

Online Python Compiler (Interpre... x

DAY-2[1][1].pdf x | +

https://www.programiz.com/python-programming/online-compiler/

Programiz

Python Online Compiler

Google Ads

...Get up to ₹60,000 in Ads credit.  
Terms Apply.

Claim Now

Programiz PRO >

main.py

Run

Output

Clear

```
1 def find_ways(m, n, N, start_row, start_col):
2     exit_count = [0]
3     def move_ball(steps, row, col):
4         if row < 0 or row >= m or col < 0 or col >= n:
5             exit_count[0] += 1
6             return
7         if steps == 0:
8             return
9         move_ball(steps - 1, row + 1, col)
10        move_ball(steps - 1, row - 1, col)
11        move_ball(steps - 1, row, col + 1)
12        move_ball(steps - 1, row, col - 1)
13    move_ball(N, start_row, start_col)
14    return exit_count[0]
15 print(find_ways(2, 2, 2, 0, 0))
16
```

6  
=== Code Execution Successful ===

80°F  
Mostly cloudy

Search


ENG  
IN


10:48  
14-10-2024

Online Python Compiler (Interpre... x

DAY-2[1][1].pdf x | +

← ↻ https://www.programiz.com/python-programming/online-compiler/ 🔍 ☆ ⚙ | 📖 ☆ 🗑 🔄 🌐





**Programiz**  
Python Online Compiler

Google Ads

...Get up to ₹60,000 in Ads credit.  
Terms Apply.

[Claim Now](#)

[Programiz PRO >](#)

main.py    Share [Run](#)

Output [Clear](#)

3  
=== Code Execution Successful ===

```
1 def rob_linear(nums):
2     prev2, prev1 = 0, 0
3     for num in nums:
4         temp = prev1
5         prev1 = max(prev1, prev2 + num)
6         prev2 = temp
7     return prev1
8 def rob(nums):
9     if len(nums) == 1:
10         return nums[0]
11     elif len(nums) == 2:
12         return max(nums)
13     return max(rob_linear(nums[:-1]), rob_linear(nums[1:]))
14 nums = [2, 3, 2]
15 print(rob(nums))
```

JS

GO

80°F Mostly cloudy



ENG IN 10:56 14-10-2024

Online Python Compiler (Interpre... x

DAY-2[1][1].pdf x | +

https://www.programiz.com/python-programming/online-compiler/

Programiz

Python Online Compiler

Programiz PRO >

main.py

Share

Run

```
1 def climb_stairs(n):
2     if n == 1:
3         return 1
4     if n == 2:
5         return 2
6     a, b = 1, 2
7     for i in range(3, n + 1):
8         a, b = b, a + b
9     return b
10 print(climb_stairs(4))
```

Output

Clear

5

=== Code Execution Successful ===

80°F Mostly cloudy

Search

ENG IN

10:59 14-10-2024

main.py

1 import math

2 def unique\_paths(m,n):

3 return math.comb(m+n-2,m-1)

4 print (unique\_paths(7,3))

Share

Run

Output

Clear

28

=== Code Execution Successful ===

```
main.py
1- def large_group_positions(s):
2     res = []
3     n = len(s)
4     i = 0
5     while i < n:
6         j = i
7         while j < n and s[j] == s[i]:
8             j += 1
9         if j - i >= 3:
10            res.append([i, j - 1])
11        i = j
12    return res
13 s = "abbxxxxzzy"
14 output = large_group_positions(s)
15 print(output)
16
```

Output

[[3, 6]]

=== Code Execution Successful ===

JS

GO

82°F

Search

ENG  
IN

12:32  
14-10-2024

main.py

Run

Share

```
1- def game_of_life(board):
2-     if not board:
3-         return
4-     m, n = len(board), len(board[0])
5-     next_board = [[0] * n for _ in range(m)]
6-     for i in range(m):
7-         for j in range(n):
8-             live_neighbors = 0
9-             for x in range(-1, 2):
10-                 for y in range(-1, 2):
11-                     if (x != 0 or y != 0) and 0 <= i + x < m
                        and 0 <= j + y < n:
12-                         live_neighbors += board[i + x][j +
                            y]
13-                 if board[i][j] == 1:
14-                     if live_neighbors < 2:
15-                         next_board[i][j] = 0
16-                     elif live_neighbors <= 3:
17-                         next_board[i][j] = 1
18-                 else:
```

Output

Clear

```
[[0, 0, 0], [1, 0, 1], [0, 1, 1], [0, 1, 0]]

=== Code Execution Successful ===
```

Programiz

Python Online Compiler

Google Ads

Get up to \$50,000 in AdSense credit. Terms Apply.

Claim Now

Programiz PRO >

main.py

Run

Share

```
12         live_neighbors += board[i + x][j + y]
13     if board[i][j] == 1:
14         if live_neighbors < 2:
15             next_board[i][j] = 0
16         elif live_neighbors <= 3:
17             next_board[i][j] = 1
18         else:
19             next_board[i][j] = 0
20     else:
21
22         if live_neighbors == 3:
23             next_board[i][j] = 1
24     for i in range(m):
25         for j in range(n):
26             board[i][j] = next_board[i][j]
27 board = [[0, 1, 0], [0, 0, 1], [1, 1, 1], [0, 0, 0]]
28 game_of_life(board)
29 print(board)
```

Output

Clear

```
[[0, 0, 0], [1, 0, 1], [0, 1, 1], [0, 1, 0]]
=== Code Execution Successful ===
```

81°F Monthly cloudy

Search

14-10-2024 12:23

ENG IN

main.py

```
1 poured = 2
2 query_row = 1
3 query_glass = 1
4 current_row = [0] * (query_row + 1)
5 current_row[0] = poured
6 for row in range(query_row):
7     next_row = [0] * (row + 2)
8     for glass in range(row + 1):
9         if current_row[glass] > 1:
10             overflow = (current_row[glass] - 1) / 2.0
11             next_row[glass] += overflow
12             next_row[glass + 1] += overflow
13     current_row = next_row
14 result = min(1, current_row[query_glass])
15 print(result)
```

Output

0.5

=== Code Execution Successful ===

Clear