🏠 > My Courses > OCAJP 8 Training Course -1Z0 808 > Practice Test III > **Report**

Search Courses 🔍

Practice Test III

Attempt	**Completed on**
1	Wednesday , 04 September 2019 , 06:42 PM
Marks Obtained	**Time Taken**
0 / 70	00 H 00 M 24 S
Your score is	**Result**
0.0%	Fail
Mode	
Exam	

Domains wise Quiz Performance Report

No	1
Domain	Creating and Using Arrays
Total Question	10
Correct	0
Incorrect	0
Unattempted	10
Marked for review	0

No	2
Domain	Handling Exceptions
Total Question	7
Correct	0
Incorrect	0
Unattempted	7
Marked for review	0
No	3
Domain	Java Basics
Total Question	8
Correct	0
Incorrect	0
Unattempted	8
Marked for review	0
No	4
Domain	Working with Inheritance
Total Question	10
Correct	0
Incorrect	0
Unattempted	10
Marked for review	0
No	5
Domain	Working with Methods and Encapsulation
Total Question	11
Correct	0
Incorrect	0
Unattempted	11
Marked for review	0

No	6
Domain	Using Loop Constructs
Total Question	4
Correct	0
Incorrect	0
Unattempted	4
Marked for review	0
No	7
Domain	Using Operators and Decision Constructs
Total Question	9
Correct	0
Incorrect	0
Unattempted	9
Marked for review	0
No	8
Domain	Working with Selected classes from the Java API
Total Question	10
Correct	0
Incorrect	0
Unattempted	10
Marked for review	0
No	9
Domain	Working with Methods and Encapsulation
Total Question	1
Correct	0
Incorrect	0
Unattempted	1
Marked for review	0
Total	Total
All Domain	All Domain
Total Question	70
Correct	0
Incorrect	0
Unattempted	70
Marked for review	0

Review the Answers

Sorting by

All

Question 1   Unattempted

Domain :Creating and Using Arrays

Which of the following is valid array initialization ?

     A.   int [ ] array [ ] = new int [ ][ ];

     B.   int [ ][ ] two = new int [ ][5];

     C.   int #one [ ] = new int [5];

     D.   int [3][ ] three = new int [ ][ ];

     E.   int [ ] one = new int [6];   ✅

Explanation:

Option E is the correct answer.

You must include the size of an array when you construct it (using new) if it is not an anonymous array. It is never legal to include the size of an array in the declaration. So, option D is incorrect. Options A and B are incorrect since we haven't given the size for the first dimension.

Option C is incorrect since we can't use # for the variable names.

Reference : http://docs.oracle.com/javase/tutorial/java/nutsandbolts/arrays.html

---

Ask our Experts

Rate this Question? 🙂 🙁

Question 2   Unattempted

Domain :Creating and Using Arrays

What will be the output of this program?

    1.   class Whiz {

    2.   public static void main(String[] args) {

3.   int a[ ] [] = { {1,2,4} , {5,2,1},{0,43,2}};

4.   int b[ ] = a[2];

5.   System.out.print(b[1]);

6.   }

7. }

A.   4

B.   5

C.   43   ✅

D.   2

E.   Compilation fails due to an error at line 4

Explanation:

Explanation:
Option C is the correct answer.

At line 3, we created a two-dimensional array with 3 rows. At line 4, we assigned the third row of the two-dimensional array to the one-dimensional int array b. Note that the array indexes are starting with 0 so 2 means the third row above. At line 5, the output would be the 2nd element of the array b, which is 43 hence option C is correct.

Reference : http://docs.oracle.com/javase/tutorial/java/nutsandbolts/arrays.html

Ask our Experts
Rate this Question? 🙂 🙁

Question 3   Unattempted

                                                                    Domain :Creating and Using Arrays

What will be the output of this program code?

1.   class Whiz {

2.   public static void main(String[] args) {

3.   int a[ ] = { 1,2,053,4};

4.   int b[ ][ ] = { {1,2,4} , {2,2,1},{0,43,2}};

5.   System.out.print(a[3]==b[0][2]);

6.   System.out.print(" " + (a[2]==b[2][1]));

7.   }

8. }

A.   true false

B.   false false

C.   false true

D.   true true ✓

E.   Compilation fails

Explanation:

Explanation:
Option D is the correct answer.

Indexing of array elements begins with zero. So, index position one refers to the second element in an array. So, here a[3] refers to the fourth element of array a. and we have assigned an octal value to a[2] so the value of the element is 43 in decimals. Henc,e both lines 5,6 print true so option D is correct.

Reference : http://docs.oracle.com/javase/tutorial/java/nutsandbolts/arrays.html

Ask our Experts

Rate this Question? ☺ ☹

Question 4   Unattempted

Domain :Creating and Using Arrays

What will be the output of this program code?

```
1.   class Whiz {

2.         public static void main(String[] args) {

3.               int a[ ][ ] = new int[3][];

4.                a[1] = new int[]{1,2,3};

5.                a[2] = new int[]{4,5};

6.               System.out.print(a[1][2]);

7.         }

8.   }
```

A. **2**

B. **3** ✅

C. **Exception will be thrown at runtime.**

D. **Compilation fails due to an error at line 3.**

E. **Compilation fails due to an error at line 4.**

Explanation:

Explanation:
Option B is the correct answer.

At line 3, we created a two-dimensional array with the first dimension as 3 so there can be three rows, then at lines 4 and 5, we assigned two anonymous one-dimensional arrays to last two rows of the two-dimensional array. At line 6, we tried to print the second-row third element which is 3. Note here we haven't given any value to the first row of the two-dimensional array. So, option B is correct.

Reference : http://docs.oracle.com/javase/tutorial/java/nutsandbolts/arrays.html

Ask our Experts

Rate this Question? 🙂 🙁

Question 5   Unattempted

Domain :Creating and Using Arrays

What will be the output of this program code?

```
1.      import java.util.Arrays;

2.

3.      public class Whiz {

4.          public static void main(String[] args) {

5.              int [ ][ ] ints = new int[2][];

6.

7.              Arrays.sort(ints[1]);

8.

9.              System.out.print(Arrays.toString(ints[1]));

10.         }

11.     }
```

A. [0, 0, 0]

B. [null,null,null]

C. null

D. NullPointerException ✅

E. Compilation fails

Explanation:

Explanation:

Option D is the correct answer.

Methods in the Arrays class throw a NullPointerException, if the specified array reference is null. Here we haven't initialized the ints[1], so it is null. Hence passing it to Arrays sort method throws a NullPointerException and option D is correct.

Reference : http://docs.oracle.com/javase/tutorial/java/nutsandbolts/arrays.html

---

Ask our Experts

Rate this Question? 😊 ☹️

Question 6   Unattempted

Domain :Creating and Using Arrays

What will be the output of this program code?

```
1.      public class Whiz{

2.        public static void main(String[] args) {

3.            long size = 0xB;

4.            int []ints = new int[size];

5.

6.            ints[5] = 10;

7.

8.            System.out.print(ints[5] + 10);

9.        }

10.   }
```

A.   **20**

B.   **15**

C.   **10**

D.   A NullPointerException is thrown.

E.   Compilation fails. ✅

Explanation:

Explanation:

Option E is the correct answer.

The code fails as we have passed long value instead int as the size of the array, which is illegal, so option E is correct.

Reference   :http://docs.oracle.com/javase/tutorial/java/nutsandbolts/arrays.html

---

Ask our Experts

Rate this Question?   ☺   ☹

Question 7   Unattempted

Domain :Creating and Using Arrays

What will be the output of this program code?

```
1.      import java.util.Arrays;
2.      public class Whiz{
3.          public static void main(String[] args) {
4.              int[ ] ints = {3,6,1,4,0};
5.              Arrays.sort(ints,0,4);
6.                  for(int i :  ints) {
7.                      System.out.print(i);
8.                  }
9.          }
10.    }
```

A.   01436

B.   01346

C.   13460   ✅

D.   An Exception is thrown.

E.   Compilation fails.

Explanation:

Explanation:

Option C is the correct answer.

One version of the sort method of Arrays class sorts the specified range of the array into ascending order. The range to be sorted extends from the index fromIndex, inclusive, to the index toIndex, exclusive.

public static void sort(int[] a,int fromIndex,int toIndex)

Here we have passed range 0 to 4, which means elements from first element (inclusive) to 4th element. So only they will be sorted i.e. 3,6,1,4 will be sorted and the final output is 13460. Hence, option C is correct.

Reference : http://docs.oracle.com/javase/tutorial/java/nutsandbolts/arrays.html

---

Ask our Experts

Rate this Question? ☺ ☹

Question 8   Unattempted

Domain :Handling Exceptions

Given the following set of classes:
class A extends Exception {}
class B extends A {}
class C extends B {}

What is the correct sequence of catch blocks for the following try block?
try {
 // codes
}

A. Catch Exception, A, B and C

B. Catch Exception, C, B, and A

C. Catch A, B, C, and Exception

D. Catch C, B, A and Exception ✓

E. None of these

Explanation:

Explanation:

Option D is the correct answer.

The derived most class should be caught first. The order of classes at the same hierarchy level is not important.

So, correct order should be C, B, A and Exception.

REFERENCE   :http://docs.oracle.com/javase/tutorial/essential/exceptions/catch.html

Ask our Experts

Rate this Question?   ☺   ☹

Question 9   Unattempted

Domain :Handling Exceptions

Which of the following error/exception is typically thrown by JVM?

- A.   **IllegalStateException**

- B.   **AssertionError**

- C.   **StackOverflowError**   ✓

- D.   **ArrayIndexOutOfBoundsException**   ✓

Explanation:

Option C and D are the correct answer.

StackOverflowError, ArrayIndexOutOfBoundsException will be thrown by JVM. Hence Options C, D are correct. Options A, B are incorrect because IllegalStateException, AssertionError can be thrown programmatically .

REFERENCE :

https://coderanch.com/t/252956/certification/JVM-Exception-Programmatic-Exception

Ask our Experts

Rate this Question?   ☺   ☹

Domain :Handling Exceptions

What will be the output of this program code?

```
1.   class Whiz {
2.        public static void main(String [] args)throws Exception {
3.             try {
4.                   System.out.print("try Started ");
5.                   throw new Exception();
6.             } catch(Exception e) {
7.                   throw e;
8.             } finally {
9.                   System.out.print("Finally Completed");
10.            }
11.       }
12.  }
```

A.   Compilation fails.

B.   try Started Finally Completed followed by an exception . ✅

C.   try Started followed by an exception Finally Completed .

D.   An exception is thrown at runtime without printing try Started or Finally Completed.

Explanation:

Explanation:

Option B is the correct answer.

If a method (or catch clause) does not handle an exception, the finally block is executed before the exception is propagated. Here catch clause could handle the exception thrown but in catch block it re-

throw the exception, hence it propagate. So, finally executes and then uncaught exception pass to the calling method. Hence, option B is correct.

REFERENCE   :http://docs.oracle.com/javase/tutorial/essential/exceptions/catchOrDeclare.html

---

Ask our Experts

Rate this Question?   ☺   ☹

Question 11   Unattempted

Domain :Handling Exceptions

Consider the following code:

1. class Whiz{

2. public static void main(String [] args){

3.   try{

4.   int i = Integer.parseInt(args[0]);

5.   System.out.print(i);

6.   }catch(ArrayIndexOutOfBoundsException e){

7.   System.out.println("Array Index Out Of Bounds Exception");

8.   }catch(NumberFormatException e){

9.   System.out.println("Number Format Exception");

10.   }

11. }

12. }

And the command-line invocations are

1.   java Whiz

2.   java Whiz 12

3.   java Whiz A

4.   java Whiz 12 24

Which of the following are true? *(Choose 3)*

A.   Invocation 1 will print Array Index Out Of Bounds Exception.   ✓

B.   Invocation 2 will print will print 12.   ✓

C.   Invocation 3 will print Array Index Out Of Bounds Exception.

D.   Invocation 3 will print Number Format Exception.   ✓

E.   Invocation 4 will print Array Index Out Of Bounds Exception.

Explanation:

Explanation:

Options A, B and D are the correct answers.

Integer.parseInt method is used to convert a String to an int. If the String is not a valid number then the method throws a NumberFormatException. ArrayIndexOutOfBoundsException occurs if you use a bad index value.

REFERENCE   :http://docs.oracle.com/javase/tutorial/essential/exceptions/catchOrDeclare.html

---

Ask our Experts

Rate this Question?   ☺   ☹

Question 12   Unattempted

Domain :Handling Exceptions

Which of the following is checked exception?

A.   ClassCastException

B.   NullPointerException

C.   ExceptionInInitializerError

D.   IllegalArgumentException

E.   None of the above.   ✓

Explanation:

Explanation:

Option E is the correct answer.

There are three kinds of exceptions in Java.
1. checked exception
2. error
3. runtime exception

1. Checked exception:
These are exceptional conditions that a well-written application should anticipate and recover from.
All exceptions are checked exceptions, except for those indicated by Error, RuntimeException, and their subclasses.

2. Error:
These are exceptional conditions that are external to the application, and that the application usually cannot anticipate or recover from. All direct or indirect subclasses of Error are errors. Ex : ExceptionInInitializerError etc

3. Runtime exception:
These are exceptional conditions that are internal to the application, and that the application usually cannot anticipate or recover from.All direct or indirect subclasses of RuntimeException are the runtime exceptions.
Ex : ClassCastException, NullPointerException, IllegalArgumentException, etc.

Errors and runtime exceptions are collectively known as unchecked exceptions.

So, option E is correct.

REFERENCE   :http://docs.oracle.com/javase/tutorial/essential/exceptions/catchOrDeclare.html

---

Ask our Experts

Rate this Question?   ☺   ☹

Question 13   Unattempted

                                                                          Domain :Handling Exceptions

Which of the following will be thrown by the JVM when a method calls itself too many times?

  A.   **NullPointerException**

B.   NumberFormatException

C.   IllegalArgumentException

D.   ArrayIndexOutOfBoundsException

E.   None of the above   ✅

Explanation:

Explanation:

Option E is the correct answer.

StackOverflowError is thrown by the JVM when a method calls itself too many times (this is called infinite recursion because the method typically calls itself without end). So ,option E is correct.

The NullPointerException is thrown by the JVM when there is a null reference where an object is required. So, option A is incorrect.

The NumberFormatException is thrown by the programmer when an attempt is made to convert a string to a numeric type but the string doesn't have an appropriate format. So option B is incorrect.

Option C is incorrect since the IllegalArgumentException is thrown by the programmer to indicate that a method has been passed an illegal or inappropriate argument.

The ArrayIndexOutOfBoundsException is thrown by the JVM when code uses an illegal index to access an array. Hence, option D is incorrect.

REFERENCE   : http://docs.oracle.com/javase/8/docs/api/java/lang/Exception.html

Ask our Experts

Rate this Question? ☺ ☹

Question 14   Unattempted

Domain :Handling Exceptions

What will be the output of this program code?

1.   **public class Whiz{**

2.   **public static void main(String[] args) {**

3.   **try {**

```
 4.                     print();
 5.                 }catch(Exception e){}
 6.            }
 7.
 8.          static void print() {
 9.              try {
10.                  throw new ClassCastException();
11.              }catch(ClassCastException e) {
12.                  System.out.print("A  ");
13.              }finally {
14.                  System.out.print("B ");
15.              } finally {
16.                  System.out.print("B 2");
17.              }
18.              System.out.print("OCAJP ");
19.          }
20.      }
```

A.   A B B2 OCAJP

B.   AB 2 OCAJP

C.   B OCAJP

D.   B B2

E.   B followed by a NullPointerException.

F.   Compilation fails. ✅

Explanation:

Explanation:

Option F is the correct answer.

While using finally block, we need to remember that, for each try, there could be zero or one finally block but not more than one, hence code fails to compile. However, it is possible to have more than one catch blocks per try.

REFERENCE   :http://docs.oracle.com/javase/tutorial/essential/exceptions/catch.html

---

Ask our Experts

Rate this Question? ☺ ☹

Question 15   **Unattempted**

Domain :Java Basics

What will be the output of this program code?

```
1.   class Whiz{
2.          int a=10;
3.          public static void main(String[] args) {
4.                  new Whiz().print();
5.          }
6.          public void print(){
7.                  int a = 8;
8.                  System.out.print(a + " ");
9.          }
10.  }
```

A.   **10**

B.   **18**

C.   8   ✓

D.   Compilation fails

Explanation:

Explanation:

Option C is the correct answer.

Inside a class method, when a local variable has the same name as one of the instance variables, the local variable shadows the instance variable inside the method block. So, the value of the "a" variable declared in the method print() can be seen as the output.

REFERENCE : http://docs.oracle.com/javase/tutorial/java/nutsandbolts/variables.html

---

Ask our Experts

Rate this Question? ☺ ☹

Question 16   Unattempted

                                                        Domain :Working with Inheritance

Which will compile successfully when inserted at line 3?

1.   abstract interface Movable{

2.   int x = 10;

3.   //insert code here

4.   void run();

5.   }

A.   private int x = 10;

B.   abstract int i = 5;

C.   final static float c = 6.0;

D.   final short s=10;   ✓

E.   None of the above.

Explanation:

Explanation:

Option D is the correct answer.

Option A is incorrect since at line 2 we already defined a variable calls 'x' so we can't have another variable with the same name in the same scope. Also, the private variables are not allowed in interfaces.

Option B is incorrect as an abstract modifier is not valid for a variable.

Option C is incorrect since float literal should be ended with 'f' when it has decimal points.

Option D is correct as we can assign int literal within the range of -32768 – 32767 to a short.

REFERNCE   : http://docs.oracle.com/javase/tutorial/java/nutsandbolts/variables.html

Ask our Experts

Rate this Question?   ☺   ☹

Question 17   Unattempted

Domain :Java Basics

Which of the following uses a fully qualified name in java?

A.   **class MyDate extends java.util.Date{ //statement; }**   ✓

B.   **import java.util.*; class MyDate extends Date{ //statement; }**

C.   **import java.util.Date.*; class MyDate extends Date{ //statement; }**

D.   **None of the above.**

Explanation:

Explanation :
Option A is the correct answer

A fully qualified means using the complete package details when accessing a Java class. For example, if you are using java.lang.String is a fully qualified name for the class String. Here "java.lang" is the package where String class is declared and implemented. Other options are using the wild card "*" to import the classes and not using the fully qualified name to access the classes.

So, in this case, option A uses fully qualified name for the date class.

REFERENCE : http://docs.oracle.com/javase/tutorial/java/package/packages.html

---

Ask our Experts

Rate this Question?   🙂 ☹️

Question 18   Unattempted

Domain :Java Basics

What will be the output of this program code?

```
1.      import java.util.*;

2.

3.      class Whiz {

4.              public static void main(String args[]){

5.                      System.out.println(new Date());

6.              }

7.      }
```

A.   **Prints today date**   ✅

B.   **An Exception**

C.   **Compile Error at line 1**

D.   **Compile Error at line 5.**

Explanation:

Explanation:

Option A is the correct answer.

Date class is present in Java since JDK 1.0. It is present in java.util package. It will contain current date, time, day name, time, and zone information. So, at line 5, statement prints current date, time, day name, time, and zone information.

REFERENCE : http://docs.oracle.com/javase/tutorial/java/package/packages.html

---

Ask our Experts

Rate this Question? ☺ ☹

Question 19   **Unattempted**

Domain :Java Basics

Which of the following import statement will be inserted at line 1?

1.   **// insert here**

2.

3.   **public class Whiz {**

4.   **public static void main(String args[ ]) {**

5.   **System.out.println(pow(5,5));**

6.   **}**

7.   **}**

A.   **import java.lang.Math.*;**

B.   **import static java.Math.*;**

C.   **import static java.lang.Math.pow;** ✅

D.   **import java.Math.pow;**

E.   **No import statement is needed.**

Explanation:

Explanation:

Option C is the correct answer.

In given code, we have used the static pow method of Math class directly, so we have to import statically that method or all static members of the class. Correct syntax to import pow method statically is;

import static java.lang.Math.pow;

So, option C is correct.

REFERNCE　　　:　http://docs.oracle.com/javase/tutorial/java/package/packages.html

Ask our Experts

Rate this Question?　☺　☹

Question 20　　　　　　　　　　　　　　　　　　　　**Unattempted**

Domain :Java Basics

A java source code file can have -

　A.　**Only two non-public classes.**

　B.　**Few public classes.**

　C.　**Few package statements.**

　D.　**Methods with same names in different classes**　✓

　E.　**None of the above.**

Explanation:

Explanation:

Option D is the correct answer.

We can have methods with same name and same scope. This is overloading of a method in the same class. So, option D is correct.

Java source can have any number of non-public classes, but only one public class. When there is a public class its name should be same as the file name. Only one package statement is allowed per source code file.

REFERNCE   :   http://docs.oracle.com/javase/tutorial/getStarted/application/index.html

---

Ask our Experts

Rate this Question?   🙂   🙁

Question 21   **Unattempted**

Domain :Java Basics

What will be the output of this program?

```
1.   class Whiz{

2.          public static void main(String[] args){

3.                  int n = 808;

4.                  System.out.format("%d", n);

5.                  System.out.format("%06d", n);

6.          }

7.   }
```

A.   808000808   ✅

B.   808000000808

C.   808808

D.   An Exception is thrown.

E.   Compilation fails.

Explanation:

Explanation:

Option A is the correct answer.

The question is related to Format output. The Sun micro Systems Inc. has introduced formatted output from J2SE5.0.(It is a compile Error for previous versions). The format method concatenates with the 0's (the number of zeros is a specified value in the format specified – number of digits on the variable). Here while the statement 4 prints n as normal, line 5 statement will make the number of digits 6 by filling leading zeros so option A is correct.

REFERNCE   : http://docs.oracle.com/javase/tutorial/getStarted/application/index.html

Ask our Experts

Rate this Question?   ☺   ☹

Question 22   **Unattempted**

Domain :Working with Methods and Encapsulation

What will be the output of this program?

```
1.     class Whiz {

2.

3.          static int i;

4.          int j;

5.          Whiz() { j=i++; }

6.

7.          public static void main(String args[]) {

8.                 Whiz s = new Whiz();

9.                 Whiz s1= new Whiz();

10.                Whiz s2= new Whiz();

11.                System.out.print( "i = "+s.i);

12.                System.out.print( ", j = "+s.j);

13.         }

14.    }
```

What is the result?

A.   i = 3, j = 3

B.   i = 1, j = 0

C.   i = 3, j = 0   ✓

D.   An Exception is thrown.

E.   Compilation fails.

Explanation:

Explanation:
Option C is the correct answer.

In the above snippet, the variable i is static and j is in non-static variables. A static variable is associated with the class has only one copy per class but not for each object, Whereas a Non-static variable will have one copy each per object. Each instance of a class will have one copy of non-static variables. Since the variable i is common for every instance it will get increment each time instance is created so the values of i is 3 and j is 0. (While creating an instance, i is incremented and retains its value)

REFERENCE   : http://docs.oracle.com/javase/tutorial/java/nutsandbolts/variables.html

Ask our Experts

Rate this Question? 🙂 🙁

Question 23   Unattempted

Domain :Working with Methods and Encapsulation

What will be the output of this program?

1.   class Whiz{

2.

3.   final int i;

4.

5.   public static void main(String args[]){

6.   Whiz s= new Whiz();

7.   System.out.println("i = "+s.i);

8.   }

9.   }

    A.   Prints i = 0

    B.   Prints i = 1

    C.   Prints i = Null

    D.   Runtime Exception is thrown.

    E.   Compile-time Error.   ✅

Explanation:

Explanation:

Option E is the correct answer.

The compiler complains "variable i has not been initialized in the default constructor". A final variable is equivalent to a constant entity in java. Once it is initialized it may not be changed. But, it has to be initialized at the time of declaration. It can be initialized in 3 ways.

A final variable is initialized at the time of declaration

final int i=100;

It can be initialized in the constructor

Whiz(){   i=100;}

It can be initialized in instance block of initialization

REFERENCE   : http://docs.oracle.com/javase/tutorial/java/nutsandbolts/variables.html

---

Ask our Experts

Rate this Question?   🙂   🙁

Domain :Java Basics

Consider this program -

```
1.  class Whiz {

2.

3.      public static void main(String [ ] args) {

3.          for (int x = 0; args.length > x++;) {

4.              System.out.print(args[x]+ " ");

5.          }

6.      }

7.  }
```

And the command-line invocations are?

I.   java Whiz
II.   java Whiz 10
III.   java Whiz 10 11

Which of the following is true?

A. Only the invocation I will complete without throwing exceptions ✓

B. Only Invocation I will throw an exception.

C. Invocation II will produce he output 12.

D. Invocation III will produce the output 12 32.

E. All the invocations will not throw exceptions.

Explanation:

Explanation:
Option A is the correct answer.

Here if we provide any command line argument it will throw an exception at the last iteration since we used args.length>x++. In the final iteration when it reaches to line 4, args[x] will throw *ArrayIndexOutOfBoundsException as it tries to accept one argument more than provided.* But in the first invocation args.length is equal to zero so for loop condition is false, and never reach to line 4. Hence, option A is correct.

REFERNCE   : http://docs.oracle.com/javase/tutorial/getStarted/application/index.html

Ask our Experts

Rate this Question? ☺ ☹

Question 25   **Unattempted**

Domain :Creating and Using Arrays

What will be the output of this program?

1.   class Whiz{

2.   public static void main(String args[]) {

3.   new Whiz().iterator(new int[]{10,12,13});

4.   }

5.   void iterator(int [ ]i) {

6.   for(int x=0;x<i.length;System.out.print(i[x] + " "))x++;

7.   }

8.   }

A.   **10 12 13**

B.   **12 13**

C.   **10 12.**

D.   **12 13 followed by an exception** ✅

E.   **Compilation fails**

Explanation:

Explanation:
Option D is the correct answer.

At line 3, we passed an anonymous array with three int values to the iterator method. Inside the iterator method for loop iterates trough the elements. Here inside the for loop as the third element we have given printing statement which is legal, so firstly, it will execute the increment statement and then goes for printing. So, 12 will be printed as the first element but in the last iteration, it will try to access 4th element in the array which is not there hence after printing 12 and 13 it will throw an exception.

As Explained above option D is correct.

Reference : http://docs.oracle.com/javase/tutorial/java/nutsandbolts/for.html

---

Ask our Experts

Rate this Question? 🙂 🙁

Question 26   Unattempted

Domain :Creating and Using Arrays

What will be the output of this program?

```
1.    class Whiz {

2.        public static void main(String args[ ]) {

3.            int [ ] a = {1,2,3,4};

4.

5.            for (int j : a) {

6.                if ( j==2 ) continue;

7.                    for (int x = 0; x < 3; System.out.print(x)) { x++;  }

8.            }

9.        }

10.    }
```

A.   **123**

B.   **123123**

C.   **123123123**   ✅

D.   **Compilation fails.**

E.   **Will print 123 and then goes to never ending loop.**

Explanation:

Explanation:
Option C is the correct answer.

In each iteration of the outer loop, inner loop prints elements of a. In the inner loop, we have placed printing statement in for loop updating portion which is legal. When j is equal to two, current iteration stops as if calls continue. So when the j is 2 it won't print anything and skip to next iteration. Hence there will be 123 on first, third and fourth iteration process. So, option C is correct.

Reference : http://docs.oracle.com/javase/tutorial/java/nutsandbolts/branch.html

Ask our Experts

Rate this Question?   ☺   ☹

Question 27   Unattempted

                                                                              Domain :Using Loop Constructs

What will be the output of this program?

1.   class Whiz {

2.   public static void main(String args [ ]) {

3.

4.   Loop1: for (int x = 0;x<5; x++) {

5.   if (y==3) break Loop1;

6.   for (int y = 0; y<3; y++) {

7.   if (y==2) continue Loop1;

8.   System.out.print(y);

9.   }

10.   }

11.   }

12.   }

A.   **0101010101**

B.   **123123**

C.   **123123123**

D.   **Compilation fails.** ✅

E.   **Will print 123 and then goes to never ending loop.**

Explanation:

Explanation:
Option D is the correct answer.

This is nested loop. In outer for loop, there is no variable defined as y. Outer loop can't see the variables in inner loop so trying to use inner loop variable y at line 5 causes a compile time error. Hence, option D is correct.

Reference : http://docs.oracle.com/javase/tutorial/java/nutsandbolts/branch.html

---

Ask our Experts

Rate this Question? 🙂 🙁

Question 28   Unattempted

Domain :Using Loop Constructs

What will be the output of this program?

1.   class Whiz {

2.   public static void main(String args [ ]) {

3.

```
4.              Loop1:for (int x = 0; x < 5 ;x++) {
5.                          if (x == 3) continue;
6.                          for (int y = 0; y < 3; y++) {
7.                                if (y == 1) continue Loop1;
8.                                  System.out.print(y);
9.                          }
10.                      }
11.              }
12.      }
```

A.   **012012012012**

B.   **0000**   ✓

C.   **00000**

D.   **Compilation fails.**

E.   **Will print 000 and then goes to never ending loop.**

Explanation:

Explanation:
Option B is the correct answer.

There is a labeled loop called Loop1 as the outer loop and it won't iterate the inner for loop when variable x is 3. So, it causes the inner for loop to iterate only 4 times. In inner for loop, every time when y equals to 0, it will stop iterating and continue Loop1.

Reference : http://docs.oracle.com/javase/tutorial/java/nutsandbolts/for.html

---

Ask our Experts

Rate this Question?   ☺   ☹

Question 29   Unattempted

Domain :Using Loop Constructs

What will be the output of this program?

```
1.    class Whiz {
2.        public static void main(String args [ ]) {
3.
4.            Loop1: for (int x = 0; x < 3; x++) {
5.                for (int y = 0; y < 3; y++) {
6.                    if ( y==2 ) continue;
7.                    if ( y==2 && x==2 ) break Loop1;
8.                    System.out.print(y);
9.                }
10.               }
11.          }
12.    }
```

A.   **01010101**

B.   **000**

C.   **010101**   ✅

D.   **Compilation fails.**

Explanation:

Explanation:
Option C is the correct answer.

Every time y equals 2, the loop will not reach to second if statement. So, there is no influence of second if statement to the output. So, "01" will be printed three times.

Reference : http://docs.oracle.com/javase/tutorial/java/nutsandbolts/for.html

Question 30   Unattempted

Domain :Creating and Using Arrays

Which of the following can be used to iterate through all elements of this array?
double [] dlbs = {1,5.1,2.0,7};

    A.   **for(double d : double dlbs[]){ }**

    B.   **for(double d ; dlbs){ }**

    C.   **for(dbs ; dlbs){ }**

    D.   **for(dbs : double dlbs){ }**

    E.   **for(double d : dlbs){ }** ✓

Explanation:

Explanation:

Option E is the correct answer.

Syntax for enhanced for loop is

       for([data type] [name] : [array or collection name])

So in given options, only E follows correct syntax.

Options A, C, and D are incorrect as it is invalid syntax in collection name area. Option B is incorrect because we need to use a colon, not the semi-colon.

Reference   :http://docs.oracle.com/javase/tutorial/java/nutsandbolts/for.html

Question 31   Unattempted

Domain :Using Loop Constructs

What will be the output of this program?

```
1.  public class Whiz {

2.      public static void main(String [ ] args) {

3.          do {

4.              int i = 3;

5.              System.out.print(i++);

6.          } while (i <= 6);

7.      }

8.  }
```

A.   333

B.   3456

C.   333333

D.   Infinite loop of 3

E.   Compilation fails   ✓

Explanation:

Explanation:
Option E is the correct answer.

The variable i is declared within the body of the do-while statement, so it is out of scope on line 6.
Line 6 generates a compiler error, so option E is correct.

Reference : http://docs.oracle.com/javase/tutorial/java/nutsandbolts/while.html

---

Ask our Experts

Rate this Question? ☺ ☹

Question 32   Unattempted

Domain :Using Operators and Decision Constructs

What will be the output of this program?

```
1.   public class Whiz {

2.

3.        public static void main(String [ ] args) {

4.             String out = "";

5.             z: for (int i = 3; i< 8; i++) {

6.                  if (i == 3)

7.                       continue;

8.                  if (i== 3)

9.                       break z;

10.                  out += i;

11.             }

12.             System.out.println(out);

13.        }

14.   }
```

A.   4

B.   4567   ✅

C.   34567

D.   No output will be produced.

E.   Compilation fails due to multiple errors.

Explanation:

Option B is the correct answer.

For loop at line 5 iterates till the value of i reaches 8, so the value of i will be increased 1 by 1 from 3 to 8. When the value of the variable i is 3, first "if" test will execute and skip concatenating the value of i to the String out. Since the first if statement, second if statement does not execute so loop won't break there. So, the String out will consist of concatenated values from 4 to 7 i.e. "4567". Hence, option B is correct

Reference : http://docs.oracle.com/javase/tutorial/java/nutsandbolts/branch.html

Ask our Experts

Rate this Question?   ☺   ☹

Question 33   Unattempted

Domain :Using Operators and Decision Constructs

What will be the output of this program?

```
1.   class Whiz {
2.       public static void main(String [ ] args) {
3.           for (int x = 10; x > 5; x++) {
4.               if (x == 16) x -= 11;
5.                   System.out.print(x + " ");
6.           }
7.       }
8.   }
```

A.   10 11 12 13 14 15 3.

B.   10 11 12 13 14 15

C.   The program will hang without ever completing.   ✅

D.   Compilation fails due to an error at line 4.

E.   Compilation fails due to multiple errors.

Explanation:

Explanation:
Option C is the correct answer.

Option C is correct since the code will produce 10 11 12 13 14 15 and when x= 16, the value of x will change to 5. So, that will create an endless loop.

Reference : http://docs.oracle.com/javase/tutorial/java/nutsandbolts/for.html

---

Ask our Experts

Rate this Question?   ☺   ☹

Question 34   **Unattempted**

Domain :Using Operators and Decision Constructs

Which of the following has higher precedence than "=="?

    A.   !=

    B.   &

    C.   ^

    D.   +=

    E.   -- ✓

Explanation:

Explanation:

Option E is the correct answer.

Option A is incorrect since "!=" has the same precedence as "==". So, option A is incorrect.

Options B, C, and D are incorrect as &, += and ^ have lower precedence than ==.

Option E is correct since "--" has higher precedence than "==".

Reference : http://docs.oracle.com/javase/tutorial/java/nutsandbolts/operators.html

Ask our Experts

Rate this Question?   ☺   ☹

Question 35   Unattempted

Domain :Using Operators and Decision Constructs

Which of the following will compile and produce BC when inserted at line 4?

```
1.    class Whiz {

2.          public static void main(String [ ] args) {

3.                final int s = 2;

4.                // insert code here

5.                switch(s) {

6.                      case 1    : System.out.print("A");break;

7.                      case x-1  :System.out.print("B");

8.                      case x    : System.out.print("C");break;

9.                      case x+1  : System.out.print("D");break;

10.                     default   : System.out.print("F");

11.               }

12.         }

13.   }
```

A.   int x=2;

B.   final int x=2;

C.   int x=3;

D.   private final int x = 3;

E.   None of the above. ✅

Explanation:

Explanation:
Option E is the correct answer.

The case constant has to be a compile-time constant, so we can't use option A and C. Option D is incorrect since we can't use private access modifier inside the local scope.

Option B is incorrect since it will produce two cases with same value 1 which is illegal. Hence, option E is correct.

Reference   : http://docs.oracle.com/javase/tutorial/java/nutsandbolts/switch.html

---

Ask our Experts

Rate this Question? ☺ ☹

Question 36   Unattempted

Domain :Using Operators and Decision Constructs

Which of the following is an arithmetic operator?

    A.   % ✅

    B.   =

    C.   !=

    D.   >

    E.   **None of the above**

Explanation:

Explanation:

Option A is the correct answer.

Option A is correct as % is an arithmetic operator.

Option B is an assignment operator, and !=, > are relational operators, so they are incorrect.

Reference : http://docs.oracle.com/javase/tutorial/java/nutsandbolts/expressions.html

---

Ask our Experts

Rate this Question? ☺ ☹

Question 37　　　　　　　　　　　　　　　　　　　　　　　　　　　　　　　　**Unattempted**

Domain :Using Operators and Decision Constructs

Consider this given program code and choose the correct option.

```
1.    1.    class Whiz {
      2.         public static void main(String args[ ]) {
      3.              final int i = 0;
      4.              final int j;
      5.              j=2;
      6.              int x= (int)(Math.random() * 3);
      7.
      8.              switch(x) {
      9.                   case i : {System.out.print("A");}
     10.                   case 1: System.out.print("B");break;
     11.                   case j : System.out.print("C");
     12.              }
     13.         }
     14.    }
```

Note: Math.random() * 3 will assign value 0,1 or 2.x .

A. The output will be A

B. The output will be AB

C. The output will be B

D.   The output will be BC

E.   Compilation fails. ✅

Explanation:

Option E is the correct answer.

A case constant must be a compile-time constant. Here integer j is not a compile time constant.

Reference : http://docs.oracle.com/javase/tutorial/java/nutsandbolts/switch.html

---

Ask our Experts

Rate this Question? ☺ ☹

Question 38   Unattempted

Domain :Using Operators and Decision Constructs

Select the correct statement.

A.   The only legal expression in an if statement is a boolean expression ✅

B.   Curly braces are essential for if blocks.

C.   Switch statement can evaluate float data type.

D.   The default keyword is must for switch statement.

E.   double can be used as case constant.

Explanation:

Option A is the correct answer.

Option A is correct as if statement can evaluate only boolean expressions

Option E is incorrect since the switch statements can evaluate only to enums, byte, short, int and char data types, and String.

The default keyword is optional hence option D is incorrect. The curly braces are optional for if blocks so option B is incorrect.

Reference : http://docs.oracle.com/javase/tutorial/java/nutsandbolts/switch.html

---

Ask our Experts

Rate this Question? ☺ ☹

Question 39   Unattempted

Domain :Working with Inheritance

Which of the following statement(s) is/are true?
I.   The overridden equals method of a class checks reference equality.
II.   The "==" invokes the equals method when used to check objects equality.
III.   The equals method is defined in the Object class.

A.   Only I

B.   Only III   ✓

C.   Only I and II

D.   Only I and III

E.   All of the above

Explanation:

Explanation:
Option B is the correct answer.

Statement I is incorrect since equals() method is overridden. It checks two objects for equality as per its implementation. The equals() method provided in the Object class uses the identity operator (==) to determine whether two objects are equal. For primitive data types, this gives the correct result. For objects, however, it does not. The equals() method provided by Object tests whether the object references are equal—that is, if the objects compared are the exact same object so statement III is correct.

Statement II is incorrect since == doesn't invoke any methods.

Reference : http://docs.oracle.com/javase/tutorial/java/IandI/objectclass.html

---

Ask our Experts

Rate this Question? ☺ ☹

Domain :Using Operators and Decision Constructs

What will be the output of this program?

```
1.    class Whiz {
2.         public static void main(String args[ ]) {
3.              int x = 0, y =10;
4.
5.              if (x++ > 1 && ++y >10)
6.                   System.out.print(x + y);
7.              if (++y > 10 || ++x >10)
8.                   System.out.print(x + y);
9.         }
10.   }
```

A.   No output

B.   **1112**

C.   **1012**

D.   **12** ✅

E.   **13**

Explanation:

Explanation:
Option D is the correct answer.

the first if statement evaluates the value of x and when saw it is false it won't evaluate the value of y.
So in this, if statement only the value of x changes but the value of y won't. In second if statement
evaluating the value of y return true so it won't evaluate the value of x, so the value of x won't change.
So, the final value of x will be 1 and y will be 11.

Reference : http://docs.oracle.com/javase/tutorial/java/nutsandbolts/if.html

---

Ask our Experts

Rate this Question? ☺ ☹

Question 41   Unattempted

Domain :Using Operators and Decision Constructs

What will be the output of this program?

```
1.    class Whiz {

2.        public static void main(String[ ] args) {

3.            int i = 3, j = 2;

4.            System.out.println(i-- + --j + ++i);

5.        }

6.    }
```

A.   8

B.   7   ✓

C.   6

D.   5

E.   Compilation fails.

Explanation:

Explanation:
Option B is the correct answer.

The increment/decrement operators can be applied before or after the operand. At Line 4   i-- and --j will end in i and j being decremented by one respectively. Now, i is 2 and j is 1. ++i will increment the value of i by one. Now, i is 3 and j is 1. So, the statement at line 4 becomes as follows

System.out.println(3 + 1 + 3);

Hence output will be 7, option B is correct.

Reference   : http://docs.oracle.com/javase/tutorial/java/nutsandbolts/op2.html

Ask our Experts

Rate this Question?   ☺   ☹

Question 42   Unattempted

Domain :Working with Inheritance

What will be the output of this program?

1.   class Animal {

2.   public void eat()throws NullPointerException {

3.   System.out.print("Animal eats ");

4.   }

5.   }

6.   class Dog extends Animal {

7.   void eat(String s)throws Exception{

8.   System.out.print("Dog eats "+ s);

9.   }

10.   }

11.

12.   public class Whiz {

13.   public static void main (String [ ] args)throws Exception {

14.   Animal a= new Dog();

15.   Dog d = (Dog)a;

16.   a.eat();

17.   d.eat("Meat");

18.   }

19.   }

A.   Animal eats Dog eats Meat   ✓

B.   Animal eats Animal eats

C.   Compilation fails due to an error at line 15.

D.   An exception will be thrown at runtime.

E.   Compilation fails due to an error at line 17.

Explanation:

Explanation:

Option A is the correct answer.

When we overload a method we must change the argument list and we can throw different exceptions. We can have different access modifiers. So the eat method doesn't cause any problem. Though variable has Animal reference type, it is Dog object and there is no issue when we cast it to be a dog object.

Reference   :   http://docs.oracle.com/javase/tutorial/java/javaOO/objectcreation.html

http://docs.oracle.com/javase/tutorial/java/javaOO/usingobject.html

---

Ask our Experts

Rate this Question?   ☺   ☹

Question 43   Unattempted

Domain :Working with Inheritance

Which of following represents a correct interface?

A.   interface I{ protected int x = 10; }

B.   interface I{int x = 10; void print(){ } }

C.   abstract interface I{ default void print(){ } }   ✓

D.   interface I{ default void print(); }

E.   None of the above.

Explanation:

Explanation:

Option C is the correct answer.

Option C is correct as the interface is 100% abstract so they can be marked with the abstract keyword but it is optional and from java SE 8 we are allowed to add non-abstract default methods to interfaces.

Option A is incorrect as the Interface variables are implicitly public, static and final so the modifier protected is not allowed here.

Option B is incorrect as the interface methods are implicitly abstract (except for static and default methods). An abstract method can't have a body.

Option D is incorrect as default method should be non-abstract.

REFERENCE   :http://docs.oracle.com/javase/tutorial/java/IandI/abstract.html

Ask our Experts

Rate this Question?   ☺   ☹

Question 44   Unattempted

Domain :Working with Inheritance

Which of the following statement is true?

A.   The up casting doesn't limit the capabilities of the object.

B.   The down casting can't be occurred implicitly.   ✓

C.   Casting subclass object to superclass object is known as down casting.

D.   With up casting, a ClassCastException is possible.

E.   None of the above.

Explanation:

Explanation:

Option B is the correct answer

Option B is correct as down casting should be done explicitly.

Option A is incorrect as the when we casting the object becomes more general so it will remove object's unique actions. Therefore, object capabilities are reduced.

Option C is incorrect as the casting subclass object to super class object is known as up casting.

Option D is incorrect as with down casting, a ClassCastException is possible but not with up casting.

Reference   :http://docs.oracle.com/javase/tutorial/java/IandI/subclasses.html

---

Ask our Experts

Rate this Question? ☺ ☹

Question 45   Unattempted

Domain :Working with Inheritance

Given:

  1.   **class A { }**

  2.   **class B extends A { }**

  3.   **class Whiz {**

  4.   **public static void main(String [ ] args) {**

  5.   **A aa = new B();**

  6.   **A bb = new B();**

  7.   **A aaa = new A();**

  8.   **}**

  9.   **}**

Which of the following statement is true about the above code fragment?
I.   **Object type of the variable "aa" is B.**
II.   **Reference type of the variable "bb" is B.**

III.   Object type of the variable "aaa" is A.

   A.   Only I.

   B.   Only II.

   C.   Only I and II.

   D.   Only I and III.   ✅

   E.   All of the statements are true.

Explanation:

Explanation:

Option D is the correct answer.

Statement I is correct as the object type of the variable "aa" is "B" since we have used "new B()" for initializing the object.

Statement II is incorrect. The Reference type of the variable "bb" is A because we declared the variable "bb" with "A" reference.

Statement III is correct the object type of the variable "aaa" is "A" since we have used "new A()" for initializing the object.

So, option D is correct as only the statement I and III are correct.

Reference   :http://docs.oracle.com/javase/tutorial/java/javaOO/objectcreation.html

                     :http://docs.oracle.com/javase/tutorial/java/javaOO/usingobject.html

---

Ask our Experts

Rate this Question? ☺ ☹

Question 46   Unattempted

                                        Domain :Working with Selected classes from the Java API

What will be the output of this program?

   1.   class Access{

2.   native int AccessCode;

3.   transient Integer AccessCode2 = 1000;

4.   Access(int i){

5.   AccessCode = i;

6.   }

7.   public static void main(String args[]){

8.   System.out.print(new Access(1020).AccessCode);

9.   }

10. }

A.   0

B.   1020

C.   Compilation fails due to an error at line 2.   ✅

D.   Compilation fails due to an error at line 3.

Explanation:

Explanation:

Option C is the correct answer.

This code fails the compilation because here, we used the keyword native with variable and it is illegal. The native keyword is applied to a method to indicate that the method is implemented in native code using JNI(Java Native Interface).

Reference :
https://stackoverflow.com/questions/3544919/what-are-transient-and-volatile-modifiers

https://stackoverflow.com/questions/6101311/what-is-the-native-keyword-in-java-for

Ask our Experts

Rate this Question?   ☺   ☹

Question 47   Unattempted

Domain :Working with Methods and Encapsulation

What will be the output of this program?

```
1.   class Whiz{
2.           int i = 10;
3.           public static void main(String args[]){
4.                   int i;
5.                   if(new Whiz().go(10))   i = 5;
6.                           System.out.print(i);
7.           }
8.           boolean go(int y){
9.                   if(y>5) return true;
10.                  else return false;
11.          }
12.  }
```

A.   5

B.   10

C.   An exception will be thrown at runtime.

D.   Compilation fails due to an error at line 4.

E.   Compilation fails due to an error at line 6.   ✓

Explanation:

Explanation:

Option E is the correct answer.

Local variables must be initialized before using them, here we have used if block to initialize the value for variable i. So, the compiler knows that initialization might not happen and produce compile-time error due to line 6.

REFERNCE   : http://docs.oracle.com/javase/tutorial/java/nutsandbolts/datatypes.html

---

Ask our Experts

Rate this Question? ☺ ☹

Question 48   **Unattempted**

Domain :Working with Inheritance

What will be the output of this program?

```
1.  class Person{
2.        int age;
3.        String name;
4.        public void read(){
5.              System.out.print("Person is reading ");
6.        }
7.        private void other(){}
8.  }
9.  class Livera extends Person{
10.       Livera(String s ,int i){
11.             age = i;
12.       name = s;
13.       }
14.       public void read(){
```

15.   System.out.print(name +" is reading ");

16.   }

17.   public void other()throws Exception{}

18.   }

19.   public class Whiz{

20.   public static void main(String [] args){

21.   Livera s = new Livera("Livera" , 22);

22.   Person p =s;

23.   s.read();

24.   p.read();

25.   }

26.   }

A.   Livera is reading Livera is reading ✅

B.   Livera is reading Person is reading

C.   An exception will be thrown at runtime.

D.   Compilation fails due to an error at line 22

E.   None of the above.

Explanation:

Explanation:

Option A is the correct answer.

Variable p and s refer to same object type Livera at though they have the different reference type. So, invoking read methods invokes the same overridden method. The other method is not overridden because the other method in the SuperClass (Person) is marked as private.

REFERENCE : http://docs.oracle.com/javase/tutorial/java/javaOO/usingobject.html

---

Ask our Experts

Rate this Question? 🙂 🙁

Question 49 **Unattempted**

Domain :Working with Inheritance

What will be the output of this program?

1. class SuperClass{

2.   public void method(){

3.   System.out.print("SuperClass");

4.   }

5. }

6. final class Sub extends SuperClass{

7.   public void method(){

8.   System.out.print("SubClass");

9.   }

10. }

11. public class SubSub extends Sub{

12.   public static void main(String args[]){

13.   ((SuperClass)new SubSub()).method();

14.   }

15.   public void method(){

16.   System.out.print("SubsubClass");

17.   }

18. }

    A.   The output will produce SubsubClass

B.   The output will produce SubClass

C.   The output will produce SuperClass

D.   An exception will be thrown at runtime.

E.   Compilation fails.   ✅

Explanation:

Explanation:

Option E is the correct answer.

Everything is fine in this code but a final class cannot be extended. So compilation fails because Sub Subclass extends Subclass which has marked with the final.

REFERNCE   : http://docs.oracle.com/javase/tutorial/java/nutsandbolts/datatypes.html

---

Ask our Experts

Rate this Question?   ☺ ☹

Question 50   Unattempted

Domain :Working with Methods and Encapsulation

What will be the output of this program?

```
1.   class Whiz{

2.       public static void main(String args[]) throws Exception{

3.           System.out.print(new Whiz().check(10));

4.       }

5.       boolean check(short x){

6.           if(x<10) return true;

7.           else return false;

8.       }

9.   }
```

A.   false.

B.   true

C.   An exception is thrown at runtime

D.   Compilation fails due to an error at line 3   ✓

E.   Compilation fails due to multiple errors

Explanation:

Explanation:

Option D is the correct answer.

At line 3, we pass int literal to the method but the check method accepts only short variable so the compilation fails. If we use a cast as (short)10 then the code will work fine. This is because we pass int literal as the parameter to a method, however, if we assign 10 to short value as follows it is legal.

        short s = 10;

REFERENCE   : http://docs.oracle.com/javase/tutorial/java/nutsandbolts/variables.html

---

Ask our Experts

Rate this Question?   ☺   ☹

Question 51   Unattempted

Domain :Working with Methods and Encapsulation

What will be the output of this program?

1.   class Whiz{

2.   public static void main(String[] args) {

3.   Double d = 8.08;

4.   System.out.print(d instanceof Object);

5.   }

6.   }

A.   true ✅

B.   false

C.   Compilation fails due to an error at line 3.

D.   Compilation fails due to an error at line 4.

Explanation:

Explanation:

Option A is the correct answer.

The code will compile successfully and produce true as output. Here at line 3 instead of primitive, we have used the wrapper version of the double primitive. Wrappers are the type of object so it returns true.

REFERENCE   : http://docs.oracle.com/javase/tutorial/java/nutsandbolts/variables.html

---

Ask our Experts

Rate this Question?   ☺ ☹

Question 52   Unattempted

Domain :Working with Methods and Encapsulation

What will be the output of this program?

1.   class Whiz{

2.   public static void main(String args[]){

3.   double d = new Divider().divide(30,3);

4.   System.out.print(d);

5.   }

6.   }

7.

8.   class Divider{

9.   double divide(int i, int j){

10.   return i/j;

11.   }

12.   }

A.   **10.0** ✅

B.   **10**

C.   **0.0**

D.   **Compilation fails due to an error at line 3.**

E.   **An exception is thrown.**

Explanation:

Explanation:

Option A is the correct answer.

At line 3, we have created anonymous intake of Divide class and then invoke the divide method by passing to integers. So, the code will return a double result after dividing two numbers, so option A is correct.

REFERNCE   : http://docs.oracle.com/javase/tutorial/java/javaOO/usingobject.html

---

Ask our Experts

Rate this Question? ☺ ☹

Question 53   Unattempted

Domain :Working with Methods and Encapsulation

Which of the following is/are true?

1.   The variables declared inside a method are called local variable.

2. **The local variables are initialized to its default value.**

3. **The variable "s" in the declaration "short s = 10" should be an instance variable.**

A. **Only I.** ✅

B. **Only II.**

C. **Only III.**

D. **Only I and II.**

E. **Only I and III.**

Explanation:

Explanation:

Option A is the correct answer.

Statement I is correct since the variables inside a method are called as local variables.

Statement II is incorrect as the instance variables and class variables are initialized to its default value but not local variables.

Statement III is incorrect as we can't say anything about the variable "s" as it could be a local variable or instance variable because it doesn't use any modifier. If It has used a modifier, except "final" then it should be an instance variable. Option A is correct as only the statement I is correct.

REFERENCE   : http://docs.oracle.com/javase/tutorial/java/nutsandbolts/variables.html

Ask our Experts

Rate this Question? ☺ ☹

Question 54   **Unattempted**

Domain :Java Basics

Which of the following set contains only primitive literals?

A. **1, 'c', "a"**

B.   **1, 1.5f, True**

C.   'BF', **10**, "Sure"

D.   **1.2D, 1f, 'c'** ✅

E.   **None of the above.**

Explanation:

Explanation:

Option D is the correct answer.

Option D is correct since all are primitive literals, they are double, float and char.

Option A is incorrect as "a" is a String literal.

Option B is incorrect as True is incorrect literal is should be true.

Option C is incorrect as 'BF' is illegal; char literal can only have one letter.

REFERNCE   : http://docs.oracle.com/javase/tutorial/java/nutsandbolts/datatypes.html

---

Ask our Experts

Rate this Question?   ☺   ☹

Question 55   **Unattempted**

Domain :Working with Methods and Encapsulation

What will be the output of this program?

1.   **class Whiz{**

2.   **int x = 012;**

3.   **public static void main(String [] args){**

4.   **Whiz pr = new Whiz();**

5.   **pr.go(20);**

6.   **}**

7.   **void go(final int i){**

8.   System.out.print(i/x);

9.   }

10. }

A.   1

B.   2 ✅

C.   0

D.   Compilation fails due to an error at line 2.

E.   Compilation fails due to an error at line 7.

Explanation:

Explanation:

Option B is the correct answer.

Option B is correct. At line 2, instead of using Decimal literals, we have used octal literals. So, the decimal value of x is 10 and the output is 2.

REFERENCE   : http://docs.oracle.com/javase/tutorial/java/nutsandbolts/variables.html

---

Ask our Experts

Rate this Question? ☺ ☹

Question 56   Unattempted

                                                          Domain :Working with Inheritance

What will be the output of this program?

1.   class MainClass{

2.   MainClass(){   System.out.print("MainClass ");   }

3.   }

4.   class SubClass extends MainClass{

```
5.        {System.out.print("I ");}

6.        static{System.out.print("S ");}

7.        SubClass(int i){

8.              this();

9.              System.out.print("SubClass ");

10.       }

11.

12.       SubClass(){

13.             super();

14.             System.out.print("SubClass ");

15.       }

16.  }

17.  public class SubSubClass extends SubClass{

18.       SubSubClass(String s){

19.             super();

20.             System.out.print("SubSubClass ");

21.       }

22.       public static void main(String [] args){

23.             new SubSubClass("ABC");

24.       }

25.  }
```

A.   MainClass S I SubClass SubSubClass

B.   S MainClass I SubClass SubSubClass ✅

C.   S MainClass I SubClass SubClass SubSubClass

D.   SubSubClass SubClass S I MainClass

E.   Compilation fails

Explanation:

Explanation:

Option B is the correct answer.

To get the correct answer, you should know, Static initialization blocks run once when the class is first loaded. Instance initialization blocks run every time a new instance is created. They run after all super-constructors and before the constructor's code have run.

Here first 'S' will be printed as the static block executes when the class is loading, then at line 23 when the instance of SubSubClass is created, last super class constructor in the class hierarchy which is MainClass will be executed and print "MainClass". When it comes to next subclass it will first execute non-static code block and then its constructor, so "I" and "SubClass" will print. Finally, the SubSubclass constructor will execute and print 'SubSubClass'. Hence, option B is correct.

Reference   :http://docs.oracle.com/javase/tutorial/java/javaOO/arguments.html

Ask our Experts

Rate this Question?   ☺   ☹

Question 57   *Unattempted*

Domain :Working with Methods and Encapsulation

Which of the following statement is invalid?

A.   Overloaded methods may change the argument list.   ✅

B.   Overridden methods should not change the argument list.

C.   Overloaded methods must change the argument list.

D.   Overloaded methods can change return type.

E.   In Overridden methods, object type determines which method is selected.

Explanation:

Option A is the correct answer.

Only option A is invalid, as we should change the argument list when overloading method. So option C is a valid statement. We can change the return type also; hence option D is a valid statement.

When method overriding methods we must not change the argument list, so option B is valid.

In Overridden methods, object type determines which method selected so option E is valid.

Reference   :http://docs.oracle.com/javase/tutorial/java/javaOO/returnvalue.html

---

Ask our Experts

Rate this Question?   ☺ ☹

Question 58   Unattempted

                                                    Domain :Working with Inheritance

What will be the output of this program?

```
1.    class Person{

2.        Person() {

3.            System.out.print("CP ");

4.        }

5.        static{ System.out.print("SP ");}

6.    }

7.

8.    class Student extends Person {

9.        Student() {

10.            System.out.print("CS ");

11.        }

12.    }

13.

14.    class Teacher extends Person {

15.        Teacher() {

16.            System.out.print("CT ");
```

```
17.            }

18.                private Teacher(String s){

19.                    System.out.print("OCT ");

20.                }

21.        }

22.

23.    class Whiz {

24.            public static void main(String [ ] args) {

25.                Person p1 = new Teacher("name");

26.                Student s1 = new Student();

27.            }

28.        }
```

A.   SP OCT CP SP CS

B.   SP CP IT OCT CP CS

C.   SP CP OCT CP CS

D.   SP CP OCT CT SP CP CS

E.   Compilation fails.   ✅

Explanation:

Explanation:
Option E is the correct answer.

Teacher class has a constructor which can take String as an argument but it is marked as private so it can't be seen by the Whiz class. So, this code fails to compile.

Reference   :   http://docs.oracle.com/javase/tutorial/java/javaOO/arguments.html

Ask our Experts

Rate this Question?   :)   :(

Question 59   Unattempted

Domain :Working with Methods and Encapsulation

Which of the following is the correct method signature of overloaded version of this method?
public void method() { / *codes */}

A.   public int method()

B.   private void method()

C.   private int method(String s)   ✓

D.   public String method()

E.   None of the above

Explanation:

Explanation:

Option C is the correct answer.

While overloading the method, argument list should be changed. So, option C is correct since it changes the argument list. It is legal to change the return type when we overload methods.

Reference : http://docs.oracle.com/javase/tutorial/java/javaOO/returnvalue.html

Ask our Experts

Rate this Question?   :)   :(

Question 60   Unattempted

Domain :Working with Methods and Encapsulation

What will be the output of this program?

1.   public class Whiz {

2.

3.   int x = 10;

4.   static int y = x;

5.

6.   Whiz() {

7.   y++;

8.   }

9.

10.   public static void main(String args[ ]) {

11.   System.out.println(y + new Whiz().x);

12.

13.   }

14.   }

A.   20

B.   21

C.   22

D.   23

E.   Compilation fails. ✅

Explanation:

Explanation:
Option E is the correct answer.

Static content can't access instance content, so at line 4, trying to access instance variable to assign a value for static variable y, results in a compile-time error, hence option E is correct.

Reference : http://docs.oracle.com/javase/tutorial/java/javaOO/classvars.html

Ask our Experts

Rate this Question?   🙂   ☹

Question 61   Unattempted

Domain :Working with Methods and Encapsulation

What will be the output of this program?

1.   class Whiz {

2.   static {

3.   x = 10;

4.   y = 5;

5.   }

6.   int x;

7.   final static int y ;

8.   public static void main(String args[ ]) {

9.   try {

10.   Whiz pr = new Whiz();

11.   int c = pr.x/y;

12.   System.out.print(c);

13.   } catch(ArithmeticException E) {

14.   System.out.print("Arithmetic Exception");

15.   }

16.   }

17.   }

A.   0

B.   2

C.   An Arithmetic Exception will be thrown.

D.   Compilation fails due to an error an line 3.   ✅

E.   Compilation fails due to multiple errors.

Explanation:

Explanation:
Option D is the correct answer.

Option D is correct since there is an error at line 3, we tried to access the non-static variable x inside a static block is illegal since static content can't access non-static content.

Reference   : http://docs.oracle.com/javase/tutorial/java/javaOO/returnvalue.html

Ask our Experts
Rate this Question?   ☺   ☹

Question 62   Unattempted

Domain :Working with Selected classes from the Java API

What will be the output of this program?

1.   class Whiz{

2.   public static void main(String[] args){

3.   StringBuilder sb =new StringBuilder("OCAJP");

4.   String s = new String(sb.toCharArray());

5.   s = s.concat(" 8");

6.   System.out.print(s);

7.   }

8.   }

A.   OCAJP 8

B.   **OCAJP**

C.   **8**

D.   **An Exception is thrown.**

E.   **Compilation fails.** ✅

Explanation:

Explanation:

Option E is the correct answer.

String class has a method called toCharArray which returns an array of characters of given string. Also, String class has a constructor which can take a char array and a string but StringBuilder class doesn't have CharArray method. Hence, code fails to compile because of line 4, so option E is correct.

REFERNCE   : 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 63   **Unattempted**

Domain :Working with Selected classes from the Java API

What will be the output of this program?

```
1.  class Whiz{

2.      public static void main(String[] args){

3.          StringBuilder sb = new StringBuilder("aAaA");

4.          sb.insert(sb.lastIndexOf("A"),true);

5.          System.out.print(sb);

6.      }

7.  }
```

A.   aAtrueaA

B.   aAatrueA ✅

C.   atrueAaA

D.   A StringIndexOutOfBoundsException is thrown

E.   Compilation fails.

Explanation:

Explanation:

Option B is the correct answer.

At line 3 we have created StringBuilder instance by passing "aAaA". Then at line 4, we have used one of the overloaded versions of the insert method of the StringBuilder

public StringBuilder insert(int offset, boolean b)

Inserts the string representation of the boolean argument into this sequence.

The overall effect is exactly as if the second argument were converted to a string by the method String.valueOf(boolean), 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 index of the last "A" as the so true will be inserted before the last occurrence of "A", hence option B is correct.

REFERNCE   : 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 64   Unattempted

Domain :Working with Selected classes from the Java API

What will be the output of this program?

```
1.   class Whiz{

2.       public static void main(String[] args){

3.            String str = "1Z0-808";

4.            str = str.replace("80", "81");

5.            str = str.substring(str.indexOf("80"), str.lastIndexOf("80"));

6.            System.out.print(str);

7.       }

8.   }
```

A.   **08**

B.   **80**

C.   **No output will be printed.**

D.   **An Exception is thrown.**   ✓

E.   **Compilation fails.**

Explanation:

Explanation:

Option D is the correct answer.

By calling the replace method, we replaced the first and only "80" occurrence with "81". Then at line 5, there won't be any occurrence of "80" in the str, so indexOf method will return -1 when we check index of the "80". At line 5, passing the -1 by calling indexOf with "80" results in an exception hence option D is correct.

OBJECTIVE   : Working with Selected classes from the Java API - Creating and manipulating Strings

REFERNCE   : http://docs.oracle.com/javase/tutorial/java/data/strings.html

Question 65   Unattempted

Domain :Working with Selected classes from the Java API

Which of the following can be used to get the number of characters in a string object?

    A.   **size()**

    B.   **capacity**

    C.   **length**

    D.   **length()** ✅

    E.   **None of the above**

Explanation:

Explanation:

Option D is the correct answer.

The String class has a method called length which returns the number of characters in given string.

public int length()

This returns the length of this string. The length is equal to the number of Unicode code units of the string.

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

Ask our Experts

Rate this Question?   🙂 ☹️

Question 66   Unattempted

Domain :Working with Selected classes from the Java API

Which of the following method of String class has been introduced in java SE 8?

     A.   split

     B.   replaceFirst

     C.   append

     D.   set

     E.   join   ✅

Explanation:

Explanation:

Option E is the correct answer.

Option E is correct since the Java SE 8 only introduced one new method called join in String class.

public static String join(CharSequencedelimiter, 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"

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

Ask our Experts

Rate this Question?   ☺   ☹

Question 67   Unattempted

Domain :Working with Selected classes from the Java API

Which of the following method was introduced in ArrayList in java SE 8?

     A.   length

B. spliterator ✅

C. retainAll

D. listIterator

E. None of the above

Explanation:

Explanation:

Option B is the correct answer.

All except length which is invalid method are valid methods which are available in the ArrayList, however from them, only the spliterator was introduced in java SE 8, others were available from earlier versions of java.

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

---

Ask our Experts

Rate this Question? 😊 ☹️

Question 68   Unattempted

Domain :Working with Selected classes from the Java API

What will be the output of this program?

1. public class Whiz{

2.   public static void main(String[] args){

3.   java.util.List<String> list = new java.util.ArrayList<>();

4.   list.add("A");

5.   list.add("C");

6.   list.add("E");

7.   list.add("D");

8.   list.add(1, "B");

9.   list.set(4, "F");

10.   System.out.println(list);

11.   }

12.   }

A.   [A, B, C, E, F]   ✓

B.   [A, C, E, F]

C.   [A, B, C, F, E]

D.   An Exception is thrown.

E.   Compilation fails.

Explanation:

Explanation:

Option A is the correct answer.

There are few versions of add method, at line 8 we have used following version,

public void add(int index, E element)

This inserts the specified element at the specified position in this list. Shifts the element currently at that position (if any) and any subsequent elements to the right (adds one to their indices).

So, here B will be added before C; and remain elements will be shifted accordingly.

At line 9, the set method will change the element of the given index position, so here D will be replaced by F, hence list will contain A, B, C, E, and F. Hence, option A is correct.

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

Ask our Experts

Rate this Question?   🙂 🙁

Question 69   Unattempted

Domain :Working with Selected classes from the Java API

Which of the following statement can be used to declare and instantiate an ArrayList to hold only Book instances with a default capacity of 10 elements?

    A.   **ArrayList<Book> book= new ArrayList<>();**

    B.   **ArrayList<Book> book= new ArrayList<>(10);**

    C.   **ArrayList<Book> book= new ArrayList<Book>(10);**

    D.   **ArrayList<Book> book= new ArrayList<Book>();**

    E.   **All of the above.** ✅

Explanation:

Option E is the correct answer.

From java SE 7, we can use the diamond operator so we can skip the generic part of the object initializing since the compiler figures out what it should be, for example following are same

ArrayList library = new ArrayList<>();

ArrayList library = new ArrayList();

The default capacity of the Array list is 10, and we can also create ArrayList with the capacity we need, so here all options are correct.

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

Ask our Experts

Rate this Question? 🙂 🙁

Question 70   Unattempted

Domain :Working with Selected classes from the Java API

Which of the following can be used to get the number of elements in an ArrayList object?

A.　**capacity**

B.　**length**

C.　**size()** ✅

D.　**length()**

E.　**None of the above.**

Explanation:

Explanation:

Option C is the correct answer.

All collection classes have a method called size which returns the number of elements in the collection, so option C is correct.

Options A and B are incorrect since there are no such variables. The length method is used with the strings.

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