## Dice-Throwing Simulator :

Write a complete program that simulates the rolling of two dice. More specifically, it simulates a user-specified number of rolls and then prints a histogram of the number of rolls for each possible pair value. In other words, it prints a histogram for the number of times that two was thrown, the number of times that three was thrown, and so on, all the way up to the number of times that twelve was thrown. Use a frequency array to keep track of the number of times each pair value is thrown. For example, the frequency[2] element holds the number of times two is thrown. The frequency[3] element holds the number of times three is thrown.

As part of your program, write a DiceSimulation class that implements these methods:

* newSimulation ─ This method clears out the frequency array (assigns all elements to zero), prompts the user for the number of rolls, and then simulates the rolls.
* additionalRolls ─ This method prompts the user for the number of rolls that should be added to the current dice-rolling simulation. It then simulates the additional rolls.
* printReport ─ This method prints the dice-rolling simulation results.

As always, you should:

* Limit your use of class variables and instance variables – only use them if appropriate.
* Use appropriate modifiers for your methods. The modifiers we’ve discussed are private, public, static, and final.
* Use helper methods if appropriate.
* Mimic the sample session precisely. In particular, note the window’s title, the prompt text, the result-message text, and the button labels.

Provide a driver class that tests your DiceSimulation class. Your driver class should contain this main method:

public static void main(String[] args)

{

Scanner stdIn = new Scanner(System.in);

String choice; // user's choice of action

boolean done = false; // user's quit flag

DiceSimulation diceSimulation = new DiceSimulation();

System.out.println(

"Welcome to the dice throwing simulator!\n");

do

{

System.out.println(

"Options: (n)ew simulation, (a)dditional rolls," +

" (p)rint, (q)uit");

System.out.print("Enter n, a, p, or q ==> ");

choice = stdIn.nextLine();

switch (choice.charAt(0))

{

case 'n': case 'N':

diceSimulation.newSimulation();

break;

case 'a': case 'A':

diceSimulation.additionalRolls();

break;

case 'p': case 'P':

diceSimulation.printReport();

break;

case 'q': case 'Q':

done = true;

break;

default:

System.out.println("Invalid selection.");

} // end switch

} while (!done);

} // end main

Sample session:

Welcome to the dice throwing simulator!

Options: (n)ew simulation, (a)dditional rolls, (p)rint, (q)uit

Enter n, a, p, or q ==> *x*

Invalid selection.

Options: (n)ew simulation, (a)dditional rolls, (p)rint, (q)uit

Enter n, a, p, or q ==> *N*

How many dice rolls would you like to simulate? *200*

Options: (n)ew simulation, (a)dditional rolls, (p)rint, (q)uit

Enter n, a, p, or q ==> *a*

How many additional rolls? *100*

Options: (n)ew simulation, (a)dditional rolls, (p)rint, (q)uit

Enter n, a, p, or q ==> *p*

DICE ROLLING SIMULATION RESULTS

Each "\*" represents 1% of the total number of rolls.

Total number of rolls = 300.

2: \*\*\*

3: \*\*\*\*\*\*\*

4: \*\*\*\*\*\*\*\*

5: \*\*\*\*\*\*\*\*\*\*\*

6: \*\*\*\*\*\*\*\*\*\*\*\*\*

7: \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

8: \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

9: \*\*\*\*\*\*\*\*

10: \*\*\*\*\*\*\*\*\*\*

11: \*\*\*\*\*

12: \*

Options: (n)ew simulation, (a)dditional rolls, (p)rint, (q)uit

Enter n, a, p, or q ==> *a*

How many additional rolls? *10000*

Options: (n)ew simulation, (a)dditional rolls, (p)rint, (q)uit

Enter n, a, p, or q ==> *p*

DICE ROLLING SIMULATION RESULTS

Each "\*" represents 1% of the total number of rolls.

Total number of rolls = 10300.

2: \*\*\*

3: \*\*\*\*\*

4: \*\*\*\*\*\*\*\*

5: \*\*\*\*\*\*\*\*\*\*\*

6: \*\*\*\*\*\*\*\*\*\*\*\*\*\*

7: \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

8: \*\*\*\*\*\*\*\*\*\*\*\*\*\*

9: \*\*\*\*\*\*\*\*\*\*\*

10: \*\*\*\*\*\*\*\*\*

11: \*\*\*\*\*\*

12: \*\*

Options: (n)ew simulation, (a)dditional rolls, (p)rint, (q)uit

Enter n, a, p, or q ==> *n*

How many dice rolls would you like to simulate? *100*

Options: (n)ew simulation, (a)dditional rolls, (p)rint, (q)uit

Enter n, a, p, or q ==> *p*

DICE ROLLING SIMULATION RESULTS

Each "\*" represents 1% of the total number of rolls.

Total number of rolls = 100.

2: \*\*\*

3: \*\*\*

4: \*\*\*\*\*\*\*\*\*\*\*

5: \*\*\*\*\*\*\*\*\*\*\*

6: \*\*\*\*\*\*\*\*

7: \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

8: \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

9: \*\*\*\*\*\*\*\*\*\*

10: \*\*\*\*\*\*\*\*\*\*\*\*\*

11: \*\*\*\*\*

12: \*\*

Options: (n)ew simulation, (a)dditional rolls, (p)rint, (q)uit

Enter n, a, p, or q ==> *q*