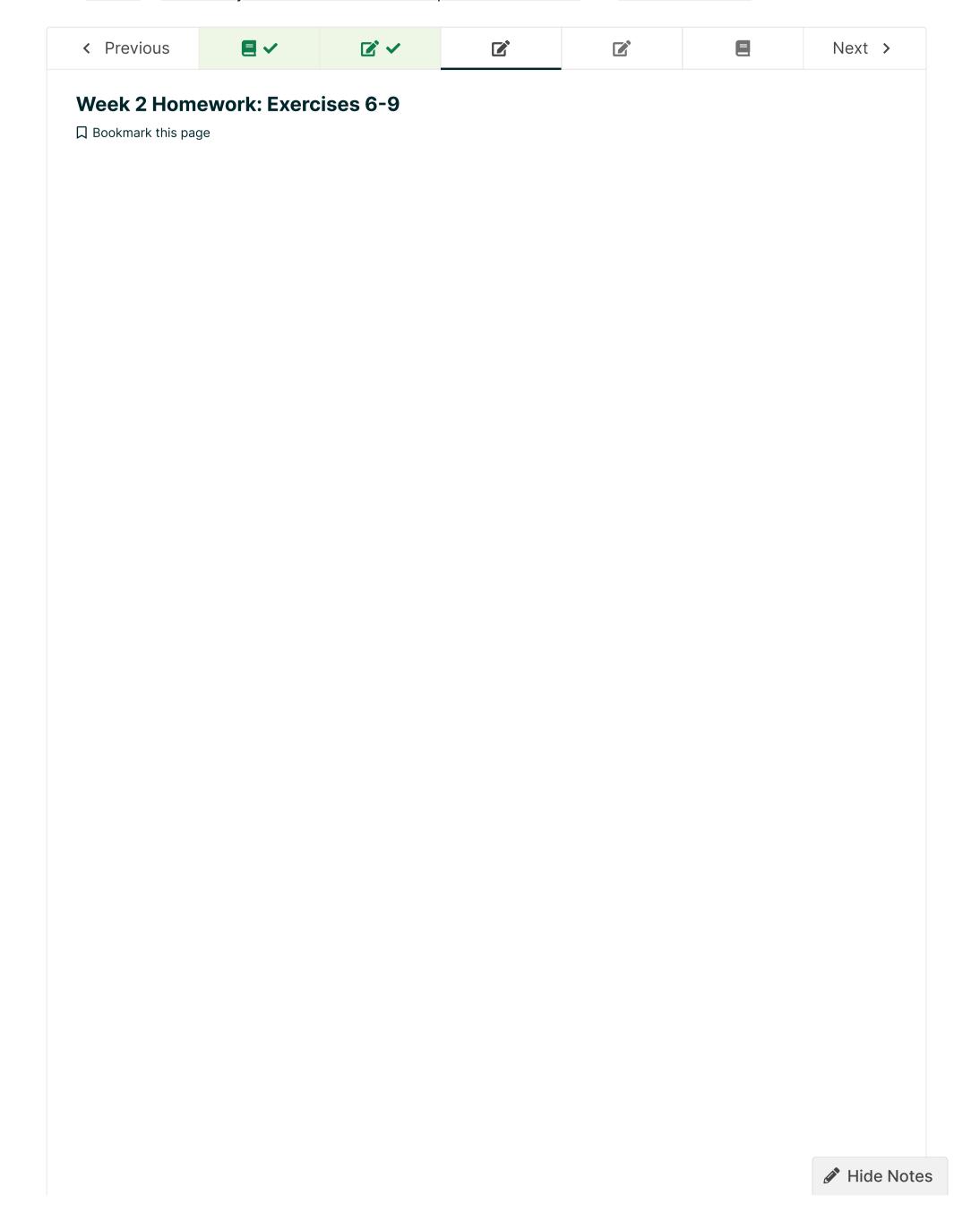


<u>Course</u> <u>Progress</u> <u>Dates</u> <u>Discussion</u> <u>Syllabus and FAQs</u> <u>Notes</u>

☆ Course / Week 2: Python Libraries and Concepts Used in Resear... / Homework: Week...



Homework due Jul 13, 2021 22:00 +06 Exercise 6 1/1 point (graded) In exercises 6 through 9, we will make functions that check whether either player has won the game. Make a function row_win(board, player) that takes the player (integer) and determines if any row consists of only their marker. Have it return True if this condition is met and False otherwise. Note that board is already defined as in Exercise 5. Call row_win to check if Player 1 has a complete row. Does Player 1 have a complete row? Yes O No Submit You have used 1 of 1 attempt ✓ Correct (1/1 point) Exercise 7 1/1 point (graded) In exercises 6 through 9, we will make functions that check whether either player has won the game. Make a function | col_win(board, player) | that takes the player (integer) and determines if any column consists of only their marker. Have it return True if this condition is met and False otherwise. Note that board is already defined as in Exercise 5. Call col win to check if Player 1 has a complete column. Does Player 1 have a complete column? Yes No

You have used 1 of 1 attempt

Submit

Exercise 8

1/1 point (graded)

In exercises 6 through 9, we will make functions that check whether either player has won the game.

Finally, create a function diag_win(board, player) that takes the player (integer) and determines if any diagonal consists of only their marker.

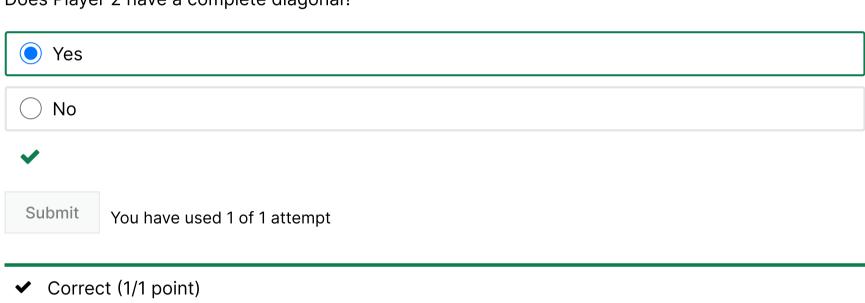
Have it return True if this condition is met and False otherwise.

Note that board is modified from Exercise 5. Call diag_win to check if Player 2 has a complete diagonal.

Use this sample code to get started:



Does Player 2 have a complete diagonal?



Exercise 9

1/1 point (graded)

In exercises 6 through 9, we will make functions that check whether either player has won the game.

Create a function <code>evaluate(board)</code> that uses <code>row_win</code>, <code>col_win</code>, and <code>diag_win</code> functions for both players. If one of them has won, return that player's number. If the board is full but no one has won, return -1. Otherwise, return 0.

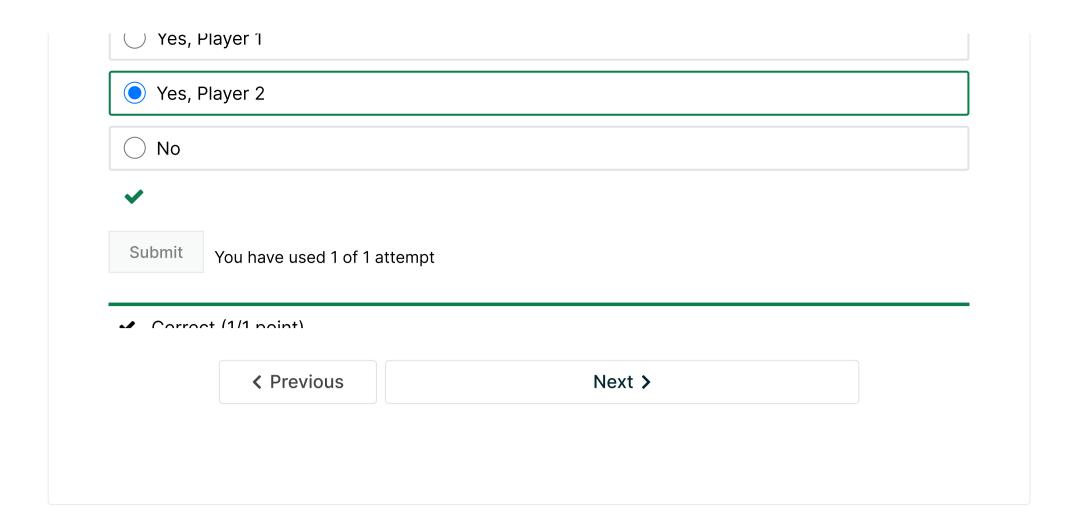
Note that board is defined as in Exercise 8. Call evaluate to see if either player has won the game yet.

Use this sample code to get started:

```
def evaluate(board):
winner = 0
for player in [1, 2]:
    # add your code here!
    pass
if np.all(board != 0) and winner == 0:
    winner = -1
return winner
```

Has anyone won the game yet?





© All Rights Reserved



edX

<u>About</u>

Affiliates

edX for Business

Open edX

Careers

<u>News</u>

Legal

Terms of Service & Honor Code

Privacy Policy

Accessibility Policy

Trademark Policy

<u>Sitemap</u>

Connect

<u>Blog</u>

Contact Us

Help Center

Media Kit

Donate















