

SOURCE CODE

utils.py


▶ ✓ □ ...

10]



1

> TIMELINE

 test.py

utils.py

```
26 self.cells = [EMPTY] * 9
```



1

> TIMELINE

 test.py

▶ ✓ □ ...

player.py > Player > get_move

```
1 # player.py
2 class Player:
3     def __init__(self, name, symbol):
4         self.name = name
5         self.symbol = symbol
6
7     def get_move(self):
8         try:
9             return int(input(f"{self.name} ({self.symbol}), enter position (1-9): ")) - 1
10        except ValueError:
11            return -1
```



> TIMELINE

EXPLORER

> CSE

> __pycache__

board.py

constants.py

game.py

main.py

player.py

test.py

utils.py

> OUTLINE

> TIMELINE

constants.py

board.py

player.py

utils.py

game.py

main.py

test.py

utils.py > validate_move

1

utils.py

2

from constants import WIN_COMBINATIONS

3

4

def check_win(board, symbol):

5

cells = board.cells

6

for a, b, c in WIN_COMBINATIONS:

7

if cells[a] == cells[b] == cells[c] == symbol:

8

return True

9

return False

10

11

def validate_move(move):

12

return 0 <= move <= 8

Ln 12, Col 26

Spaces: 4

UTF-8

LF

{ }

Python

Finish Setup

Python 3.13.7

📄

🔍

👤

🔧

🏠

🔄

🧪

🐍

EXPLORER

▼ CSE

> __pycache__

📄 board.py

📄 constants.py

📄 game.py

📄 main.py

📄 player.py

📄 test.py

📄 utils.py

> OUTLINE

> TIMELINE

constants.py

board.py

player.py

utils.py

game.py

main.py

test.py

game.py > Game > start

1 # game.py

2 from board import Board

3 from utils import check_win, validate_move

4 from constants import PLAYER_X, PLAYER_O

5 from player import Player

6

7 class Game:

8 def __init__(self):

9 print("=== Tic Tac Toe ===\n")

10 p1_name = input("Enter Player 1 Name: ")

11 p2_name = input("Enter Player 2 Name: ")

12

13 self.player1 = Player(p1_name, PLAYER_X)

14 self.player2 = Player(p2_name, PLAYER_O)

15 self.board = Board()

16

17 def play(self):

18 current_player = self.player1

19

20 while True:

21 self.board.display()

22 move = current_player.get_move()

23

24 if not validate_move(move):

25 print("Invalid input! Enter a number between 1-9.")

26 continue

27

28 if not self.board.update_cell(move, current_player.symbol):

29 print("Cell already occupied! Try again.")

30 continue

31

32 if check_win(self.board, current_player.symbol):

33 self.board.display()

34 print(f"🎉 {current_player.name} ({current_player.symbol}) WINS!")

35 break

36

37 if self.board.is_full():

38 self.board.display()

39 print("It's a DRAW!")

40 break

41

42 # switch player

43 current_player = self.player2 if current_player == self.player1 else self.player1

44

Python 3.13.7

Ln 52, Col 31

Spaces: 4

UTF-8

LF

{ }

Python

Finish Setup

▶ ✓ □ ...

game.py > Game > start

11

```
def start(self):
    while True:
        self.play()
        again = input("Play again? (y/n): ").lower()
        if again != 'y':
            print("Thanks for playing!")
            break
        self.board.reset()
```

```

1 public void Read()
2 {
3     Console.WriteLine("Enter a number: ");
4     int n = int.Parse(Console.ReadLine());
5 }
6
7 public void Print()
8 {
9     Console.WriteLine("The number is: {0}", n);
10 }
11
12 public void PrintSum()
13 {
14     Console.WriteLine("The sum of the numbers is: {0}", n + 1);
15 }
16
17 public void PrintDiff()
18 {
19     Console.WriteLine("The difference of the numbers is: {0}", n - 1);
20 }
21
22 public void PrintProd()
23 {
24     Console.WriteLine("The product of the numbers is: {0}", n * 2);
25 }
26
27 public void PrintQuot()
28 {
29     Console.WriteLine("The quotient of the numbers is: {0}", n / 2);
30 }
31
32 public void PrintRemainder()
33 {
34     Console.WriteLine("The remainder of the numbers is: {0}", n % 2);
35 }
36
37 public void PrintPower()
38 {
39     Console.WriteLine("The power of the numbers is: {0}", n * n);
40 }
41
42 public void PrintRoot()
43 {
44     Console.WriteLine("The root of the numbers is: {0}", Math.Sqrt(n));
45 }
46
47 public void PrintLog()
48 {
49     Console.WriteLine("The log of the numbers is: {0}", Math.Log(n));
50 }
51
52 public void PrintExp()
53 {
54     Console.WriteLine("The exp of the numbers is: {0}", Math.Exp(n));
55 }
56
57 public void PrintSin()
58 {
59     Console.WriteLine("The sin of the numbers is: {0}", Math.Sin(n));
60 }
61
62 public void PrintCos()
63 {
64     Console.WriteLine("The cos of the numbers is: {0}", Math.Cos(n));
65 }
66
67 public void PrintTan()
68 {
69     Console.WriteLine("The tan of the numbers is: {0}", Math.Tan(n));
70 }
71
72 public void PrintCot()
73 {
74     Console.WriteLine("The cot of the numbers is: {0}", Math.Cot(n));
75 }
76
77 public void PrintSec()
78 {
79     Console.WriteLine("The sec of the numbers is: {0}", Math.Sec(n));
80 }
81
82 public void PrintCsc()
83 {
84     Console.WriteLine("The csc of the numbers is: {0}", Math.Csc(n));
85 }
86
87 public void PrintAcos()
88 {
89     Console.WriteLine("The acos of the numbers is: {0}", Math.Acos(n));
90 }
91
92 public void PrintAsin()
93 {
94     Console.WriteLine("The asin of the numbers is: {0}", Math.Asin(n));
95 }
96
97 public void PrintAtan()
98 {
99     Console.WriteLine("The atan of the numbers is: {0}", Math.Atan(n));
100 }
101
102 public void PrintAcot()
103 {
104     Console.WriteLine("The acot of the numbers is: {0}", Math.Acot(n));
105 }
106
107 public void PrintAsec()
108 {
109     Console.WriteLine("The asec of the numbers is: {0}", Math.Asec(n));
110 }
111
112 public void PrintAcsc()
113 {
114     Console.WriteLine("The acsc of the numbers is: {0}", Math.Acsc(n));
115 }
116
117 public void PrintSinh()
118 {
119     Console.WriteLine("The sinh of the numbers is: {0}", Math.Sinh(n));
120 }
121
122 public void PrintCosh()
123 {
124     Console.WriteLine("The cosh of the numbers is: {0}", Math.Cosh(n));
125 }
126
127 public void PrintTanh()
128 {
129     Console.WriteLine("The tanh of the numbers is: {0}", Math.Tanh(n));
130 }
131
132 public void PrintCoth()
133 {
134     Console.WriteLine("The coth of the numbers is: {0}", Math.Coth(n));
135 }
136
137 public void PrintSinhArc()
138 {
139     Console.WriteLine("The sinhArc of the numbers is: {0}", Math.SinhArc(n));
140 }
141
142 public void PrintCoshArc()
143 {
144     Console.WriteLine("The coshArc of the numbers is: {0}", Math.CoshArc(n));
145 }
146
147 public void PrintTanhArc()
148 {
149     Console.WriteLine("The tanhArc of the numbers is: {0}", Math.TanhArc(n));
150 }
151
152 public void PrintCothArc()
153 {
154     Console.WriteLine("The cothArc of the numbers is: {0}", Math.CothArc(n));
155 }
156
157 public void PrintSinhInv()
158 {
159     Console.WriteLine("The sinhInv of the numbers is: {0}", Math.SinhInv(n));
160 }
161
162 public void PrintCoshInv()
163 {
164     Console.WriteLine("The coshInv of the numbers is: {0}", Math.CoshInv(n));
165 }
166
167 public void PrintTanhInv()
168 {
169     Console.WriteLine("The tanhInv of the numbers is: {0}", Math.TanhInv(n));
170 }
171
172 public void PrintCothInv()
173 {
174     Console.WriteLine("The cothInv of the numbers is: {0}", Math.CothInv(n));
175 }
176
177 public void PrintSinhExp()
178 {
179     Console.WriteLine("The sinhExp of the numbers is: {0}", Math.SinhExp(n));
180 }
181
182 public void PrintCoshExp()
183 {
184     Console.WriteLine("The coshExp of the numbers is: {0}", Math.CoshExp(n));
185 }
186
187 public void PrintTanhExp()
188 {
189     Console.WriteLine("The tanhExp of the numbers is: {0}", Math.TanhExp(n));
190 }
191
192 public void PrintCothExp()
193 {
194     Console.WriteLine("The cothExp of the numbers is: {0}", Math.CothExp(n));
195 }
196
197 public void PrintSinhLog()
198 {
199     Console.WriteLine("The sinhLog of the numbers is: {0}", Math.SinhLog(n));
200 }
201
202 public void PrintCoshLog()
203 {
204     Console.WriteLine("The coshLog of the numbers is: {0}", Math.CoshLog(n));
205 }
206
207 public void PrintTanhLog()
208 {
209     Console.WriteLine("The tanhLog of the numbers is: {0}", Math.TanhLog(n));
210 }
211
212 public void PrintCothLog()
213 {
214     Console.WriteLine("The cothLog of the numbers is: {0}", Math.CothLog(n));
215 }
216
217 public void PrintSinhLogArc()
218 {
219     Console.WriteLine("The sinhLogArc of the numbers is: {0}", Math.SinhLogArc(n));
220 }
221
222 public void PrintCoshLogArc()
223 {
224     Console.WriteLine("The coshLogArc of the numbers is: {0}", Math.CoshLogArc(n));
225 }
226
227 public void PrintTanhLogArc()
228 {
229     Console.WriteLine("The tanhLogArc of the numbers is: {0}", Math.TanhLogArc(n));
230 }
231
232 public void PrintCothLogArc()
233 {
234     Console.WriteLine("The cothLogArc of the numbers is: {0}", Math.CothLogArc(n));
235 }
236
237 public void PrintSinhLogExp()
238 {
239     Console.WriteLine("The sinhLogExp of the numbers is: {0}", Math.SinhLogExp(n));
240 }
241
242 public void PrintCoshLogExp()
243 {
244     Console.WriteLine("The coshLogExp of the numbers is: {0}", Math.CoshLogExp(n));
245 }
246
247 public void PrintTanhLogExp()
248 {
249     Console.WriteLine("The tanhLogExp of the numbers is: {0}", Math.TanhLogExp(n));
250 }
251
252 public void PrintCothLogExp()
253 {
254     Console.WriteLine("The cothLogExp of the numbers is: {0}", Math.CothLogExp(n));
255 }
256
257 public void PrintSinhLogInv()
258 {
259     Console.WriteLine("The sinhLogInv of the numbers is: {0}", Math.SinhLogInv(n));
260 }
261
262 public void PrintCoshLogInv()
263 {
264     Console.WriteLine("The coshLogInv of the numbers is: {0}", Math.CoshLogInv(n));
265 }
266
267 public void PrintTanhLogInv()
268 {
269     Console.WriteLine("The tanhLogInv of the numbers is: {0}", Math.TanhLogInv(n));
270 }
271
272 public void PrintCothLogInv()
273 {
274     Console.WriteLine("The cothLogInv of the numbers is: {0}", Math.CothLogInv(n));
275 }
276
277 public void PrintSinhLogArcExp()
278 {
279     Console.WriteLine("The sinhLogArcExp of the numbers is: {0}", Math.SinhLogArcExp(n));
280 }
281
282 public void PrintCoshLogArcExp()
283 {
284     Console.WriteLine("The coshLogArcExp of the numbers is: {0}", Math.CoshLogArcExp(n));
285 }
286
287 public void PrintTanhLogArcExp()
288 {
289     Console.WriteLine("The tanhLogArcExp of the numbers is: {0}", Math.TanhLogArcExp(n));
290 }
291
292 public void PrintCothLogArcExp()
293 {
294     Console.WriteLine("The cothLogArcExp of the numbers is: {0}", Math.CothLogArcExp(n));
295 }
296
297 public void PrintSinhLogArcInv()
298 {
299     Console.WriteLine("The sinhLogArcInv of the numbers is: {0}", Math.SinhLogArcInv(n));
300 }
301
302 public void PrintCoshLogArcInv()
303 {
304     Console.WriteLine("The coshLogArcInv of the numbers is: {0}", Math.CoshLogArcInv(n));
305 }
306
307 public void PrintTanhLogArcInv()
308 {
309     Console.WriteLine("The tanhLogArcInv of the numbers is: {0}", Math.TanhLogArcInv(n));
310 }
311
312 public void PrintCothLogArcInv()
313 {
314     Console.WriteLine("The cothLogArcInv of the numbers is: {0}", Math.CothLogArcInv(n));
315 }
316
317 public void PrintSinhLogArcExpInv()
318 {
319     Console.WriteLine("The sinhLogArcExpInv of the numbers is: {0}", Math.SinhLogArcExpInv(n));
320 }
321
322 public void PrintCoshLogArcExpInv()
323 {
324     Console.WriteLine("The coshLogArcExpInv of the numbers is: {0}", Math.CoshLogArcExpInv(n));
325 }
326
327 public void PrintTanhLogArcExpInv()
328 {
329     Console.WriteLine("The tanhLogArcExpInv of the numbers is: {0}", Math.TanhLogArcExpInv(n));
330 }
331
332 public void PrintCothLogArcExpInv()
333 {
334     Console.WriteLine("The cothLogArcExpInv of the numbers is: {0}", Math.CothLogArcExpInv(n));
335 }
336
337 public void PrintSinhLogArcInvExp()
338 {
339     Console.WriteLine("The sinhLogArcInvExp of the numbers is: {0}", Math.SinhLogArcInvExp(n));
340 }
341
342 public void PrintCoshLogArcInvExp()
343 {
344     Console.WriteLine("The coshLogArcInvExp of the numbers is: {0}", Math.CoshLogArcInvExp(n));
345 }
346
347 public void PrintTanhLogArcInvExp()
348 {
349     Console.WriteLine("The tanhLogArcInvExp of the numbers is: {0}", Math.TanhLogArcInvExp(n));
350 }
351
352 public void PrintCothLogArcInvExp()
353 {
354     Console.WriteLine("The cothLogArcInvExp of the numbers is: {0}", Math.CothLogArcInvExp(n));
355 }
356
357 public void PrintSinhLogArcInvExpInv()
358 {
359     Console.WriteLine("The sinhLogArcInvExpInv of the numbers is: {0}", Math.SinhLogArcInvExpInv(n));
360 }
361
362 public void PrintCoshLogArcInvExpInv()
363 {
364     Console.WriteLine("The coshLogArcInvExpInv of the numbers is: {0}", Math.CoshLogArcInvExpInv(n));
365 }
366
367 public void PrintTanhLogArcInvExpInv()
368 {
369     Console.WriteLine("The tanhLogArcInvExpInv of the numbers is: {0}", Math.TanhLogArcInvExpInv(n));
370 }
371
372 public void PrintCothLogArcInvExpInv()
373 {
374     Console.WriteLine("The cothLogArcInvExpInv of the numbers is: {0}", Math.CothLogArcInvExpInv(n));
375 }
376
377 public void PrintSinhLogArcInvExpInvExp()
378 {
379     Console.WriteLine("The sinhLogArcInvExpInvExp of the numbers is: {0}", Math.SinhLogArcInvExpInvExp(n));
380 }
381
382 public void PrintCoshLogArcInvExpInvExp()
383 {
384     Console.WriteLine("The coshLogArcInvExpInvExp of the numbers is: {0}", Math.CoshLogArcInvExpInvExp(n));
385 }
386
387 public void PrintTanhLogArcInvExpInvExp()
388 {
389     Console.WriteLine("The tanhLogArcInvExpInvExp of the numbers is: {0}", Math.TanhLogArcInvExpInvExp(n));
390 }
391
392 public void PrintCothLogArcInvExpInvExp()
393 {
394     Console.WriteLine("The cothLogArcInvExpInvExp of the numbers is: {0}", Math.CothLogArcInvExpInvExp(n));
395 }
396
397 public void PrintSinhLogArcInvExpInvExpInv()
398 {
399     Console.WriteLine("The sinhLogArcInvExpInvExpInv of the numbers is: {
```



1

> TIMELINE



utils.py

```
6 game.start()
```



> TIMELINE

utils.py



> TIMELINE

×

17

