Exercise 10

1/1 point (graded)

In this exercise, we will use all the functions we have written to simulate an entire game.
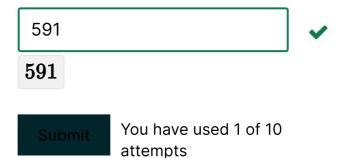
The functions `create_board()` , `random_place(board, player)` , and `evaluate(board)` are all defined as in previous exercises.

Create a function `play_game()` that:

- Creates a board.

- Alternates taking turns between two players (beginning with Player 1), placing a marker during each turn.

- Evaluates the board for a winner after each placement.

- Continues the game until one player wins (returning 1 or 2 to reflect the winning player), or the game is a draw (returning -1).

Call `play_game` 1000 times, and store the results of the game in a list called `results` . Use `random.seed(1)` so we can check your answer!

How many times does Player 1 win out of 1000 games?

591 ✔

591

Submit   You have used 1 of 10 attempts

✔ Correct (1/1 point)
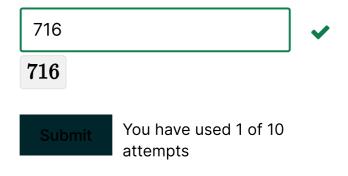
Exercise 11

1/1 point (graded)

In the previous exercise, we saw that when guessing at random, it's better to go first, as one would expect. Let's see if Player 1 can improve their strategy.

Create a function `play_strategic_game()`, where Player 1 always starts with the middle square, and otherwise both players place their markers randomly.

Call `play_strategic_game` 1000 times.

Set the seed to 1 using `random.seed(1)` again.

How many times does Player 1 win out of 1000 games with this new strategy?

716 ✔

716

Submit   You have used 1 of 10 attempts

✔ Correct (1/1 point)