

A Dice Rolling game

In this challenge, I want you to start working on a dice rolling game, similar to an existing boxed game called Yahtzee, if you've ever played it.



The objective of the game is to roll five dice, to get certain combinations of numbers, each scored differently.

The player gets multiple tries to roll the dice, but after the first roll, players can select which dice they want to roll again.

Implementing the game

In this challenge, use the **ints** method on Random, to generate a finite stream, to create a list of 5 random Integers, all between 1 and 6.

These will be the dice for your first roll.

Use the Scanner with System.in to have the user enter one of the following entries:

- The player should press Enter to keep all the dice. This will end the play in this first part of the challenge.
- The player should enter ALL, to re throw all five dice.
- The player should enter numbers, representing dice face values, for those dice to be rolled again.

Continue to loop, and throw the dice selected by the user, until the user presses enter.

Sample Output

Sample output is shown on this slide.

The system will roll the five dice, getting 5 random numbers from 1 to 6, and present these numbers to the player.

The user will be prompted with the three options they have.

```
You're dice are: [2, 3, 3, 3, 4]
Press Enter to Score.
Type "ALL" to re-roll all the dice.
List numbers (separated by spaces) to re-roll selected dice.
--> ALL
You're dice are: [1, 4, 4, 5, 5]
Press Enter to Score.
Type "ALL" to re-roll all the dice.
List numbers (separated by spaces) to re-roll selected dice.
--> 4 5
Keeping [1, 4, 5]
You're dice are: [1, 4, 5, 4, 5]
Press Enter to Score.
Type "ALL" to re-roll all the dice.
List numbers (separated by spaces) to re-roll selected dice.
-->
Game over. Real game would score and continue.
```

Sample Output

In this example, the user first enters ALL at the prompt, and all the dice are rolled again.

At the second prompt, the user enters 4 and 5 separated by a space, and only the first dice with a four, and the first dice with a 5 are rerolled.

In the last case, the user presses enter, and the play ends

```
You're dice are: [2, 3, 3, 3, 4]
Press Enter to Score.
Type "ALL" to re-roll all the dice.
List numbers (separated by spaces) to re-roll selected dice.
--> ALL
You're dice are: [1, 4, 4, 5, 5]
Press Enter to Score.
Type "ALL" to re-roll all the dice.
List numbers (separated by spaces) to re-roll selected dice.
--> 4 5
Keeping [1, 4, 5]
You're dice are: [1, 4, 5, 4, 5]
Press Enter to Score.
Type "ALL" to re-roll all the dice.
List numbers (separated by spaces) to re-roll selected dice.
-->
Game over. Real game would score and continue.
```

Bonus

If you want an extra challenge, use the GameConsole code, created in an earlier section of the course, to implement a DiceGame and DicePlayer.

Implement Scoring as shown on this slide.

The user can pick which score card item to apply their dice to, but the dice must meet certain conditions, otherwise a score of zero will be returned.

ScoreCard Item	Score	Conditions		ScoreCard Item	Score	Conditions
Aces	1 * number of 1's	none		Three of Kind	Sum of Dice	Must of three of one number, or Score = 0
Twos	2 * number of 2's	none		Four of Kind	Sum of Dice	Must of three of one number, or Score = 0
Threes	3 * number of 3's	none		Five of Kind	50 pts	Must of three of one number, or Score = 0
Fours	4 * number of 4's	none		Small Straight	30 pts	Must have 5 contiguous number, or Score = 0
Fives	5 * number of 5's	none		Large Straight	40 pts	Must have 5 contiguous numbers, or Score = 0
Sixes	6 * number of 6's	none		Full House	25 pts	Must have 3 of one value, and 2 of another, or Score = 0.

Bonus

ScoreCard Item	Score	Conditions		ScoreCard Item	Score	Conditions
Aces	1 * number of 1's	none		Three of Kind	Sum of Dice	Must of three of one number, or Score = 0
Twos	2 * number of 2's	none		Four of Kind	Sum of Dice	Must of three of one number, or Score = 0
Threes	3 * number of 3's	none		Five of Kind	50 pts	Must of three of one number, or Score = 0
Fours	4 * number of 4's	none		Small Straight	30 pts	Must have 5 contiguous number, or Score = 0
Fives	5 * number of 5's	none		Large Straight	40 pts	Must have 5 contiguous numbers, or Score = 0
Sixes	6 * number of 6's	none		Full House	25 pts	Must have 3 of one value, and 2 of another, or Score = 0.

I've included this code, implemented as an enum (`dev.lpa.chance.ScoredItem`), in the code in the resources section of this video, if you want to use that.

Each `DicePlayer` should have a `ScoreCard` with each of these items, initialized to an unscored value, when the player is created.

After rolling the dice, user should be prompted to pick from the remaining list of unscored items.

Bonus

ScoreCard Item	Score	Conditions		ScoreCard Item	Score	Conditions
Aces	1 * number of 1's	none		Three of Kind	Sum of Dice	Must of three of one number, or Score = 0
Twos	2 * number of 2's	none		Four of Kind	Sum of Dice	Must of three of one number, or Score = 0
Threes	3 * number of 3's	none		Five of Kind	50 pts	Must of three of one number, or Score = 0
Fours	4 * number of 4's	none		Small Straight	30 pts	Must have 5 contiguous number, or Score = 0
Fives	5 * number of 5's	none		Large Straight	40 pts	Must have 5 contiguous numbers, or Score = 0
Sixes	6 * number of 6's	none		Full House	25 pts	Must have 3 of one value, and 2 of another, or Score = 0.

The game ends when the user quits or completes the score card.