// ------------------------------------------

// Author: Lauren Escobedo

// Assignment: Chapter 7 Problem 7.17

// Date: 02/15/2023

// Language: Java

// File Name: Exercise7\_17.java

// Description: Exercise 7.17

// - Dice Rolling

// ------------------------------------------

import java.security.SecureRandom;

import java.util.Scanner;

public class Exercise7\_17 {

static int roll() {

// Roll the dice and return results

SecureRandom rand = new SecureRandom();

int die1 = rand.nextInt(6) + 1;

int die2 = rand.nextInt(6) + 1;

return (die1 + die2) - 2;

}

public static void main(String[] args) throws Exception {

// Declare/initialize variables

int iterations;

Scanner input = new Scanner(System.in);

// Declare and initialize results array

int[] results = new int[11];

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

results[i] = 0;

}

// User input for iteration limit

System.out.print("\n\nHow many times do you want to roll the dice? ");

iterations = input.nextInt();

System.out.println();

// "Roll" for "iterations" iterations

// Increment the index that was rolled each time

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

results[roll()] ++;

}

// Format output

System.out.print("Result ");

for (int i = 0; i < 11; i++) {

System.out.printf("| %6d ", i+2);

}

String line = "\n----------" +

"----------" +

"----------" +

"----------" +

"----------" +

"----------\nTally ";

System.out.print(line);

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

System.out.printf("| %6d ", results[i]);

}

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

input.close();

}

}

// ------------------------------------------

// Author: Lauren Escobedo

// Assignment: Chapter 7 Problem 7.27

// Date: 02/15/2023

// Language: Java

// File Name: Exercise7\_27.java

// Description: Exercise 7.27

// - Sieve of Eratosthenes

// ------------------------------------------

import java.util.Arrays;

public class Exercise7\_27 {

public static void main(String[] args) throws Exception {

// Declare and initialize results array

boolean[] results = new boolean[1000];

Arrays.fill(results, true);

// Perform sieve of eratosthenes calculations

for (int i = 2; i < results.length; i++) {

if (i > 2 && i % 2 == 0) {

results[i] = false;

}

if (i > 5 && i % 5 == 0) {

results[i] = false;

}

if (i > 3 && i % 3 == 0) {

results[i] = false;

}

}

// Output array

for (int i = 2; i < results.length; i++) {

if (results[i] == true) {

System.out.printf("%d, ",i);

}

}

}

}

// ------------------------------------------

// Author: Lauren Escobedo

// Assignment: Chapter 7 Problem 7.29

// Date: 02/15/2023

// Language: Java

// File Name: Exercise7\_29.java

// Description: Exercise 7.29

// - Fibonacci Sequence

// ------------------------------------------

import java.util.Scanner;

public class Exercise7\_29 {

// Prints fibonacci sequence up to "n"

static int fibonnaci(int n)

{

// Declare/Initialize variables

int first = 0, second = 1, counter = 0;

int third;

// Perform sequence calculations

while (counter < n) {

// Print

// System.out.print(first + " ");

// Catch overflow

try {

third = Math.addExact(second, first);

} catch(ArithmeticException e) {

System.out.println("Sorry, " + e);

break;

}

// Swap

third = second + first;

first = second;

second = third;

counter++;

}

return first;

}

public static void main(String[] args) throws Exception {

// Declare/initialize variables

int iterations;

Scanner input = new Scanner(System.in);

// User input for iteration limit

System.out.print("\n\nWhat fibonnaci number do you want to calculate? ");

iterations = input.nextInt();

System.out.println();

// Retrieve fibonacci sequence for given "n"

int result = fibonnaci(iterations);

// Print result

System.out.println(result);

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

input.close();

}

}

Text

Description automatically generated

Figure : Exercise 7.17

A picture containing graphical user interface

Description automatically generated

Figure : Exercise 7.27

Text

Description automatically generated

Figure : Exercise 7.29 - Overflow limit for int