Jaryd Meek

CSCI 4448: OOAD

Montgomery

Project 1 - Part 2

Java Version: *openjdk version "1.8.0\_312"*

**Card Dealer:**

*Code:*

import java.util.\*;

public class pt2cards {

public static void main(String[] args) {

//Scanner to read stdin

Scanner scanner = new Scanner(System.in);

//Infinite loop to keep running until a 0 gets entered

while (true) {

int numToShuffle = -1; //Store the number input

//Infinite loop to keep running until a valid input gets entered

while (true) {

System.out.println("Please Enter Number Of Cards To Deal");

//Logic from this Try catch error handling came from here

// https://stackoverflow.com/questions/23163764/error-handling-inputmismatchexception-in-java

try {

numToShuffle = scanner.nextInt();

} catch(InputMismatchException ignore){

scanner.next();

}

//Check if number is valid, or if it's time to exit

if (numToShuffle < 55 && numToShuffle > 0) {

break;

} else if (numToShuffle == 0) {

//Exit

scanner.close();

return;

}

//Loop and print error.

System.out.println("Invalid entry, please enter a valid number");

}

//Create a new deck

Deck test = new Deck();

//generate the hand

test.run(numToShuffle);

}

}

}

class Deck {

//Variables for Storage

ArrayList<String> fullDeck = new ArrayList<String>(); //Entire Deck

//Default Initializer

//Just create full deck

public Deck() {

initialize();

}

//Run

//Generate Hand with specified number of cards

//Takes number of cards to deal as parameter

//Prints hand to terminal

public void run(int numCards) {

Collections.shuffle(fullDeck, new Random()); //Shuffles the Full Deck using a random seed from the random number generator

System.out.println(fullDeck.subList(0, numCards)); //Selects the first x options in full deck after shuffling

}

//Initialize

//Initialize the Full Deck (54 Cards)

//No Parameters

//No Output

public void initialize() {

String suits[] = {"♥️","♦️","♠️","♣️"};

String rank[] = {"A", "2", "3", "4", "5", "6", "7", "8", "9", "10", "J", "Q", "K"};

for(String x: suits) {

for(String y: rank) {

fullDeck.add(y+x);

}

}

fullDeck.add("Joker1");

fullDeck.add("Joker2");

}

}

*Output:*

Please Enter Number Of Cards To Deal

-3

Invalid entry, please enter a valid number

Please Enter Number Of Cards To Deal

1

[A♦️]

Please Enter Number Of Cards To Deal

5

[4♠️, 9♠️, Q♠️, 6♥️, A♥️]

Please Enter Number Of Cards To Deal

10

[5♠️, A♠️, Q♣️, Q♦️, 2♠️, 4♦️, Q♥️, 4♠️, 5♥️, K♥️]

Please Enter Number Of Cards To Deal

20

[4♦️, 4♠️, 2♦️, 10♠️, J♥️, Q♥️, 3♥️, A♣️, J♦️, J♠️, 7♦️, Joker1, Q♠️, A♠️, 7♥️, 4♣️, 6♣️, Q♦️, 5♥️, 6♥️]

Please Enter Number Of Cards To Deal

54

[7♥️, 3♦️, 3♠️, K♥️, 9♦️, 7♠️, 10♥️, 3♥️, J♣️, 8♣️, K♣️, Joker1, 5♣️, 6♠️, 5♥️, 2♦️, 10♦️, Q♦️, 9♣️, A♦️, 6♦️, 8♥️, 6♥️, K♦️, 8♠️, J♦️, 9♠️, 5♦️, A♥️, Q♠️, Q♣️, J♠️, 2♥️, J♥️, 10♠️, 2♣️, Joker2, 10♣️, 8♦️, Q♥️, A♣️, 7♣️, 9♥️, 4♠️, K♠️, 5♠️, 4♦️, 7♦️, 3♣️, 4♥️, A♠️, 2♠️, 6♣️, 4♣️]

Please Enter Number Of Cards To Deal

55

Invalid entry, please enter a valid number

Please Enter Number Of Cards To Deal

0

Wordle game on following page

**Wordle Game:**

*Code:*

import java.util.\*;

public class pt2wordle {

public static void main(String[] args) {

//Create game then run game

Wordle game = new Wordle();

game.run();

}

}

class Wordle {

//Member Variables

private String[] wordOptions = {"ROBOT", "POINT", "FAVOR", "DRINK", "QUERY", "SLUMP", "SIEGE", "BOOST", "SOLAR", "PROXY", "PAPER", "MAJOR"};

String selected = "";

//Setup Game

public Wordle() {

initialize();

}

//Run

//Create loop that prompts user for word then finds out if that word is correct.

//No parameter

//Prints prompts and results from terminal

public void run() {

//Scanner to read input from stdin

Scanner scanner = new Scanner(System.in);

//loop to run until user quits game or gets correct answer

while(true) {

//Variables

String userInput = "";

String output = "";

//Prompt user and read input

System.out.println("Enter a 5 letter word:");

userInput = scanner.nextLine();

//If length of input is zero, quit game

if (userInput.length() == 0) {

break;

//If length of input is 5, check word and do logic for determining how far off word is

} else if (userInput.length() == 5) {

//uppercase user input

userInput = userInput.toUpperCase();

//Checks for right word entered

if (userInput.equals(selected)) {

System.out.println("Congrats! You solved it!");

return;

}

//Wrong word entered. Figure out how wrong it is.

for (int x = 0; x < 5; x++) {

if (userInput.charAt(x) == selected.charAt(x)) {

//Totally Right letter

output += " \"" + userInput.charAt(x) + "\" is a match in the correct location\n";

} else if (selected.contains( String.valueOf(userInput.charAt(x)))) {

//Right letter wrong place

output += " \"" + userInput.charAt(x) + "\" is in the word, but in a different location\n";

} else {

//Wrong letter

output += " \"" + userInput.charAt(x) + "\" is not in the word\n";

}

}

//output how wront the letter is

System.out.println(output);

} else {

//invalid entry, prompt user with error

System.out.println("Invalid entry. Please try again.");

}

}

}

//Initialize

//Pick a random word

//No Parameters

//No Output

public void initialize() {

int random = (int) (Math.random() \* 11.0);

selected = wordOptions[random];

}

}

*Output:*

Enter a 5 letter word:

aeiouy

Invalid entry. Please try again.

Enter a 5 letter word:

aeiou

"A" is in the word, but in a different location

"E" is not in the word

"I" is not in the word

"O" is a match in the correct location

"U" is not in the word

Enter a 5 letter word:

favor

Congrats! You solved it!