Stream Quiz

Due No due date **Points** 100 **Questions** 2 **Time Limit** 5 Minutes

Instructions

Exam Instructions: TIME LIMITED, AUTO-SUBMIT, AUTO-GRADING YOUR VIDEO MUST BE ON

READ CAREFULLY AND ANSWER EACH QUESTION ONE AT A TIME!

Read each question carefully and select the best choice or choices which answer that question.
Answer all questions within the time allotted.
Some questions have more than one correct answer. There is no credit for a partially answered question.
You may use your Eclipse IDE (or other instructor approved IDE).
You may not use any other materials UNLESS OTHERWISE INSTRUCTED BY THE INSTRUCTOR OR T.A.

You may NOT have any discussions with anyone except the instructor or T.A..

Taking this exam outside of these parameters will result in disqualification and a grade of zero.

Answer all questions.

EXAM WILL AUTO SUBMIT AFTER ALLOTTED TIME EXPIRES.

Attempt History

	Attempt	Time	Score
LATEST	Attempt 1	5 minutes	75 out of 100

Score for this quiz: **75** out of 100
Submitted Nov 19 at 8:37pm
This attempt took 5 minutes.

Question 1

37.5 / 50 pts

List<Integer> nList = new ArrayList<>(Arrays.asList(3,5,1,4,2));

// SCALE x 10 ODD numbers, show on console ONE LINE, comma separated , DESCENDING order (50, 30, 10,) ...

nList.Stream

.filter

.sorted

.map

.forEach

FOR THE PRECEDING CODE SEGMENT TO OUTPUT DATA AS SPECIFIED IN CODE COMMENT, Set up a Stream API pipeline by MATCHING Stream operation with it's argument:

e.g.,	LEFT SIDE	RIGHT SIDE
	.foreEach	System.out::println

Correct! **.forEach** (e) -> System.out.print(e

ou Answered **.filter** (e) -> e%2 == 2

Correct Answer e -> e%2 == 1

Correct! **.map** e -> e*10

Correct! **.sorted** Comparator.reverseOrde

Other Incorrect Match Options:

- e -> e/2 = 1
- No argument
- Comparator.reverseOrder(null)
- () -> 10
- (e) -> e%2 == 2
- () -> e*10
- null
- System.out.println()
- System.out::println

- Comparator.reverse
- Comparator.comparing(null)
- Comparator.comparing(reverse)
- System.out::print
- e -> e%2 = 1
- (e) -> e/2 == 1

Question 2

37.5 / 50 pts

List<Integer> nList = new ArrayList<>(Arrays.asList(3,5,1,4,2));

// SCALE x 101 EVEN numbers, show on console ONE LINE, comma separated , ASCENDING order (202, 404,) ...

nList.Stream

.filter

.sorted

.map

.forEach

FOR THE PRECEDING CODE SEGMENT TO OUTPUT DATA AS SPECIFIED IN CODE COMMENT, Set up a Stream API pipeline by MATCHING Stream operation with it's argument:

e.g.,	LEFT SIDE	RIGHT SIDE
	.foreEach	System.out::println

Correct!

.forEach (e) -> System.out.print(e

Correct!

.filter e -> e%2 == 0

ou Answered

.map () -> e*101

Correct Answer e -> e*101

Correct!

.sorted no argument

Other Incorrect Match Options:

- () -> 101
- e -> e%2 = 1
- () -> e*101
- Comparator.reverseOrder(null)
- (e) -> e/2 == 2
- Comparator.reverse
- Comparator.comparing(null)
- Comparator.reverseOrder()
- System.out::println
- System.out.println()
- (e) -> e%2 == 2
- e -> e/2 = 2
- Comparator.comparing(reverse)
- System.out::print
- null

Quiz Score: **75** out of 100