|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 2019\_1\_C++ \_09 | 학번 : | 20145165 | 이름 : | 정균모 |

* **강의 내용**

|  |
| --- |
| 1. 상속 |
| // 2차원 평면에서 한 점을 표현하는 클래스 Point 선언  **class Point** { //기본클래스  int x, y; //한 점 (x,y) 좌표값  public:  void set(int x, int y) { this->x = x; this->y = y; }  void showPoint() {  cout << "(" << x << "," << y << ")" << endl;  }  };  // 2차원 평면에서 컬러 점을 표현하는 클래스 ColorPoint. Point를 상속받음  class ColorPoint **: public Point** { //파생클래스  string color; // 점의 색 표현  public:  void setColor(string color) { this->color = color; }  void showColorPoint();  };  void ColorPoint::showColorPoint() {  cout << color << ":";  showPoint(); // Point의 showPoint() 호출  }  int main() {  Point p; **//기본 클래스의 객체 생성, 기본 클래스 객체로 파생클래스 멤버 접근 불가**  ColorPoint cp; **// 파생 클래스의 객체 생성**  cp.set(3,4); **//파생클래스의 객체로 기본 클래스의 멤버 호출, 상속관계**  cp.setColor("Red"); // 파생 클래스의 멤버 호출  cp.showColorPoint(); // 파생 클래스의 멤버 호출  } |
| **[실행결과]** |

|  |
| --- |
| 1. 상속과 객체 포인터 |
| int main() {  ColorPoint cp;  ColorPoint \*pDer;  **Point\* pBase = &cp; // 업캐스팅**  pBase->set(3,4);  pBase->showPoint();  **~~pBase->showColorPoint();~~ // 컴파일 오류**  **//강제 타입 변환 반드시 필요**  **pDer = (ColorPoint \*)pBase; // 다운캐스팅**  pDer->setColor("Red"); // 정상 컴파일  pDer->showColorPoint(); // 정상 컴파일  } |
| **[실행결과]** |

* 프로그램 과제

|  |
| --- |
| 1. 제시된 실행결과와 main()함수를 참고하여 프로그램을 완성하시오. 단, 포인터 사용 시 -> 연산자를 사용할 것. 동적 할당된 메모리는 반드시 반환할 것.   EMB00000e98bed4  int main() {  Family \*simpson;  string str; int digit;  cout << "가족이름과 인원수를 입력하세요 >>>> ";  **input(str, digit);**  simpson= new Family(str, digit);  cout << "가족 구성원 이름과 나이를 입력하세요 >>> "<<endl;  for (int i = 0; i < (\*(simpson)).getSize(); i++) {  cout << i + 1 << " ] ";  input(str, digit);  (\*(simpson)).setName(i, str, digit);  }  cout << "\n========가족 구성원 출력========= "<<endl;  (\*(simpson)).show();  }  class Person {  string name; //이름  int age; // 나이  public:  **//구현 하기 -**  }; |
| **[프로그램 소스 ]**  #include <iostream>  #include <string>  using namespace std;  class Person {  string name; //이름  int age; // 나이  public:  //구현 하기 -  Person() { }  void setName(string name) { this->name = name; }  void setAge(int age) { this->age = age; }  string getName() {  return this->name;  }  int getAge() {  return this->age;  }  };  class Family {  string Fname;  int total;  Person \*p;  public:  Family(string Fname, int total) { this->Fname = Fname; this->total = total; p = new Person[this->total]; }  int getSize() { return this->total; }  void setName(int i, string str, int digit);  void show();  ~Family() { delete[] p; }  };  void input(string &std, int &digit) {  cin >> std;  cin >> digit;  }  void Family::setName(int i, string str, int digit) {  (p+i)->setName(str);  (p+i)->setAge(digit);  }  void Family::show() {  cout << Fname << " 가족은 다음과 같이 " << total << "명 입니다." << endl;  for (int i = 0; i < total; i++) {  cout << "이름: " << (p)->getName() << " 나이: " << (p)->getAge() << endl;  p++;  }  }  int main() {  Family \*simpson;  string str; int digit;  cout << "가족이름과 인원수를 입력하세요 >>>> ";  input(str, digit);  simpson = new Family(str, digit);  cout << "가족 구성원 이름과 나이를 입력하세요 >>> " << endl;  for (int i = 0; i < (\*(simpson)).getSize(); i++) {  cout << i + 1 << " ] ";  input(str, digit);  (\*(simpson)).setName(i, str, digit);  }  cout << "\n========가족 구성원 출력========= " << endl;  (\*(simpson)).show();  system("pause");  } |
| **[실행결과]** |

|  |
| --- |
| 1. 교재 418p : 문제1번 |
| **[프로그램 소스 ]**  #include <iostream>  #include <string>  using namespace std;  class Circle {  int radius;  public:  Circle(int radius = 0) { this->radius = radius; }  int getRadius() { return radius; }  void setRadius(int radius) { this->radius = radius; }  double getArea() { return 3.14\*radius\*radius; }  };  class NamedCircle : public Circle {  string name;  public:  NamedCircle(int radius, string name) { setRadius(radius),this->name = name; }  void show() { cout <<"반지름이 "<< getRadius() << "인 "<<this->name << endl; }  };  int main() {  NamedCircle waffle(3, "waffle");  waffle.show();  system("pause");  } |
| **[실행결과]** |

|  |
| --- |
| 1. 교재 418p : 문제2번 |
| **[프로그램 소스]**  #include <iostream>  #include <string>  using namespace std;  class Circle {  int radius;  public:  Circle(int radius = 0) { this->radius = radius; }  int getRadius() { return radius; }  void setRadius(int radius) { this->radius = radius; }  double getArea() { return 3.14\*radius\*radius; }  };  class NamedCircle : public Circle {  string name;  public:  NamedCircle(int radius = 0, string name = "") : Circle(radius) {  this->name = name;  }  void setRName(int r, string name) { setRadius(r); this->name = name; }  string getName() { return name; }  };  int main() {  NamedCircle pizza[5];  int i, r; string name;  cout << "5 개의 정수 반지름과 원의 이름을 입력하세요" << endl;  for (i = 0; i < 5; i++) {  cout << i + 1 << ">> ";  cin >> r >> name;  pizza[i].setRName(r, name);  }  double max = pizza[0].getArea();  name = pizza[0].getName();  for (i = 1; i < 5; i++) {  if (pizza[i].getArea() > max) {  max = pizza[i].getArea();  name = pizza[i].getName();  }  }  cout << "가장 면적이 큰 피자는 " << name << "입니다" << endl;  system("pause");  } |
| **[실행결과]** |

|  |
| --- |
| 1. 교재 419p 문제3번 |
| **[프로그램 소스]** |
| **[실행결과]** |