

The Excel Logger



The Excel Logger tool was made using the openpyxl package and allowed automation of exporting PlanetWars win data to .xlsx format. This class was written to be scalable and outputs array data readable by numpy if deeper analysis is needed.

```
15 from openpyxl import Workbook
16 from openpyxl.styles import Font
17 from players import Player
18
19
20 class ExcelLogger(object):
21     # TODO: Initialize the logger
22     def __init__(self, book_name: str, sheet_name: str):
23         self.workbook = Workbook()
24         self.book_name = self.__validateBookName(book_name)
25         self.worksheet = self.__initWorksheet(sheet_name)
26
27         self.__cleanupExcelDefaults
28
29     # Summary: Adds headers to worksheet for numpy analysis
30 > def __initWorksheet(self, sheet_name: str):...
42
43     # Summary: Cleans up default excel file
44 > def __cleanupExcelDefaults(self):...
52
53     # Summary: Ammends .xlsx if needed
54 > def __validateBookName(self, given_book_name):...
62
63     # Summary: Finds empty row and calls __fillRow()
64 > def appendWinData(self, match_id: int, winner, opponent, map_name: str):...
83
84     # Summary: Appends a list of data to a row
85 > def __fillRow(self, data: list, row_id: int):...
90
91     # Summary: Tests for a tie returns true if tie occur
92 > def __tieTest(self, winner) -> bool:...
99
```

ExcelLogger class function overview.

This class will initialize and empty a new excel workbook, build sheets with headers and style the header cells.

The method *appendWinData()* is called each time a bot match ends. This function finds an empty row in the sheet, appends [*match_id*, *winner_name*, *loser_name*, *map_played*].

Additionally, the class is able to deal with tie situations (as it wouldn't be passed a *winner_name* or *loser_name* string value).