**자료구조(박요한 교수님)**

**카드 생성 후 정렬하기 과제**

컴퓨터공학과 유상현

**1. 프로그램 소스코드**

**Main.java**

|  |
| --- |
| package Card;  import java.util.Arrays;  import java.util.Scanner;  public class Main {  static void PickUpCard(Card[] cards) {  for (int i = 0; i < cards.length; i++) {  boolean IsSame = false;  cards[i] = new Card();  do {  cards[i].Reroll();  IsSame = false;  for (int k = 0; k < i; k++) {  if (cards[i].GetNumber() == cards[k].GetNumber() && cards[i].GetPatten() == cards[k].GetPatten()) {  IsSame = true;  break;  }  }  } while (IsSame);  }  }  static void PrintCard(Card[] cards) {  for (int i = 0; i < cards.length; i++)  cards[i].PrintCardValue();  }  static Card[] Sorting(Card[] cards){  Card[] sortCards = Arrays.copyOf(cards, cards.length);  int MaxIndex;  for(int i = 0; i < sortCards.length; i++){  MaxIndex = i;  for(int k = i + 1; k < sortCards.length; k++){  if (sortCards[k].GetPatten() > cards[MaxIndex].GetPatten()) {  MaxIndex = k;  }  else if (sortCards[k].GetPatten() == sortCards[MaxIndex].GetPatten()){  if(sortCards[k].GetNumber() > sortCards[MaxIndex].GetNumber()){  MaxIndex = k;  }  }  }  Card temp = sortCards[i];  sortCards[i] = sortCards[MaxIndex];  sortCards[MaxIndex] = temp;  }  return sortCards;  }  public static void main(String[] args) {  Scanner scanner = new Scanner(System.in);  System.out.printf("몇 명이 게임을 합니까? : ");  int input = scanner.nextInt();  Card[] cards = new Card[input];  PickUpCard(cards);  Card[] sortCards = Sorting(cards);  System.out.println("정렬 전 카드");  PrintCard(cards);  System.out.println("정렬 후 카드");  PrintCard(sortCards);  scanner.close();  }s  } |

**Patten.java**

|  |
| --- |
| package Card;  public enum Patten {  CLOVER("클로버"), HEART("하트"), DIAMOND("다이아몬드"), SPADE("스페이드");  private String KorName;  Patten(String KorName) {this.KorName = KorName;}  String GetKorName() {return this.KorName;}  } |

**Number.java**

|  |
| --- |
| package Card;  public enum Number {  Blank("B"), Two("2"), Three("3"), Four("4"), Five("5"),  Six("6"), Seven("7"), Eight("8"), Nine("9"), Ten("10"),  Jack("J"), Queen("Q"), King("K"), Ace(“A”);  private String RealNumber;  Number(String RealNumber) {this.RealNumber = RealNumber;}  String GetRealNumber() {return this.RealNumber;}  } |

**Card.java**

|  |
| --- |
| package Card;  import java.util.Random;  public class Card {  private int patten, number, playerNumber;  private static int playerCounter = 1;  private Random random = new Random();  // Constructor  public Card() {  patten = random.nextInt(4);  number = random.nextInt(13) + 1;  this.playerNumber = playerCounter++;  }  public Card(Patten patten) {  this.patten = patten.ordinal();  number = random.nextInt(13) + 1;  this.playerNumber = playerCounter++;  }  public Card(Number number) {  patten = random.nextInt(4);  this.number = number.ordinal();  this.playerNumber = playerCounter++;  }  public Card(Patten patten, Number number) {  this.patten = patten.ordinal();  this.number = number.ordinal();  this.playerNumber = playerCounter++;  }  public void PrintCardValue() {  System.out.println("플레이어" + this.playerNumber + " : " + Patten.values()[patten].GetKorName() + Number.values()[number].GetRealNumber());  }  public int GetPatten(){  return this.patten;  }  public int GetNumber(){  return this.number;  }  public void Reroll(){  patten = random.nextInt(4);  number = random.nextInt(13) + 1;  }  } |

**2. 실행 화면**

텍스트, 스크린샷, 폰트, 흑백이(가) 표시된 사진

자동 생성된 설명

**3. 느낀점(고찰)**

클래스를 배열로 선언하여서 자료형처럼 사용할 수 있다는 것을 알게되었다. 기존에 클래스를 다루는 것보다 훨씬 더 쉽게 다룰 수 있었던 것 같다. 클래스로 배열을 만드는 법을 몰랐다면 이번과제를 수행하면서 코드가 굉장히 길어졌을 것 같다는 생각도 들었다.