**In-Class Lab**

1. Create a Python Project with the name Lab3
2. Create a Python file called lab3.py
3. In file lab3.py Create function main in the format:

def main():

pass

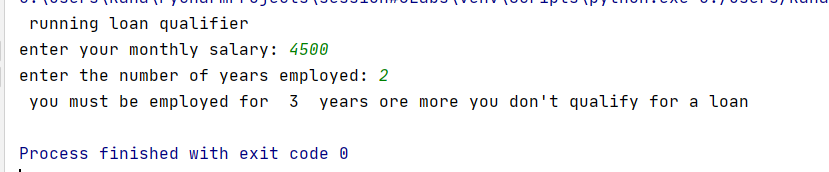
if \_\_name\_\_ = “\_\_main\_\_”:

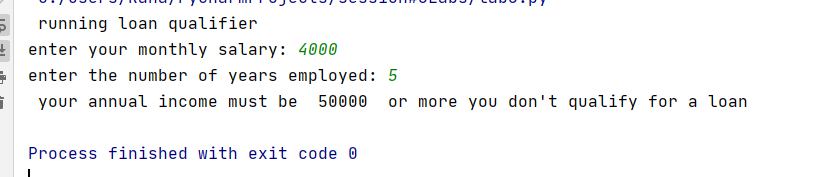
main()

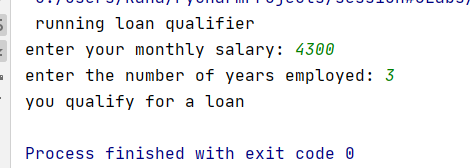
Create the following functions:

* Function loan\_qualifier(), the function asks the user for the monthly salary and the number of years employed. The function will calculate the annual income of the employee,
  + If the employee’s annual income is 50000 $ or more and he/she has been employed for 3 years or more then the function will display the message “you qualify for a loan “
  + If the annual income is less than 50000$, the message “your income must be 50000 or more, you don’t qualify for a loan” will be displayed.
  + If the employee has been employed for less than 3 years, the message “you must be employed for 3 years or more, you don’t qualify for a loan” will be displayed.

Sample output of function loan\_qualifier()

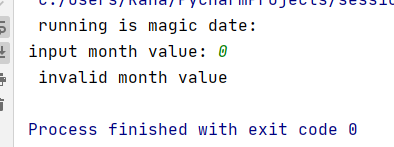


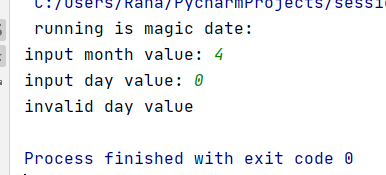


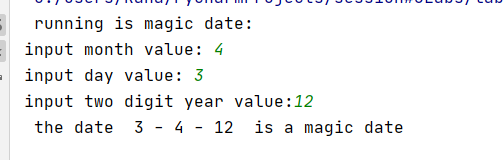


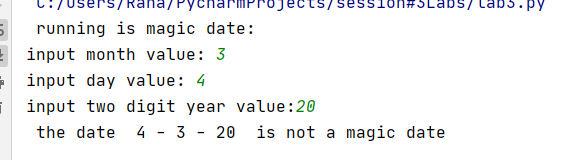
* Function is\_magic\_date(), the function prompts the user to enter the month value, the day value and two digit year value. The function will implement the following validation
  + Month value must be between 1 and 12 inclusive otherwise the message “ invalid month value” will be displayed and the function will be terminated
  + The day value must be between 1 and 31 inclusive otherwise, the message “the day value must be between 1 and 31 inclusive” will be displayed and the function will be terminated.
  + Year value must be a two digit positive number otherwise the message “the year value must be positive and it must be two digit” will be displayed and the function will be terminated.
  + If the year value was the same as the product of the day times the month value, for example if the day = 5, month = 3 and year = 15 then the message “the date 5-3-15 is a magic date” will be displayed. If the day = 4, month =3 and year = 20 the message “ the date 4-3-15 is not a magic date” will be displayed

Sample output of function is\_magic\_Date()



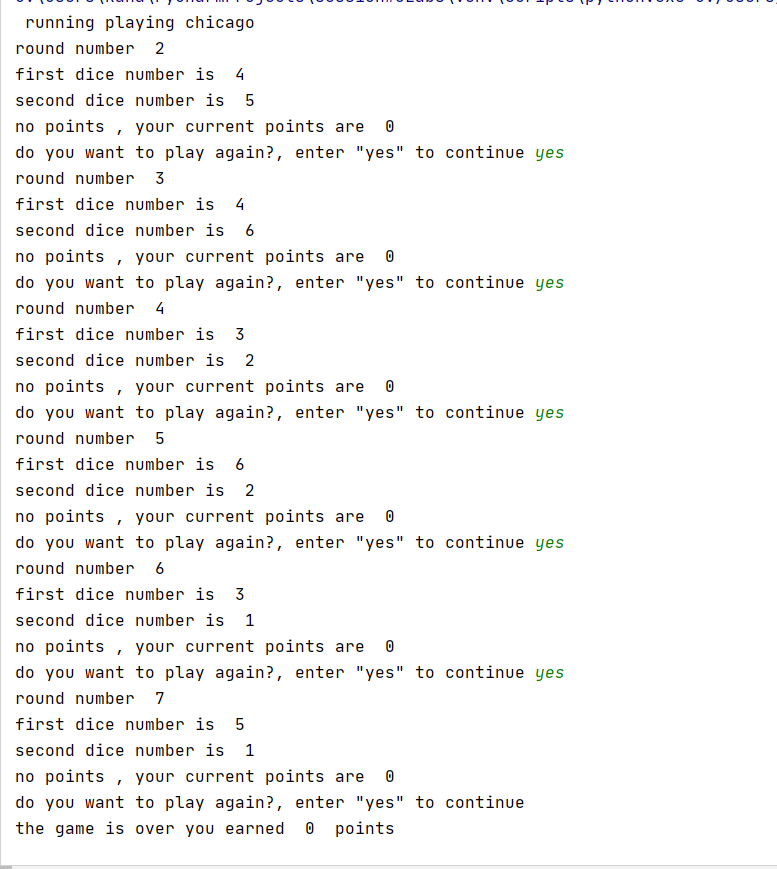


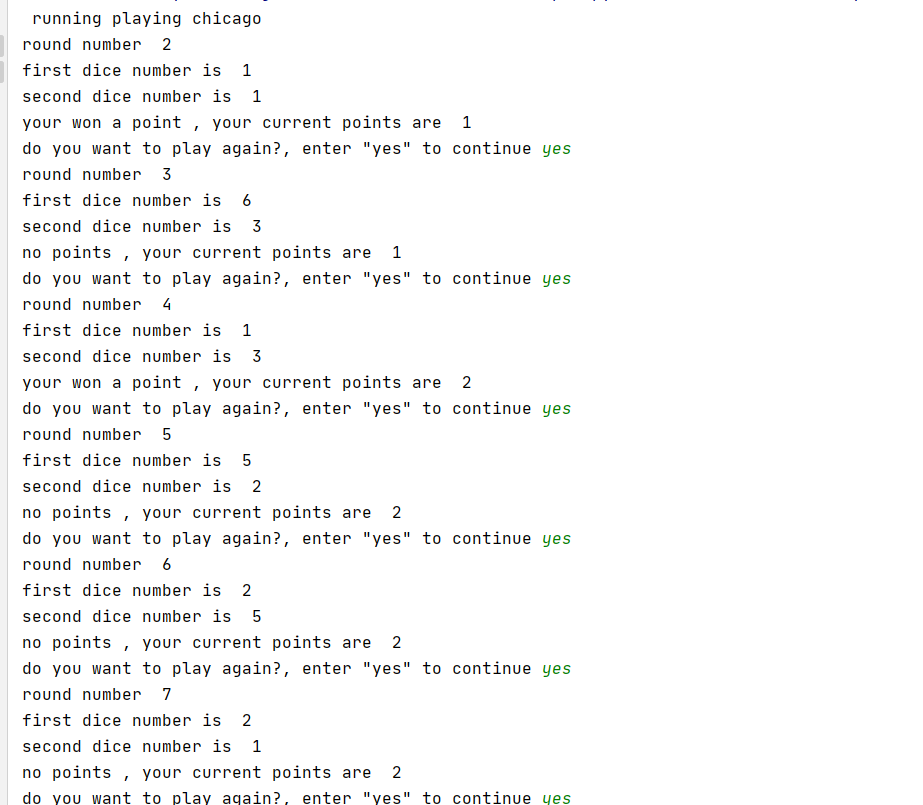




* Function play\_chicago(), the function implements the Chicago game. The game is played for 11 rounds. Round numbers start from 2 to 12. In each round two random numbers between 1 and 6 inclusive will be generated, the random numbers and the round number will be displayed to the user. If the sum of the two numbers equals the round number, the user wins a point and a message to that effect will be displayed. If the user does not win a point a message stating that will also be displayed. At the end of each round the user will be asked if he/she wants to play another round. If the user response was no or the round number is 12 the game will end and the total number of earned points will be displayed.

Sample output of play\_chicago





* In the function main(), remove the keyword pass and call functions loan\_qualifier, is\_magic\_date() and play\_chicago()

**Take-Home lab:**

Add the following functions to the file lab3.py

* Function is\_even(), the function takes an integer number as a parameter and returns True if the number is even and False if the number is odd.
* Function get\_pocket\_colour(), the function accepts an integer between 0 through 36 inclusive and return the colour of the pocket with the given number. The possible colours are green, red or black. The colours of the pockets are as follows:
  + Pocket 0 is green.
  + For pockets 1 through 10 inclusive, the odd-numbered pockets are red and the even-numbered pockets are black.
  + For pockets 11 through 18 inclusive, the odd-numbered pockets are black and the even-numbered pockets are red.
  + For pockets 19 through 28 inclusive, the odd-numbered pockets are red and the even-numbered pockets are black.
  + For pockets 29 through 36 inclusive, the odd-numbered pockets are black and the even-numbered pockets are red.
* Function play\_roulette(), on a roulette wheel, the pockets are numbered from 0 to 36. The function will ask the user to enter a pocket number and the amount of a bet. The function displays whether the pocket is green, red, or black and weather the user wins or loses.
* If the number is 0 the colour is green and the user will not win or lose.
* If the number is even and the colour is black the user wins 50% of the bet so if the bet is 100 the user will get 150.
* If the number is even and the colour is red the user wins 100% of the bet amount so if the bet is 100 the user gets 200.
* If the number is odd and the colour is black the user loses the bet.
* If the number is odd and the colour is red the user wins 50% of the bet.

A message of the prize amount or loss will be displayed

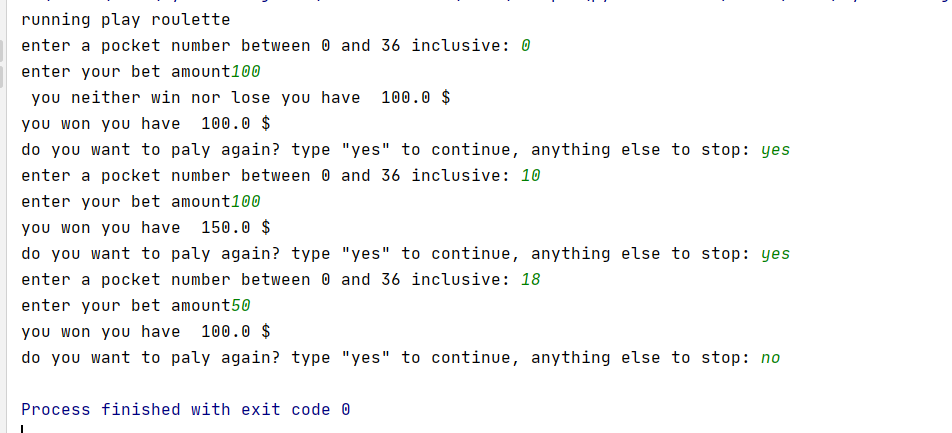
The program should display an error message if the user enters a number that is outside the range of 0 through 36.

The user will be prompted if he/she wants to play another round. The game will keep going until the user does not answer yes.

**Hint:** use function is\_even()and function get\_pocket-colour.

* Add the function calls play\_roulette() to the main function.

Sample output of function play\_roulette()



Best Practices

* Variable names are descriptive and should be lower\_snake\_case
* function names should be lower\_snake\_case
* All functions include Doc-String comments

Submission

Submit zipped folder containing project Lab3 to the dropbox before the deadline.