

### Spring 2026 COP 3330 Program #3: Casino

For this program, you will be given a framework of code to fill out. The program starts with giving the user 100 chips to bet at Arup's Casino. Then, the user is presented with a menu of three choices:

1. Play craps
2. Play roulette
3. Cash out and quit.

The user will make their selection. If it's 1 or 2, the user is first asked how many chips they want to bet, and then the game is played. Depending on the outcome, the user will either lose the chips they bet, or gain some number of chips. (Usually it's the same as what they bet, but there are two exceptions which will be detailed later. After playing each game, the outcome of the game is printed as well as the total number of chips the user left after the game.

Your code will use two static class variables:

```
public static Random rndObj;  
public static Scanner stdin;
```

This way, you can use both of these objects in multiple methods without passing them as parameters.

#### Craps

For Craps, the user will roll two standard 6 sided dice (the computer just does this automatically) and then depending on the total one of three actions will occur:

1. If the total is 7 or 11, the user wins.
2. If the total is 2, 3 or 12, the user loses.
3. Otherwise, the game continues.

In the third option, the user's roll is called their "point." The game continues with the user rolling a pair of dice. If the user gets their point a second time (before they roll a total of 7), they win. Otherwise, if they roll a 7 before they roll their point, they lose. All other rolls (not equal to 7 and not equal to their point) result in the game just continuing.

For craps, the payout is 1:1, meaning that if the user wins, they'll receive the exact number of chips they bet.

### Roulette

The version of Roulette for this assignment will be different than what is normally played. When the wheel is spun, there are 38 possible outcomes: 1 through 36, 0 and 00. When playing this, the user has four possible items they can bet on:

1. odds (1, 3, 5, ..., 35)
2. +evens (2, 4, 6, ..., 36)
3. 0
4. 00

The user will be asked what they want to bet on, and they must enter exactly one of these four strings: "odds", "+evens", "0" or "00". If they enter anything else, they'll be asked to reenter their choice.

The first two bets pay 1:1 and the last two pay 36:1. This means that if you bet "00" with 20 chips and you win, then you win 720 chips.

For roulette, the computer will spin the wheel, generating a random number in between 0 and 37, inclusive. You'll have to write a method that takes in an integer in between 0 and 37 and returns a String storing the corresponding roulette label. (0 through 36 map to their string equivalents and 37 maps to "00".) Your program should display what was spun and print out the result of the bet.

### Static Methods to Fill Out

```
// This one runs the program. Some of this is given.
public static void main(String[] args);

// This displays the menu and returns the user's menu choice.
// This method does error checking, by reprompting the user
// over and over again until they enter 1, 2 or 3. Then the
// valid menu choice is returned.
public static int getMenuChoice();

// This runs a complete game of craps and returns true if the
// user wins, and false otherwise. Note that asking what the bet
// is doesn't happen in this method, that happens in main.
public static boolean craps();

// This runs a complete game of roulette. It returns -1 if
// the user loses, 1 if the user wins a 1:1 bet and 36 if
// the user wins a 36:1 bet.
public static int roulette();

// This method takes in an integer in between 0 and 37 and
// returns the corresponding roulette space as a String.
public static String getRouletteSpace(int id);
```

```
// This method takes in the number of chips the user has
// and asks the user how many chips the user wants to bet.
// This method does error checking by reprompting the user
// to enter their bet over and over again until they enter
// an answer in between 1 and maxChips, inclusive. Then this
// number is returned.
public static int getBet(int maxChips);
```

### **Sample Program Run #1 (User Input in Bold)**

Welcome to Arup's Casino. You begin with 100 chips. Buenos suerte!  
What would you like to do?

1. Play craps.
2. Play roulette.
3. Cash out and quit.

**5**

Sorry that is invalid. Please try again.  
What would you like to do?

1. Play craps.
2. Play roulette.
3. Cash out and quit.

**2**

Time to play roulette!  
How many chips (0-100) would you like to bet?

**20**

What would you like to bet on: odds, +evens, 0 or 00?  
Pay out for odds and evens is 1:1 for 0 and 00 is 36:1.

**evens**

Sorry that's not a valid bet, please enter odds, +evens, 0 or 00.

**+evens**

You spun 19.

Sorry, your bet didn't land. Better luck next time.

Now, after the game, you have 80 number of chips.

What would you like to do?

1. Play craps.
2. Play roulette.
3. Cash out and quit.

**1**

Time to play craps!  
How many chips (0-80) would you like to bet?

**15**

You rolled 5 and 2 for a total of 7

Great, you win on with the first roll of the dice, congratulations!

Now, after the game, you have 95 number of chips.

What would you like to do?

1. Play craps.
2. Play roulette.
3. Cash out and quit.

**4**

Sorry that is invalid. Please try again.

What would you like to do?

1. Play craps.
2. Play roulette.
3. Cash out and quit.

**2**

Time to play roulette!

How many chips (0-95) would you like to bet?

**10**

What would you like to bet on: odds, +evens, 0 or 00?

Pay out for odds and evens is 1:1 for 0 and 00 is 36:1.

**00**

You spun 24.

Sorry, your bet didn't land. Better luck next time.

Now, after the game, you have 85 number of chips.

What would you like to do?

1. Play craps.
2. Play roulette.
3. Cash out and quit.

**3**

Thanks for visiting Arup's casino. You ended with 85 number of chips.

### **Sample Program Run #2 (User Input in Bold)**

Welcome to Arup's Casino. You begin with 100 chips. Buenos suerte!

What would you like to do?

1. Play craps.
2. Play roulette.
3. Cash out and quit.

**1**

Time to play craps!

How many chips (0-100) would you like to bet?

**50**

You rolled 6 and 3 for a total of 9

Now continue to roll until you either get 9 or a sum of 7.

You rolled 3 and 2 for a total of 5

The game continues, roll again.

You rolled 3 and 2 for a total of 5

The game continues, roll again.

You rolled 5 and 1 for a total of 6

The game continues, roll again.

You rolled 4 and 2 for a total of 6

The game continues, roll again.

You rolled 3 and 4 for a total of 7

Sorry, you lose by rolling 7 first before getting your original roll.

Now, after the game, you have 50 number of chips.

What would you like to do?

1. Play craps.
2. Play roulette.
3. Cash out and quit.

2

Time to play roulette!

How many chips (0-50) would you like to bet?

**30**

What would you like to bet on: odds, +evens, 0 or 00?

Pay out for odds and evens is 1:1 for 0 and 00 is 36:1.

**odds**

You spun 29.

You bet on odds and got an odd number. You get an even payout.

Now, after the game, you have 80 number of chips.

What would you like to do?

1. Play craps.
2. Play roulette.
3. Cash out and quit.

**1**

Time to play craps!

How many chips (0-80) would you like to bet?

**181**

Sorry that's not a valid number of chips to bet.

How many chips (0-80) would you like to bet?

**19**

You rolled 6 and 2 for a total of 8

Now continue to roll until you either get 8 or a sum of 7.

You rolled 4 and 3 for a total of 7

Sorry, you lose by rolling 7 first before getting your original roll.

Now, after the game, you have 61 number of chips.

What would you like to do?

1. Play craps.
2. Play roulette.
3. Cash out and quit.

**3**

Thanks for visiting Arup's casino. You ended with 61 number of chips.

### **Deliverables**

**Please submit a single .java file** for your solution to this problem via WebCourses by the designated due date: casino.java. You should create your file by adding to the file casinoframework.java and changing its name (save as).

**Please make sure to include a header comment and internal comments in your code and indent when appropriate.**