1. chapter1 pdf 답
   1. 1
      1. 2 : 예측 X
      2. 3 : 동일한 X
      3. 4 : 위임 X
   2. 3
   3. 2
   4. 2
   5. 4
2. Chap 2 답
   1. 객체지향 프로그래밍 언어 3대 주요 특징
      1. Encapsulation
      2. Inheritance
      3. Polymorphism
   2. 패키지
   3. dog->Dog
   4. args []
   5. import
   6. 3
3. Chap 3
   1. 2,4
   2. 4
   3. 1
   4. 1
   5. D
4. Chap 4
   1. 4
   2. 2
   3. 4
   4. 2
5. Chap 6
   1. 3
      1. 부모 객체는 자식 객체를 형변환(casting) 없이 가리킬 수 있다.
      2. 그러나 자식 객체는 부모 객체를 형변환(casting) 후 가리킬 수 있다.
   2. 3. 컴파일오류. 자식 클래스의 객체가 생성될 때, 부모 클래스에 기본 생성자가 없으므로, 자식 클래스에서 부모를 부를 수 없어 컴파일오류.
   3. 4
      1. 파라미터가 같아서 안 됨. 파라미터는 달라야 함
   4. 3
6. 필기
   1. class
      1. 오버라이딩 할 때 예외 규칙 : 자식에서 부모보다 더 하위의 예외 클래스를 throws 할 수 있다.
         1. O
            1. 부모 : throws Exception
            2. 자식 : throws IOException
         2. X. 컴파일 오류
            1. 부모 : throws IOException
            2. 자식 : throws Exception
      2. 오버라이딩할 때 접근제한자 규칙 : 자식이 부모보다 더 넓거나 같아야 한다
         1. O
            1. 부모 : protected
            2. 자식 : public
         2. X
            1. 부모 : public
            2. 자식 : protected
   2. Polymorphism(다양한 변형)
      1. One interface, multiple implementation
      2. Ex
         1. Employee e = new Employee();
         2. Employee mgr = new Manager();
         3. Employee clerk = new MereClerk();
         4. Employee[] emps = new Employee[3];
         5. emps[0] = new Manager();
         6. emps[1] = new MereClerk();
7. 유용한 도구
   1. 단축키
      1. Shift + alt + A : 블록 지정
   2. 버튼
      1. Refactor
         1. Extract method : 중복되는 기능을 지닌 코드들을 하나의 메소드로 합쳐줌.
         2. Extract class : 서로 연관된 클래스들을 부모와 자식 관계 등의 체계적인 클래스로 바꿔준다. 그 관계 안에서 상속, getter setter 등도 설정 가능하다.
8. 실습 제출
   1. 이메일로 : [vega2k@kakao.com](mailto:vega2k@kakao.com)
   2. 과제는 3개(30/30/40)
      1. 컴파일 오류 : -10점
      2. 실행오류(런타임 오류) : -5점
      3. 결과가 문제가 요구하는 것과 다를 경우 : -3점
      4. 로직 없이 출력만 잘 될 경우 : -10점