



You can view this report online at : <https://www.hackerrank.com/x/tests/799095/candidates/17550480/report>

Full Name:	Xi Chen
Email:	xc4@andrew.cmu.edu
Test Name:	AD - KBA - Advanced Java
Taken On:	17 Sep 2020 09:19:10 EDT
Time Taken:	73 min/ 80 min
City:	Buffalo
Course:	[TEK Boston] 2020-Java Developer 01
Invited by:	Customized
Invited on:	16 Sep 2020 18:03:49 EDT
Skills Score:	
Tags Score:	<div>Java</div> 89.29/100

89.3%  
89/100

scored in **AD - KBA - Advanced Java** in 73 min on 17 Sep 2020 09:19:10 EDT


Recruiter/Team Comments:

No Comments.

	Question Description	Time Taken	Score	Status
Q1	Which of the following... > Multiple Choice	1 min 53 sec	5/ 5	✓
Q2	Which of these... > Multiple Choice	4 min 26 sec	5/ 5	✓
Q3	Which statement... > Multiple Choice	6 min 15 sec	5/ 5	✓
Q4	Which of the following... > Multiple Choice	10 min 11 sec	5/ 5	✓
Q5	Which of the following... > Multiple Choice	4 min	5/ 5	✓
Q6	Call Static Method of interface > Multiple Choice	7 min 53 sec	0/ 5	✗
Q7	default method Interface > Multiple Choice	9 min 34 sec	0/ 5	✗
Q8	What is a Functional... > Multiple Choice	1 min 16 sec	5/ 5	✓
Q9	What is a Functional Method... > Multiple Choice	4 min 27 sec	5/ 5	✓
Q10	Which of the following... > Multiple Choice	1 min 18 sec	5/ 5	✓
Q11	Streams > Multiple Choice	3 min 41 sec	5/ 5	✓
Q12	Intermediate Operations > Multiple Choice	1 min 58 sec	4.29/ 5	⚠
Q13	skip() > Multiple Choice	1 min 53 sec	5/ 5	✓

Q14	<a href="#">StringJoiner merge()</a> > Multiple Choice	58 sec	5/ 5	✓
Q15	<a href="#">What is a stream...</a> > Multiple Choice	1 min 22 sec	5/ 5	✓
Q16	<a href="#">Is main a Thread?</a> > Multiple Choice	36 sec	5/ 5	✓
Q17	<a href="#">getState()</a> > Multiple Choice	55 sec	5/ 5	✓
Q18	<a href="#">sleep() from MyThread vs sleep() from another Thread</a> > Multiple Choice	1 min 28 sec	5/ 5	✓
Q19	<a href="#">sleep() vs wait()</a> > Multiple Choice	7 min	5/ 5	✓
Q20	<a href="#">Use Multithreading</a> > Multiple Choice	44 sec	5/ 5	✓

QUESTION 1



Correct Answer

Score 5

Which of the following... > Multiple Choice Java

QUESTION DESCRIPTION

Which of the following Java Collection Framework (JCF) classes implement Set<E>?


CANDIDATE ANSWER

Options: (Expected answer indicated with a tick)

☐ MapSet<E>
☒ TreeSet<E>
☐ EnumSet<E>
☒ HashSet<E>
☐ UpSet<E>

No Comments

QUESTION 2



Correct Answer

Score 5

Which of these... > Multiple Choice Java

QUESTION DESCRIPTION

Which of these JDK packages contains the primary Collection classes and interfaces?


CANDIDATE ANSWER

Options: (Expected answer indicated with a tick)

☐ java.awt
☐ java.collection
☐ java.lang
☒ java.util

No Comments

QUESTION 3



Correct Answer

Score 5

Which statement... > Multiple Choice 


Java

QUESTION DESCRIPTION

Which statement best describes the Java Collections Framework's Set<E> ?

CANDIDATE ANSWER

Options: (Expected answer indicated with a tick)

☒

A Set is a collection that contains no duplicate elements.

☐

A Set is a collection of values which must not intersect with any other Set.

☐


A Set is a collection of no more than 6 tennis games, not counting tie-breakers.

☐

A Set is a collection that must contain duplicate values.

No Comments

QUESTION 4



Correct Answer

Score 5

Which of the following... > Multiple Choice 

Java

QUESTION DESCRIPTION

Which of the following best describes a *map* in the Java Collections Framework?

CANDIDATE ANSWER

Options: (Expected answer indicated with a tick)

☐


A map is a collection of unique values where each value is associated with one or more keys.

☐

A map is a special collection used for geo-location.


☐


A map is a collection of unique keys where each key is associated with one or more values.

☒

A map is a collection of unique keys where each key is associated with a value.

No Comments

<b>QUESTION 5</b>  Correct Answer	<b>Which of the following...</b> > Multiple Choice <span>Java</span>
Score 5	<p><u>QUESTION DESCRIPTION</u></p> <p>Which of the following collection classes implements the List&lt;E&gt; interface?</p>
	<p><b>CANDIDATE ANSWER</b></p> <p><b>Options:</b> (Expected answer indicated with a tick)</p> <p><input checked="" type="checkbox"/> <input type="radio"/> ArrayList&lt;E&gt;</p> <p><input type="radio"/> Map&lt;K,V&gt;</p> <p><input checked="" type="checkbox"/> <input type="radio"/> LinkedList&lt;E&gt;</p> <p><input type="radio"/> Set&lt;E&gt;</p> <p>No Comments</p>

<b>QUESTION 6</b>  Wrong Answer	<b>Call Static Method of interface</b> > Multiple Choice <span>Java</span>
Score 0	<p><u>QUESTION DESCRIPTION</u></p> <p>In Java 8, can a Static Method <b>of</b> an Interface be called by using an instance of a class that implements that Interface, but that does not override the Static Method?</p>
	<p><b>CANDIDATE ANSWER</b></p> <p><b>Options:</b> (Expected answer indicated with a tick)</p> <p><input checked="" type="checkbox"/> <input type="radio"/> No, a Static Method of an Interface must be called by using the interface name preceding the method name.</p> <p><input type="radio"/> <input checked="" type="radio"/> Yes, in this aspect, calling a Static Method within an interface is identical to calling one in a class.</p> <p><input type="radio"/> Yes, but only if the Static Method is not abstract</p> <p><input type="radio"/> Yes, but only if the Static Method is final</p> <p>No Comments</p>

## QUESTION 7



Wrong Answer

Score 0

## default method Interface &gt; Multiple Choice

Java

## QUESTION DESCRIPTION

What is the most likely output of this program?

```
interface Cat {  
  
    default String says() {  
  
        return "miaw";  
  
    }  
  
}  
  
interface Lion {  
  
    default String says() {  
  
        return "ROAR!";  
  
    }  
  
}  
  
class Tiger implements Cat, Lion{  
  
}  
  
public class Main{  
  
    public static void main(String args[]) {  
  
        System.out.println(new Tiger().says());  
  
    }  
  
}
```


## CANDIDATE ANSWER

Options: (Expected answer indicated with a tick)

- ☒ ☐ Unresolved compilation problem: Duplicate default methods named says with the parameters () and () are inherited from the types Lion and Cat
- ☐ miaw
- ☐ ROAR
- ☒ Unresolved compilation problem: The method says() is undefined for the type Tiger

No Comments

QUESTION 8



Correct Answer

Score 5

What is a Functional... > Multiple Choice 


Java

QUESTION DESCRIPTION

What is a Functional Interface in Java?

CANDIDATE ANSWER

Options: (Expected answer indicated with a tick)



☒ An interface that contains only one abstract method

☐

An interface that contains one or more abstract method

☐


An interface that contains no abstract method

☐

An interface that contains more than one abstract method

No Comments

QUESTION 9



Correct Answer

Score 5

What is a Functional Method... > Multiple Choice 


Java

QUESTION DESCRIPTION

What is a Functional Method in Java?

CANDIDATE ANSWER

Options: (Expected answer indicated with a tick)



☒ The single Abstract Method of a Functional Interface

☐

Any Method of a Functional Interface

☐


Any Abstract Method of an Interface

☐

The single Non-Abstract Method of a Functional Interface

No Comments

QUESTION 10



Correct Answer

Score 5

Which of the following... > Multiple Choice 


Java

QUESTION DESCRIPTION

Which of the following is not a valid declaration of a lambda expression?

CANDIDATE ANSWER

Options: (Expected answer indicated with a tick)



☒

MyFunctionalInterface fi0 = sum(int a, int b) { return a + b;};

☐

MyFunctionalInterface fi1 = (int a, int b) ->{ return a + b;};

☐


MyFunctionalInterface fi2 = (int a, int b) -> a + b;

☐

MyFunctionalInterface fi3 = (a, b) -> a + b;

No Comments

QUESTION 11



Correct Answer

Score 5

Streams > Multiple Choice 


Java

QUESTION DESCRIPTION

Which of the following is/are true about Streams in Java?

CANDIDATE ANSWER

Options: (Expected answer indicated with a tick)



☒

A Stream is not a data type or data structure by itself, instead, it takes input from the Collections, Arrays or I/O channels.

☐

A Stream of the same data type or data structure of the Collections, Arrays or I/O channels that it takes input from.

☐

Changes made on Stream reflect on the original source when the stream operations are completed.

☐

None of the above

No Comments

**QUESTION 12**

Correct Answer

Score 4.29

**Intermediate Operations** > Multiple Choice

Java

**QUESTION DESCRIPTION**

Which of the following are examples of Intermediate Operations?

**CANDIDATE ANSWER****Options:** (Expected answer indicated with a tick)

- ☒ ☒ map()
- ☒ ☒ filter()
- ☒ ☐ unsorted()
- ☒ ☒ peek()
- ☒ ☒ sorted()
- ☒ ☒ distinct()
- ☒ ☒ limit()
- ☐ sum()
- ☐ min()
- ☐ max()
- ☐ count()
- ☐ average()
- ☐ collect()
- ☐ reduce()

No Comments



**QUESTION 13**

Correct Answer

Score 5

skip() > Multiple Choice Java**QUESTION DESCRIPTION**

What is the most likely output of this program?

```
public class Main {  
  
    public static void main(String[] args) {  
  
        Integer arr[] = {1,3,5,7,9,2,4,6,8};  
  
        Arrays.stream(arr).skip(2).sorted().skip(3)  
  
            .forEach(x -> System.out.println(x));  
  
    }  
  
}
```

**CANDIDATE ANSWER****Options:** (Expected answer indicated with a tick)

- ☒ 6  
☐ 7  
☐ 8  
☐ 9  
  
☐ 2  
☐ 4  
☐ 6  
☐ 8  
  
☐ This code could not compile  
☐ ArrayIndexOutOfBoundsException

No Comments

## QUESTION 14



Correct Answer

Score 5

## StringJoiner merge() &gt; Multiple Choice Java

## QUESTION DESCRIPTION

What is the most likely output of this program?

```
public class Main {  
  
    public static void main(String[] args) {  
  
        StringJoiner sj1 = new StringJoiner(":", "{", "}");  
        StringJoiner sj2 = new StringJoiner(",", "[", "]");  
        sj1.add("Name").add("Tony");  
        sj2.add("LastName").add("Strak");  
        System.out.println(sj1.merge(sj2).toString());  
  
    }  
}
```

## CANDIDATE ANSWER

Options: (Expected answer indicated with a tick)

- ☒ ☐ {Name:Tony:LastName,Strak}
- ☐ {Name:Tony:[LastName,Strak]}
- ☐ Unresolved compilation problem: The method merge(":", "{",}") in the type StringJoiner is not applicable for the arguments ("", "[", "]")
- ☐ {[LastName,Strak,Name:Tony]}

No Comments

**QUESTION 15**

Correct Answer

Score 5

**What is a stream...** > Multiple Choice Java**QUESTION DESCRIPTION**

What is a stream in Java?

**CANDIDATE ANSWER****Options:** (Expected answer indicated with a tick)

A stream is a sequence of elements that supports various methods for sequential and parallel aggregate operations



A Stream is a framework that provides an architecture to store and manipulate the group of objects.



A Stream is a container object that holds a fixed number of values of a single type.



A Stream is a sequence of characters that exist as a single object.

No Comments

**QUESTION 16**

Correct Answer

Score 5

**Is main a Thread?** > Multiple Choice Java**QUESTION DESCRIPTION**

What is the most likely output of this program? Is main a Thread?

```
public class Main {  
  
    public static void main(String[] args) {  
  
        System.out.println(Thread.currentThread().getName());  
  
    }  
  
}
```

**CANDIDATE ANSWER****Options:** (Expected answer indicated with a tick)

main



Thread-0



java.lang.Thread



java.lang.IllegalStateException

No Comments

**QUESTION 17**

Correct Answer

Score 5

**getState()** > Multiple Choice Java**QUESTION DESCRIPTION**

What is the most likely output of this program?

```
public class Main extends Thread {  
  
    public static void main(String[] args) {  
  
        System.out.println(Thread.currentThread().getState());  
  
    }  
  
}
```

**CANDIDATE ANSWER****Options:** (Expected answer indicated with a tick)

RUNNABLE



NEW



TERMINATED



Unresolved compilation problem: The method getState() is undefined for the type Thread

No Comments

## QUESTION 18



Correct Answer

Score 5

sleep() from MyThread vs sleep() from another Thread > Multiple Choice Java

## QUESTION DESCRIPTION

What is the most likely output of this program?

```
class MyThread extends Thread {  
  
    public void run() {  
  
        try {  
  
            MyThread.sleep(2000);  
  
            System.out.println("I took a long nap");  
  
        } catch (InterruptedException e) {  
  
        }  
  
    }  
  
}  
  
public class Main {  
  
    public static void main(String[] args){  
  
        System.out.println("I am awake");  
  
        new MyThread().start();  
  
        System.out.println("And I feel alive");  
  
    }  
  
}
```

## CANDIDATE ANSWER

Options: (Expected answer indicated with a tick)

- ☒ ☐ I am awake And I feel alive I took a long nap
- ☐ I am awake I took a long nap And I feel alive
- ☐ Unresolved compilation problem: The method Sleep(int) is undefined for the type MyThread
- ☐ I am awake And I feel alive

No Comments

**QUESTION 19**

Correct Answer

Score 5

**sleep() vs wait()** > Multiple Choice Java**QUESTION DESCRIPTION**

In Java, what are some differences between the Thread.sleep() and Thread.wait()?

**CANDIDATE ANSWER**

**Options:** (Expected answer indicated with a tick)

- ☒ ☐ sleep() is used for time-synchronization whereas wait() is used for multi-thread-synchronization.
- ☒ ☐ wait() is called on an object and the current thread must synchronize on that same object. sleep() is always called on the currently executing thread.
- ☐ wait() is used for time-synchronization whereas sleep() is used for multi-thread-synchronization.
- ☐ sleep() is called on an object and the current thread must synchronize on that same object. wait() is always called on the currently executing thread.

No Comments

**QUESTION 20**

Correct Answer

Score 5

**Use Multithreading** > Multiple Choice Java**QUESTION DESCRIPTION**

In Java, what do we need to do to use multithreading in our programs?

**CANDIDATE ANSWER**

**Options:** (Expected answer indicated with a tick)

- ☒ ☐ By Extending the Thread class
- ☒ ☐ By implementing the Runnable Interface
- ☐ By running your program multiple time
- ☐ Nothing, multithreading is automatic in Java

No Comments