

Note: Yahtzee™는 Hasbro 가 제작 한 게임의 상표 이름입니다.

배열 Arrays, arrays, everywhere...

배열은 재미있는 프로그램을 작성할 수 있는 능력을 크게 향상시킵니다. 과제는 다양한 상황에서 배열을 용하여 인기있는 멀티 플레이어 주사위 게임을 구현합니다. 주사위를 위한 배열, 주사위를 다시 배열하기위한 배열, 플레이어 이름의 배열, 플레이어의 점수 배열, 전체 스코어 카드를 처리 할 수 있는 패널 배열이 있습니다. 프로그램이 끝나면 배열의 개념을 익힐 수 있습니다.

게임 목표

목표는 게임 Yahtzee™의 컴퓨터 버전을 만드는 것입니다. 이미 게임을 했을 수도 있지만 그렇지 않은 사람들도 배우기가 쉽습니다. 5 개의 주사위와 1 ~ 10 명의 선수가 있습니다. 게임의 라운드는 각 플레이어가 차례대로 진행합니다. 각각의 턴에 플레이어는 13 개의 카테고리 (Dice Categories)중 하나에 해당하는 구성으로 그들을 가져갈 수 있도록 5 개의 주사위를 굴립니다. 첫 번째 굴림 후에 플레이어는 주사위의 일부 또는 전부를 다시 굴릴 수 있습니다. 두 번째 굴림 후에, 플레이어는 다시 한 번 일부 또는 모든 주사위를 굴릴 수 있습니다. 그러나 세 번째 주사위 굴림이 끝날 때까지 플레이어는 스코어 카드의 13 개 카테고리 중 하나에 최종 주사위 구성을 지정해야 합니다. 주사위 구성이 해당 카테고리의 기준을 충족하면 플레이어는 해당 카테고리에 대해 적절한 점수를 받습니다. 그렇지 않은 경우 해당 카테고리의 점수는 0 입니다. 13 개의 카테고리가 있고 각 카테고리가 정확히 한 번 사용되므로 게임은 13 라운드로 구성됩니다. 13 라운드가 끝나면 모든 플레이어가 모든 카테고리의 점수를 받게 됩니다. 점수가 가장 높은 플레이어가 승자로 선언됩니다.

주사위 카테고리(Dice categories): 주사위 구성과 점수의 13 가지 카테고리는 다음과 같습니다.

1. **Ones.** 모든 주사위 구성이 이 카테고리에 유효합니다. 점수는 주사위에 표시된 모든 1 의 합계와 같으며 1 이 표시되지 않으면 0 입니다.
- 2-6. **Twos, Threes, Fours, Fives, Sixes.** (위와 동일하지만 다른 값에 대해). 모든 주사위 구성이 이 카테고리에 유효합니다. 점수는 주사위에 표시되는 2, 3, 4, 5, 6 등의 합계와 같습니다.
7. **Three of a Kind.** 적어도 3 개의 주사위는 동일한 값을 보여 주어야합니다. 점수는 주사위에 표시된 모든 값의 합계와 같습니다.
8. **Four of a Kind.** 적어도 4 개의 주사위는 동일한 값을 보여 주어야합니다. 점수는 주사위에 표시된 모든 값의 합계와 같습니다.
9. **Full House.** 주사위는 3 개가 같은 값을 가지며 2 개가 또 다른 같은 값을 가져야합니다. 점수는 25 점입니다.

10. **Small Straight.** 주사위는 2-3-4-5 와 같이 적어도 네 개의 연속적인 값을 가져야합니다. 점수는 30 점입니다.
11. **Large Straight.** 주사위는 1-2-3-4-5 와 같은 5 개의 연속적인 값을 가져야합니다. 점수는 40 점입니다.
12. **Yahtzee!** 모든 주사위는 같은 값을 가져야 합니다. 점수는 50 점입니다..
13. **Chance.** 모든 주사위 구성이 이 카테고리에 유효합니다. 점수는 주사위에 표시된 모든 값의 합계와 같습니다.

상단항목(Upper section)		
카테고리	설명	예시
1.Ones	1 이 나온 주사위 눈의 합	 점수 3
2.Twos	2 가 나온 주사위 눈의 합	 점수 6
3.Threes	3 이 나온 주사위 눈의 합	 점수 12
4.Fours	4 가 나온 주사위 눈의 합	 점수 8
5.Fives	5 가 나온 주사위 눈의 합	 점수 5
6.Sixes	6 이 나온 주사위 눈의 합	 점수 18
Upper Bonus	Upper Section 에서 합이 63 점이 넘을 경우 35 점을 추가한다. [5]	

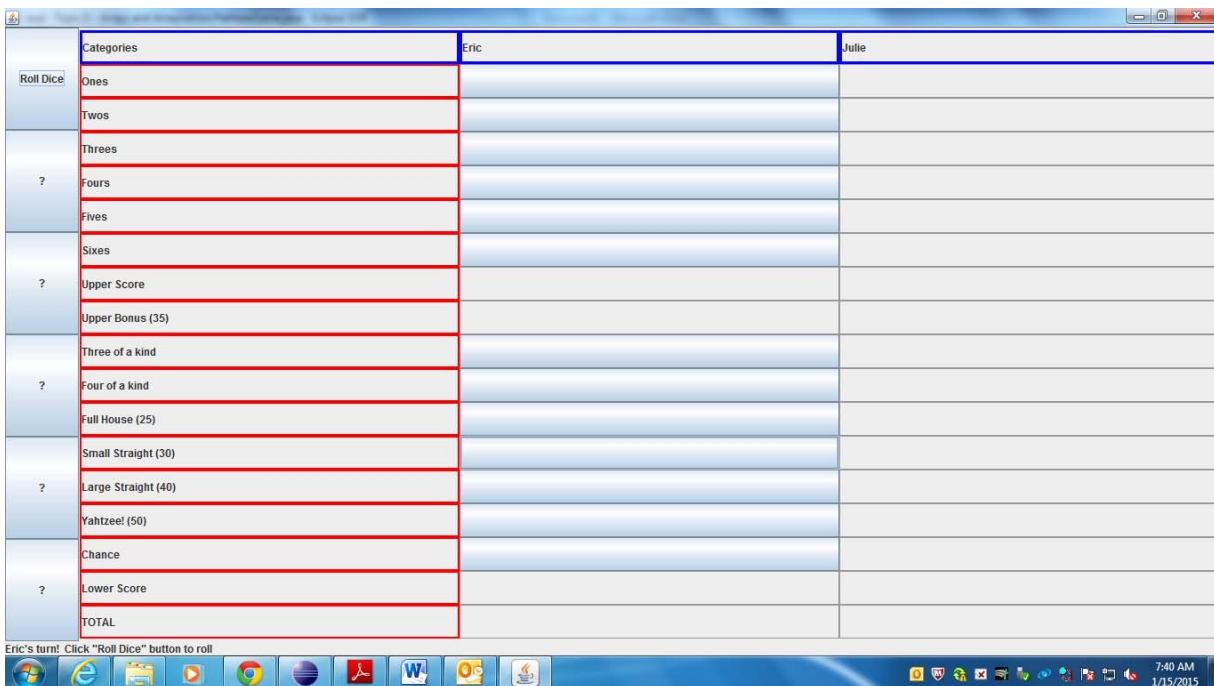
하단항목(Lower section)			
카테고리	설명	점수	예제
7.Three-Of-A-Kind	적어도 3 개의 주사위가 같아야 합니다.	모든 주사위의 합계	 점수 17
8.Four-Of-A-Kind	적어도 4 개의 주사위가 같아야 합니다.	모든 주사위의 합계	 점수 24
9.Full House	하나는 숫자 3 개, 다른 하나는 2 개	25	 점수 25

10.Small Straight	4 개의 순차 주사위 (1-2-3-4, 2-3-4-5 또는 3-4-5-6)	30	 점수 30
11.Large Straight	5 개의 순차 주사위 (1-2-3-4-5 또는 2-3-4-5-6)	40	 점수 40
12.Yahtzee	5 개의 주사위 모두 동일	50	 점수 50
13.Chance	모든 조합	모든 주사위의 합계	 점수 13

실행 예시

프로그램이 시작되면 시작 메시지가 표시되고 사용자에게 플레이어 수를 입력하라는 메시지가 표시됩니다. 그런 다음 사용자에게 한 번에 하나씩 플레이어 이름을 입력하라는 메시지를 표시합니다. 에릭 (Eric)과 줄리 (Julie)라는 두 명의 선수가 참가한다고 가정 해 보겠습니다. 팝업 대화 상자를 사용하여 Eric 과 Julie 라는 이름을 입력하면 그림 1 과 같이 그래픽 창에 시작 Yahtzee 스코어 카드와 주사위가 표시됩니다.

그림 1 ("Eric"및 "Julie"플레이어가 있는 새로운 2 인 게임 구성 후)



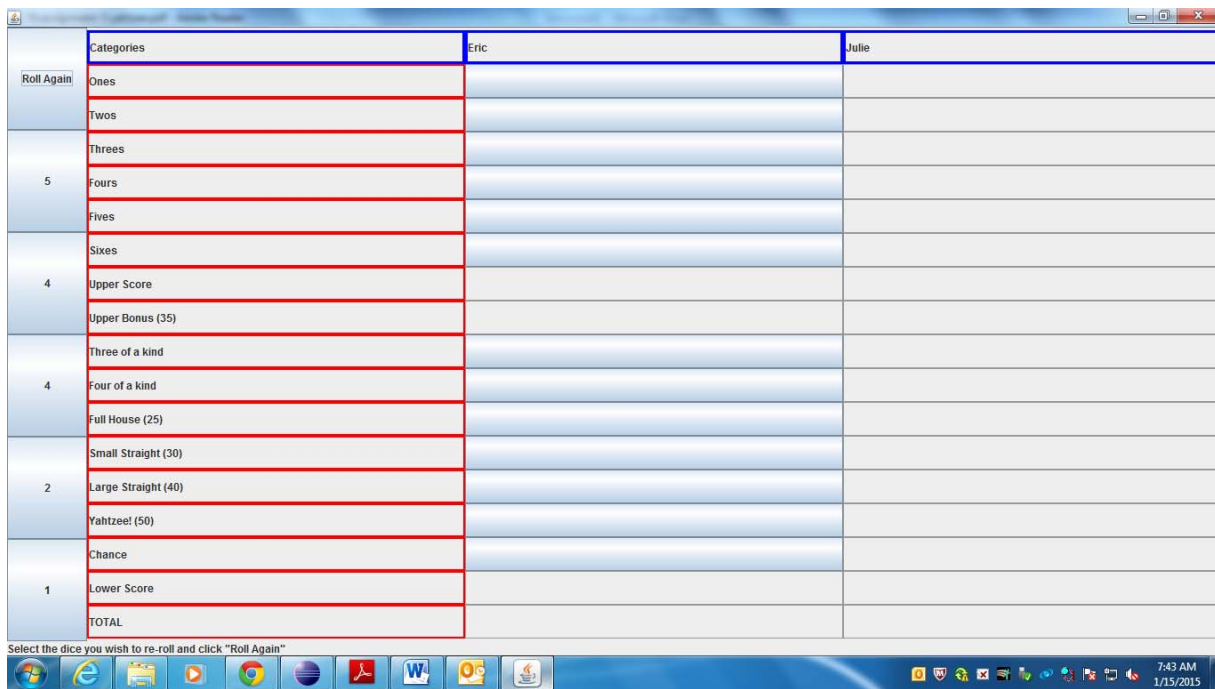
Yahtzee 스코어 카드

Yahtzee 스코어 카드를 보는 데 1-2 분 정도 소요됩니다. 게임을 구성하는 13 개의 카테고리는 두 부분으로 나뉩니다. 위쪽 섹션에는 Ones, Twos, Threes 등의 카테고리가 있습니다. 게임이 끝나면 이 카테고리의 값이 합산되어 Upper Score 라고 표시된 항목에 값이 생성됩니다. 또한, 상위 섹션에 대한 플레이어의 점수가 총 63 점 이상인 경우, 그 플레이어는 다음 라인 (Upper Bonus)에서 35 점 보너스를 받게 됩니다. 스코어 카드의 하단 섹션에 있는 점수도 합쳐져서 Lower Score 라고 표시된 항목을 생성합니다. 그런 다음 Upper Score, Upper Bonus (있는 경우) 및 Lower Score 를 합산하여 각 선수의 총점(Total)을 계산합니다.

샘플 게임하기

이제 그림 1 의 게임을 시작할 준비가 되었습니다. 에릭이 먼저이므로, 스코어 카드에서 그의 버튼을 사용할 수 있게 됩니다. 스코어 카드에는 또한 에릭의 차례가 표시됩니다. 주사위를 굴리기 위해 "주사위 굴리기(Roll Dice)"버튼을 클릭하십시오. Eric 이 주사위 버튼을 클릭하면 주사위가 무작위로 굴려서 그림 2 와 같은 화면이 나타납니다.

그림 2 (Eric 의 첫 번째 주사위 굴리기 이 후)



	Categories	Eric	Julie
Roll Again	Ones		
	Twos		
5	Threes		
	Fours		
	Fives		
4	Sixes		
	Upper Score		
	Upper Bonus (35)		
4	Three of a kind		
	Four of a kind		
	Full House (25)		
2	Small Straight (30)		
	Large Straight (40)		
	Yahtzee! (50)		
1	Chance		
	Lower Score		
	TOTAL		

언뜻 보기에, 이 구성은 2 개의 4 가 꽤 잘 보입니다! 그는 Yahtzee 에 대한 기회가 있다고 생각합니다! 두 번째 굴리기에서 Eric 은 1, 2 및 5 를 다시 굴리려고 합니다. 이 선택을 나타내기 위해 Eric 은 보유하고 싶은 주사위를 클릭해야 합니다. 그렇게 클릭한 주사위는 다시 굴릴 수 없고 클릭하지 않은 주사위는 다시 굴릴 준비가 됩니다.

그림 3 (Eric 은 보유하고 싶은 4 를 2 개 선택 했으며, 1,2,5 는 다시 굴릴 준비가 되었습니다.)

	Categories	Eric	Julie
Roll Again	Ones		
	Twos		
5	Threes		
	Fours		
	Fives		
4	Sixes		
	Upper Score		
	Upper Bonus (35)		
4	Three of a kind		
	Four of a kind		
	Full House (25)		
2	Small Straight (30)		
	Large Straight (40)		
	Yahtzee! (50)		
1	Chance		
	Lower Score		
	TOTAL		

주사위를 다시 굴리기 위해서는 플레이어가 다시 굴림 (Roll Again)버튼을 클릭 해야합니다.

항상 자신감이 넘친 Eric 은 Yahtzee 에 가기로 결정하고 1, 2 및 5 를 다시 굴렸습니다. Eric 은 두 번째 굴리기에서 4 를 더 이상 얻지 못해서, 마지막 세 번째 굴리기에 똑같이 1,2,5 주사위를 다시 굴렸습니다. 운 좋게도 더 많은 4 를 2 개 더 얻었습니다. Eric 은 그림 4 와 같이 4 개의 4 와 1 개의 2 로 끝납니다.

그림 4 (Eric 의 첫 번째 라운드의 마지막 굴리기 이후, Four of a Kind 선택 예정)

	Categories	Eric	Julie
Roll Again	Ones		
	Twos		
4	Threes		
	Fours		
	Fives		
4	Sixes		
	Upper Score		
	Upper Bonus (35)		
4	Three of a kind		
	Four of a kind		
	Full House (25)		
4	Small Straight (30)		
	Large Straight (40)		
	Yahtzee! (50)		
2	Chance		
	Lower Score		
	TOTAL		

Eric 은 Yahtzee 를 얻지 못했지만 합리적으로 괜찮은 Four of a Kind 를 얻었습니다. 카테고리를 선택하라는 요청을 받으면 Eric 은 Four of a Kind 를 클릭하고 18 점을 점수에 기록합니다.

이제는 줄리의 차례 . 줄리는 에릭보다 더 나은 게임 운을 가졌습니다. 그림 5 는 3 번 굴리기가 끝날 때의 주사위의 구성을 보여줍니다 : 5 개의 주사위 3 개와 2 개의 4 개의 주사위. 줄리는 매우 기쁘게 생각하며 Full House 카테고리를 사용할 계획입니다. Full House 는 25 점의 가치가 있으며 초반 리드를 가져옵니다.

그림 5 (Julie 의 첫 번째 라운드 마지막 굴리기 이후, 카테고리를 선택할 때)

	Categories	Eric	Julie
Roll Again	Ones		
	Twos		
4	Threes		
	Fours		
	Fives		
5	Sixes		
	Upper Score		
	Upper Bonus (35)		
5	Three of a kind		
	Four of a kind	18	
	Full House (25)		
4	Small Straight (30)		
	Large Straight (40)		
	Yahtzee! (50)		
5	Chance		
	Lower Score		
	TOTAL		

이제 라운드 # 2 입니다. 에릭은 7 점 뒤진 상태입니다. 그의 첫 번째 굴리기는 4, 1, 6, 2, 4 입니다. 에릭은 4 또는 6 을 3 으로 바꾸면 Small Straight 를 가질 수 있다고 봅니다. 그는 두 번 더 주사위를 굴립니다. 마지막 주사위 구성이 1, 1, 5, 2, 4 인 Eric 은 그림 6 과 같이 Ones 를 선택합니다.

그림 6 (Eric 은 Ones 선택으로 2 점 획득)

	Categories	Eric	Julie
Roll Dice	Ones	2	
	Twos		
?	Threes		
	Fours		
	Fives		
?	Sixes		
	Upper Score		
	Upper Bonus (35)		
?	Three of a kind		
	Four of a kind	18	
	Full House (25)		25
?	Small Straight (30)		
	Large Straight (40)		
	Yahtzee! (50)		
?	Chance		
	Lower Score		
	TOTAL		

Julie's turn! Click "Roll Dice" button to roll

줄리는 첫 번째 굴리기에서 Large Straight 를 얻습니다. 당연히, 줄리는 주사위를 다시 굴리지 않고 Large Straight 를 선택합니다.

그림 7 (3 라운드 시작 시점의 스코어 카드 상태)

	Categories	Eric	Julie
Roll Dice	Ones	2	
	Twos		
?	Threes		
	Fours		
	Fives		
?	Sixes		
	Upper Score		
	Upper Bonus (35)		
?	Three of a kind		
	Four of a kind	18	
	Full House (25)		25
?	Small Straight (30)		
	Large Straight (40)		40
	Yahtzee! (50)		
?	Chance		
	Lower Score		
	TOTAL		

Eric's turn! Click "Roll Dice" button to roll

게임은 비슷한 방식으로 계속됩니다. 각 라운드마다 플레이어는 다음을 수행해야 합니다.

1. 주사위 굴리기 (Roll Dice) 버튼을 클릭하여 주사위 다섯 개 모두의 초기 굴림을 설정하십시오.
2. 보관할 주사위들을 클릭 선택한 다음 선택하지 않은 주사위를 다시 굴립니다.

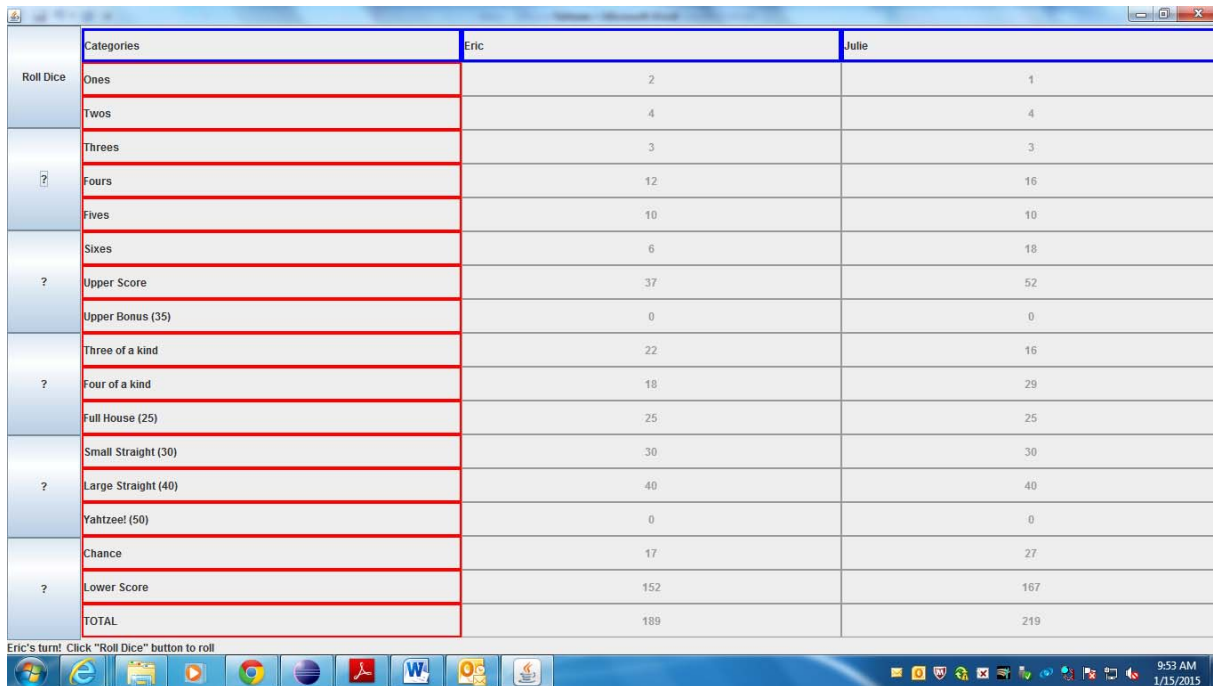
3. 2 단계를 반복하여 최종 주사위 구성을 생성하십시오.

4. 카테고리를 클릭하여 해당 상자에 점수를 저장하십시오.

플레이어는 적절한 카테고리가 없기 때문에 주사위의 구성과 일치하지 않는 카테고리를 선택해야 할 때가 있습니다. 이 경우 플레이어는 선택한 상자에서 단순히 0 점을 얻습니다.

13 개 카테고리를 선택할 때까지 게임 끝까지 진행합니다.

그림 9 (게임이 끝날 때)



	Categories	Eric	Julie
Roll Dice	Ones	2	1
	Twos	4	4
?	Threes	3	3
	Fours	12	16
	Fives	10	10
?	Sixes	6	18
	Upper Score	37	52
	Upper Bonus (35)	0	0
?	Three of a kind	22	16
	Four of a kind	18	29
	Full House (25)	25	25
?	Small Straight (30)	30	30
	Large Straight (40)	40	40
	Yahtzee! (50)	0	0
?	Chance	17	27
	Lower Score	152	167
	TOTAL	189	219

게임이 끝났으므로 Upper Score 와 Lower Score 가 계산되고 적용 가능한 경우 Upper Bonus 가 주어집니다. 줄리도 에릭도 두 보너스를 모두 받지 못했기 때문에 Upper Bonus 를 받지 못했습니다. 줄리는 219 점으로 189 점 에릭을 이겼습니다.

Java 언어용 프로그램 가이드라인

Dice 클래스– 각 주사위에 대해 굴림 된 값을 저장하는 데 사용되는 간단한 클래스

- 변수: int roll
- 함수
 - public void rollDie() – 주사위의 값을 1-6 의 랜덤 정수로 변경합니다.
 - public int getRoll() – 현재 주사위 값을 반환합니다.

Configuration 클래스– 주사위 구성에 대해 점수를 매기고 테스트하는 정적 메서드 모음

- 변수: static String [] configs 각 주사위 카테고리 이름들의 문자열 정적 배열 (실행 예시 참조)
- 함수
 - public static String[] getConfigs() – configs 배열을 반환합니다.
 - public static int score(int row, Dice[] d) – row 에 따라 주사위 점수를 매깁니다. 예를 들어, row 가 0 이면 "Ones"가 채점되어야 함을 의미합니다. row 가 2 이면, "Threes"가 득점되어야 함을 의미합니다. row 가 득점 (scored)하지 않아야 하는 버튼 (즉, UpperScore, UpperBonus, LowerScore, Total 등)을 나타내는 경우 -1 을 반환합니다.
 - private static int scoreUpper(Dice[] d, int num) – Upper Section 구성 (Ones, Twos, Threes, ...)에 대해 주사위 점수를 매 깁니다. 예를 들어, num 이 1 이면 "Ones"구성의 주사위 점수를 반환합니다.
 - private static int scoreThreeOfAKind(Dice[] d) – 유효한 Three of a Kind 주사위를 나타내는 경우 모든 주사위의 합을 반환하고, 그렇지 않으면 0 을 반환합니다.
 - private static int scoreFourOfAKind(Dice[] d) – 유효한 Four of a Kind 주사위를 나타내는 경우 모든 주사위의 합을 반환하고, 그렇지 않으면 0 을 반환합니다.
 - private static int scoreFullHouse(Dice[] d) – 유효한 Full House 주사위를 나타내는 경우 25 를 반환하고, 그렇지 않으면 0 을 반환합니다.
 - private static int scoreYahtzee(Dice[] d) – 유효한 Yahtzee 주사위를 나타내는 경우 50 을 반환하고, 그렇지 않으면 0 을 반환합니다.
 - private static int scoreSmallStraight(Dice[] d) – 유효한 Small Straight 주사위를 나타내는 경우 30 을 반환하고, 그렇지 않으면 0 을 반환합니다.
 - private static int scoreLargeStraight(Dice[] d) – 유효한 Large Straight 주사위를 나타내는 경우 40 을 반환하고, 그렇지 않으면 0 을 반환합니다.
 - private static int sumDie(Dice[] d) – 모든 주사위의 합을 반환하는 helper 메소드

- `private static Dice[] sortDice(Dice[] d)` – 주사위 값을 오름 차순 정렬해서 반환
- `private static boolean[] existValueDice(Dice[] d)` – 주사위 값 -1 을 인덱스로 True 로 설정한 길이가 6 인 배열 반환, 즉 주사위 값이 1,2,5,3,6 이면 `[True, True, True, False, True, True]` 반환

Player 클래스 – 플레이어의 점수와 사용 된 카테고리 점수를 관리하는 클래스

- 변수
 - `private String name`
 - `private static final int UPPER = 6;` //the number of configurations in the upper portion
 - `private static final int LOWER = 7;` //the number of configurations in the lower portion
 - `private boolean[] used` //size of used is LOWER + UPPER
 - `private int[] scores` //size of scores is LOWER + UPPER
 - `private int upperScore`
 - `private int lowerScore`
 - `private int totalScore`
- 함수
 - 생성자– 플레이어 명수를 인자로 받음
 - `public void setScore(int score, int index)` – index 인덱스의 scores 배열의 값을 score 로 설정
 - `public int getUpperScore()` – UpperScore 값 반환
 - `public int getLowerScore()` – LowerScore 값 반환
 - `public boolean[] getUsed()` – used 배열 반환
 - `public int getTotalScore()` – Total 값 반환
 - `public String toString()` – 플레이어의 이름 반환
 - `public boolean allUpperUsed()` – 모든 Upper Section 이 사용되었으면 True, 아니면 False 반환
 - `public boolean allLowerUsed()` – 모든 Lower Section 이 사용되었으면 True, 아니면 False 반환
 - `public void setAtUsed(int index)` – index 인덱스의 used 배열의 값을 True 로 설정

YahtzeeBoard 클래스– the class that sets up the main interface and contains all ActionListeners

- 변수
 - `private int numPlayers` - 입력받은 플레이어 수
 - `private Player[] players` - 다수 플레이어 배열
 - `private Dice[] dice` - 5 개 주사위 객체 배열
 - `private DicePanel dicePanel`
 - `private PlayerPanel playerPanel`
 - `private JButton[] diceButtons` - 5 개 주사위 버튼 배열
 - `private JButton[][] fields` - 다수 플레이어들의 카테고리 버튼 2 차원 배열

- private CategoryListener cl - 카테고리 버튼 이벤트 처리 함수
- private DiceListener dl - 주사위 버튼 이벤트 처리 함수
- private RollDiceListener rl - RollDice 버튼 이벤트 처리 함수
- private JButton rollDice - RollDice 버튼
- private JLabel bottomLabel - 윈도우 하단 라벨
- private static int player //0 으로 초기화, 어떤 플레이어 차례인지 관리
- private static int roll //0 으로 초기화, 3 번의 굴리기 차례를 관리
- private static int round //0 으로 초기화, 13 번 라운드 차례를 관리
- private final int UPPERTOTAL = 6 //fields 2 차원 배열의 upper total 인덱스
- private final int UPPERBONUS = 7 // fields 2 차원 배열의 upper bonus 인덱스
- private final int LOWERTOTAL = 15 // fields 2 차원 배열의 lower total 인덱스
- private final int TOTAL = 16 // fields 2 차원 배열의 total 인덱스

• 함수

- 생성자- 인수로 플레이어의 수를 받아 들인다; 변수 초기화, 초기 인터페이스 생성
- public Player[] getPlayer() – returns players array
- public Dice[] getDice() – returns dice array
- public Button[][] getFieldsButtons() – returns fields array
- public void playerTurn(int index) – 이 기능은 플레이어의 차례가 시작될 때 일어날 필요가있는 모든 것을 처리합니다. index 는 플레이어의 인덱스를 나타냅니다 (첫 번째 플레이어 인덱스 = 0 등). 하단 라벨을 텍스트 " _____의 순서대로 설정합니다! 굴리기 "주사위 굴리기"버튼을 클릭하십시오. ". 주사위 주사위 버튼의 텍스트를 "주사위 굴림"텍스트로 설정합니다. 새로운 주사위 인 경우 라운드 변수를 증가시킵니다. 13 라운드가 모두 끝나면 finalScore ()를 호출합니다. 현재 플레이어의 버튼에서 모두 사용하도록 설정하고 다른 플레이어에서는 사용하지 않도록 설정합니다. 각 주사위 버튼의 텍스트를 "?"로 설정하여 모든 주사위 버튼을 활성화하고 주사위 배열의 새 주사위를 초기화합니다. 주사위 버튼을 활성화로 설정하고 롤 변수를 다시 0 으로 설정합니다.
- public int getCatIndex(int i) – 인덱스 i 의 카테고리 번호를 반환합니다 (전달 된 경우 카테고리 버튼 배열에서 행 인덱스라고 가정 할 수 있습니다. 예를 들어, i 가 0~5 이면 단순히 i 의 값을 반환합니다. i 가 8~14 사이 이면 , 2 보다 작은 i 를 돌려 주어, 그렇지 않은 경우는 -1 을 돌려 준다.
- public boolean validCatIndex(int i) – i 가 0 ~ 5 (포함) 또는 8 ~ 14 (포함)이면 true 를, 그렇지 않으면 false 를 반환합니다.
- public void finalScore() – 각 플레이어의 점수를 계산합니다. 점수 (UpperBonus 포함)를 나타내는 모든 필드의 텍스트를 설정합니다. 승자를 식별하고 하단 라벨을 "축하합니다. _____, 총 점수가 ____ 인 승자입니다."라고 말합니다.

- Java 언어용 내부 클래스
 - `private class DicePanel extends JPanel`
 - 생성자: 인터페이스의 가장 왼쪽에있는 패널처럼 보이는 패널을 만듭니다. 하나는 "RollDice"버튼이고, 아래 주사위 버튼은 시작하려면 "?"표시가 있습니다. "?"가있는 모든 버튼에는 `DiceListener` 이벤트 처리 함수가 바인드 되어야 합니다.
 - `private class PlayerPanel extends JPanel`
 - 변수:
 - `private final int ROWS = 18`
 - `private JLabel[] titles` //initialized to length ROWS
 - 생성자: 모든 플레이어의 패널 및 카테고리 레이블 (인터페이스 참조)에 대한 초기 인터페이스를 설정합니다. 각 플레이어에게 이름을 물어볼 수도 있습니다.
 - `private class CategoryListener implements ActionListener` – 카테고리 버튼 중 하나를 클릭하면 이 함수가 수행합니다. 이것은 플레이어의 차례가 끝났을 때 일어날 모든 것을 나타냅니다. 선택한 카테고리의 점수를 매기십시오 (2 차원 배열의 인덱스를 기반으로 함). 점수가 -1 이면 (즉, 사용자가 선택해서는 안되는 버튼을 클릭) 아무 것도 하지 않습니다. 그렇지 않으면 선택한 구성 요소의 텍스트를 해당 구성의 점수로 설정하십시오. 클릭 한 버튼을 비활성화하십시오. 현재 플레이어에 대해 선택한 구성의 점수를 설정합니다. 플레이어 변수를 증가시킵니다. 다음 플레이어의 차례가되도록 설정하십시오.
 - `private class RollDiceListener implements ActionListener` – 이것은 "Roll Dice"버튼을 클릭했을 때 수행하는 이벤트 함수입니다. roll 증가, 이미 선택되지 않은 모든 주사위를 굴립니다. 그것이 플레이어의 마지막 굴림이라면, 모든 주사위를 비활성화로 설정하고 roll 을 0 으로 설정하고 하단 라벨의 텍스트를 "카테고리 선택"으로 설정하십시오. 마지막 roll 이 아닌 경우 하단 라벨을 "보관할 주사위 선택 후 "Roll Again" 클릭" 으로 설정하십시오. "Roll Dice 버튼의 텍스트를"Roll Again"으로 설정하십시오.
 - `private class DiceListener implements ActionListener` – 주사위 버튼을 클릭할 때 수행되는 이벤트 처리 함수로써 주사위를 비활성화 시켜서 다시 굴릴 수 없도록 함.

Hints

- Three of a Kind, Four of a Kind, Yahtzee, Full House 사이의 유효성 결정 간에는 큰 차이가 없습니다
- Small Straight, Large Straight 사이의 유효성을 결정하는 데는 큰 차이가 없습니다.
- 모든 주사위 구성은 Ones, Twos, Threes, Fours, Fives, Sixes 및 Chance 에 유효합니다.
- 요구 사항을 충족시키지 못하는 카테고리의 주사위 구성은 점수가 0 입니다.
- 플레이어가 다음에 해야 할 것을 알려주기 위해서 하단 라벨을 변경해야 합니다 (Roll Dice 해야 할 때, Roll Again 할 때 주사위를 선택하는 시기, 카테고리를 선택 할 시기, 승자가 되는 등).

- 플레이어가 주사위 카테고리를 선택하면 오류가 있는지 확인하십시오. 사용자는 이전 카테고리를 다시 사용할 수 없습니다.
- 매 라운드마다 플레이어는 주사위를 세 번 굴립니다. 플레이어가 후속 굴리기에서 아무 것도 변경하고 싶지 않다면, 플레이어는 주사위를 모두 선택하여 비활성화하고 Roll Again 버튼을 클릭해야 합니다.
- 게임이 끝나면 Upper Bonus(Upper Score 가 63 점 이상인 경우 35 점), Upper Score, Lower Score 의 최종 합계를 계산하고 Total 에 할당하는 것을 잊지 마십시오.
- 객체지향언어의 Encapsulation 을 위해서 모든 클래스 변수는 private 으로 설정하고 public 메소드로 접근합니다.

확장

1 차 목표는 Yahtzee 게임을 완벽하게 구현하는 것이며, 그 후 확장을 위한 아이디어를 찾을 수 있습니다.

다음은 최소한 몇 가지 아이디어입니다.

- 높은 점수 파일 저장 기능을 추가하십시오. 상위 10 개의 최고 점수와 이름을 파일에 저장하고 게임 실행 사이에 관리 되도록 합니다. 게임 시작 전에 최고 상위 점수 파일을 읽어서 프린트하고 게임 종료 전에 상위 10 위 안에 드는 기록이 있으면 파일을 업데이트합니다.
- 게임에서 Yahtzees 보너스 점수를 추가하십시오. Yahtzee 카테고리에 0 을 입력하지 않는 한, 게임에서 추가 Yahtzee 를 얻으면 100 점의 보너스 점수를 얻을 수 있습니다.

항상 그렇듯이 모든 기본 요구 사항을 완료하고 철저하게 테스트 한 후에 만 확장 기능을 수행해야 합니다. 확장 버전을 만들려면 기본 기능을 쉽게 확인할 수 있도록 기본 버전과 확장 버전을 모두 준비하십시오. 기본 기능을 방해하지 않는 작은 확장 기능은 한 버전에 포함시키는 것이 좋습니다.