교재 528p : 문제 10번

|  |
| --- |
| **[프로그램 소스]**  #include <iostream>  #include <cstdlib>  #include <vector>  #include <String>  #include<ctime>  using namespace std;  class Nation {  string nation, capital;  public :  Nation() {};  void insert(string na,string cp) {  this->nation = na;  this->capital = cp;  }  string getna(){ return nation; }  string getcp() { return capital; }  };  int main() {  bool tf = true;  string nation, capital;  Nation na;  int ok = 0,no=0,set, choice = 0;;  vector<Nation> v;  na.insert("한국", "서울");  v.push\_back(na);  na.insert("미국", "워싱턴");  v.push\_back(na);  na.insert("스페인", "마드리드");  v.push\_back(na);  na.insert("일본", "도쿄");  v.push\_back(na);  na.insert("중국", "베이징");  v.push\_back(na);  int menu,size=v.size()+1;  srand((unsigned)time(0));  cout << "\*\*\*\*\*나라의 수도 맞추기 게임을 시작합니다." << endl;  while (tf) {  cout << "정보 입력: 1,퀴즈 2, 종료 3 >>";  cin >> menu;  switch (menu) {  case 1:  cout << "현재 " << size << "개 의 나라가 입력되어 있습니다." << endl;  cout << "나라의 수도를 입력하세요. (no no 이면 입력 끝!)" << endl;  cout << size << " >> ";  cin >> nation >> capital;  for (int i = 0; i < v.size(); i++) {  if (v[i].getna() == nation && v[i].getcp() == capital) {  cout << "이미 존재 하는 나라 입니다." << endl;  choice = 1;  break;  }  else if (nation == "no" && capital == "no") {  break;  }  else if (choice == 0) {  na.insert(nation, capital);  v.push\_back(na);  size++;  break;  }choice = 0;  }  break;  case 2:  while (true) {  set = rand() % v.size();  cout << v[set].getna() << " 의 수도는 ?? >> " ;  cin >> capital;  if (capital == "exit") {  cout << "정답 개수 :" << ok << "오답 :" << no << "입니다" << endl;  break;  }  if (v[set].getcp() == capital) {  cout << "정답 입니다!!"<<endl; ok++;  }  else {  cout << "오답 입니다 ㅠㅠ"<< endl; no++;  }  }  break;  case 3:  tf = false;  break;  }  }  char cs = getchar();  } |
| **[실행 결과]** |

1. 교재 530p : 문제 14번

|  |
| --- |
| **[프로그램 소스]**  #include <iostream>  #include <cstdlib>  #include <map>  #include <string>  using namespace std;  int main() {  map<string, string> pw;  bool tf = true,tf\_2=true; int menu, size = 0;  string name, passwerd;  cout << "\*\*\*\*\* 암호 관리 프로그램 WHO를 시작합니다. \*\*\*\*\*" << endl;  while (tf) {  cout << "삽입:1, 검사:2, 종료:3 >> "; cin >> menu;  switch (menu) {  case 1:  cout << "이름 , 암호>> "; cin >> name>>passwerd;  pw.insert(make\_pair(name, passwerd));  break;  case 2:  name = ""; passwerd = "";  cout << "이름 ? "; cin >> name;  while (true) {  cout << "암호 ? "; cin >> passwerd;  if (pw[name] == passwerd) {  cout << "통과 !!"<<endl;  break;  }  else {  cout << "실패~~"<<endl;  }  }  break;  case 3:  tf = false;  break;  }  }  char c = getchar();  return 0;  } |
| **[실행 결과]** |