

EXTENDED OFFER | Flat 15% OFF SITEWIDE | Use Coupon - WHIZOFFER15

[Home](#) > [My Courses](#) > [OCAJP 8 Training Course -1Z0 808](#) >[Quiz : Working with Selected classes from the Java API](#) > **Report**

Search Courses

**Quiz : Working with Selected classes from the Java API****Attempt**

1

**Completed on**

Wednesday , 04 September 2019 , 06:37 PM

**Marks Obtained**

0 / 20

**Time Taken**

00 H 00 M 05 S

**Your score is**

0.0%

**Result**

Fail

**Mode**

Exam

**Domains wise Quiz Performance Report**

No	1
Domain	Other
Total Question	20
Correct	0
Incorrect	0
Unattempted	20
Marked for review	0
Total	Total
All Domain	All Domain
Total Question	20
Correct	0
Incorrect	0
Unattempted	20
Marked for review	0

## Review the Answers

Sorting by

All

### Question 1


Unattempted

Domain : Other

Given :

```
1 public class Whiz {  
2     public static void main(String [] args) {  
3         StringBuilder sb = new StringBuilder(10);  
4         StringBuilder sb1 = new StringBuilder("ABCD");  
5         System.out.println(sb.capacity()+sb1.capacity());  
6     }  
7 }
```

What is the result?

- A. 14
- B. 20
- C. 26
- D. 30 
- E. Compilation fails

### Explanation:

Explanation:

Option D is the correct answer.

At line 3 we have passed integer 10 as the constructor argument when creating the string builder which will result a string builder with initial capacity of 10.

At line 4 we have created another string builder by passing string with 4 characters. In such cases capacity is calculated by java by adding 16 to passed string length, hence here capacity will be 20.

At line 5, 30 will be printed since the string builders have the capacities of 10 and 20. Hence option D is correct.

**REFERENCE :** <http://docs.oracle.com/javase/7/docs/api/java/lang/StringBuilder.html>

Ask our Experts

Rate this Question? 😊 😞

Question 2

Unattempted

Domain : Other

Given

```
1    import java.util.ArrayList;
2    import java.util.List;
3
4    public class Whiz{
5        public static void main(String[] args){
6            List<Integer> ints = new ArrayList<>();
7            ints.add(1);
8            ints.add(3);
9            ints.add(4);
10           ints.add(2,3);
11           ints.removeIf(e -> e < 3);
12           System.out.println(ints);
13       }
14   }
```

What is the output?

A. [1, 3, 3, 4]

B. [3, 3, 4] ✓

- C. [3, 4]
- D. [1]
- E. Compilation fails

---

**Explanation:****Explanation:**

Option B is the correct answer.

The removeIf method will remove all elements where elements match the given predicate, so here it will remove elements which are less than 3.

At line 10, we insert value 3 to index position 2. So ints will initially contain 1,3,3 and 4. Since only the 1 matches the given predicate at line 11, it will be removed, so option B is correct.

**REFERENCE**

:

<https://docs.oracle.com/javase/8/docs/api/java/util/function/Predicate.html>

---

Ask our Experts


Rate this Question? 😊 😞

Question 3

Unattempted

Domain : Other

Which of the following is the functional method of java.util.function.Predicate<T> interface?

- A. Boolean accept(T t)
- B. Boolean apply(T t)
- C. T get()
- D. boolean test(T t) 
- E. None of above.

---

**Explanation:**

**Explanation :**

Option D is the correct answer.

The Predicate represents a predicate (boolean-valued function) of one argument. Its functional method is

`boolean test(T t)`

This method evaluates this predicate on the given argument. So option D is correct

**REFERENCE :**

<https://docs.oracle.com/javase/8/docs/api/java/util/function/Predicate.html>


---

Ask our Experts

Rate this Question? 😊 😞

**Question 4****Unattempted****Domain : Other**

A Functional interface should :

- A. Contain a static method.
- B. Contain one or more abstract methods.
- C. Contain only one abstract method. 
- D. Not contain abstract methods.
- E. Contain the `@FunctionalInterface` annotation.

**Explanation:****Explanation :**

Option C is the correct answer.

A functional interface should have only one abstract method so option C is correct.

It is optional to include default or static method in functional interface, so option A is incorrect.

It is optional to have `@FunctionalInterface` annotation, for a functional interface. So option E is incorrect.

REFERENCE : <https://docs.oracle.com/javase/8/docs/api/java/util/function/Predicate.html>

<https://docs.oracle.com/javase/tutorial/java/javaOO/lambdaexpressions.html#approach5>

Ask our Experts

Rate this Question? 😊 😞

### Question 5


Unattempted

Domain : Other

Given :

```
1  import java.util.ArrayList;
2  import java.util.List;
3
4  public class Whizlab {
5
6      public static void main(String args[]){
7
8          List<String> strings = new ArrayList<>();
9
10         strings.add("A");
11         strings.add("B");
12         strings.add("C");
13         String[] arr = new String[2];
14         arr = strings.toArray(arr);
15         for(String s : arr){
16             System.out.print(s);
17         }
18     }
19 }
```

What is the output?

- A. AB
- B. ABC 
- C. nullnull
- D. An Exception is thrown.
- E. Compilation fails.

---

**Explanation:**

Explanation :

Option B is the correct answer.

The ArrayList class has following method which we can use to convert list to an array of correct type.

```
public<T> T[] toArray(T[] a)
```

Returns an array containing all of the elements in this list in proper sequence (from first to last element); the runtime type of the returned array is that of the specified array. If the list fits in the specified array, it is returned therein. Otherwise, a new array is allocated with the runtime type of the specified array and the size of this list.

If the list fits in the specified array with room to spare (i.e., the array has more elements than the list), the element in the array immediately following the end of the collection is set to null. (This is useful in determining the length of the list only if the caller knows that the list does not contain any null elements.)

So here a new array with all elements will be returned and it will be assigned to the arr reference, so option B is correct.

**REFERENCE :** <https://docs.oracle.com/javase/8/docs/api/java/util/ArrayList.html>

---

Ask our Experts

Rate this Question?  

Question 6


Unattempted

Domain : Other

Given :

```
1  import java.util.ArrayList;
2  import java.util.List;
3
4  public class Whizlab{
5
6      public static void main(String[] args){
7
8          List list = new ArrayList<>();
9
10         list.add(5);
11
12         list.add(5);
13
14         System.out.println(Integer.max(list.get(0),1));
15
16     }
17 }
```

What is the output?

- A. true
- B. 10
- C. 1
- D. An Exception is thrown.
- E. Compilation fails.. 

**Explanation:**

Explanation :

Option E is the correct answer.

Here we haven't given any generic type for the array list, so get method will return object and also we can add any sort of object. Here at line 10 trying to invoke max method of the Integer class by passing object (list.get(0)) causes a compile time error, So option E is correct.



REFERENCE : <https://docs.oracle.com/javase/8/docs/api/java/util/ArrayList.html>

---

Ask our Experts

Rate this Question? 😊 😞

Question 7

Unattempted

Domain : Other

Given

```
LocalDate ld = LocalDate.of(2016, 3, 21);
```

```
_____ lt = ld.atTime(10,10);
```

Which of the following can be used to fill the blank?

A. Instant

B. LocalDateTime 

C. LocalDate

D. LocalTime

E. Date

---

**Explanation:**

Explanation :

Option B is the correct answer.

First here we have LocalDate instance and then invoking the atTime will combine passed time with LocalDate value, and result will be LocalDateTime object, so we have to have LocalDateTime reference, hence option B is correct.

REFERENCE : <https://docs.oracle.com/javase/8/docs/api/java/time/LocalDate.html>

---

Ask our Experts

Rate this Question? 😊 😞

Question 8


Unattempted

Domain : Other

Given :

```
1  import java.time.LocalDate;
2
3  public class Whizlab{
4
5      public static void main(String[] args) {
6
7          LocalDate ld = LocalDate.of(2015, 2, 27).plusDays(3);
8
9          System.out.println(ld.getMonth());
10
11     }
12 }
```

What is the result?

- A. 2
- B. 3
- C. MARCH 
- D. An Exception.
- E. Compilation fails.

---

### Explanation:

Explanation :

Option C is the correct answer.

At line 5 we have created a LocalDate with date 2015/2/27, and then we have invoked the plusDays method by passing 3 so finally ld will contain 2015/3/2 as the date. At line 6 invoking the getMonth method will result the enum value of the month not the integer value of the month so finally 'MARCH' will be printed. Hence option C is correct.

**Reference** : <https://docs.oracle.com/javase/8/docs/api/java/time/LocalDate.html>

---

Ask our Experts

Rate this Question?  

## Question 9


Unattempted

Domain : Other

Given :

```
1  import java.time.Period;
2
3  public class Whizlab{
4
5      public static void main(String[] args) {
6
7          Period p = Period.ofMonths(13);
8
9          System.out.println(p.normalized());
10
11     }
12 }
```

What is the output?

- A. P1Y1M 
- B. P13M
- C. P1Y
- D. An Exception.
- E. Compilation fails.

**Explanation:**

Explanation :

Option A is the correct answer.

At line 5 we have created a period of 13 months. Java knows that 12 months means 1 year, so java can identify that period consist of 1 year and 1 month. For such a conversion we can use the normalized method which normalizes the years and months units, leaving the days unit unchanged. The months unit is adjusted to have an absolute value less than 12, with the years unit being adjusted to compensate. For example, a period of "1 Year and 15 months" will be normalized to "2 years and 3 months". So in this case output will be P1Y1M. Hence option A is correct.

**Reference** : <https://docs.oracle.com/javase/8/docs/api/java/time/Period.html>

[Ask our Experts](#)

Rate this Question? 😊 😞

## Question 10

Unattempted

Domain : Other

Given :

```
1  class Whizlab {  
2      public static void main(String [] args) {  
3          String s = "Whiz";  
4          s.concat("Lab");  
5          s.toLowerCase();  
6          System.out.print(s);  
7          //how many?  
8      }  
9  }
```

How many String objects are created when code reaches to line 7?

- A. 1
- B. 2
- C. 3
- D. 4 ☒
- E. 5

**Explanation:**

Explanation :

Option D is the correct answer.

Line 3 creates a new String object and gives it the value "Whiz". At line 4, two String objects are created ("Lab" and "WhizLab"). At line 5, a new String object is created ("whiz"). So four objects are created hence option D is correct and others are incorrect.

**REFERENCE** : <http://docs.oracle.com/javase/tutorial/java/data/strings.html>

---

Ask our Experts

Rate this Question? 😊 😞

Question 11


Unattempted

Domain : Other

Given :

```
1 class Whizlab {  
2     public static void main(String args[]) {  
3         StringBuilder sb = new StringBuilder("Whizlab");  
4         char[] ch = new char[4];  
5         sb.getChars(1, 5, ch, 1);  
6         for(char c : ch) System.out.print(c);  
7     }  
8 }
```

What is the result?

- A. hizl
- B. hizl
- C. izla
- D. An exception. 
- E. Compilation fails.

---

**Explanation:**

**Explanation :**

**Option D is the correct answer.**

```
public void getChars(int srcBegin,int srcEnd,char[] dst,int dstBegin);
```

The getChars method copy characters from this sequence into the destination character array dst. The first character to be copied is at index srcBegin; the last character to be copied is at index srcEnd-1.

The total number of characters to be copied is srcEnd-srcBegin. The characters are copied into the subarray of dst starting at index dstBegin and ending at index:

$\text{dstbegin} + (\text{srcEnd} - \text{srcBegin}) - 1$

So here at line 5 we have given 1 as the start point and 5 as the end point, so there will be 4 chars extracted which is the number of elements that can be hold in the array created at line 4, so adding one offset makes it 5 elements which results an `ArrayIndexOutOfBoundsException`. Hence option D is correct.

**REFERENCE :** <http://docs.oracle.com/javase/7/docs/api/java/lang/StringBuilder.html>

<http://docs.oracle.com/javase/tutorial/java/data/buffers.html>

---

Ask our Experts

Rate this Question? 😊 😞

Question 12


Unattempted

Domain : Other

Given :

```
1 public class Whiz {  
2     public static void main(String [ ] args) {  
3         long l = 1;  
4         int i = 0;  
5         StringBuilder sb = new StringBuilder("10");  
6         sb.insert(i, l);  
7         System.out.println(sb.toString());  
8     }  
9 }
```

What is the result?

- A. 10
- B. 110 
- C. 100
- D. An exception
- E. Compilation fails

---

**Explanation:**

**Explanation:**

**Option B is the correct answer.**

At line 5 we have created StringBuilder instance by passing "10". Then at line 6, we have used one of the overloaded versions of the insert method of the StringBuilder

```
public StringBuilder insert(int offset,long l)
```

It inserts the string representation of the long argument into this sequence.

The overall effect is exactly as if the second argument were converted to a string by the method `String.valueOf(long)`, and the characters of that string were then inserted into this character sequence at the indicated offset. The offset argument must be greater than or equal to 0, and less than or equal to the length of this sequence.

Here we have passed the 0 as offset, which is the value of the variable i, and then we have passed long value 1, so 1 will be inserted into sb at the indicated offset 0, hence option B is correct.

**REFERENCE :** <http://docs.oracle.com/javase/7/docs/api/java/lang/StringBuilder.html>

<http://docs.oracle.com/javase/tutorial/java/data/buffers.html>

---

Ask our Experts

Rate this Question?  

Question 13


Unattempted

Domain : Other

**Given**

```
1  class Whizlab {  
2  
3      public static void main(String args[]){  
4  
5          StringBuilder sb = new StringBuilder("Whizlab");  
6          int x = sb.capacity();  
7          sb.trimToSize();  
8          int y =sb.capacity();  
9          System.out.print(x*y);  
10     }  
11 }
```

**What is the output?**

- A. 32
- B. 46
- C. 30 
- D. 39
- E. Compilation fails.

**Explanation:**

Explanation :

Option C is the correct answer.

Default capacity of the stringbuilder is length of the current content plus 16. So here when we invoke the method capacity at line 5 it will assign 23 to x. Then at line 6 we invoke the trimToSize method which result the capacity to change 7 which is the length of the current content. Hence option C is correct.

**REFERENCE**

: <http://docs.oracle.com/javase/7/docs/api/java/lang/StringBuilder.html>

<http://docs.oracle.com/javase/tutorial/java/data/buffers.html>



---

Ask our Experts


Rate this Question? 😊 😞

Question 14

Unattempted

Domain : Other

Which of the following will create valid string instance?

- A. `new String();`
- B. `new String(new byte[]{88,89});`
- C. `new String(new char[]{'1','2','3'});`
- D. `new String(new StringBuilder("ABCD"));`
- E. All of above. 

---

**Explanation:**

Explanation:

**Option E is the correct answer**

Given all are valid uses of String class constructors. So option E is correct.

`String()` : Initializes a newly created String object so that it represents an empty character sequence.

`String(byte[] bytes)` : Constructs a new String by decoding the specified array of bytes using the platform's default charset.

`String(StringBuilder builder)` : Allocates a new string that contains the sequence of characters currently contained in the string builder argument.

`String(char[] value)` : Allocates a new String so that it represents the sequence of characters currently contained in the character array argument.

**REFERENCE** : <http://docs.oracle.com/javase/tutorial/java/data/strings.html>

---

Ask our Experts

Rate this Question? 😊 😞

## Question 15

Unattempted

Domain : Other

Given :

```
1 public class Whiz {  
2     public static void main(String [ ] args) {  
3         String s = String.join("-", "1", "1");  
4         System.out.println(s);  
5     }  
6 }
```

What is the output?

- A. 1-1 ☒
- B. -11
- C. 11
- D. o
- E. Compilation fails due to line 3

**Explanation:**

Explanation:

Option A is the correct answer

From java Se 8, we have new method called join in String class.

```
public static String join(CharSequence delimiter,CharSequence... elements)
```

Returns a new String composed of copies of the CharSequence elements joined together with a copy of the specified delimiter.

For example,

```
String message = String.join("-", "Java", "is", "cool");
```

```
// message returned is: "Java-is-cool"
```

So here two "1" will be joined by "-", so output will be "1-1". Hence option A is correct.

**REFERENCE :** <http://docs.oracle.com/javase/tutorial/java/data/strings.html>

Ask our Experts

Rate this Question? 😊 😞

### Question 16

Unattempted

Domain : Other

Given

```
1  import java.time.LocalDate;
2  import java.time.LocalTime;
3
4  public class Whiz {
5      public static void main(String [ ] args) {
6          LocalDate date = LocalDate.of(2015,12,12);
7          LocalTime time = date.atTime(10, 22);
8          System.out.println(date.getDayOfWeek() + " > " + time.getHour()+":"+time.getMinute());
9      }
10 }
```

What is the result?

- A. SATURDAY > 10:22
- B. 6 > 10:22
- C. A DateTimeException

## D. Compilation fails

**Explanation:**

Explanation:

Option D is the correct answer.

The code fails to compile since at line 7, invoking the `atTime` method on date instance will result a `LocalDateTime` instance not and `LocalTime` instance so trying to assign it to a `LocalTime` result a compile time error. So option D is correct.

**REFERENCE :**

<https://docs.oracle.com/javase/8/docs/api/java/time/LocalDateTime.html>

Ask our Experts

Rate this Question? 😊 😞

**Question 17**


Unattempted

Domain : Other

Given :

```
1  import java.time.LocalDate;
2  import java.time.MonthDay;
3  import java.time.Year;
4
5  public class Whiz {
6      public static void main(String[] args) {
7          Year y = Year.of(2015);
8          LocalDate date = y.atMonthDay(MonthDay.of(4,31));
9          System.out.println(date);
10     }
11 }
```

What is the output?

- A. 2015-03-31
- B. 2014-05-01
- C. 2014-04-31
- D. An Exception is thrown 
- E. Compilation fails

---

**Explanation:**

Explanation:

Option D is the correct answer.

At line 7, we have created an year instance by passing 2015 as the year. Then at line 8 we have tried to get local date instance by invoking the atMonthDay method of the year class.

We have passed 4 as the month and 31 as the day arguments for the of method which causes an exception because month April doesn't have such a day. Hence option D is correct.

**REFERENCE** : <https://docs.oracle.com/javase/8/docs/api/java/time/LocalTime.html>

---

Ask our Experts

Rate this Question?  

Question 18

Unattempted


Domain : Other

Given

```
1    import java.util.*;
2
3    class Whiz{
4
5        public static void main(String [] args){
6
7            ArrayList<String> alist = new ArrayList<String>();
8
9            alist.add("A");
10
11           alist.add("B");
```

```
7.         alist.add("A");
8.         alist.add("C");
9.         alist.add("B");
10.        System.out.println(alist);
11.        Collections.sort(alist);
12.        System.out.println(alist);
13.    }
14. }
```

What is the result?

- A. [A,B,C] [A,B,C]
- B. [A,B,A,C,B] [A,A,B,B,C] 
- C. [A,B,A,C,B] [A,B,A,C,B]
- D. [A,B,A,C,B] [A,B,C]
- E. Compilation fails

---

#### Explanation:

Explanation:

Option B is the correct answer.

Here we have used the method called sort from java.util.Collections class. The sort method has sorted them alphabetically. Line 10 will print unsorted list according to the order we added elements. After sorting them they will be printed by the print method at line 12. So the answer is B.

Option A is incorrect as with Lists we can keep duplicate elements.

Option C is incorrect as the sort method of the "java.util.Collections" class sorted elements.

There is no problem in this code, it compiles fine so E is incorrect.

**Reference** : <http://docs.oracle.com/javase/7/docs/api/java/util/ArrayList.html>

Ask our Experts

Rate this Question? 😊 😞

## Question 19


Unattempted

Domain : Other

Given :

```
1      import java.util.*;
2
3      class Whizl
4
5          public static void main(String [] args){
6
7              List lst = new ArrayList(1);
8
9              lst.add(1);
10             lst.add("A");
11             lst.add(new Integer(5));
12
13             System.out.print(lst);
14
15         }
16     }
```

Which is true?

- A. Compilation fails due to error on line 4
- B. Compilation fails due to multiple errors
- C. Compilation succeeds without any warning
- D. An Exception is thrown at the runtime
- E. The output will be [1, A, 5] 

**Explanation:**

Explanation:

Option E is the correct answer.

Option E is correct as the code compiles and runs without any exception and prints [1, A, 5].

Option C is incorrect since we have not specified what can List lst hold, so it can take any kind of object except primitive. (This is called as the non-generic collection) The compiler will warn you about using the non-generic collections as there are risks by allowing anything to be put in to the collection.

Option D is incorrect. Not like arrays, with ArrayList size of the array can resize accordingly. So here even we have created an ArrayList for holding one element, it will dynamically change its size when we add more elements. So there won't be an ArrayIndexOutOfBoundsException.

**Reference :** <http://docs.oracle.com/javase/7/docs/api/java/util/ArrayList.html>

Ask our Experts


Rate this Question? 😊 😞

Question 20

Unattempted

Domain : Other

Which of following can be considered as a functional interface?

- A. `public interface Movable { static void print(){ } void move(); }` 
- B. `@FunctionalInterface public interface Movable { }`
- C. `@FunctionalInterface public interface Movable { default void move(); }`
- D. `public interface Movable { static void move(); }`

**Explanation:**

**Explanation:**

**Option A is the correct answer.**

Functional interfaces provide target types for lambda expressions and method references. Each functional interface has a single abstract method (SAM), called the functional method for that functional interface, to which the lambda expression's parameter and return types are matched or adapted. It can also include other methods such as static and default methods.

Also we should remember that interfaces can have default and/or static methods, but they should be non abstract , this is not just for functional interfaces, this applies for all java interfaces.



As explained above, option A satisfies requirements for the function interfaces.

All other answers are invalid interfaces.

**REFERENCE**

:

<https://docs.oracle.com/javase/8/docs/api/java/util/function/Predicate.html>

---

Ask our Experts

Rate this Question? 😊 😞

Finish Review

**Certification**

Cloud Certification

Java Certification

PM Certification

Big Data Certification

**Company**

Support

Discussions

Blog

**Follow us**

---

© Copyright 2019. Whizlabs Software Pvt. Ltd. All Right Reserved.