Module: Core Java

Session 26: Assessment

Trainer can conduct online/offline assessment using following MCQs

OO Thinking

-----------

What is Java technology?

A. programming language

B. development environment

C. application environment

D. All of the above

ANSWER: D

What is byte code in the context of Java?

A. The type of code generated by a Java compiler

B. The type of code generated by a Java Virtual Machine

C. It is another name for a Java source file

D. It is the code written within the instance methods of a class.

ANSWER: A

Which of the following option is true about Java bytecode?

A. It is Platform Independent

B. It is Platform Dependent

C. Can not Say

D. None of the above

ANSWER: A

Which of the following option is true about jdk?

A. It is platform Independent

B. It is platform Dependent

C. both A and B

D. dependence upon the program code

ANSWER: B

JVM provides definitions for what?

A. Instruction set

B. Class file format

C. Garbage-collected heap

D. All of the above

ANSWER: D

Which statements are true about Garbage Collection in java?

A. It has to be invoked manually

B. It can be Controlled by the Programmer

C. Its implementation is same for all platforms

D. It is done automatically

ANSWER: D

The Class Loader is Responsible for one of the following activities. Which one?

A. Compiling files

B. Executing files

C. Preventing spoofing

D. None of the above

ANSWER: C

What is the purpose of the main method?

A. To build a user interface.

B. To hold the APIs of the application.

C. To create buttons and scrollbars.

D. To act as the entry point for the program.

ANSWER: D

Which statement(s) are true?

A. Cohesion is the OO principle most closely associated with hiding implementation details

B. Cohesion is the OO principle most closely associated with making sure that classes know about other classes only through their APIs

C. Cohesion is the OO principle most closely associated with making sure that a class is designed with a single, well focused purpose

D. Cohesion is the OO principle most closely associated with allowing a single object to be seen as having many types

ANSWER: C

What will happen when you attempt to compile and run this code?

1. public class MyMain{

2. public static void main(String argv){

3. System.out.println("Hello World");

4. }

5. }

A. The compiler will complain that main is a reserved word and cannot be used for a class

B. The code will compile and when run will print out "Hello World"

C. The code will compile but will complain at run time that no constructor is defined

D. The code will compile but will complain at run time that main is not correctly defined

ANSWER: B

What will be the output of the following program?

1. class StringLiteral{

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

3. String str="local\national";

4. System.out.print(str);

5. }

6. }

A. local\national

B. local ational

C. local

ational

D. Compiler error

ANSWER: C

What will be the output for the following program?

1. public class Identifier {

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

3. double strictfp=5.02;

4. strictfp+=.333;

5. System.out.print(strictfp);

6. }

7. }

A. 5.353

B. 5.353D

C. 5.353d

D.Compilation error

ANSWER: D

What is the advantage of using import statements?

A. To avoid having to declare variables.

B. To refer to a class without using prefixes.

C. To avoid calling methods.

D. To import the images you want to use.

ANSWER: B

Which of the following are primitive types?

A. byte

B. String

C. integer

D. Float

ANSWER: A

Which of these are legal identifiers?

A. number\_1

B. -number\_a

C. -$1234

D. -volatile

ANSWER: A

Which of these are not legal identifiers?

A. 1alpha

B. \_abcd

C. xy+abc

D. transient

ANSWER: B

Which of the following are keywords in Java?

A. Value

B. NULL

C. variable

D. synchronized

ANSWER: D

Which of these are valid signature for the main method?

A. public void main(){}

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

C. static public void main(String){}

D. public static void main(String a){}

ANSWER: B

What happens when the following program is compiled and executed with the command: java test?

1. class test {

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

3. if(args.length > 0)

4. System.out.println(args.length);

5. }

6. }

A.The program compiles and runs but does not print anything.

B.The program compiles and runs and prints 0

C.The program compiles and runs and prints 1

D.The program compiles and runs and prints 2

ANSWER: A

What is the result of compiling and running this program?

1. public class test {

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

3. int i, j;

4. int k = 0;

5. j = 2;

6. k = j = i = 9;

7. System.out.println(k);

8. }

9. }

A.The program does not compile because of the statement k = j = i = 1;

B.The program compiles and runs printing 0.

C.The program compiles and runs printing 9.

D.The program compiles and runs printing 2.

ANSWER: C

Classes, Objects and Constructors

-------------------------------

What does Analysis describe?

A. How the System will work

B. How the System will be used

C. What the system needs to do

D. None of the above

ANSWER: C

What does Design describe?

A. How the system does it

B. What the system will do

C. What the system should not do

D. None of the above

ANSWER: A

Which is a correct class declaration?

A. class Super {}

B. private class Super1{}

C. public Class super2 {}

D. public class class{};

ANSWER: A

Which is a correct way of declaring an object for class ClassName?

A. ClassName cn = new className();

B. ClassName();

C. new ClassName();

D. ClassName = ClassName();

ANSWER: C

Without which of the following, it is impossible to declare a

method?

A. return type

B. access modifier

C. argument list

D. None of the above

ANSWER: A

There are objects of String s="Java" and s2="java".Which of the following lines can be written to check whether the contents of strings are equal?

A. if(s.equalsIgnoreCase(s2))

B. if(s.equals(s2)

C. if(s==s2)

D. if(s.noCaseMatch(s2))

ANSWER: A

What will happen if you try to compile the following code?

1. class Test{

2. void add(int a){System.out.println(a);}

3. void add(int a,int b,int c){System.out.println("Total of three :"+(a+b+c));}

4. void add(int...a){

5. int res=0;

6. for(int i:a){res=res+i;}

7. System.out.println("Total from variable argument:"+res);}

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

9. Test ob=new Test();

10. ob.add(4,5,6);}}

A. Will successfully compile and the output is 'Total from variable arguments :15'

B. Will successfully compile and the output is 'Total of three :15'

C. Will not compile

D. Will compile but run time error

ANSWER: B

A top level class may have only the following access modifier?

A. package

B. friendly

C. private

D. public

ANSWER: D

Select the one most appropriate answer. A top level class without any modifier is accessible to which of the following class?

A. any class

B. any class within the same package

C. any class within the same file

D. any subclass of this class

ANSWER: B

Which two cause a compiler error?

A. int[] scores = {3, 5, 7};

B. int [][] scores = {2,7,6}, {9,3,45};

C. String cats[] = {â€œFluffyâ€, â€œSpotâ€, â€œZeusâ€};

D. String[] dogs = new String[]{new String(â€œFidoâ€),new String(â€œSpikeâ€), new String(â€œAikoâ€)};

Answer: B

Given:

1. public class ReturnIt {

2. return Type methodA(byte x, double y) {

3. return (long)x / y \* 2;

4. }

5. }

What is the narrowest valid returnType for methodA in line2?

A. int

B. byte

C. double

D. short

ANSWER: C

Given

1. class Alien{

2. String invade(short ships) { return "a few"; }

3. String invade(short... ships) { return "many"; }

4. }

5.

6. class Defender {

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

8. System.out.println(new Alien().invade(7));}

9. }

What is the result?

A. many

B. a few

C. Compilation fails

D. The output is not predictable

E. An exception is thrown at runtime

ANSWER: C

Which two cause a compiler error?

A. int[] scores = {3, 5, 7};

B. int [][] scores = {2,7,6}, {9,3,45};

C. String cats[] = {â€œFluffyâ€, â€œSpotâ€, â€œZeusâ€};

D. String[] dogs = new String[]{new String(â€œFidoâ€),new String(â€œSpikeâ€), new String(â€œAikoâ€)};

Answer: B

Given:

1. class A {

2. }

3. class Alpha {

4. private A myA = new A();

5.

6. void dolt( A a ) {

7. a = null;

8. }

9. void tryIt() {

10. dolt( myA );

11. }

12. }

Which statement is correct?

A. There are no instanced of A that will become eligible for garbage collection.

B. Explicitly setting myA to null marks that instance to be eligible for garbage collection.

C. Any call on tryIt() causes the private instance of A to be marked for garbage collection.

ANSWER: B

Given:

1. int i = 1,j = 10;

2. do{

3. if (i>j) {

4. continue;

5. }

6. j--;

7. } while (++i <6);

8. System.out.println(â€œi = â€œ +i+â€ and j = â€œ+j);

What is the result?

A. i = 6 and j = 5

B. i = 5 and j = 5

C. i = 6 and j = 4

D. i = 5 and j = 6

ANSWER: A

Given:

1. public class A {

2. void A() {

3. System.out.println(â€œClass Aâ€);

4. }

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

6. new A();

7. }

8. }

What is the result?

A. Class A

B. Compilation fails.

C. An exception is thrown at line 2.

D. The code executes with no output.

ANSWER: D

Given:

1. class Bar { }

2. class Test {

3. Bar doBar() {

4. Bar b = new Bar();

5. return b;

6. }

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

8. Test t = new Test();

9. Bar newBar = t.doBar();

10. System.out.println(â€œnewBarâ€);

11. newBar = new Bar();

12. System.out.println(â€œfinishingâ€);

13. }

14. }

At what point is the Bar object, created on line 3, eligible for garbage collection?

A. After line 8.

B. After line 10.

C. After line 4, when doBar() completes.

D. After line 11, when main() completes.

ANSWER: B

Which statement is true?

A. Programs will not run out of memory.

B. Objects that will never again be used are eligible for garbage collection.

C. Objects that are referred to by other objects will never be garbage collected.

D. Objects that can be reached from a live thread will never be garbage collected.

E. Objects are garbage collected immediately after the system recognizes they are eligible.

ANSWER: D

Given:

1. public class X {

2. private static int a;

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

4. modify(a);

5. System.out.println(a);}

6. public static void modify(int a){

7. a++;}

8. }

What is the result?

A. 0

B. 1

C. Compilation fails.

D. An exception is thrown at runtime.

ANSWER: A

Given:

1. public class Test {

2. public static void add3 (Integer i) {

3. int val = i.intValue();

4. val += 3;

5. i = new Integer(val);}

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

7. Integer i = new Integer(0);

8. add3(i);

9. System.out.println(i.intValue());}

10. }

What is the result?

A. 0

B. 3

C. Compilation fails.

D. An exception is thrown at runtime.

ANSWER: A

\_\_\_\_\_\_\_is a special method that has the same name as the class and is invoked automatically whenever an object of the class is instantiated?

A. constructor

B. setter

C. getter

D. static method

ANSWER: A

Which of the following is TRUE?

A. In java, an instance field declared public generates a compilation error.

B. int is the name of a class available in the package java.lang

C. Instance variable names may only contain letters and digits.

D. A class has always a constructor (possibly automatically supplied by the java compiler).

E. The more comments in a program, the faster the program runs.

ANSWER: D

A constructor is a special type of?

A. class

B. variable

C. method

D. object

ANSWER: C

A default constructor?

A. has no return type

B. has no argument

C. has one argument

D. has one argument but no return type.

ANSWER: B

Which is a valid Constructor for the class called 'Cat'?

A. public void Cat() {}

B. cat() {}

C. void Cat() {}

D. public Cat() {}

ANSWER: D

Can you identify the default Constructor for the class called 'SqlStar'?

A public sqlstar() {}

B SqlStar() {}

C void SqlStar() {}

D private sqlstar() {}

ANSWER : B

A constructor is used to?

A. Free memory.

B. Initialize a newly created object.

C. Import packages.

D. Create a JVM for applets.

ANSWER: B

Given

1. class Mixer {Mixer(){}

2. Mixer(Mixer m){ m1 = m; }

3. Mixer m1;

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

5. Mixer m2 = new Mixer();

6. Mixer m3 = new Mixer(m2);

7. m3.go();

8. Mixer m4 = m3.m1;

9. m4.go();

10. Mixer m5 = m2.m1;

11. m5.go();

12. }

13. void go() { System.out.print("hi "); }}

What is the result?

A. hi hi hi

B. Compilation fails

C. hi, followed by an exception

D. hi hi, followed by an exception

ANSWER: D

Read the following program and select the answer. The following program is in test1.java file.

1. public class test1 {

2. int i = 10;

3. public test1( ) {

4. ++(i + 1);

5. System.out.println( i );

6. }

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

8. new test1( );

9. }

10. }

A. Nothing is wrong with code.

B. Program will not compile.

C. Program will compile but doesn’t run.

ANSWER: B

What is wrong with the following code?

1. public class test3 {

2. public test3( ) {

3. int sum = 1, x = 0;

4. while ( x <= 10) {

5. sum = ++x;

6. +sum;

7. }

8. System.out.println(sum);

9. }

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

11. new test3( );

12. }

13. }

A. Nothing is wrong with code.

B. Program will not compile, because Line 6 is not an expression.

C. Program will compile but doesn’t run.

D. Line 8 will print 12.

ANSWER: B

What will be the output for the given program

1. class A{

2. A()

3. {

4. this(10);

5. System.out.println("constructor of class A");

6. }

7. A(int a)

8. {

9. System.out.println("parameterized constructor");

10. }

11. public static void main(String a[])

12. {

13. new A(10);

14. }

15. }

A. "parameterized constructor

B. "parameterized constructor" followed by "constructor of class A"

C. "constructor of class A" followed by "parameterized constructor"

D. variable a already defined

ANSWER: A

What will be the output for the given program?

1. class A{

2. A()

3. {

4. this.method1();

5. System.out.println("constructor of class A");

6. }

7. void method1()

8. {

9. System.out.println("method in class A");

10. }

11. public static void main(String a[])

12. {

13. new A();

14. }

15. }

A. compilation error

B. "method in class A" followed by "constructor of class A"

C. "constructor of class A" followed by "method in class A"

D. runtime error

ANSWER: C

1. class A{

2. A()

3. {

4. System.out.println("constructor of class A");

5. }

6. void method1()

7. {

8. System.out.println("method in class A");

9. }

10. public static void main(String a[])

11. {

12. new A();

13. this.method1();

14. }

15. }

A. compilation error

B. "method in class A" followed by "constructor of class A"

C. "constructor of class A" followed by "method in class A"

D. runtime error

ANSWER: A

1. class A{

2. void A()

3. {

4. System.out.println("constructor of class A");

5. }

6. public static void main(String a[])

7. {

8. A ref = new A();

9. ref.A();

10. }

11. }

A. compilation error

B. "constructor of class A"

C. runtime error

D. Program executes with a warning

ANSWER: B

What is wrong with the following program?

1. public class test4 {

2. int i = 1;

3. public test4( ) {

4. for (int i = 1; i < 10; i++) {

5. i++;

6. }

7. System.out.println( i );

8. }

9. public static void main(String agrs[ ] ) {

10. new test4( );

11. }

12. }

A. Nothing is wrong with code.

B. Program will not compile, variable "i" is declared 2 times.

C. Program will compile but don't run.

ANSWER: A

Read the following program and select the correct answer

1. public class test4 {

2. public test4( ) {

3. int i = 1;

4. for (i = 1; i < 10; i++) {

5. i = i << 2;

6. }

7. System.out.println( i );

8. }

9. public static void main(String agrs[ ]) {

10. new test4( );

11. }

12. }

A. Nothing is wrong with code. ---- Program will print 21.

B. Program will print 22.

C. Program will print 20.

ANSWER: A

\_\_\_ keyword is used to refer to the current object

A. super

B. this

C. new

D. volatile

ANSWER: B

Can we use this keyword with respect to the variables?

A. True

B. False

C. True only with non-static variables

D. True only with static variables

ANSWER: A

Can we use this keyword in static block?

A. True

B. False

ANSWER: B

Given:

1. class TestA {

2. TestB b;

3. TestA() {

4. b = new TestB(this);

5. }

6. }

7. class TestB {

8. TestA a;

9. TestB(TestA a) {

10. this.a = a;

11. }

12. }

13. class TestAll {

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

15. new TestAll().makeThings();

16. // ...code continues on

17. }

18. void makeThings() {

19. TestA test = new TestA();

20. }

21. }

Which two statements are true after line 15, before main completes?

A. Line 15 causes a stack overflow.

B. An exception is thrown at runtime.

C. The object referenced by b is not eligible for garbage collection.

D. The object referenced by b is eligible for garbage collection.

E. The object referenced by a is not eligible for garbage collection.

ANSWER: D

Class Design & Encapsulation

----------------------------

Which of the following way is correct to create a package at compilation time?

A. javac -d . classname.java

B. javac \_d .classname.java

C. javac-d. classname.java

D. javac -d. classname.java

ANSWER: A

Which of the following way is correct to execute a program in package?

A. java -d . classname

B. java packagename.classname

C. java -d packagename.classname

D. java classname

ANSWER: B

\_\_\_\_\_\_\_\_ symbol denotes the current working directory?

A. .(dot)

B. \*

C. -d

D. (space)

ANSWER: A

In Java, multi-leveled packages can be created.?

A. TRUE

B. FALSE

ANSWER: A

what is the default access specifier for a class?

A. default

B. protected

C. public

D. private

ANSWER: A

User-defined package can also be imported just like the standard packages.

A. True

B. False

Ans: A

All standard classes of Java are included within a package called \_\_\_\_\_.?

A. java.lang.Class

B. java.util.Name

C. java.lang.Object

D. no package includes

ANSWER: A

An \_\_\_\_\_\_\_\_\_ determines which features of a class may be used by other classes?

A. specifierno

B. inheritanceno

C. implementationno

D. Access specifierno

ANSWER: D

Which methods can access to private attributes of a class?

A. Only Static methods of the same class

B. Only instances of the same class

C. Only methods those defined in the same class

D. Only classes available in the same package.

ANSWER: C

Which of the following statements declare class Sample to belong to the payroll.admindept package?

A. package payroll;package admindept;

B. import payroll.\*;

C. package payroll.admindept.Sample;

D. import payroll.admindept.\*;

E. package payroll.admindept;

ANSWER: E

Which methods can access to prOtected attributes of a class?

A. Only Static methods of the same class

B. Only instances of the same class

C. Only methods those defined in the same class and sub class

D. Only classes available in the same package.

ANSWER: C

Given

1. import java.util.\*;

2. package utility;

3. public class A{

4. public static void main(String arg[])

5. { System.out.println("executed"); }

6. }

What happens when run the above program?

A. compilation error

B. Run time Error

C. Exception

D. executed

ANSWER: A

Given

1. package utils;

2. public class Repetition {

3. public static String twice(String s) { return s + s; }}

Demo.java

1. // insert code here

2. public class Demo {

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

4. System.out.println(twice("pizza"))}}

Which code should be inserted at line 1 of Demo.java to compile and run Demo to print"pizzapizza"

A. import utils.\*;

B. static import utils.\*;

C. import utils.Repetition.twice();

D. import static utils.Repetition.twice;

E. static import utils.Repetition.twice;

ANSWER: D

//ClassOne.java:

1. package com.abe.pkg1;

2. public class ClassOne {

3. private char var = â€˜aâ€™;

4. char getVar() { return var; }

5. }

//classTest.java:

1. package com.abe.pkg2;

2. import com.abc.pkg1.ClassOne;

3. public class ClassTest extends ClassOne {

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

5. char a = new ClassOne().getVar();

6. char b = new ClassTest().getVar();

7. }

}

What is the result?

A. Compilation fails.

B. Compilation succeeds and no exceptions are thrown.

C. An exception is thrown at line 5 in ClassTest.java.

D. An exception is thrown at line 6 in ClassTest.java.

ANSWER: A

//A.java.

1. package com.ts;

2. public class A {

3. public void m1() {System.out.print("A.m1, ");}

4. private void m2() {System.out.print("A.m2, ");}

5. }

//D.java.

1. package com.ts.other;

2. import com.ts.A;

3. public class D {

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

5. A a = new A();

6. a.m1();

7. a.m2();

8. }}

What is the result of attempting to compile and run the program?

A. Prints: A.m1, A.m2,

B. Compile-time error at 6 in D.java

C. Compile-time error at 7 in D.java

D. None of the above

Answer: C

//A.java.

1. package com.ts;

2. public class A {

3. public void m1() {System.out.print("A.m1, ");}

4. void m2() {System.out.print("A.m2, ");}

5. }

//D.java.

1. package com.ts;

2. import com.ts.A;

3. public class D {

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

5. A a = new A();

6. a.m1();

7. a.m2();

8. }}

What is the result of attempting to compile and run the program?

A. Prints: A.m1, A.m2,

B. Compile-time error at 6 in D.java

C. Compile-time error at 7 in D.java

D. None of the above

Answer: C

//A.java.

1. package com.ts;

2. public class A {

3. public void m1() {System.out.print("A.m1, ");}

4. protected void m2() {System.out.print("A.m2, ");}

5. }

//D.java.

1. package com.ts.other;

2. import com.dan.chisholm.A;

3. public class D {

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

5. A a = new A();

6. a.m1(); // 1

7. a.m2(); // 2

8. }}

What is the result of attempting to compile and run the program?

A. Prints: A.m1, A.m2,

B. Compile-time error at 6 in D.java

C. Compile-time error at 7 in D.java

D. None of the above

Answer: C

//ParentUtil.java

1. package testpkg.p1;

2. public class ParentUtil

3. {public int x = 420;protected int doStuff() { return x; }}

//ChildUtil.java

1. package testpkg.p2;

2. import testpkg.p1.ParentUtil;

3. public class ChildUtil extends ParentUtil

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

5. {new ChildUtil().callStuff();}

6. void callStuff()

7. { S.o.p("this " + this.doStuff() ); /\* Line 18 \*/

8. ParentUtil p = new ParentUtil();

9. S.o.p(" parent " + p.doStuff() ); /\* Line 20 \*/

10. }}

Which statement is true?

A. The code compiles and runs, with output this 420 parent 420

B. If line 7 is removed, the code will compile and run.

C. If line 9 is removed, the code will compile and run.

D. An exception is thrown at runtime.

Answer: C

//A.java.

1. package com.ts;

2. public class A {

3. A(){ }

4. public void m1() {System.out.print("A.m1, ");}

5. }

//D.java.

1. package com.ts.other;

2. import com.dan.chisholm.A;

3. public class D {

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

5. A a = new A();

6. a.m1();

7. }}

What is the result of attempting to compile and run the program?

A. Prints: A.m1

B. Compile-time error at 5

C. Compile-time error at 6

D. None of the above

Answer: B

Given:

1. public class Outer{

2. public void someOuterMethod() {

3. // Line 3

4. }

5. public class Inner{}

6. public static void main( String[]argv ) {

7. Outer o = new Outer();

8. // Line 8

9. }

10. }

Which instantiates an instance of Inner?

A. new Inner(); // At line 3

B. new Inner(); // At line 8

C. new o.Inner(); // At line 8

D. new Outer.Inner(); // At line 8

ANSWER: A

Inheritance & polymorphism

---------------------------

How can you access members of one class into another class without creating object of that class?

A. Inheritance

B. Interface

C. Abstraction

D. Encapsulation

ANSWER: A

What type of inheritance does Java have?

A. single inheritance

B. double inheritance

C. multiple inheritance

D. class inheritance

ANSWER: A

Can an object be a subclass of another object?

A. Yes---as long as single inheritance is followed.

B. No---inheritance is only between classes.

C. Only when one has been defined in terms of the other.

D. Yes---when one object is used in the constructor of another.

ANSWER: B

How many objects of a given class can there be in a program?

A. One per defined class.

B. One per constructor definition.

C. As many as the program needs.

D. One per main() method.

ANSWER: C

Which property of Java allows the creation of hierarchical classifications?

A. Interface

B. Inheritance

C. Robust

D. Distributed

ANSWER: B

What produces a compiler error?

A. class A {

public A(int x) {}

}

B. class A {

}

class B extends A {

B() {}

}

C. class A {

A() {}

}

class B {

public B() {}

}

D. class Z {

public Z(int) {}

}

class A extends Z {

}

ANSWER: D

Given:

1. class A{

2. A() { System.out.print("superclass constructor, ");}

3. public void m1() { System.out.print("method in superclass, "); } }

4. class B extends A{

5. B(){

6. super();

7. super.m1();

8. System.out.print("subclass constructor"); }

9. B(int a) {

10. System.out.println("parameterized constructor, "); }

11. public static void main(String args[])

12. {

13. B b = new B(); } }

What will be the result?

A. compilation error

B. Runtime Error

C. superclass constructor, method in superclass, subclass constructor

D. superclass constructor, subclass constructor

ANSWER: C

Given:

1. class A{

2. A() { System.out.print("superclass constructor, ");} }

3. class B extends A{

4. B(){

5. super();

6. this(6);

7. System.out.println("subclass constructor"); }

8. B(int a) {

9. System.out.println("parameterized constructor, "); }

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

11. {

12. B b = new B(); } }

A. superclass constructor, subclass construcotr, parameterized constructor

B. compilation error

C. runtime error

D. superclass constructor, parameterized constructor

ANSWER: A

Given:

1. public class Main{

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

3. B obj=new S();

4. obj.dis(); }}

5. class B{

6. static void dis(){

7. System.out.println("From Base"); }}

8. class S extends B{

9. void dis(){

10. System.out.println("From Sub"); } }

What is the result of attempting to compile and run the program?

A. Prints From Sub

B. Prints From Base

C. Compile-time error

D. Run-time error

ANSWER: C

What will be printed when you execute the code?

1. class A {

2. A() {

3. System.out.println("Class A Constructor");

4. }

5. }

6. public class B extends A {

7. B() {

8. System.out.println("Class B Constructor");

9. }

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

11. B b = new B();

12. }

13. }

A) "Class A Constructor" followed by "Class B Constructor"

B) "Class B Constructor" followed by "Class A Constructor"

C) Compile time error

D) Run time error

ANSWER: B

What restriction is there on using the super reference in a constructor?

A. Only one child class can use it.

B. It can only be used in the parent's constructor.

C. It must be used in the first statement of the constructor.

D. It must be used in the last statement of the constructor.

ANSWER: C

Given:

1. class Top {

2. public Top(String s) { System.out.print("B"); }

3. }

4. public class Bottom2 extends Top {

5. public Bottom2(String s) { System.out.print("D"); }

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

7. new Bottom2("C");

8. System.out.println(" "); } }

What is the result?

A. BD

B. DB

C. BDC

D. Compilation fails

ANSWER: D

Given:

1. class A{

2. A() { System.out.print("superclass constructor, ");} }

3. class B extends A{

4. A() { System.out.print("superclass overriden constructor"); }

5. B(int a) { System.out.print("parameterized constructor "); }

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

7. B b = new B(10); } }

What will be the output?

A. compilation error

B. Runtime Error

C. superclass constructor, superclass overriden constructor, parameterized constructor

D. superclass overriden constructor, parameterized constructor

ANSWER: A

Given:

1. class Clidder {

2. private final void flipper() { System.out.println("Clidder"); }

3. }

4. public class Clidlet extends Clidder {

5. public final void flipper() { System.out.println("Clidlet"); }

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

7. new Clidlet().flipper(); } }

What is the result?

A. Clidlet

B. Compilation fails

C. Clidder Clidlet

D. Clidlet Clidder

ANSWER: A

Given

1. class Building {

2. Building() { System.out.print("b "); }

3. Building(String name) {

4. this(); System.out.print("bn " + name); } }

5. public class House extends Building {

6. House() { System.out.print("h "); }

7. House(String name) {

8. this(); System.out.print("hn " + name); }

10. public static void main(String[] args) { new House("x "); } }

What is the result?

A. h hn x

B. hn x h

C. b h hn x

D. b hn x h

ANSWER: C

Given:

1. class X { void do1() { } }

2. class Y extends X { void do2() { } }

3. class Chrome {

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

5. X x1 = new X();

6. X x2 = new Y();

7. Y y1 = new Y();

8. // insert code here } }

Which, inserted at line 9, will compile? (Choose all that apply.)

A. x2.do2();

B. (Y)x2.do2();

C. ((Y)x2).do2();

D. None of the above statements will compile

ANSWER: C

Given:

1. class A {

2. final public int method1(int a, int b) {return 0; } }

3. class B extends A {

4. public int method1(int a, int b) { return 1; } }

5. public class Test {

6. public static void main(Strings args[]) {

7. B b;

8. System.out.println(â€œx = â€œ + b.method1(0, 1)); } }

What is the result?

A. x=0

B. x=1

C. Compilation fails.

D. En exception is thrown at runtime.

ANSWER: C

Given:

1. class Super {

2. public int getLenght() { return 4; } }

3. public class Sub extends Super {

4. public long getLenght() { return 5; }

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

6. Super sooper = new Super();

7. Sub sub = new Sub();

8. System.out.println(

9. sooper.getLenght() + â€œ,â€ + sub.getLenght() ); } }

What is the output?

A. 4,4

B. 4,5

C. 5,4

D. Compilation fails.

ANSWER: D

Given:

1. class Super {

2. public Integer getLenght() { return new Integer(4); } }

3. public class Sub extends Super {

4. public Long GetLenght() { return new Long(5); }

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

6. Super sooper = new Super();

7. Sub sub = new Sub();

8. System.out.println(

9. sooper.getLenght().toString() + â€œ,â€ +

10. sub.getLenght().toString() ); } }

What is the output?

A. 4,4

B. 4,5

C. 5,4

D. 5,5

ANSWER: A

Given:

1. public class Hotel{

2. public int bookings;

3. public void book() {bookings++;}}

4. public class SuperHotel extends Hotel {

5. public void book() {bookings--;}

6. public void book(int size) {

7. book();

8. super.book();

9. bookings += size;}

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

11. Hotel hotel = new SuperHotel();

12. hotel.book(2);

13. System.out.print(hotel.bookings);}}

What is the result?

A. Compilation Error

B. Runtime Error

C. -1

D. 0

ANSWER: A

The term \_\_\_\_\_\_\_\_\_\_ means the ability to take many forms?

A. inheritance

B. polymorphism

C. member function

D. encapsulation

ANSWER: B

What is Polymorphism?

A. Is not supported by Java

B. Refers to the ability of two or more objects belonging to different classes to respond to exactly the same message in different class-specific ways

C. Simplifies code maintenance

D. Refers to the ability of two or more objects belonging to different classes to respond to exactly the same message in different class â€“specific ways and simplifies code maintenance.

ANSWER: D

A polymorphic reference variable can refer to either an object of their own class or an object of the subclasses inherited from its class.

A. True

B. False

ANSWER: A

Can we have overloaded constructor in a class?

A. True

B. False

ANSWER: A

Polymorphism is extensively used in implementing\_\_\_\_?

A. Encapsulation

B. data hiding

C. inheritance

D. interface

ANSWER: C

The process in which the code to be link with the procedure call is not know till execution time it is called as\_\_\_\_?

A. late Binding

B. early binding

C. static binding

D. dynamic binding

ANSWER: D

Can we have overrided constructor in classes defined?

A. True

B. Fase

C. depends upon the program requirements

ANSWER: B

When the code to be linked with the call is known at compile time that situation is called as\_\_\_\_\_?

A. Binding

B. late binding

C. static binding

D. dynamic binding

ANSWER: C

An overloaded method consists of,

A. The same method name with different types of parameters

B. The same method name with different number of parameters

C. The same method name and same number and type of parameters with different return type

D. Both (a) and (b) above

E. (a), (b) and (c) above.

ANSWER: D

When an overridden method is called from within a subclass, it will always refer to the version of that method defined by the

A. Super class

B. Subclass

C. Compiler will choose randomly

D. Interpreter will choose randomly

ANSWER: B

Which of the following does not belong: If a class inherits from some other class, it should

A. Make use of the parent class's capabilities

B. Over-ride or add the minimum to accomplish the derived class' purpose

C. Over-ride all the methods of its parent class

D. Make sure the result "IS-A-KIND-OF" its base class

ANSWER: C

Can we override main() method?

A. static methods cannot be overriden

B. we can only overload main()

C. we can override main()

D. none of these

ANSWER: A

Given:

//SuperOver.java

1. class SuperOver{

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

3. { System.out.print("superover class, "); } }

//SubOver.java

1. class SubOver extends Over {

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

3. {

4. SuperOver so = new SubOver();

5. System.out.print("subover class"); } }

What will be the output for the above program?

A. superover class, subover class

B. superover class

C. subover class

D. Compilation error

ANSWER: C

Given:

//Test.java

1. class Test{

2. Test(){ System.out.print("Hello, "); }

3. public void showItem(){ } }

//A.java

1. abstract class A extends Test {

2. A(){ System.out.print("Hi everyOne"); }

3. abstract public void showItem(); }

//Dont.java

1. class Dont extends A {

2. public void showItem(){ }

3. public static void main(String[] args) { new Dont(); } }

What will be the output when we execute the above program?

A. Compile time error

B. an exception may be thrown

C. Hello, Hi everyone

D. no showItem() defined,so no output

ANSWER: C

Output of this program is 35. Which method executes, & whether it is method Overloading or method Overriding?

//Base1.java

1. class BaseI{

2. public int doSum(short a, short b){ return(a+b); } }

//Derived1.java

1. public class DerivedI extends BaseI{

2. public int doSum(int a,short b) { return(a+b); }

3. public int doSum(short a, short b) { return(a+b); }

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

5. DerivedI d = new DerivedI();

6. short x =15;

7. short y =20;

8. System.out.println(d.doSum(x,y)); } }

A. superclass method. No overloading or overriding

B. overriding

C. overloading

D. overriding AND overloading

E. None of the above

ANSWER: C

Given:

1. class Ret {

2. public long tryIt() {

3. long num =25;

4. return num; } }

5. class ReturnType extends Ret {

6. public long tryIt() {

7. char ch='a';

8. long num=50;

9. return ch; }

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

11. Ret r = new ReturnType();

12. System.out.println(r.tryIt()); } }

What will be output of the above program?

A. compilation Error

B. 25

C. 50

D. 97

ANSWER: D

Given:

1. public class Tester {

2. static void call(Long x, Long y)

3. { System.out.println("long x, long y");}

4. static void call(int x)

5. { System.out.println("int x"); }

6. static void call(Number x, Number y)

7. { System.out.println("number x, number y");}

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

9. int val=3;

10. call(val,val); } }

What will be the output for the above program?

A. long x, long y

B. int x

C. number x, number y,

D. compilation error

ANSWER: C

What is the result of compiling and running the following code?

1. class Base {

2. public final int getNext(int i) { return ++i; } }

3. public class Derived extends Base { public int getNext(int i) { return i++; }

4. public static void main(String[] args) { int result = new Derived().getNext(3);

5. System.out.print(result);

6. result = new Base().getNext(3);

7. System.out.print(result); } }

A. 33

B. 34

C. 44

D. compilation error

ANSWER: D

What is the result of compiling and running the following code?

1. class Base {

2. public void method(final int x) { System.out.print("Base"); } }

3. public class Derived extends Base { public void method(int x) { System.out.print("Derived"); }

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

5. Base b = new Derived();

6. b.method(3); } }

A. will compile fine and wil print "Derived"

B. will compile fine and wil print "Base"

C. Won't compile because of line 1.Can be corrected by marking x as final,then the output will be "Derived"

D. Won't compile because of line 1.Can be corrected by marking x as final,then the output will be "Base"

ANSWER: A

What modification is necessary to produce the following output : SubType?

1. class Type {

2. public Object getTypeName() { return "type"; } }

3. class SubType extends Type {

4. public String getTypeName() { return "subType"; } }

5. public class Tester1 {

6. public static void main(String [] ar) {

7. Type first = new SubType();

8. System.out.println(first.getTypeName()); } }

A. No modification is needed

B. At line 7 change the return type from String to Object to be a correct</p>\n<p>overloading of getTypeName()

C. At line 4 change Type to SubType

D. At line 8 cast first.getTypeName() to String

ANSWER: A

Abstract Classes & Interfaces

-----------------------------

Through which concept we achieve late binding?

A. interface

B. abstract class

C. polymorphism

D. Inheritence

ANSWER: A

Given:

1. abstract class AbstractIt {

2. abstract float getFloat();

3. }

4. public class AbstractTest extends AbstractIt {

5. private float f1 = 1.0f;

6. private float getFloat() { return f1; }

7. }

What is the result?

A. Compilation succeeds.

B. An exception is thrown.

C. Compilation fails because of an error at line 2.

D. Compilation fails because of an error at line 6.

ANSWER: D

When a program class implements an interface, it must provide behavior for?

A. two methods defined in that interface

B. any methods in a class

C. only certain methods in that interface

D. all methods defined in that interface

ANSWER: D

A relationship between classes represents \_\_\_\_\_\_\_\_\_ and an is relationship between classes represents \_\_\_\_\_\_\_\_\_?

A. static class, non-static class

B. composition, inheritance

C. inheritance, composition

D. overriding, overloading

ANSWER: B

Which is the default access modifier for an interface method?

A. privae

B. protected

C. default

D. public

ANSWER:D

What type of a variable Can we define an interface?

A. final and static

B. final

C. static

D. final and non-static

ANSWER: A

Is it necessary to implement all the methods of an interface while implementing the interface?

A. True

B. False

ANSWER: B

When a class implements an interface, it must

A. redefine each constant from the interface.

B. implement each method listed in the interface.

C. declare a variable for each constant in the interface.

D. include a private method for each method in the interface.

ANSWER: B

Given:

1. interface Beta {}

//Alpha.java

1. class Alpha implements Beta {

2. String testIt() {

3. return â€œTestedâ€;

4. }

5. }

//Main1.java

1. public class Main1 {

2. static Beta getIt() {

3. return new Alpha();

4. }

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

6. Beta b = getIt();

7. System.out.println( b.testIt() );

8. }

9. }

What is the result?

A. Tested

B. Compilation fails.

C. The code runs with no output.

D. An exception is thrown at runtime

ANSWER: B

Which three demonstrate an â€œis aâ€ relationship?

A. public class X { }

public class Y extends X { }

B. public class Person { }

public class Employee {

public Employee(Person person) { }

C. public interface Color { }

public class Shape { private Color color; }

D. public interface Species { }

public class Animal { private Species species; }

ANSWER: A

What makes to the class for not making any subclasses?

A. final

B. finally

C. static

D. finalize

ANSWER: A

Given:

1. abstract public class Employee {

2. protected abstract double getSalesAmount();

3. public double getCommision() {return getSalesAmount() \* 0.15;}

4. }

5. class Sales extends Employee { //insert method here}

Which two methods, inserted independently at line 5, correctly complete the Sales class?

i. double getSalesAmount() { return 1230.45; }

ii. public double getSalesAmount() { return 1230.45; }

iii. private double getSalesAmount() { return 1230.45; }

iv. protected double getSalesAmount() { return 1230.45; }

A. i and ii

B. i and iii

C. ii and iv

D. iii and iv

ANSWER: C

Given

1. abstract class AirPlane {

2. abstract void fly();

3. void land() { System.out.print("Landing..");}}

4. class AirJet extends AirPlane {

5. AirJet() {super(); }

6. void fly() {System.out.print("Flying..");}

7. abstract void land() ;}

The above code contains a compilation error , what can be done to fix this error - independently?

A. Remove abstract from line 7 and add body to method land()

B. add abstract to line 6

C. Remove super() call at line 5

D. Remove abstract at line 1 and line 2

ANSWER: A

To resolve the compilation error(s) in the following code, what can be done independently

1. interface Movable {

2. public abstract void m1();

3. void m2();

4. public void m3();

5. abstract void m4(); }

6. class Chair implements Movable {

7. public void m1() { }

8. void m2() { } }

A. mark class Chair "abstract"

B. mark Chair "abstract" and mark m2() "public"

C. implement m3() and m4() in Chair (with public access modifier)

D. implement the methods m3() and m4() in Chair and mark m2() in Chair "public"

ANSWER: B

Given:

1. public abstract class AbstractTest {

2. public int getNum() { return 45; }

3. }

4. public abstract class Bar {

5. public int getNum() { return 38; }

6. }

7. public static void main(String[] args) { AbstractTest t = new AbstractTest()

8. { public int getNum() { return 22; } };

9. Bar f = new Bar()

10. { public int getNum()

11. { return 57; } };

12 System.out.println(f.getNum() + " " + t.getNum()); } }

What is the result?

A. 57 22

B. 45 38

C. 45 57

D. Compilation fails

ANSWER: A

Given:

1. abstract class Vehicle { public int speed() { return 0; } }

2. class Car extends Vehicle { public int speed() { return 60; } }

3. class RaceCar extends Car { public int speed() { return 150; }

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

5. RaceCar racer = new RaceCar();

6. Car car = new RaceCar();

7. Vehicle vehicle = new RaceCar();

8. System.out.println(racer.speed() + ", " + car.speed() + ", " + vehicle.speed()); } }

What is the result?

A. 0, 0

B. 150, 60, 0

C. Compilation fails

D. 150, 150, 150

ANSWER: D

Given:

1. abstract class C1 {

2. public C1() { System.out.print(1); } }

3. class C2 extends C1 {

4. public C2() { System.out.print(2); } }

5. class C3 extends C2 {

6. public C3() { System.out.println(3); } }

7. public class Ctest { public static void main(String[] a) { new C3(); } }

What is the result?

A. Compilation fails

B. 23

C. 32

D. 123

ANSWER: D

Given:

1. public interface A { public void m1(); }

2. class B implements A { }

3. class C implements A { public void m1() { } }

4. class D implements A { public void m1(int x) { } }

5. abstract class E implements A { }

6. abstract class F implements A { public void m1() { } }

7. abstract class G implements A { public void m1(int x) { } }

What is the result?

A. Compilation succeeds

B. Exactly one class does NOT compile

C. Exactly two classes do NOT compile

D. Exactly four classes do NOT compile

E. Exactly three classes do NOT compile

ANSWER: D

Given:

1. interface DoStuff2 {float getRange(int low, int high); }

2. interface DoMore { float getAvg(int a, int b, int c); }

3. abstract class DoAbstract implements DoStuff2, DoMore { }

4. class DoStuff implements DoStuff2 {

5. public float getRange(int x, int y) { return 3.14f; } }

6. interface DoAll extends DoMore { float getAvg(int a, int b, int c, int d); }

What is the result?

A. The file will compile without error

B. Compilation fails. Only line 3 contains an error

C. Compilation fails. Only line 5 contains an error

D. Compilation fails. Only line 6 contains an error

ANSWER: A

Given:

1. public class Electronic implements Device { public void doIt() { } }

2. abstract class Phone1 extends Electronic { }

3. abstract class Phone2 extends Electronic { public void doIt(int x) { } }

4. class Phone3 extends Electronic implements Device { public void doStuff() { } }

5. interface Device { public void doIt(); }

What is the result?

A. Compilation fails with an error on line 3

B. Compilation fails with an error on line 1

C. Compilation succeeds

D. Compilation fails with an error on line 5

ANSWER: C

Exception handling

------------------

The exception class is in \_\_\_\_ package?

A. java.file

B. java.io

C. java.lang

D. java.util

ANSWER: C

\_\_\_\_\_\_ is a superclass of all exception classes.

A. Exception

B. Throwable

C. RuntimeException

D. IOException

ANSWER: B

What will happen to the Exception object after exception handling?

A. It will go for Garbage Collector. And frees the memory.

B. it will continueous the execution of the program

C. It waits for the user command

D. JVM will not understand

ANSWER: A

The toString ( ) method in the user-defined exception class is overridden.

A. True

B. False

ANSWER: A

Which cannot be thrown using the throw statement?

A. Error

B. Event

C. Object

D. RuntimeException

Answer: A

When a method can throw an exception then it is specified by \_\_\_\_\_ keyword

A. finally

B. throw

C. throws

D. catch

ANSWER: C

Which of the following options is true?

A. For one try block, you can have multiple catch block

B. finally block can be wrtten without try block

C. Non checked exceptions requires throws clause

D. You can not write multiple nested try-catch block

ANSWER: A

What must A method declare to throw?

A. Unchecked exceptions

B. Checked exceptions

C. Error

D. RuntimeException

ANSWER: B

What are checked exceptions?

A. checked by java compiler

B. checked by java virtual machine

C. above two

D. none of the above

ANSWER: A

What are un checked exceptions?

A. checked by java compiler

B. checked by java virtual machine

C. above two

D. none of the above

ANSWER: B

\_\_\_\_\_\_\_\_\_\_ exception will be thrown if you try to access the array element beyond its index value?

A. ArrayIndexOutOfBounds

B. FileNotFound

C. NumberFormate

D. ArrayIndexOutOfBound

ANSWER: A

Which cannot be thrown using the throw statement? (Choose four)

A. Error

B. Event

C. Object

D. RuntimeException

Answer: A

\_\_\_\_\_\_\_\_ and \_\_\_\_\_\_\_\_\_\_ classes are derived from Throwable class?

A. Error and Exception

B. Throws and throw

C. try and catch

D. final and finalise

ANSWER: A

Each catch clause is like a little function that takes a --------Argument`s of one particular type?

A. many

B. two

C. single

D. either A) or B)

ANSWER: C

Which of the following is true?

A. If an exception is uncaught in a method, the method will terminate and normal execution will resume

B. An overriding method must declare that it throws the same exception classes as the method it overrides

C. The main() method of a program can declare that it throws checked exceptions

D. finally blocks are executed if, and only if, an exception gets throws while inside the corresponding try block

ANSWER: C

Given:

1. public class X {

2. public X aMethod() { return this;} }

1. public class Y extends X { }

Which two methods can be added to the definition of class Y?

A. public void aMethod() {}

B. private void aMethod() {}

C. public void aMethod(String s) {}

D. private Y aMethod() { return null; }

ANSWER: C

Analyze the following code:

1. public class Test {

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

3. int[] oldList = {1, 2, 3, 4, 5};

3. reverse(oldList);

4. for (int i = 0; i < oldList.length; i++)

5. System.out.print(oldList[i] + " "); }

6. public static void reverse(int[] list) {

7. int[] newList = new int[list.length];

8. for (int i = 0; i < list.length; i++)

9. newList[i] = list[list.length - 1 - i];

10. list = newList; } }

A. The program displays 1 2 3 4 5

B. The program displays 1 2 3 4 5 and then raises an ArrayIndexOutOfBoundsException

C. The program displays 5 4 3 2 1

D. The program displays 5 4 3 2 1 and then raises an ArrayIndexOutOfBoundsException

ANSWER: A

Suppose the following Java statements:

1. char ch = ' ';

2. try {

3. do {

4. ch = (char) System.in.read(); }

5. while ( ch != 'G'); }

6. catch(Exception i) {

7. System.out.println(ch); }

are executed and applied to the following input which is typed on the keyboard:

g MnGyZ

The final value of the variable ch will be?

A. 'g'

B. ' '

C. 'n'

D. 'G'

ANSWER: D

Given:

1. public class ExceptionTest {

2. class TestException extends Exception {}

3. public void runTest() throws TestException {}

4. public void test() { runTest(); } }

At Point X on line 4, which code is necessary to make the code compile?

A. No code is necessary.

B. throws Exception

C. catch ( Exception e )

D. throws RuntimeException

ANSWER: B

Given:

1. class A {

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

3. Object error = new Error();

4. Object runtimeException = new RuntimeException();

5. System.out.print((error instanceof Exception) + ",");

6. System.out.print(runtimeException instanceof Exception); } }

What is the result of attempting to compile and run the program?

A. false,true

B. false,false

C. true,false

D. compile-time error

ANSWER: A

To explicitly throw an exception, \_\_\_\_\_\_ keyword is used.

A. try

B. catch

C. throw

D. throwing

ANSWER: C

To create our own exception class, we have to \_\_\_\_\_\_\_

A. Extend exception class

B. Create our own try and catch block

C. use finally block

D. Use throws keyword

ANSWER: A

What will be the result of attempting to compile and run the following program?

1. public class MyClass {

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

3. RuntimeException re = null;

4. throw re; } }

A. The code will fail to compile, since the main() method does not declare that it throws Runtime Exception in its declaration

B. The program will fail to compile, since it cannot throw re

C. The program will compile without error and will throw java.lang.RuntimeException when run

D. The program will compile without error and will throw java.lang.NullpointerException when run

ANSWER: D

To create our own exception class , which class we have to extends?

A. Error

B. RuntimeException

C. implement Runnable

D. None of these

ANSWER: B

When we extends RuntimeException, do you have to use throw keyword in try and catch block?

A. True

B. False

C. Depends upon requirement

D. optional

ANSWER: D

When we extends Throwable, do you have to use throw keyword in try and catch block?

A. True

B. False

C. Depends upon requirement

D. optional

ANSWER: D

Given:

1. public class X {

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

3. try {

4. badMethod();

5. System.out.print(â€œAâ€); }

6. catch (Exception ex) { System.out.print(â€œCâ€); }

7. finally { System.out.print(â€œBâ€); }

8. System.out.print(â€œDâ€);

9. }

10. public static void badMethod() { throw new Error(); } }

What is the result?

A. ABCD

B. Compilation fails.

C. C is printed before exiting with an error message.

D. BC is printed before exiting with an error message.

ANSWER: B

Given:

1. class Exc0 extends Exception { }

2. class Exc1 extends Exc0 { }

3. public class Test {

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

5. try { throw new Exc1(); }

7. catch (Exc0 e0) {

8. System.out.println(â€œEx0 caughtâ€); }

9. catch (Exception e) {

10. System.out.println(â€œexception caughtâ€); } } }

What is the result?

A. Ex0 caught

B. exception caught

C. Compilation fails because of an error at line 2.

D. Compilation fails because of an error at line 6.

ANSWER: A

Given:

1. try {

2. if ((new Object))(.equals((new Object())))

3. { System.out.println(â€œequalâ€); }

4. else { System.out.println(â€œnot equalâ€); } }

5. catch (Exception e) { System.out.println(â€œexceptionâ€); }

What is the result?

A. equal

B. not equal

C. exception

D. Compilation fails.

ANSWER: D

Given:

1. class Exc extends Exception { }

2. class Exc1 extends Exc { }

3. public class Test {

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

5. try { throw new Exc1(); }

6. catch (Exc e0) { System.out.println("Exc caught"); }

7. catch (Exception e) { System.out.println("exception caught"); } } }

A. Exc caught

B. exception caught

C. Compilation fails because of an error at line 2.

D. Compilation fails because of an error at line 5.

ANSWER: A

Given

1. class MyException extends Exception {

2. private int x;

3. MyException(int a)

4. { x=a; }

5. public String toString()

6. { return("MyException [ " + x + "]"); } }

7. class ExceptionDemo1{

8. static void compute(int a) throws MyException

9. {

10. System.out.println(" called compute : "+a);

11. if(a>10)

12. throw new MyException(a);

13. System.out.println("Exit");

14. }

15. public static void main(String args[])

16. {

17. try

18. { compute(11); }

19. catch(MyException e)

20. { System.out.println("Error : "+e); } } }

What will be the output?

A. compilation error

B. Error exception

C. called compute : 11 Error : MyException [ 11]

D. cannot predict

ANSWER: C

Given:

1. import java.\*;

2. class Myexception extends Exception {

3. Myexception(String n)

4. { super(n); } }

//Throwdemo.java

1. class Throwdemo {

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

3. Myexception me=new Myexception("This is from me");

4. try { throw me; }

5. catch(Myexception g) { System.out.println(g); } } }

What will be the output?

A. This is form me

B. Myexception: This is form me

C. compilation erro

D. object not found

ANSWER: B

Given

1. public class Main{

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

3. try { throw new Exception("throwing an exception"); }

4. catch (Exception e) { System.out.println(e.getMessage()); } } }

What happens while we are executing the program?

A. compilation error

B. Exception at line4

C. throwing an exception

D. Exception occurs

ANSWER: C

Given

1. public class Main {

2. double method(int i) throws Exception{ return i/0; }

3. boolean method(boolean b) { return !b; }

4. static double method(int x, double y) throws Exception { return x + y ; }

5. static double method(double x, double y) { return x + y - 3; }

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

7. Main mn = new Main();

8. try{

9. System.out.println(method(10, 20.0));

10. System.out.println(method(10.0, 20));

11. System.out.println(method(10.0, 20.0));

12. System.out.println(mn.method(10));

13. }

14. catch (Exception ex){ System.out.println("exception occoure: "+ ex); } } }

What will be the output when we execute the program?

A. run time Error

B. 30.0 27.0 27.0 exception occoure: java.lang.ArithmeticException: / by zero

C. compilation error

D. cant predict the output

ANSWER: B

Given

1. import java.\*;

2. class UserExce extends Exception {

3. UserExce(String n) { super(n); } }

4. class ThrowUser {

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

UserExce me=new UserExce("This is from me");

throw me; } }

What will be the output?

A. compilation error

B. runtime error

C. my exception: this is from me

D. no output, but executes successfully

ANSWER: A

Given

1. import java.\*;

2. class ExceUser extends RuntimeException {

3. ExceUser(String n) { super(n); } }

4. class ThrowUser {

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

6. ExceUser me=new ExceUser("This is from me");

7. throw me; } }

What will be the output?

A. compilation error

B. runtime error and prints ExceUser: This is frome me

C. ExceUser: This is from me

D. no output, but executes successfully

ANSWER: B

Given

1. import java.\*;

2. class ExceUser extends RuntimeException {

3. ExceUser(String n) { super(n); } }

4. class ThrowUser {

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

6. ExceUser me=new ExceUser("This is from me");

7. try{ throw me; }

8. catch(ExceUser e){ System.out.println(e.getMessage());} } }

What will be the output?

A. compilation error

B. runtime error and prints ExceUser: This is frome me

C. ExceUser: This is from me

D. This is from me

ANSWER: D

Given

1. import java.\*;

2. class ExceUser extends Throwable {

3. ExceUser(String n) { super(n); } }

4. class ThrowUser {

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

6. ExceUser me=new ExceUser("This is from me");

7. try{ throw me; }

8. catch(ExceUser e){ System.out.println(e.getMessage());} } }

What will be the output?

A. compilation error

B. runtime error and prints ExceUser: This is frome me

C. ExceUser: This is from me

D. This is from me

ANSWER: D

Given

1. import java.\*;

2. class ExceUser extends Throwable {

3. ExceUser(String n) { super(n); } }

4. class ThrowUser {

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

6. ExceUser me=new ExceUser("This is from me");

7. throw me;

8. } }

What will be the output?

A. compilation error

B. runtime error and prints ExceUser: This is frome me

C. ExceUser: This is from me

D. This is from me

ANSWER: D

Given

1. import java.\*;

2. class ExceUser extends CompiletimeException {

3. ExceUser(String n) { super(n); } }

4. class ThrowUser {

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

6. ExceUser me=new ExceUser("This is from me");

7. try{ throw me; }

8. catch(ExceUser e){ System.out.println(e.getMessage());} } }

What will be the output?

A. compilation error

B. runtime error and prints ExceUser: This is frome me

C. ExceUser: This is from me

D. This is from me

ANSWER: A

Threads

--------

What are the three parts of a Thread?

A. CPU,Bytecode,Data

B. CPU,Code only

C. CPU,Code,Data

D. None of the above

ANSWER: C

Which Can be used to create a new Thread?

A. Extend java.lang.Thread and override the run method.

B. Extend java.lang.Runnable and override the start method.

C. Implement java.lang.thread and implement the run method.

E. Implement java.lang.Thread and implement the start method.

ANSWER: A

Which method of the Runnable interface must be overwritten while implementing this interface?

A. public void running()

B. public void runnable()

C. public void startrunning()

D. public void run()

ANSWER: D

Which two are valid constructors for Thread?

1. Thread(Runnable r, String name)

2. Thread()

3. Thread(int priority)

4. Thread(Runnable r, ThreadGroup g)

5. Thread(Runnable r, int priority)

A. 1 and 3

B. 2 and 4

C. 1 and 2

D. 2 and 5

ANSWER: C

What is the state of a newly created Thread at initial stages?

A. New

B. Blocked

C. Running

D. Runnable

ANSWER: A

Which of the following methods starts a Thread?

A. run()

B. startrun()

C. start()

D. begin()

ANSWER: C

Runnable is a?

A. Class

B. Method

C. Variable

D. Interface

ANSWER: D

Which method must be defined by a class implementing the java.lang.Runnable

interface?

A. void run()

B. public void run()

C. public void start()

D. void run(int priority)

ANSWER: B

What is the functionality of the sleep() method?

A. Causes the all threads to sleep for the specified number of milliseconds.

B. Causes the target thread to sleep for the specified number of milliseconds.

C. Causes the currently executing thread to sleep for the specified number of milliseconds.

D. None

ANSWER: C

What happens when thread X executes a wait() method on object A, without owning object A lock?

A. Compilation fails.

B. An exception is thrown.

C. The wait() method has no effect.

D. Thread X receives the lock immediately.

ANSWER: B

Which method in the Thread class is used to create and launch a new thread of Execution?

A. Run();

B. Start();

C. Run(Runnable r);

D. Start(Runnable r);

E. Execute(Thread t);

ANSWER: B

Which statement is true?

A. To call the wait() method, a thread most own the lock of the current thread.

B. To call the wait() method, a thread must own the lock of the object on which the call is to be made.

C. To call the join() method, a thread must own the lock of the object on which the call is to be made.

D. To call the sleep() method, a thread must own the lock of the object which the call is to be made.

ANSWER: B

Suspend thread can be revived by using

A. start() method

B. Suspend() method

C. resume() method

D. yield() method

ANSWER: C

What can cause a thread to become non-runnable?

A. Exiting from a synchronized block.

B. Calling the wait method on an object.

C. Calling the notify method on an object.

D. Calling the notifyAll method on an object.

ANSWER: B

Given

1. class Main extends Thread{

2. public void run(String s){

3. System.out.println("thread"); }

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

5. Main m=new Main();

6. m.start(); } }

What is the result of attempting to compile and run the program?

A. Prints thread

B. Prints Nothing

C. Compile-time error

D. Run-time error

ANSWER: A

Given

1. class Main extends Thread{

2. public void run(){

3. System.out.println("thread"); }

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

5. Main m=new Main();

6. m.start(); } }

What is the result of attempting to compile and run the program?

A. Prints thread

B. Prints Nothing

C. Compile-time error

D. Run-time error

ANSWER: A

Given:

1. class MyThread extends Thread {

2. public void run() { System.out.println(â€œAAAâ€); }

3. public void run(Runnable r) { System.out.println(â€œBBBâ€); }

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

5. new Thread(new MyThread()).start();

6. } }

What is the result?

A. AAA

B. BBB

C. Compilation fails.

D. The code runs with no output.

ANSWER: A

Given:

1. public class Foo implements Runnable {

2. public void run (Thread t) {

3. system.out.printIn(â€œRunning.â€);

4. }

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

6. new thread (new Foo()).start();

7. }

8. }

What is the result?

A. An exception is thrown.

B. The program exists without printing anything.

C. An error at line 1 causes compilation to fail.

D. An error at line 2 causes the compilation to fail.

ANSWER: D

Which statement is true?

A. If only one thread is blocked in the wait method of an object, and another thread executes the modify on that same object, then the first thread immediately resumes execution.

B. If a thread is blocked in the wait method of an object, and another thread executes the notify method on the same object, it is still possible that the first thread might never resume execution.

C. If a thread is blocked in the wait method of an object, and another thread executes the notify method on the same object, then the first thread definitely resumes execution as a direct and sole consequence of the notify call.

D. If two threads are blocked in the wait method of one object, and another thread executes the notify method on the same object, then the first thread that executed the wait call first definitely resumes execution as a direct and sole consequence of the notify call.

ANSWER: B

Which CANNOT directly cause a thread to stop executing?

A. Calling the yield method.

B. Calling the wait method on an object.

C. Calling the notify method on an object.

D. Calling the start method on another Thread object.

ANSWER: C

Ad.Thread Concepts

-------------------

\_\_\_keyword in Java provides locking, which ensures mutual exclusive access of shared resource and prevent data race?

A. Synchronized

B. Volatile

C. transient

D. none of these

ANSWER: A

Which kind of variable would you prefer to synchronize on?

A. A member variable of a primitive type

B. A member variable that is an object reference

C. A method local variable that is a reference to an instance that is created within the method

D. None of the above

ANSWER: B

What is the default thread at the time of starting the program?

A. main()

B. class

C. object creation

D. constructor

ANSWER: A

The methods wait() and notify() are defined in

A. java.lang.Object

B. java.lang.Runnable

C. java.lang.Thread

D. java.lang.ThreadGroup

ANSWER: A

Garbage collector thread belongs to which priority?

A. high priority

B. low priority

C. no priority

D. normal priority

ANSWER: B

When two threads are waiting on each other and can't proceed the program is said to be in a\_\_\_\_\_?

A. deadlock

B. synchronized

C. Dameon thread

D. locked threads

ANSWER: A

To which blocks of the program we use synchronize?

A. class

B. methods

C. constructors

D. variables

ANSWER: B

If a class has a synchronised method and non-synchronised method, can multiple threads execute the non-synchronised methods?

A. True

B. False

C. depends on the requirement

D. depends on number of threads

ANSWER: A

Can a thread call multiple synchronized methods on the object of which it hold the lock?

A. True

B. False

C. thread calls but when it releases lock

D. None of these

ANSWER: A

Can static methods be synchronized?

A. Yes

B. No

ANSWER: A

\_\_\_\_ is the methods which when called on the object the thread releases the locks held on that object.

A. notify()

B. join()

C. yield()

D. wait()

ANSWER: D

Given:

1. class A1 extends Thread

2. { String[] sa;

3. public A1(String[] sa)

4. { this.sa = sa; }

5. public void run()

6. { synchronized (sa) { System.out.print(sa[0] + sa[1] + sa[2]); } } }

7. class B1

8. { private static String[] sa = new String[]{"X","Y","Z"};

9. public static void main (String[] args)

10. {

11. synchronized (sa) {Thread t1 = new A1(sa); t1.start();

12. sa[0] = "A"; sa[1] = "B"; sa[2] = "C"; } } }

A. ABC

B. XYZ

C. ABC XYZ

D. error

ANSWER: A

Given:

1. public class A extends Thread {

2. A() { setDaemon(true); }

3. public void run() { (new B()).start();

4. try { Thread.sleep(60000); }

5. catch (InterruptedException x) {}

6. System.out.println(A done); } }

//B.java

1. class B extends Thread {

2. public void run() {

3. try { Thread.sleep(60000); }

4. catch (InterruptedException x) {}

5. System.out.println(B done); } }

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

7. new A()).start(); } }

What is the result?

A. A done

B. B done

C. There is no exception that the application will print anything.

D. B done A done

ANSWER: C

Suppose you need to make a method Thread safe. Which modifier should you add to its signature?

A. static

B. transient

C. synchronize

D. synchronized

ANSWER: D

Thread Z holds the lock on object A. Thread X is blocked inside a wait call on ObjectA.

What allows thread X to become runnable?

A. Thread X is interrupted.

B. Thread X is interrupted.

C. Thread X wait() times out.

D. Thread Z releases the lock on A and calls the notifyAll() method on objectA.

ANSWER: D

Given:

1. public class SyncTest {

2. private int x;

3. private int y;

4. private synchronized void setX( int i) { x = i; }

5. private synchronized void setY( int i) { y = i; }

6. public void setXY( int i ) { setX(i); setY(i); }

7. public synchronized boolean check() {return x != y; }

8. }

Under which condition will check return true when called from a different class?

A. check can never return true.

B. check can return true when setXY is called by multiple threads.

C. check can return true when multiple threads call setX and setY separately.

D. check can return true only if SyncTest is changed to allow x and y to be set separately.

ANSWER: B

1. public class X implements Runnable {

2. private int x;

3. private int y;

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

5. X that = new X();

6. (new Thread(that)) . start( );

7. (new Thread(that)) . start( ); }

8. public synchronized void run( ) (

9. for (;;) {

10. x++;

11. y++;

12. System.out.printIn(x = + x + y = + y);

13. } } }

What is the result?

A. An error at line 8 causes compilation to fail.

B. Errors at lines 6 and 7 cause compilation to fail.

C. The program prints pairs of values for x and y that might not always be the same on the same line (for example, x=2, y=1)

D. The program prints pairs of values for x and y that are always the same on the same line (for example, x=1, y=1. In addition, each value appears twice (for example, x=1, y=1 followed by x=2s, y=2)

ANSWER: D

Given:

1. public class X implements Runnable {

2. private int x;

3. private int y;

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

5. X that = new X();

6. (new Thread( that )).start();

7. (new Thread( that )).start();

8. }

9. public void run() {

10. for (;;) {

11. synchronized (this) {

12. x++;

13. y++;

14. }

15. x = + x + y = + y);

16. } } }

What is the result?

A. Compilation fails.

B. The program prints pairs of values for x and y that might not always be the same on the same line (for example, x = 2, y = 1).

C. The program prints pairs of values for x and y that are always the same on the same line (for example, x = 1, y = 1). In addition, each value appears only once (for example, x = 1, y = 1 followed by x = 2, y = 2). The thread name at the start of the line shows that both threads are executing concurrently.

D. The program prints pairs of values for x and y that are always the same on the same line (for example, x = 1, y = 1). In addition, each value appears only once (for example, x = 1, y = 1 followed by x = 2, y = 2). The thread name at the start of the line shows that only a single thread is actually executing.

ANSWER: D

What is garbage collection in the context of Java?

A. The operating system periodically deletes all of the java files available on the system.

B. Any package imported in a program and not used is automatically deleted.

C. When all references to an object are gone, the memory used by the object is automatically reclaimed.

D. The JVM checks the output of any Java program and deletes anything that doesn't make sense.

ANSWER: C

Which cannot directly cause a thread to stop executing?

A. Calling the SetPriority() method on a Thread object.

B. Calling the wait() method on an object.

C. Calling notify() method on an object.

D. Calling read() method on an InputStream object.

ANSWER: C

Basics of Collection Framework

------------------------------

All collection classes are available in

A. java.io package

B. java.lang package

C. java.awt package

D. java.util package

ANSWER: D

Here you will find an unordered collection in which no duplicates are permitted. Which one?

A. Set

B. Collection

C. SortedSet

D. List

ANSWER: A

Which of the following may have duplicate elements?

A. Collection

B. List

C. Queue

D. Set

ANSWER: B

Which of the following methods of Map class returns a set of all the key-value pairs?

A. values()

B. keySet()

C. entrySet()

D. get()

ANSWER: C

Which of the following methods of Map returns a Set of all the keys in the map?

A. Aget()

B. entrySet()

C. values()

D. keySet()

ANSWER: D

Which interface does java.util.Hashtable implement?

A. java.util.Map

B. java.util.List

C. java.util.Hashable

D. java.util.Collection

ANSWER: A

Which statement is true for the class java.util.ArrayList?

A. The elements in the collection are ordered.

B. The collection is guaranteed to be immutable.

C. The elements in the collection are guaranteed to be unique.

D. The elements in the collection are accessed using a unique key.

ANSWER: A

Which of the following provide the capability to store objects using a key-value pair?

A. Java.util.Map and Java.util.StoredMap.

B. Java.util.Set.

C. Java.util.List.

D. Java.util.StoredSet.

ANSWER: A

TreeMap class is used to implement which collection interface?

A. Set

B. SortedSet

C. List

D. SortedMap

ANSWER: D

Which of the following Collection interface implemented by the Vector class?

A. Set

B. List

C. Array

D. Vector

ANSWER: B

Which of the following Collection interface implemented by the Hashtable class?

A. Set

B. List

C. Map

D. Vector

ANSWER: C

Which of the following Collection interface implemented by the HashSet class?

A. Set

B. List

C. Array

D. Vector

ANSWER: A

Which of these are interfaces in the collection framework?

A. Set

B. Array

C. Vector

D. LinkedList

ANSWER: A

Which of these interfaces are in the collection framework?

A. HashMap

B. ArrayList

C. Collection and SortedMap

D. TreeMap

ANSWER: C

Which collection class allows you to access its elements by associating a key with an element's value, and provides synchronization?

A. java.util.SortedMap

B. java.util.TreeMap

C. java.util.TreeSet

D. java.util.Hashtable

ANSWER: D

Which collection class allows you to associate its elements with key values, and allows you to retrieve objects in FIFO (first-in, first-out) sequence?

A. java.util.ArrayList

B. java.util.LinkedHashMap

C. java.util.HashMap

D. java.util.TreeMap

ANSWER: B

Given:

1. import java.util.\*;

2. class Ques{

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

4. String s1 = "abc";

5. String s2 = "def";

6. Vector v = new Vector();

7. v.add(s1);

8. v.add(s2);

9. String s3 = v.elementAt(0) + v.elementAt(1);

10. System.out.println(s3); } }

What changes are needed to make the above program to compile?

A. Declare Ques as public

B. Cast v.elementAt(0) to a String

C. Cast v.elementAt(1) to an Object.

D. Import java.lang

ANSWER: B

What happens when the following class is compiled?

1. import java.util.\*;

2. class x {

3. public HashMap dothis(){return new HashMap(); } }

4. class y extends x{

5. public Object dothis(){return new Object(); } }

A. Above code will compile without any error.

B. Above code will not compile because of syntax error.

C. Above code will not compile with error of dothis() in y cannot override dothis() in x; attempting to use incompatible return type.

D. Above code will compile but will generate Exception at run time.

ANSWER: C

Given

1. import java.util.\*;

2. public class Mixup {

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

4. Object o = new Object();

5. // insert code here

6. s.add("o");

7. s.add(o); } }

And these three fragments:

I. Set s = new HashSet();

II. TreeSet s = new TreeSet();

III. ArrayList s = new ArrayList();

When fragments I, II, or III are inserted, independently, at line 7, which is false?

A. Fragment I,II will get compiled,Fragment I executes without exception

B. Fragment II,III will get compiled.Fragment II executes without exception

C. Fragment III,I will get compiled,Fragment III,I executes without exception

D. none of these works

ANSWER: C

Given:

1. import java.util.\*;

2. public class Lists {

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

4. List list = new LinkedList();

5. list.add("["); list.add("A"); list.add("]");

6. System.out.println(list);

7. ListIterator it = list.listIterator();

8. while(it.hasNext()) {

9. if ("[".equals(it.next()) || "]".equals(it.next()))

10. it.remove();

11. else

12. it.add("\*"); }

13. System.out.println(list); } }

What is the output for the above program?

A. [[, A, ]] and [A]

B. [A] and [[, A, ]]

C. [[, A ] and [A ]]

D. [, A ,]] AND [[ A ]

ANSWER: A

Synchronized resizable-array implementation of the List interface is \_\_\_\_\_\_\_\_\_\_\_\_\_.

A. Vector

B. ArrayList

C. Hashtable

D. HashMap

ANSWER: A

How to sort a list of objects?

A. Arrays.sort();

B. comparable.sort();

c. comparator.sort();

D. sort();

ANSWER: A

Can we create a synchronized collection from given collection?

A. True

B. False

ANSWER: A

What are common algorithms implemented in Collections Framework?

A. sorting and searching

B. only searching

C. replacing and searching

D. only sorting

ANSWER: A

Which of the following is the correct signature of compareTo() method?

A. int compareTo(Object o)

B. int compareTo(Object o1, Object o2)

C. void compareTo(Object o1, Object o2)

D. void compareTo(Object o1)

ANSWER: A

Given a class whose instances, when found in a collection of objects, are sorted by using the compareTo() method, which statement is true?

A. The class implements java.lang.Comparable.

B. The class implements java.lang.Comparator.

C. The interface used to implement sorting allows this class to define only one sort sequence.

D. The interface used to implement sorting allows this class to define many different sort sequences.

ANSWER: A

To implement int compare(a,b), which interface has to implemented by the user?

A. Comparator

B. Comparable

ANSWER: A

Which of the following method is not related to Iterator interface?

A. hasNext()

B. next()

C. remove()

D. hasRemove()

ANSWER: D

What is the output of the following program?

1. import java.util.\*;

2. class Ques{

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

4. HashSet set = new HashSet();

5. String s1 = "abc";

6. String s2 = "def";

7. String s3 = "";

8. set.add(s1);

9. set.add(s2);

10. set.add(s1);

11. set.add(s2);

12. Iterator i = set.iterator();

13. while(i.hasNext()) {

14. s3 += (String) i.next();

15. }

16. System.out.println(s3);

17. } }

A. abcdef

B. fedcba

C. defabc

D. defabcdefabc

ANSWER: A

Given:

1. import java.util.\*;

2. public class SortOf {

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

4. ArrayList<Integer> a = new ArrayList<Integer>();

5. a.add(1);

6. a.add(5);

7. a.add(3);

8. Collections.sort(a);

9. a.adad(2);

10. Collections.reverse(a);

11. System.out.println(a); } }

What will be the output?

A. [1, 2, 3, 5]

B. [2, 1, 3, 5]

C. [2, 5, 3, 1]

D. [5, 3, 2, 1]

ANSWER: C

Given:

1. import java.util.\*;

2. public class SortOf {

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

4. ArrayList<Integer> a = new ArrayList<Integer>();

5. a.add(1);

6. a.add(5);

7. a.add(3);

8. Arrays.sort(a);

9. a.adad(2);

10. Arrays.reverse(a);

11. System.out.println(a); } }

What will be the output?

A. compilation Error

B. Exception

C. nothing prints

D. None of these

ANSWER: A

Given:

1. import java.util.Comparator;

2. public class MyIntComparable implements Comparator<Integer>{

3. public int compare(Integer o1, Integer o2) {

4. return (o1>o2 ? -1 : (o1==o2 ? 0 : 1));

5. } }

//Simple2

1. import java.util.\*;

2. public class Simple2 {

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

4. List<Integer> list = new ArrayList<Integer>();

5. list.add(5);

6. list.add(4);

7. list.add(3);

8. list.add(7);

9. list.add(2);

10. list.add(1);

11. Collections.sort(list, new MyIntComparable());

12. for (Integer integer : list) {

13. System.out.print(integer); } } }

What will be the output?

A. 754321

B. 7654321

C. 654321

D. 764321

ANSWER: A

To implement int compareTo(a,b), which interface has to implemented by the user?

A. Comparator

B. Comparable

ANSWER: B

If you want to cycle through a collection in bidirectional then which interface should you implement?

A. Iterator

B. ListIterator

C. Enumarator

D. Comparable

ANSWER: B

Which method is used to sort according to the specified comparator?

A. sort()

B. sort(List<T>,Comparator<? super T>)

C. none of these

ANSWER: B

\_\_\_\_\_is a SortedMap extended with navigation methods returning the closest matches for given search targets.

A. NavigableMap<k,v>

B. NavigableSet<E>

C. map<k,v>

D. sortedmap<k,v>

ANSWER: A

\_\_\_\_\_is a Set that further provides a total ordering on its elements.

A. SortedSet<E>

B. SortedMap<K,V>

C. NavigableSet<E>

D. set<E>

ANSWER: A

Which of the following is the one of the constructors for the TreeSet?

A. TreeSet(Collection<? extends E> c)

B. TreeSet(Collection())

C. TreeSet(Collection<E> c)

D. all the above

ANSWER: A

Which of the following is the correct signature of compare() method?

A. int compare(Object o)

B. int compare(Object o1, Object o2)

C. void compare(Object o1, Object o2)

D. void compare(Object o1)

ANSWER: B

If you want to sort objects based on natural order then which interface you use?

A. Comparable

B. COmparator

C. Iterator

D. Enumerator

ANSWER: A

Generics and Legacy Classes

---------------------------

Why do you use Generic classes in collection framework?

A. type-safety and avoid ClassCastException at runtime.

B. coding standards

C. to get an exception

D. none of these

ANSWER: A

How to initialize a Generic Class?

A. GenericClassName<type1>

B. GenericClassName(type1)

C. <type1>GenericClassName

D. (type1)GenericClassName

ANSWER: A

What is Bounded and Unbounded wildcards in Generics?

A. <? extends T> and <? super T>

B. Generics which has limit to access

C. <? super T> and <? extends T>

ANSWER: A

LinkedList<? extends E>, what it represents?

A. Any subtype of E

B. Any supertype of E

C. Any type

D. Generic Type of E

ANSWER: A

LinkedList<? super E>, what it represents?

A. Any subtype of E

B. Any supertype of E

C. Any type

D. Generic Type of E

ANSWER: B

LinkedList<?>, what it represents?

A. Any subtype of E

B. Any supertype of E

C. Any type

D. Generic Type of E

ANSWER: C

Which of the following are true?

A. The elements of a Collection class can be ordered by using the sort method of the Collection interface

B. You can create an ordered Collection by instantiating a class that implements the List interface

C. The Collection interface sort method takes parameters of A or D to change the sort order, Ascending/Descending

D. The elements of a Collection class can be ordered by using the order method of the Collection interface

ANSWER: B

How can you remove an element from a Vector?

A. delete method

B. cancel method

C. clear method

D. remove method

ANSWER: D

What is the output for the below code?

1. public class Test {

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

3. ArrayList list = new ArrayList();

4. ArrayList listStr = list;

5. ArrayList listBuf = list;

6. listStr.add(0, "Hello");

7. StringBuffer buff = listBuf.get(0);

8. System.out.println(buff.toString());

9. } }

A. Hello

B. Compile error

C. java.lang.ClassCastException

D. null

ANSWER: C

Given:

1. public class Gen<G> {

2. G g;

3. Gen(G g)

4. { this.g =g; }

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

6. {

7. Gen<String> arr[] = new Gen[5]; //line 1

8. arr[0] = new Gen("Java"); //line 2

9. arr[1] = new Gen(1); //line 3

10. arr[2] = (Gen<String>)new Gen(1); //line 4

11. arr[3] = (Gen<String>)new Gen<Integer>(1); //line 5

12. for(Gen o:arr)

13. { System.out.println(o); }

14. } }

What will be the output if you attempt to execute the program?

A. Compile time Error at line 1

B. Compile time Error at line 3

C. Compile time Error at line 4

D. Compile time Error at line 5

ANSWER: D

Given:

1. import java.util.\*;

2. class Vehicle {}

3. class Car extends Vehicle {}

4. class Bus extends Vehicle {}

5. class TestSamp {

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

7. ArrayList<Car> a = new ArrayList<Car>();

8. a.add(new Car());

9. ArrayList b = a;

10. ArrayList<Bus> c = (ArrayList<Bus>)b;

11. c.add(new Bus());

12. for (Object obj : b)

13. System.out.println(obj);

14. }

15. }

What will be the output if you attempt to execute the program?

A. compiler error

B. compiles with warning and gives some output

C. compiles without warning and gives some output

D. copiles and run with no output

ANSWER: C

Given:

1. import java.util.\*;

2. class Test {

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

4. Set vals = new TreeSet<String>();

5. vals.add("one");

6. vals.add(1);

7. vals.add("two");

8. System.out.println(vals);

9. }

10. }

What will be the output if you attempt to execute the program?

A. Does not Compile

B. Compiles with warning and prints output [one, 1, two]

C. Compiles without warning and prints output [one, 1, two]

D. Compiles with warning and throws exception at runtime

E. Compiles without warning and throws exception at runtime

ANSWER: D

Given:

1. import java.util.\*;

2. class CheckTest {

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

4. Hashtable ht = new Hashtable<String, String>();

5. ht.put("chec", "check");

6. ht.put(1000, "check"); // line 1

7. ht.put("check", 20.01); // line 2

8. System.out.print(ht.get("chec") + " ");

9. System.out.print(ht.get(1000) + " "); //line 3

10. System.out.print(ht.get("check")); //line 4

11. } }

What will be the output if you attempt to execute the program?

A. Compilation fails due to error at line 1,2,3,4

B. Compilation fails due to error at line 3,4

C. Compiles fine and exception is thrown at runtime.

D. Compiles fine and prints "check check 20.01"

E. Compilation fails due to error at line 1,2

ANSWER: D

Can we use Generics with arrrays?

A. True

B. False

ANSWER: A

Given:

1. import java.util.\*;

2. public class Vec{

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

4. Vec v = new Vec();

5. v.amethod(); }

6. public void amethod(){

7. Vector mv = new Vector();

8. mv.addElement("Hello");

9. mv.addElement(new Integer(99));

10. for(int i=0; i< mv.size(); i++){

11. System.out.print(mv.elementAt(i)+", ");

12. } } }

What will be the output for the above program?

A. compilation Error

B. Hello, 99,

C. Hello 99

D. Exception

ANSWER: B

Can Enums have constructors and methods?

A. Yes

B. No

ANSWER: A

Which of the following are true statements?

1. The Iterator interface declares only three methods: hasNext, next and remove.

2. The ListIterator interface extends both the List and Iterator interfaces.

3. The ListIterator interface provides forward and backward iteration capabilities.

4. The ListIterator interface provides the ability to modify the List during iteration.

5. The ListIterator interface provides the ability to determine its position in the List.

A. 2, 3, 4 and 5

B. 1, 3, 4 and 5

C. 3, 4 and 5

D. 1, 2 and 3

ANSWER: B

Which method is related to Iterator?

A. hasMoreElements()

B. nextElement()

C. next()

D. delete()

ANSWER: C

can we add an element to the iterator?

A. Yes

B. NO

ANSWER: B

What are the inherited method of Enums from Enum class?

A. final boolean equals(Object other)

B. EnumTypeName[] values()

C. EnumTypeName valueOf(String name)

ANSWER: A

Suppose s1 and s2 are two strings. Which of the following statements or expressions are incorrect?

A. String s = new String("new string");

B. String s3 = s1 + s2;

C. s1 >= s2

D. int i = s1.length();

ANSWER: C

Which of the following methods are methods of the String class?

A. delete( )

B. append( )

C. reverse( )

D. replace( )

ANSWER: D

Which of these method of class StringBuffer is used to reverse sequence of characters?

A. reverse()

B. reverseall()

C. Reverse()

D. reverseAll()

ANSWER: A

Which of the following is not a wrapper class?

A. String

B. Integer

C. Boolean

D. Character

ANSWER: A

Which of the following are immutable objects?

A. String Buffer

B. String

C. Integer

D. Boolean

ANSWER: B

Which one does not have a valueOf(String) method?

A. Integer

B. Boolean

C. Character

D. Long

E. Short

ANSWER: C

Given:

1. public class TestLocal {

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

3. String s[] = new String[6];

4. System.out.print(s[6]); } }

what will be the output?

A. null

B. compile time error

C. Exception

D. null followed by 0

ANSWER: C

Given:

1. class output {

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

3. {

4. StringBuffer c = new StringBuffer("Hello ");

5. System.out.println(c.length());

6. } }

what will be the output?

A. 4

B. 5

C. compilation erro

D. 6

ANSWER: D

What is the output of the following code?

1. class Test

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

3. { String s1 = new String("Welcome to Java!");

4. String s2 = s1;

5. if (s1 == s2)

6. System.out.println("s1 is equal to s2");

7. else

8. System.out.println("s1 is not equal to s2");

9. } }

A. s1 is equal to s2

B. s1 is not equal to s2

ANSWER: A

What is the output of the following code?

1. class Test

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

3. { String s1 = new String("Welcome to Java!");

4. String s2 = new String("Welcome to Java!");

5. if (s1 == s2)

6. System.out.println("s1 is equal to s2");

7. else

8. System.out.println("s1 is not equal to s2");

9. } }

A. s1 is equal to s2

B. s1 is not equal to s2

ANSWER: B

What is the output of the following code?

1. class Test

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

3. { StringBuffer s1 = new StringBuffer("Welcome to Java!");

4. StringBuffer s2 = s1;

5. if (s1 == s2)

6. System.out.println("s1 is equal to s2");

7. else

8. System.out.println("s1 is not equal to s2");

9. } }

A. s1 is equal to s2

B. s1 is not equal to s2

ANSWER: A

What is the output of the following code?

1. class Test

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

3. { String s1 = new String("Welcome to Java!");

4. String s2 = new String("Welcome to Java!");

5. if (s1.equals(s2))

6. System.out.println("s1 is equal to s2");

7. else

8. System.out.println("s1 is not equal to s2");

9. } }

A. s1 is equal to s2

B. s1 is not equal to s2

ANSWER: A

Given:

1. public class EqualsOper {

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

3. if("String ".trim() == "String")

4. System.out.println("Equal");

5. else

6. System.out.println("Not Equal"); } }

What will be the output for the above program?

A. the code will compile an print Equal.

B. the code will compile an print Not Equal.

C. the code will cause a compiler error

D. excutes but no output

ANSWER: B

What happens when you execute the given program?

1. public class MethodTest{

2. public void method(Object o){

3. System.out.println("Object Verion"); }

4. public void method(String s){

5. System.out.println("String Version"); }

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

7. MethodTest test = new MethodTest();

8. test.method(null); } }

A. The code does not compile.

B. The code compiles cleanly and shows â€œObject Versionâ€.

C. The code compiles cleanly and shows â€œString Versionâ€

D. The code throws an Exception at Runtime.

ANSWER: C

What will happen if you attempt to execute the program?

1. public class MethodTest{

2. public void method(StringBuffer sb){

3. System.out.println("StringBuffer Verion"); }

4. public void method(String s){

5. System.out.println("String Version"); }

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

7. MethodTest test = new MethodTest();

9. test.method(null); } }

A. compilation error.

B. The code compiles cleanly and shows â€œStringBuffer Versionâ€.

C. The code compiles cleanly and shows â€œString Versionâ€

D. The code throws an Exception at Runtime.

ANSWER: A

What will be the output for the given program?

1. class Mystery {

2. public static String mystery(String s) {

3. String s1 = s.substring(0,1);

4. String s2 = s.substring(1, s.length() - 1);

6. String s3 = s.substring(s.length() - 1);

7. if (s.length() <= 3)

8. return s3 + s2 + s1;

9. else

10. return s1 + mystery(s2) + s3; }

11. public static void main(String arg[]) {

12. System.out.println(mystery("deliver")); } }

A. deliver

B. elive

C. del

D. deliv

ANSWER: A

What is the output of the following program?

1. import java.util.StringTokenizer;

2. class TestStringTokenizer

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

4. { String s = "I am learning Java.";

5. StringTokenizer st = new StringTokenizer(s);

6. while (st.hasMoreTokens())

7. System.out.print(st.nextToken()+" "); } }

A. I am learning Java.

B. I, am ,learning Java.

C. I. am. learning Java

D. None of the above

ANSWER: A

Which method can be used to find out the total allocated capacity of a StrinBuffer? Ans: capacity( ) method

A. valueOf()

B. capacity()

C. equals()

D. length()

ANSWER: B

Which of these method of class StringBuffer is used to extract a substring from a String object?

A. substring()

B. Substring()

C. SubString()

D. None of the mentioned

ANSWER: A

What will be the output if you attempt the program to execute?

1. import java.util.Date;

2. public class DateDemo {

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

4. Date date = new Date();

5. String str = String.format("Current Date/Time : %tc", date );

6. System.out.printf(str); } }

A. prints current date and time

B. printf not found, error

C. exception

D. none of these

ANSWER: A

Java IO

-------

Which of the following statements are true? UTF characters are all 8-bits.

A. UTF characters are all 16-bits.

B. UTF characters are all 24-bits.

C. Unicode characters are all 16-bits.

D. Bytecode characters are all 16-bits.

ANSWER: D

Which class provides a formatted input function?

A. System

B. Reader

C. Scanner

D. Writer

ANSWER: C

Name the exception thrown by the read method defined in InputStream class.

A. IOException

B. SQLException

C. InputStreamException

D. None of the above

ANSWER: A

Which of the following are valid constructors for the DataInputStream class?

A. DataInputStream(FileInputStream in, int size)

B. DataInputStream(FileInputStream in)

C. DataInputStream(File f)

D. DataInputStream(String s)

ANSWER: B

Which method is used to rename a file?

A. renameTo(File newName)

B. rename(File newName)

C. renameTo(String newName)

D. renameTo(String newName)

ANSWER: A

Which class provides readLine() method?

A. OutputStream

B. InputStream

C. BufferedReader

D. BufferedWriter

ANSWER: C

How can you change the current working directory using an instance of the File class called FileName?

A. FileName.chdir("DirName").

B. FileName.cd("DirName").

C. FileName.cwd("DirName").

D. The File class does not support directly changing the current directory.

ANSWER: D

which method of File class is used to get the entire path of the file?

A. getName()

B. getPath()

C. getAbsolutePath()

D. getParent()

ANSWER: C

Which abstract class is the super class of all classes used for reading bytes?

A. Reader

B. FileReader

C. ByteReader

D. InputStream

ANSWER: D

Which abstract class is the super class of all classes used for writing characters?

A. Writer

B. FileWriter

C. CharWriter

D. OutputStream

ANSWER: B

DataInput is?

A. an abstract class defined in java.io.

B. a class we can use to read primitive data types.

C. an interface that defines methods to open files.

D. an interface that defines methods to read primitive data types.

ANSWER: D

Which of these are legal ways of accessing a File named "file.tst" for reading?

A. FileReader fr = new FileReader("file.tst");

B. FileReader fr = new FileReader("file.tst", "UTF8");

C. InputStreamReader isr = new InputStreamReader("file.tst");

ANSWER: A

You execute the code below in an empty directory. What is the result?

File f1 = new File("dirname");

File f2 = new File(f1, "filename");

A. No directory is created, and no file is created.

B. A new directory called dirname is created in the current working directory. A new file called filename is created in directory dirname.

C. A new directory called dirname and a new file called filename are created, both in the current working directory.

D. A new file called filename is created in the current working directory.

ANSWER: A

Which of these classes are not abstract?

A. FilterWriter

B. Reader

C. InputStream

D. CharArrayReader

ANSWER: D

Which of the following are true?

A. The InputStream and OutputStream classes are byte-oriented.

B. The ObjectInputStream and ObjectOutputStream do not support serialized object input and output.

C. The Reader and Writer classes are byte/character-oriented.

D. The Reader and Writer classes are the preferred solution to serialized object output.

ANSWER: A

How do you create a Reader object from an InputStream object? Use the static createReader( ) method of InputStream class.

A. Use the static createReader( ) method of Reader class.

B. Create an InputStreamReader object, passing the InputStream object as an argument to the InputStreamReader constructor.

C. Create an OutputStreamReader object, passing the InputStream object as an argument to the OutputStreamReader constructor.

ANSWER: C

The\_\_\_\_\_\_\_\_ method helps in clearing the buffer.

A. flush()

B. trim()

C. clean()

D. remove()

ANSWER: A

Given:

1. import java.io.\*;

2. public class Hit{

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

4. Hit p = new Hit();

5. p.go(); }

6. public void go(){

7. try{

8. DataInputStream dis = new DataInputStream(System.in);

9. dis.read();

10. } catch(Exception e){}

11. System.out.println("Continuing");

12. } }

Which of the following statements are true?

A. The code will compile and pause until a key is hit

B. The code will not compile because System.in is a static class

C. The code will compile and run to completion without output

D. The code will not compile because System.in is not a valid constructor for DataInputStream

ANSWER: A

Given:

1. import java.io.\*;

2. public class Fos{

3. String s = new String("Hello");

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

5. Fos f = new Fos();

6. f.amethod(); }

7. public void amethod(){

8. FileOutputStream fos = new FileOutputStream("Out.txt");

9. fos.write(10); } }

What will be the output?

A. Compile time error

B. Runtime error

C. Creation of a file called Out.txt containing the text "10"

D. Creation of a file called Out.txt

ANSWER: A

Reading or Writing can be done even after closing the input/output source

A. True

B. False

C. sometimes it is true

ANSWER: B

The isFile() method returns a boolean value depending on whether the file object is a file or a directory

A. True

B. False

ANSWER: A

Ad.Java IO

----------

How is information represented in a file?

A. Analog form

B. Binary form

C. Various special forms depending on the type of file

D. Machine levele form

ANSWER: B

In JAVA end of file (EOF) is indicated by?

A. -1

B. 0

C. 1

D. False

ANSWER: A

What is the need of Serialization?

A. To send state of one or more objects state over the network through a socket.

B. To save the state of an object in a file.

C. An objectâ€™s state needs to be manipulated as a stream of bytes.

D. All the above

ANSWER: D

Which of the following methods cannot be invoked by an instance of StreamTokenizer?

A. nextToken()

B. toString()

C. equals()

D. close()

ANSWER: D

Which of the following can you perform using the File class?

A. Change the current directory

B. Return the name of the parent directory

C. Move a file to another directory

D. Find if a file contains text or binary information

ANSWER: B

How to declare a variable if that should not be part serializable object?

A. transient

B. private

C. protectd

D. default

ANSWER: A

Which of the following is the legal mode for creating a new RandomAccessFile stream?

A. "w"

B. 'r'

C. "rw"

D. "rwx"

ANSWER: C

Given:

1. import.java.io.\*;

2. class Echo {

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

4. {

5. String line;

6. BufferedReader stdin = new BufferedReader(new InputStreamReader( System.in ) );

7. System.out.println("Enter your input:");

8. line = stdin.readLine();

9. System.out.println( "You typed: " + line ); } }

How many lines of data does the program read?

A. One

B. Two

C. Three

D. As many as we enter

ANSWER: A

What will be the output of the given program?

1. import java.io.\*;

2. class Test

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

4. { try

5. { RandomAccessFile raf = new RandomAccessFile("test.dat", "r");

6. int i = raf.readInt(); }

7. catch(IOException e)

8. { System.out.println("runtime exception"}); } }

A. The program does not compile because raf is not created correctly.

B. The program does not compile because readInt() is not implemented in RandomAccessFile.

C. The program compiles, but throws IOException because the file test.dat doesn't exist. The program displays runtime exception.

D. The program compiles and runs fine, but nothing is displayed on the console.

ANSWER: C

What will be the output of the given program?

1. import java.io.\*;

2. class filesinputoutput {

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

4. InputStream obj = new FileInputStream("inputoutput.java");

5. System.out.print(obj.available()); } }

A. true

B. false

C. prints number of bytes in file

D. prints number of characters in the file

ANSWER: C

Given:

1. import java.io.\*;

2. class WriteData {

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

4. { try {

5. BufferedReader br = new BufferedReader(newInputStreamReader(System.in));

6. System.out.println("enter the data");

7. String s=br.readLine();

8. FileOutputStream fos = new FileOutputStream("file1.txt");

9. FileOutputStream fos = new FileOutputStream("file1.txt",true);

10. byte ba[] = s.getBytes();

11. fos.write(ba);

12. fos.close(); }

13. catch(Exception e){ System.out.println(e); } } }

What is the difference between line 8 and line 9?

A. line 8,whenever we open the same file for second time, the file overwritten as the file pointer points to Zero location.line9, to point the file pointer for second time at the end of the file we use true

B. no difference, in both the statments files get overridden

C. in both the lines,for second time the file pointer points at the end of the file we use true

D. in both the lines, whenever we open the same file for second time, the file overwritten as the file pointer points to Zero location

ANSWER: A

What will be the output for the given program?

1. import java.io.\*;

2. public class filesinputoutput {

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

4. String obj = "abc";

5. byte b[] = obj.getBytes();

6. ByteArrayInputStream obj1 = new ByteArrayInputStream(b);

7. for (int i = 0; i < 2; ++ i) {

8. int c;

9. while ((c = obj1.read()) != -1) {

10. if(i == 0) { System.out.print((char)c); } } }

11. }

12. }

A. abc

B. ABC

C. ab

D. AB

ANSWER: A

To serialize an array or a collection all the members of it must be serializable?

A. True

B. False

ANSWER: A

What is the output of this program?

1. import java.io.\*;

2. public class filesinputoutput {

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

4. String obj = "abc";

5. byte b[] = obj.getBytes();

6. ByteArrayInputStream obj1 = new ByteArrayInputStream(b);

7. for (int i = 0; i < 2; ++ i) {

8. int c;

9. while ((c = obj1.read()) != -1) {

10. if (i == 0) {

11. System.out.print(Character.toUpperCase((char)c));

12. obj2.write(1);

13. }

14. }

15. System.out.print(obj2);

16. }

17. } }

A. AaBaCa

B. ABCaaa

C. AaaBaaCaa

D. AaBaaCaaa

ANSWER: D

If a class is serializable but its superclass in not , what will be the state of the instance variables inherited from super class after deserialization?

A. they will get the original values which are assigned by the default constructor

B. values will remains same

C. when we call those variables, we get an exception

D. none of these

ANSWER: A

Given:

1. import java.io.\*;

2. class AddUpAll

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

4. { int value; // the value of the current integer

5. int limit; // the number of integers to add up

6. int sum = 0; // initialize sum

7. String line;

8. BufferedReader stdin = new BufferedReader( new InputStreamReader( System.in ) );

9. System.out.println("Enter how many integers:");

10. line = \_\_\_\_\_\_// Insert code here

11. limit = Integer.parseInt( line.trim() );

12. int count = 1; // initialize count

13. while ( count <= limit )

14. { System.out.println("Enter a number:");

15. line = \_\_\_\_\_\_\_\_\_ // insert code here

16. value = Integer.parseInt( line.trim() );

17. sum = sum + value; // add to the sum

18. count = count + 1; // increment count }

19. System.out.println( "Grand Total: " + sum ); } }

Fill with the sutable code in the blank to read the values?

A. stdin.readLine();

B. stdin.write();

C. stdin.read();

D. stdin.writeLine();

ANSWER: A

What is the purpose of implementing the writeObject() and readObject() method?

A. to store the transient variables state as a part of the serialized object

B. to write into the file

C. to read from the file

D. to store the values in the variables

ANSWER: A

Are the static variables saved as the part of serialization?

A. True

B. False

ANSWER: A

Given

1. public class NonSerial { //This is a non-serializable class }

2. public class MyClass implements Serializable{

3. private static final long serialVersionUID = 1L;

4. private NonSerial nonSerial;

5. MyClass(NonSerial nonSerial){

6. this.nonSerial = nonSerial; }

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

8. NonSerial nonSer = new NonSerial();

9. MyClass c = new MyClass(nonSer);

10. try {

11. FileOutputStream fs = new FileOutputStream("test1.ser");

12. ObjectOutputStream os = new ObjectOutputStream(fs);

13. os.writeObject(c);

14. os.close();

15. } catch (Exception e) { e.printStackTrace(); }

16. try {

17. FileInputStream fis = new FileInputStream("test1.ser");

18. ObjectInputStream ois = new ObjectInputStream(fis);

19. c = (MyClass) ois.readObject();

20. ois.close();

21. } catch (Exception e) {

22. e.printStackTrace();

23. } } }

What will be the output?

A. Compilation erro

B. Exception

C. test1.ser file creats

D. None of these

ANSWER: A

Do we need to implement any method of Serializable interface to make an object serializable?

A. True

B. False

C. depends on the requirements

D. none of these

ANSWER: B