JAVA 과제 #2

201810754 조윤직

제출일: 2019. 9. 30

기억 매칭 카드 게임

문제정의 : 맨처음에 1~8의 두 장씩 있는 카드를 무작위로 4X4 배치를 하고 일정 시간 동안 보여준 뒤 다시 뒤집어 가린 다음에 서로 일치하는 카드를 골랐을 때만 다시 뒤집어 준다.

문제 분석 및 해결책 :

cards 클래스 맴버 변수:

card: int 형 2차원 4\*4배열 카드의 값을 담을 변수

card\_hide: bool형 2차원배열 4\*4 :카드를 출력 할 때 카드가 숨겨졌는지 안숨겨 졌는지 알기 위해

match\_count: 16개 카드중 몇 개를 맞추었는지 게임 종료를 위한 int변수

소스코드:

private int card[][];

private Boolean card\_hide[][];

int match\_count =0;

맴버 함수:

cards(): 카드 클래스의 생성자 card와 card\_hide에 new로 동적할당을 해주고 이중for문으로 card\_hide에는 모두 false를, card에는 기본 값으로 초기화 해준다.

cards() {

card =new int[4][4];

card\_hide= new Boolean[4][4];

for(int i=0;i<card.length;i++)

for(int j = 0; j<card[i].length;j++) {

card\_hide[i][j]=false;

card[i][j] = i\*2 + j/2 +1;

}

}

printcard(): 카드 상태를 출력해주는 함수, 이중 for문과 if를 사용하여 카드 하나씩 출력, 숨겨져있으면 \*로 아니면 card의 값을 출력해줌

public void printcard() {

System.***out***.printf("%5s","| ");

System.***out***.println("1 2 3 4");

System.***out***.println("======================");

for(int i=0;i<card.length;i++) {

System.***out***.printf("%d%4s",i+1 ,"| ");

for(int j=0;j<card[i].length;j++) {

if(card\_hide[i][j])

System.***out***.print("\* ");

else

System.***out***.print(card[i][j] + " ");

}

System.***out***.println();

}

}

static cmd\_cls() : 콘솔창 지우는 부분을 함수로 만들어 둠.

static void cmd\_cls() {

try {

new ProcessBuilder("cmd", "/c", "cls").inheritIO().start().waitFor();

} catch (InterruptedException | IOException e) {

e.printStackTrace();

}

}

hide\_card(): 전체 카드를 모두 뒤집어주는(숨겨주는)

public void hide\_card(boolean hide) {

for(int i=0;i<card.length;i++)

for(int j = 0; j<card[i].length ;j++) {

card\_hide[i][j]=hide;

}

}

mix\_card(): 카드를 초기 값에서 섞어주는 함수, 랜덤하게(random클래스 이용) 두카드를 선택하여 서로 바꾸는걸 카드의 개수만큼 실행하여 카드를 초기값에서 무작위로 섞어 주었다.

public void mixcard() {

System.***out***.println("카드 섞기 완료");

for(int i=0; i<16;i++) {

int x1= *rand*.nextInt(4);

int x2= *rand*.nextInt(4);

int y1= *rand*.nextInt(4);

int y2= *rand*.nextInt(4);

int temp =card[x1][y1];

card[x1][y1]=card[x2][y2];

card[x2][y2] =temp;

}

hide\_card(false);

checkcard(int x1, int y1,int x2, int y2): 먼저 함수안에 try catch로 인덱스를 벗어날경우 인덱스오류 메시지와 함께 함수를 종료 하게 하고, 두카드의 위치를 매개 변수로 받아 혹시 같은 카드를 입력했을 때는 종료, 두 카드가 모두 현재 뒤집혀(숨겨져) 있을 경우 잠시 뒤집어서 보여주고 다를 경우 다시 뒤집고 종료 하는 함수.

public boolean checkcard(int x1,int y1,int x2,int y2) {

try {

if(x1==x2 &&y1 ==y2) {

System.***out***.println("에러 . 같은 카드를 고르셨습니다. \n카드를 두 개 선택해 주십시요");

return false;

}

if(card\_hide[x1][y1]&&card\_hide[x2][y2]) {

*cmd\_cls*();

card\_hide[x1][y1]=card\_hide[x2][y2] =false;

printcard();

if(card[x1][y1]==card[x2][y2]) {

match\_count +=2;

System.***out***.println("성공! 두카드는 일치 합니다.");

return true;

}

else{

card\_hide[x1][y1] =card\_hide[x2][y2]= true;

System.***out***.println("두 카드는 일치 하지 않습니다");

}

}

else {

System.***out***.println("이미 뒤집힌 카드 입니다.");

}

return false;

} catch (IndexOutOfBoundsException e) {

System.***out***.println("인덱스 에러 입니다. 정수 4개 모두 1~4의 수를 입력 해주세요 ");

}

return false;

}

}

gamestart(): 게임의 실행함수, 맨 처음 콘솔을 지운 뒤 카드를 섞고 나서 Tread.sleep을 이용하여 잠시 보여주고, match\_count가0에서 16이 될 때까지 숨겨진 상태로 카드를 출력해주고 두 카드를 입력 받고 checkcard실행, 을 반복한다.

public void gamestart() throws InterruptedException {

*cmd\_cls*();

System.***out***.println("게임 시작 ");

match\_count=0;

mixcard();

printcard();

System.***out***.println("카드는 3초간만 보여집니다");

Thread.*sleep*(3000);

hide\_card(true);

while(true) {

*cmd\_cls*();

printcard();

int x1,x2,y1,y2;

while(true) {

try {

System.***out***.print("첫 번 째 카드를 선택 하세요(ex.2 3) :");

x1= *scan*.nextInt()-1;

y1= *scan*.nextInt()-1;

break;

} catch (InputMismatchException e) {

System.***out***.println("에러 ! 에러 !");

System.***out***.println("정수 두 개를 띄어서 제대로 입력해 주세요.");

*scan*.nextLine();

}

}

while(true) {

try {

System.***out***.print("두 번 째 카드를 선택 하세요(ex.2 3) :");

x2= *scan*.nextInt()-1;

y2= *scan*.nextInt()-1;

break;

} catch (InputMismatchException e) {

System.***out***.println("에러 ! 에러 !");

System.***out***.println("정수 두 개를 띄어서 제대로 입력해 주세요.");

*scan*.nextLine();

}

}

boolean match= checkcard(x1,y1,x2,y2);

if(match) {

if(match\_count==16) {

*cmd\_cls*();

printcard();

System.***out***.println("게임 성공 ");

return;

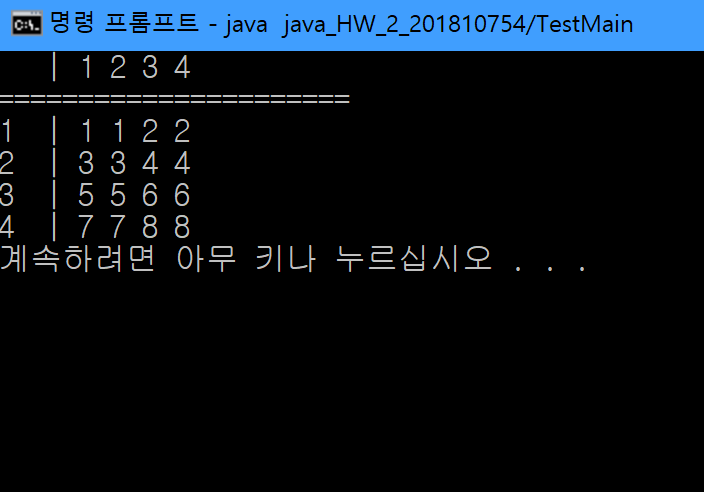
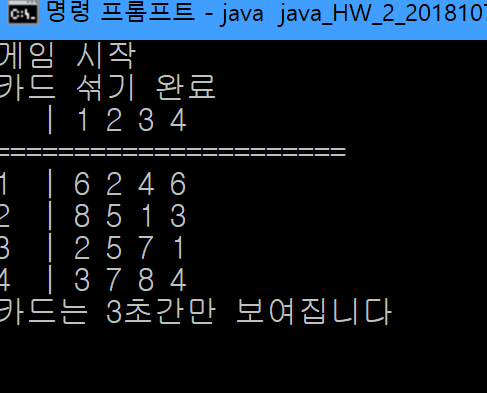
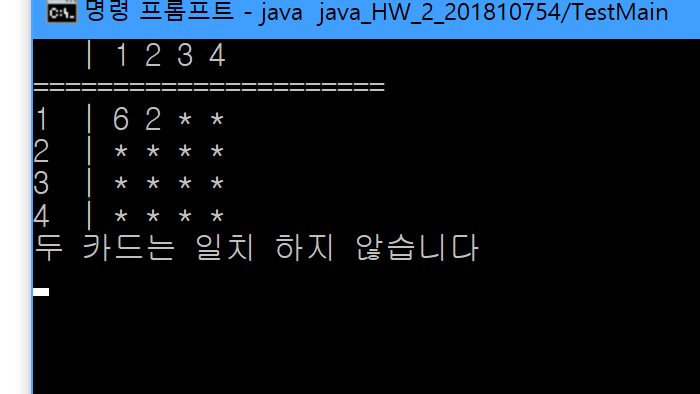
}

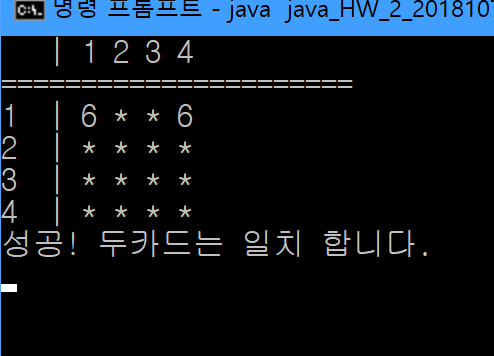
}

Thread.*sleep*(1000);

}

}

-실행 결과

검은색이(가) 표시된 사진

자동 생성된 설명

느낀점과 토의사항:

자바에서의 콘솔 명령어 실행. (cls, pause,…), Tread.sleep(), 프로그램이 종료되지 않고 계속 실행되기 위해서는, 정상적으로 작동되기 위해서는 예외 처리가 필수,