**자료구조(박요한 교수님)**

**트럼프카드 비교하기 과제**

컴퓨터공학과 유상현

**1. 프로그램 소스코드**

**Main.java**

|  |
| --- |
| package cardSystem;  public class Main {  public static void main(String[] args) throws Exception {  Card card1 = new Card(1);  Card card2 = new Card(2);  card1.PickUpCard();  card2.PickUpCard();  while (card1.Compare(card2)) {  card2.PickUpCard();  }  card1.PrintCardValue();  card2.PrintCardValue();  card1.Game(card2);  }  } |

**Card.java**

|  |
| --- |
| package cardSystem;  import java.util.Random;  public class Card {  public int patten, number, player;  // Constructor  Card(int input) {  player = input;  }  // Method  public void PrintCardValue() {  System.out.println("Player" + this.player + " : " + Patten.values()[patten].GetKorName() + Number.values()[number].GetRealNumber());  }  public void PickUpCard(){  Random random = new Random();  patten = random.nextInt(4);  number = random.nextInt(13);  }  public int GetPatten(){  return this.patten;  }  public int GetNumber(){  return this.number;  }  public boolean Compare(Card card){  if(this.number == card.number && this.patten == card.patten){  return true;  }  else {  return false;  }  }  public void Game(Card card){  boolean Issame = false;  if (this.number > card.number) {  System.out.println("우승자는 " + this.player + "번 입니다.");  return;  } else if (this.number < card.number) {  System.out.println("우승자는 " + card.player + "번 입니다.");  return;  }  else if (this.number == card.number) {  Issame = true;  }  if (Issame) {  if (this.patten > card.patten) {  System.out.println("우승자는 " + this.player + "번 입니다.");  return ;  }  else {  System.out.println("우승자는 " + card.player + "번 입니다.");  return ;  }  }  }  } |

**Patten.java**

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

**2. 실행 화면**

텍스트, 폰트, 스크린샷, 타이포그래피이(가) 표시된 사진

자동 생성된 설명

텍스트, 폰트, 스크린샷, 그래픽이(가) 표시된 사진

자동 생성된 설명

**3. 느낀점(고찰)**

클래스를 왜 사용하는지에 대해서 의문이 조금 있었는데 파일을 분할해보면서 코드의 가독성이 올라가는 것이 좋았고 나중에 코드의 재사용성도 올라갈 것 같아서 코드를 작성하면서 파일을 분할하는 습관을 들여야 할 것 같다.