|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 2021\_1\_C++\_05\_개념활용 | 학번 : | 20185309 | 이름 : | 황명원 |

* 개념 활용

1. 제시된 main()함수를 참고하여 Sample 클래스를 구현하고 테스트하는 프로그램을 작성하시오.

int main() {

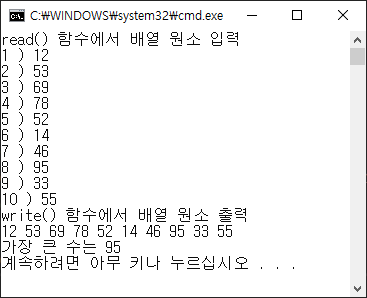
Sample s(10); // 10개 정수 배열을 가진 Sample 객체 생성, 단 정수 배열은 동적 할당

s.read(); // 키보드에서 정수 배열 읽기

s.write(); // 정수 배열 출력

cout << "가장 큰 수는 " << s.big() << endl; // 가장 큰 수 출력

}



|  |
| --- |
| [프로그램 소스]  #include <iostream> using namespace std;  class Sample{  int \*arr; //배열 동적할  int size;  public:  Sample(int n){  size=n; //배열의 크기를 size 로 생성  arr=new int[size]; //size 크기만큼 배열 생성  }  ~Sample();  void write();  void read();  int big(); //big 메소드 int 로 생성 }; Sample::~Sample(){  delete arr; } void Sample::read() { //배열 입력하는 함수 만들기  cout<<"read() 함수에서 배열원소 입력"<<endl;  for(int i=0 ; i<size ; i++){  cout<<i+1<<" ) ";  cin>>arr[i];  } } void Sample::write() { //배열 출력하는 함수 만들기  cout<<"write() 함수에서 배열 원소 출력"<<endl;  for(int i=0 ; i<size ; i++)  cout<<arr[i]<<" ";  cout<<endl; } int Sample::big(){ //배열안의 원소중 큰수를 리턴하는 함수 만들기  int big\_num;  big\_num=arr[0];  for(int i=0 ; i<size ; i++){  if(arr[i]>big\_num) //생성해놨던 big\_num이랑 비교했을때 다른 원소가 더크면 그걸 big\_num으로 초기화  big\_num=arr[i];  }  return big\_num; }  int main() {  Sample s(10); // 10개 정수 배열을 가진 Sample 객체 생성, 단 정수 배열은 동적 할당  s.read(); // 키보드에서 정수 배열 읽기  s.write(); // 정수 배열 출력  cout << "가장 큰 수는 " << s.big() << endl; // 가장 큰 수 출력 } |
| [실행결과]  텍스트이(가) 표시된 사진  자동 생성된 설명 |

1. 다음과 같은 Person 클래스가 있다. Person 클래스와 main()함수를 작성하여 객체 배열을 동적으로 생성하고 입력값으로 초기화 한 후 이름을 출력하고 검색하는 프로그램을 작성하시오

class Person {

string name;

string tel;

public:

Person();

string getName() { return name; }

string getTel() { return tel; }

void set(string name, string tel);

};

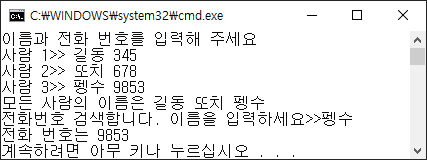
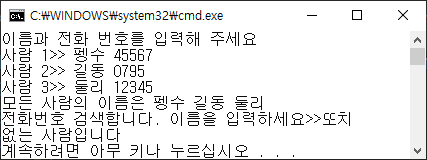
int main() {

PersonManager manager(3);

manager.show();

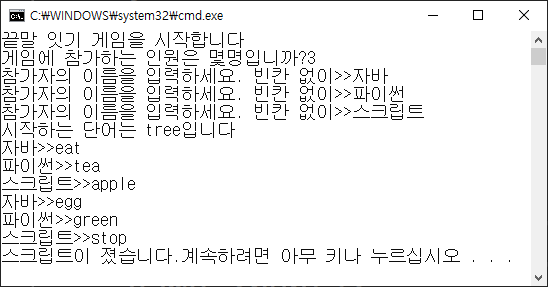
manager.search();

}

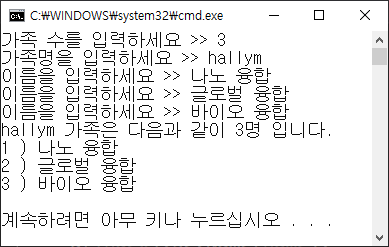
|  |
| --- |
| [프로그램 소스]  #include <iostream> using namespace std;  class Person {  string name;  string tel; public:  string getName() { return name; } //받은 이름 리턴받기   string getTel() { return tel; } //받은 전화번호 리턴받기   void set(string name, string tel){ //이름과 전화번호 입력받기  this->name=name;  this->tel=tel;  }; };  class PersonManager{  string \*arr1; //이름 저장하는 배열 생성   string \*arr2; //전화번호 저장하는 배열 생성   int size; //배열의 크기 저장하는 변수 생성  public:  PersonManager(int n){  size=n; //받은 숫자를 size에 저장   arr1=new string[size]; //size 만큼 이름배열 생성   arr2=new string[size]; //size 만큼 전화번호배열 생성   }  ~PersonManager(); //소멸자  void show(); //이름 전화번호 입력받고 이름출력하는 메소드 생성   void search();//이름을 검색하고 전화번호 알려주는메소드 생성  };  PersonManager::~PersonManager() {  delete []arr1; //이름배열 delete  delete []arr2; //전화번호배열 delete  }  void PersonManager::show() {  Person p[size];//객체 배열 생성   cout<<"이름과 전화 번호를 입력해 주세요"<<endl;  for(int i=0 ; i<size ; i++){  cout<<"사람 "<<i+1<<">>";  cin>>arr1[i]>>arr2[i];  p[i].set(arr1[i],arr2[i]);//입력받은 이름과 번호 입력   }  cout<<"모든 사람의 이름은 ";  for(int i=0 ; i<size ; i++){  cout<<p[i].getName()<<" "; //getName()으로 이름들 출력   }  cout<<endl; } void PersonManager::search() {  string in\_name;  cout<<"전화번호를 검색합니다. 이름을 입력하세요>>";  cin>>in\_name;   for(int i=0 ; i<size ; i++){  if(in\_name==arr1[i]) //조건이 true면 전화번호 나오게 출력   cout<<"전화번호는 "<<arr2[i];  } }  int main() {  PersonManager manager(3); //몇명받을 건지 생성하기   manager.show(); //이름 과 전화번호 입력   manager.search(); //이름으로 전화번호 찾기  } |
| [실행결과]  텍스트이(가) 표시된 사진  자동 생성된 설명 |

1. 제시된 결과처럼 끝말잇기 게임을 실행하는 프로그램을 작성하시오. 단, 끝말 잇기 게임에 참여하는 선수의 수를 입력 받고 선수 각 사람의 이름을 입력 받은 후 게임을 시작한다



|  |
| --- |
| [프로그램 소스]  #include <iostream> using namespace std; #include <string> #include <locale>   class Game{ //Game 클래스 생성  int size;// 배열 크기 변수생성  string word; //단어를 입력받는 변수생성  string word2; //입력받는 단어랑 비교할 단어를 저장하는 변수생성  string \*arr\_name; //이름을 동적으로 할당받는 배열로 만들기 public:  Game(int n){ //몇명이 게임을 참여할지 생성자 만들기   size=n;  arr\_name=new string[n];  };  ~Game();//소멸자   void Name(); //이름 입력받는 메소드생성   void Play\_Game(); //끝말잇기 게임하는 메소드 생성   };  Game::~Game() {   delete []arr\_name; //할당받은 이름배열을 delete } void Game::Name() {  for(int i=0 ; i<size ; i++){  cout<<"참가자의 이름을 입력하세요. 빈칸없이>>";  cin>>arr\_name[i]; //배열로 이름 입력받고 저장   } } void Game::Play\_Game() {  int word\_size;  cout<<"시작하는 단어는 tree 입니다."<<endl;  word2="tree";  int i=0;  while(true){  cout<<arr\_name[i]<<">>";  cin>>word; //단어 입력받기  word\_size=word2.length(); //단어의 크기 저장    if(word2.at(word\_size-1)==word.at(0)){ //맨뒤글자와 입력받은단어의 맨 앞글자가 같다는 조건생성   word2=word; //tree를 없애고 입력받은 단어를 넣기   }  else{  cout<<arr\_name[i]<<"가 졌습니다"; //단어가 다르면 졌다고 하기   break;//종료   }  i++; //배열 1씩증가해서 이름바꾸기   if(i>=size){ //입력받은 이름 다끝나면 다시 맨처음 이름으로 돌아오게만들기   i=0;  }  } }  int main(){  int person\_size;  cout<<"끝말잇기 게임을 시작합니다."<<endl;  cout<<"게임에 참가하는 인원은 몇명입니까?";  cin>>person\_size;   Game game(person\_size); //참가하는 수 생성  game.Name();//이름 입력하기   game.Play\_Game();//게임 시작     } |
| [실행결과]  텍스트이(가) 표시된 사진  자동 생성된 설명 |

1. Person, Family 클래스에 필요한 멤버를 추가하여 제시된 결과처럼 실행되도록 프로그램을 완성 하시오



class Person {

string name;

public:

Person(string name) { this->name = name; }

string getName() { return name; }

};

class Family {

Person\* p; // Person 배열 포인터

int size; // Person 배열의 크기. 가족 구성원 수

public:

Family(string name, int size); // size 개수만큼 Person 배열 동적 생성

void show(); // 모든 가족 구성원 출력

~Family();

};

int main() {

Family \*simpson;

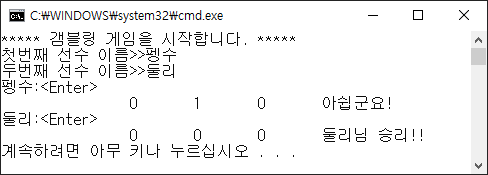
delete simpson;

}

|  |
| --- |
| [프로그램 소스]  #include <iostream> #include <string> using namespace std;  class Person {  string name; public:  Person(){};  Person(string name) { this->name = name; }  void setName(string name){this->name=name;}  string getName() { return name; } };  class Family {  Person\* p; // Person 배열 포인터  int size; // Person 배열의 크기. 가족 구성원 수  string name; public:  Family(string name, int size); // size 개수만큼 Person 배열 동적 생성  void set\_fam(int num,string name); //숫자와 이름 입력받  void show(); // 모든 가족 구성원 출력  ~Family(); //소멸자 생성 };  Family::Family(string name,int size){ //생성자 만들기  p=new Person[size]; //p 의 크기입력받고 생성  this->name=name;  this->size=size; } Family::~Family(){  delete []p; //p를 delete } void Family::set\_fam(int num,string name) {  p[num].setName(name); //입력받은 숫자와 이름을 Person 에 setName에다 설장  } void Family::show(){// 결과 출력 메소드  cout<<name<<"가족은 다음과같이 "<<size<<"명 입니다."<<endl;  for(int i=0 ; i<size ; i++){  cout<<i+1<<" ) "<<p[i].getName()<<endl; //이름 순서대로 출력  } }  int main() {   int fam\_num;  string fam\_name;  string per\_name;   cout<<"가족 수를 입력하세요 >> ";  cin>>fam\_num; //가족 수 입력받기  cout<<"가족명을 입력하세요 >>";  cin>>fam\_name; //가족명 입력받기  cin.ignore(); //교수님이 알려주신대로 getline을 쓸려했으나 출력때 오류떠서 이걸썼습니다.(입력버퍼를 비우기위함)  Family \*simpson=new Family(fam\_name,fam\_num); //simpson 생성   for(int i=0; i<fam\_num ;i++){  cout<<"이름을 입력하세요 >>";  getline(cin,per\_name); //띄어쓰기 가능하게 입력  simpson->set\_fam(i,per\_name);// 입력받은걸 설정  }   simpson->show(); //결과 출력  delete simpson; } |
| [실행결과]  텍스트이(가) 표시된 사진  자동 생성된 설명 |

1. 갬블링 게임을 조건대로 작성하시오

* n명의 사람이 게임을 진행하며 선수 이름을 초기에 입력 받는다
* 선수가 번갈아 자신의 차례에서 <enter>키를 치면 랜덤한 3개의 수를 생성하고 3개의 수가 일치하면 승자가 된다
* 숫자의 범위는 1~3
* 선수는 Player 클래스로 작성하고 n명의 선수는 배열로 구성한다



|  |
| --- |
| [프로그램 소스]  #include <iostream> #include <string> #include <ctime> #include <cstdlib> using namespace std; class Player{ //Player 클래스 만들기  string name; public:  Player(){};  Player(string name){this->name=name;}; //이름 생성  void setName(string name){this->name=name;}; //이름 설정  string getName(){return name;};//이름 리턴 };  class Gambling{  int size; //배열의 크기  Player \*p; //n명의 선수를 배열로 구성 하기위해 클래스 포인터배열 선언  int rand\_num[3];//랜덤값 배열  string enter; //enter 입력받는 변수 public:  Gambling(int num){  size=num; //배열 크기 입력한 숫자로받기  p=new Player[size];//Player 배열 크기 선언  cout<<"\*\*\*\*\* 겜블링 게임을 시작합니다. \*\*\*\*\*"<<endl;  };  ~Gambling(){  delete []p;//Player배열 delete  };  void write(); //이름 입력하는 메소드  void game(); //게임 시작하는메소드만들기 };  void Gambling::write(){  string name;  for(int i=0; i<size ; i++){ //받은 크기만큼 for문 실행  cout<<i+1<<"번째 선수 이름 >>";  cin>>name; //이름 입력  p[i].setName(name); //Player배열에 이름설정  } } void Gambling::game(){     int i=0;  while(true){   for(int i=0 ; i<3 ; i++){  rand\_num[i]=rand()%3+1; //랜덤 생성  }  cout<<p[i].getName()<<":";  cin>>enter; //enter 입력받기  if(enter!="<Enter>"){ //<Enter>가 아닐경우 다시입력하고 처음으로 돌아가게 설정  cout<<"다시 입력하십시오"<<endl;  continue;  }  for(int j=0 ; j<3 ; j++){  cout<<" "<<rand\_num[j]; //랜덤 1~3 을 3개 출력  }  if((rand\_num[0]==rand\_num[1]) and (rand\_num[1]==rand\_num[2])){//랜덤 숫자가 3개같으면 조건을 생성  cout<<" "<<p[i].getName()<<"님 승리!";  break;//3개 일치하면 반복문 종료  }  else  cout<<" 아쉽군요!"<<endl;   i++;  if(i>=size)//이름을 계속 부르기위해 조건 생성  i=0;   }  }  int main(){  Gambling gb(2);// n명으로 생성  gb.write(); //사람 이름 입력  gb.game(); //게임시작 } |
| [실행결과]    텍스트이(가) 표시된 사진  자동 생성된 설명 |