**자료구조(박요한 교수님)**

**바이트스트림 입출력**

컴퓨터공학과 유상현

**1.** **프로그램 소스코드**

App.java

|  |
| --- |
| package Copy;  import java.io.\*;  import java.util.\*;  class Member implements Serializable{  private int ID;  private String name;  Member (int ID, String name){  this.ID = ID;  this.name = name;  }  public int getID(){  return this.ID;  }  public String getName(){  return this.name;  }  }  class ByteCopy{  void copy(String filename) throws IOException{  FileInputStream fin = new FileInputStream(filename);  FileOutputStream fout = new FileOutputStream("/Users/sanghyun/Develop/java/IOstream/src/Copy/copybyte.pdf");  int readCount = -1;  while ((readCount = fin.read()) != -1) {  fout.write(readCount);  }  fin.close();  fout.close();  }  }  class BufferCopy{  void copy(String filename) throws IOException{  BufferedInputStream bin = new BufferedInputStream(new FileInputStream(filename));  BufferedOutputStream bout = new BufferedOutputStream(new FileOutputStream("/Users/sanghyun/Develop/java/IOstream/src/Copy/copybuffer.pdf"), 40);  int c;  while ((c = bin.read()) != -1) {  bout.write((char) c);  }  bout.close();  bin.close();  }  }  class fileCopy{  private ByteCopy bc;  private BufferCopy bfc;  private String filePath;  public fileCopy(){  bc = new ByteCopy();  bfc = new BufferCopy();  }    public void setFilePath(String filepath){  this.filePath = filepath;  }  public void ByteCopyMethod() throws IOException{  System.out.println("1 바이트씩 복사");  long beforeTime = System.currentTimeMillis();  bc.copy(filePath);  long afterTime = System.currentTimeMillis();  long diffTime = afterTime - beforeTime;  System.out.println("실행 시간(ms): " + diffTime);  }  public void BufferCopyMethod() throws IOException{  System.out.println("버퍼 복사");  long beforeTime = System.currentTimeMillis();  bfc.copy(filePath);  long afterTime = System.currentTimeMillis();  long diffTime = afterTime - beforeTime;  System.out.println("실행 시간(ms): " + diffTime);  }  }  class Program{  private HashSet<Member> hashSet;  private String filePath;  Program(){  hashSet = new HashSet<Member>();  }  public void setFilePath(String filePath){  this.filePath = filePath;  }  public void addMember(Member input){  hashSet.add(input);  }  public void removeMember(int ID){  Iterator<Member> iter = hashSet.iterator();  while(iter.hasNext()){  Member tmp = iter.next();  if(tmp.getID() == ID){  System.out.println("ID" +tmp.getID()+ "delete complete");  iter.remove();  break;  }  }  }  public void printAll(){  Iterator<Member> iter = hashSet.iterator();  while(iter.hasNext()){  Member tmp = iter.next();  System.out.println("ID: " +tmp.getID() + " Name: "+tmp.getName());  }  }  public void writeClass() throws IOException, ClassNotFoundException{  ObjectOutputStream oos = new ObjectOutputStream(new FileOutputStream(filePath));  oos.writeObject(hashSet);  oos.close();  }  public void readClass() throws IOException, ClassNotFoundException{  ObjectInputStream ois = new ObjectInputStream(new FileInputStream(filePath));  Object obj = ois.readObject();  if (obj instanceof HashSet) {  hashSet = (HashSet<Member>) obj;  } else {  System.out.println("Wrong type Object");  }  ois.close();  }  }  public class App {  public static void main(String[] args) throws IOException, ClassNotFoundException{  // 임의의 파일을 복사하는 코드  System.out.println("파일 복사");  String filename = "/Users/sanghyun/Develop/java/IOstream/src/Copy/14(a) 파일입출력\_바이트스트림.pdf";  fileCopy fc = new fileCopy();  fc.setFilePath(filename);  fc.ByteCopyMethod();  fc.BufferCopyMethod();  System.out.println("");  // 객체 파일로 저장해서 다시 불러오는 코드  System.out.println("객체 복사");  Program program = new Program();  Program test = new Program();  program.setFilePath("/Users/sanghyun/Develop/java/IOstream/src/Copy/objectcopy.txt");  test.setFilePath("/Users/sanghyun/Develop/java/IOstream/src/Copy/objectcopy.txt");  program.addMember(new Member(0, "Hello"));  program.addMember(new Member(1, "World"));  program.addMember(new Member(2, "Java"));  program.writeClass();  test.readClass();  test.printAll();  }  } |

**2. 실행 화면**

텍스트, 폰트, 스크린샷, 번호이(가) 표시된 사진

자동 생성된 설명

텍스트, 소프트웨어, 멀티미디어 소프트웨어, 스크린샷이(가) 표시된 사진

자동 생성된 설명



**3. 느낀 점(고찰)**

바이트 단위를 이용해서 파일을 복사할 수 있다는 것이 재미있었고 조금은 특이했던 것이 객체 복사였는데 객체를 단순히 소스파일에서만 작업해놓는 것이 아니라 객체를 바이트를 이용해서 저장할 수 있고 이를 다시 불러올 수 있다는 것이 활용성이 높아보였다.