|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 2019\_1\_C++ \_03 | 학번 : |  | 이름 : |  |

* **강의 내용**

|  |
| --- |
| 1. 위임생성자와 타겟 생성자 & 멤버변수 초기화 |
| #include <iostream>  using namespace std;  class Point {  int x, y;  public:  Point();  Point(int a, int b);  void show() { cout << "(" << x << ", " << y << ")" << endl; }  };  **Point::Point() : Point(0, 0) { }** // 위임 생성자  **Point::Point(int a, int b) : x(a), y(b) { }** // 타겟 생성자    int main() {  Point origin;  Point target(10, 20);  origin.show();  target.show();  } |

|  |
| --- |
| 1. C++ 구조체 - **클래스와 동일. 단, 접근 지정자 생략 시 public이 디폴트** |
| #include <iostream>  using namespace std;  struct StructCircle { // C++ 구조체 선언  private:  int radius;  public:  StructCircle(int r) { radius = r; } // 구조체의 생성자  double getArea();  };  double StructCircle::getArea() {  return 3.14\*radius\*radius;  }  int main() {  StructCircle waffle(3); **//구조체 변수 선언 시 struct 키워드 생략**  cout << "면적은 " << waffle.getArea();  } |
| **[실행결과]** |

|  |
| --- |
| 1. 조건부 컴파일로 헤더 파일 중복 문제 해결 – Main.cpp를 완성하시오. |
| **//Adder.h**  **#ifndef ADDER\_H**  **#define ADDER\_H**  **class Adder** { // 덧셈 모듈 클래스  int op1, op2;  public:  Adder(int a, int b);  int process();  };  **#endif**  **//Adder.cpp**  #include “Adder.h”  Adder::Adder(int a, int b) {  op1 = a; op2 = b;  }  int Adder::process() {  return op1 + op2;  }  **//Main.cpp – 완성된 소스를 아래에 추가하시오.** |
| **[실행결과]** |

|  |
| --- |
| 1. 객체 포인터 |
| #include <iostream>  using namespace std;  **class Circle** {  int radius;  public:  Circle() { radius = 1; }  Circle(int r) { radius = r; }  double getArea();  };  double Circle::getArea() {  return 3.14\*radius\*radius;  }  int main() {  Circle donut;  Circle pizza(30);  // 객체 이름으로 멤버 접근  cout << donut.getArea() << endl;  // 객체 포인터로 멤버 접근 : -> 사용  **Circle \*p;**  **p = &donut;**  cout << p->getArea() << endl; // donut의 getArea() 호출  cout << (\*p).getArea() <<endl; // donut의 getArea() 호출  **p = &pizza;**  cout << p->getArea() << endl; // pizza의 getArea() 호출  cout << (\*p).getArea() << endl; // pizza의 getArea() 호출  } |
| **[실행결과]** |

|  |
| --- |
| 1. 객체 배열 |
| #include <iostream>  using namespace std;  class Circle {  int radius;  public:  Circle() { radius = 1; }  Circle(int r) { radius = r; }  void setRadius(int r) { radius = r; }  double getArea();  };  double Circle::getArea() {  return 3.14\*radius\*radius;  }  int main() {  Circle circleArray[3]; // (1) Circle 객체 배열 생성, default 생성자 호출  // 배열의 각 원소 객체의 멤버 접근 (2)  circleArray[0].setRadius(10); circleArray[1].setRadius(20);  circleArray[2].setRadius(30);  for(int i=0; i<3; i++) // 배열의 각 원소 객체의 멤버 접근(2)  cout << "Circle " << i << "의 면적은 " << circleArray[i].getArea() << endl;  Circle \*p; // 객체 배열을 가리키는 포인터 선언 (3)  p = circleArray; // 초기화 (4)  for(int i=0; i<3; i++) { // 객체 포인터로 배열 접근 (5)  cout << "Circle " << i << "의 면적은 " << p->getArea() << endl;  p++; }  } |
| **[실행결과]** |

|  |
| --- |
| 1. 동적 메모리 할당 |
| #include <iostream>  using namespace std;  int main() {  cout << "입력할 정수의 개수는?";  int n;  cin >> n; **// 정수의 개수 입력**  if(n <= 0)  return 0;  **int \*p = new int[n]; // n 개의 정수 배열 동적 할당**    if(!p) {  cout << "메모리를 할당할 수 없습니다.";  return 0;  }  for(int i=0; i<n; i++) {  cout << i+1 << "번째 정수: "; **// 프롬프트 출력**  **cin >> p[i]; } // 키보드로부터 정수 입력**    int sum = 0;  for(int i=0; i<n; i++)  **sum += p[i];**  cout << "평균 = " << sum/n << endl;  **delete [] p; // 배열 메모리 반환**  } |
| **[실행결과]** |

* 내용 점검 – 3장 연습문제

|  |  |
| --- | --- |
| 번호 | 문제풀이 |
| 2 |  |
| 4 |  |
| 5 |  |
| 6 |  |
| 7 |  |
| 8 |  |
| 9 |  |
| 10 |  |
| 11 | (1)  (2)  (3) |
| 12 |  |
| 13 |  |
| 14 |  |
| 15 |  |
| 16 |  |
| 17 |  |
| 18 |  |
| 19 |  |
| 20 |  |

* 프로그램 과제

|  |
| --- |
| 1. 교재 150p 문제2번   힌트) 문자열 길이를 반환하는 인스턴스 메소드 : lenth() |
| **[프로그램 소스]** |
| **[실행결과]** |

|  |
| --- |
| 1. 교재 151p 문제4번 |
| **[프로그램 소스]** |
| **[실행결과]** |

|  |
| --- |
| 1. 교재 152p 문제7번   힌트) 디폴트 생성자에서 time() 함수를 사용하여 시드값 설정 – 매번 다른 난수를 생성 |
| **[프로그램 소스]** |
| **[실행결과]** |

|  |
| --- |
| 1. 교재 153p 문제8번 |
| **[프로그램 소스]** |
| **[실행결과]** |

|  |
| --- |
| 1. 교재 154p 문제10번 |
| **[프로그램 소스]** |
| **[실행결과]** |

|  |
| --- |
| 1. 교재 156p 문제12번 |
| **[프로그램 소스]** |
| **[실행결과]** |