

CSC 1060 Review Project (250 points)

For this project, you will be programming a simplified version of a popular dice game called Craps. Your program will do the following:

- Prompt user for name and use in code
- Use print statements to display an image of 2 dice (using print statements)
 - This could just be generic but you could also use a method to have it print out specific numbers
- Roll two dice (using math. Random) – you must keep track of both die individually.
 - Each die has six faces representing values 1, 2, ..., and 6, respectively.
 - Print out dice roll values.
- Check the sum of the two dice.
 - If the sum is 2, 3, or 12 (called craps), you lose
 - If the sum is 7 or 11 (called natural), you win
 - If the sum is another value (i.e., 4, 5, 6, 8, 9, or 10), a point is established. Continue to roll the dice until either a 7 or the same point value is rolled. If 7 is rolled, you lose. Otherwise, you win. Rolls of 2, 3 or 12 do not cause you to lose if a point has been established.
 - If the point is an even number and the roll is doubles matching the point (3-3, 4-4 or 5-5), print out that the player rolled doubles matching the point and not only won but won big or something special.
- Continue to play until player chooses to stop (via prompt after each win or loss) (you should not ask if a player wants to continue in the middle of a point roll, only after a win or loss has been determined).
- Track number of wins, wins with doubles and losses
- Store results by round in an array and print out as part of statistics (for example, first roll is 2, 5 – store the 2 dice rolls and win in array – 3 different arrays might be one way)
- At game end, print out game statistics.

Your program acts as a single player. Make sure to only use concepts used in the first 8 chapters of your book – points will be deducted for using concepts not in the first 8 chapters of your book. This is also an individual effort so make sure that the work submitted is yours and yours alone.

Make sure to run your program before submitting and make sure that it is working correctly.

Name your file CS1060_ReviewProj_*yourlastname*. Submit your Java file to D2L.

Your code must contain:

- Named constants
- At least one loop structure
- If and/or Switch statements
- Generate random numbers using `math.random ()`
- Well formatted output
- Complex Boolean statements (something that contains an AND and/or an OR)
- Multiple methods created and called properly
- At least one array

You will be graded on the following criteria:

- Complete comment block at beginning
- Comments throughout code (explaining your program flow)
- Code compiles and runs (failure to compile and compile will be an automatic 50% deduction)
- Code works and produces desired output described above
- Variables and constants are named by the standards previously provided
- Only concepts from chapters 1 through 8 of your book are used (concepts used that are not in these chapters may have up to a 50% deduction).

The following is my planned grading rubric:

<i>Concept</i>	Points
Use of arrays	20
comments	20
2 dice	15
dice roll	15
win/lose (if)	20
doubles	15
No extraneous prompts	5
continue (loop)	20
statistics	15
complex Boolean statements	20
variable names	15
constants	15
correctness	35
well formatted	20
total	250

Programming / Coding Standards (this should be review):

What are Coding Standards? Think of coding standards as **a set of rules, techniques, and best practices to create cleaner, more readable, more efficient code with minimal errors**. They offer a uniform format by which software engineers can use to build sophisticated and highly functional code. The concept of Coding Standards is universal throughout industry, albeit that many companies and governmental organizations have significantly stricter and narrowly defined requirements for their software development teams.

As you write your code, use indenting (as appropriate), add white space to make code readable and add comments. Your comments will include preamble (or comment block) and throughout your code to explain what is happening.

At the very beginning of every program file, you'll need to add the following data as comments. (Also known as metadata – data about data.)

```
//Name: <your name here>
//Date: <date finished>
//Program: <assignment name>
//Program Description: <what does this program do – this may be one line or an entire
paragraph depending on the complexity of your program>
//Inputs: < what are the inputs to this program>
//Outputs: <what does this program produce>
```