**자료구조(박요한 교수님)**

**ArrayList를 활용하여 학생 정보 관리하기 과제**

컴퓨터공학과 유상현

**1.** **프로그램 소스코드**

App.java

|  |
| --- |
| package Collection2;  import java.util.\*;  class Member {  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;  }  @Override  public boolean equals(Object o){  if (this == o)  return true;  if (!(o instanceof Member))  return false;  Member member = (Member) o;  if( ID == member.getID() && Objects.equals(ID, member.getID()) ){  System.out.printf("ID %d is exist!!\n", ID);  return true;  }  else  return false;  }  @Override  public int hashCode(){  return Objects.hash(ID);  }  }  class Program{  private HashSet<Member> hashSet;  Program(){  hashSet = new HashSet<Member>();  }  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("delete complete");  iter.remove();  break;  }  }  }  public void printAll(){  Iterator<Member> iter = hashSet.iterator();  while(iter.hasNext()){  Member tmp = iter.next();  System.out.printf("ID : %d Name: %s\n",tmp.getID(), tmp.getName());  }  }  }  public class App {  public static void main(String[] args) {  Program program = new Program();  program.addMember(new Member(12, "kim"));  program.addMember(new Member(12, "you"));  program.addMember(new Member(25, "park"));  program.addMember(new Member(37, "song"));  program.printAll();  program.removeMember(12);  program.printAll();  }  } |

**2. 실행 화면**

**텍스트, 폰트, 스크린샷, 번호이(가) 표시된 사진

자동 생성된 설명**

**3. 느낀 점(고찰)**

HashSet을 이용하여 중복되는 값을 개발자가 직접 찾는 것이 아닌 hash 값을 통해 편리하게 구현할 수 있다는 것이 편리했고, 또 hash 값에 대해 오버라이딩하여 쓸 수 있다는 것이 편리했습니다.