

Challenge 1:

Predicted Output: Sum: 10

Variable	Value
arr	[1, 2, 3, 4]
sum	0 (initial value)
arr[i]	arr[0], arr[1], arr[2], arr[3]
sum (first iteration)	0 + 1 = 1
sum (second iteration)	1 + 2 = 3
sum (third iteration)	3 + 3 = 6
sum (fourth iteration)	6 + 4 = 10
calculateSum(numbers)	10

Output:

Sum: 10

Challenge 2:

Predicted Output:

true

false

true

Variable	Value
Function Call: isEven(4)	
num	4
4 % 2 === 0	0 === 0 → True
Output	true
Function Call: isEven(4)	
num	7
7 % 2 === 0	1 === 0 → False
Output	false
Function Call: isEven(4)	
num	0
0 % 2 === 0	0 === 0 → True
Output	true

Output:

true

false

true

Challenge 3:

Predicted Output:

Hello, Alice!

Hello, Bob!

Hello, Charlie!

Variable	Value
names	["Alice", "Bob", "Charlie"]
i	[0, 1, 2]
names[i]	"Alice" "Bob" "Charlie"
greet(names[i])	"Hello, Alice!" "Hello, Bob!" "Hello, Charlie!"

Output:

Hello, Alice!

Hello, Bob!

Hello, Charlie!

Challenge 4:

Predicted Output:

[30, 20, 10]

[10, 20, 30]

Variable	Value
arr	[10, 20, 30]
reverse	[] (initially empty)
i	[0, 1, 2]
arr[i] = arr[2] arr[i] = arr[1] arr[i] = arr[0]	30 20 10
reversed	[30] [30, 20] [30, 20, 10]
return	[30, 20, 10]
reverseArray(originalArray)	[30, 20, 10]
(originalArray)	[10, 20, 30]

Output:

[30, 20, 10]

[10, 20, 30]

Challenge 5:

Predicted Output:

[[2, 4], [6, 8]]

Variable	Value
matrix	[[1, 2], [3, 4]] (initial value)
(i = 0, j = 0) matrix[0][0]	1 * 2 = 2
Updated Matrix	[[2, 2], [3, 4]]
(i = 0, j = 1) matrix[0][1]	2 * 2 = 4
Updated Matrix	[[2, 4], [3, 4]]
(i = 1, j = 0) matrix[0][1]	3 * 2 = 6
Updated Matrix	[[2, 4], [6, 4]]
(i = 1, j = 1) matrix[0][1]	4 * 2 = 8
Updated Matrix	[[2, 4], [6, 8]]
return	[[2, 4], [6, 8]]

Output:

[[2, 4], [6, 8]]