Java Questions & Answers – Character and Boolean Data Types

This Section of our 1000+ Java MCQs focuses on Character and Boolean Datatypes of Java Programming Language.

1. What is the numerical range of a char data type in Java?  
a) -128 to 127  
b) 0 to 256  
c) 0 to 32767  
d) 0 to 65535  
View Answer

Answer: d  
Explanation: Char occupies 16-bit in memory, so it supports 2^16 i:e from 0 to 65535.

2. Which of these coding types is used for data type characters in Java?  
a) ASCII  
b) ISO-LATIN-1  
c) UNICODE  
d) None of the mentioned  
View Answer

Answer: c  
Explanation: Unicode defines fully international character set that can represent all the characters found in all human languages. Its range is from 0 to 65536.

3. Which of these values can a boolean variable contain?  
a) True & False  
b) 0 & 1  
c) Any integer value  
d) true  
View Answer

Answer: a  
Explanation: Boolean variable can contain only one of two possible values, true and false.

4. Which of these occupy first 0 to 127 in Unicode character set used for characters in Java?  
a) ASCII  
b) ISO-LATIN-1  
c) None of the mentioned  
d) ASCII and ISO-LATIN1  
View Answer

Answer: d  
Explanation: First 0 to 127 character set in Unicode are same as those of ISO-LAIN-1 and ASCII.

5. Which one is a valid declaration of a boolean?  
a) boolean b1 = 1;  
b) boolean b2 = ‘false’;  
c) boolean b3 = false;  
d) boolean b4 = ‘true’  
View Answer

Answer: c  
Explanation: Boolean can only be assigned true or false literals.

6. What is the output of this program?

1. **class** array\_output {
2. **public** **static** **void** main(String args[])
3. {
4. **char** array\_variable [] = **new** **char**[10];
5. **for** (**int** i = 0; i < 10; ++i) {
6. array\_variable[i] = 'i';
7. System.out.print(array\_variable[i] + "" );
8. i++;
9. }
10. }
11. }

a) i i i i i  
b) 0 1 2 3 4  
c) i j k l m  
d) None of the mentioned  
View Answer

Answer: a  
Explanation: None.  
output:  
$ javac array\_output.java  
$ java array\_output  
i i i i i

7. What is the output of this program?

1. **class** mainclass {
2. **public** **static** **void** main(String args[])
3. {
4. **char** a = 'A';
5. a++;
6. System.out.print((**int**)a);
7. }
8. }

a) 66  
b) 67  
c) 65  
d) 64  
View Answer

Answer: a  
Explanation: ASCII value of ‘A’ is 65, on using ++ operator character value increments by one.  
output:  
$ javac mainclass.java  
$ java mainclass  
66

8. What is the output of this program?

1. **class** mainclass {
2. **public** **static** **void** main(String args[])
3. {
4. **boolean** var1 = **true**;
5. **boolean** var2 = **false**;
6. **if** (var1)
7. System.out.println(var1);
8. **else**
9. System.out.println(var2);
10. }
11. }

a) 0  
b) 1  
c) true  
d) false  
View Answer

Answer: c  
Explanation: None.  
output:  
$ javac mainclass.java  
$ java mainclass  
true

9. What is the output of this program?

1. **class** booloperators {
2. **public** **static** **void** main(String args[])
3. {
4. **boolean** var1 = **true**;
5. **boolean** var2 = **false**;
6. System.out.println((var2 & var2));
7. }
8. }

a) 0  
b) 1  
c) true  
d) false  
View Answer

Answer: d  
Explanation: boolean ‘&’ operator always returns true or false. var1 is defined true and var2 is defined false hence their ‘&’ operator result is false.  
output:  
$ javac booloperators.java  
$ java booloperators  
false

10. What is the output of this program?

1. **class** asciicodes {
2. **public** **static** **void** main(String args[])
3. {
4. **char** var1 = 'A';
5. **char** var2 = 'a';
6. System.out.println((**int**)var1 + " " + (**int**)var2);
7. }
8. }

a) 162  
b) 65 97  
c) 67 95  
d) 66 98  
View Answer

Answer: b  
Explanation: ASCII code for ‘A’ is 65 and for ‘a’ is 97.  
output:  
$ javac asciicodes.java  
$ java asciicodes  
65 97

Java Questions & Answers – Data Type-Enums

This set of Java Multiple Choice Questions & Answers (MCQs) focuses on “Data Structures-Enums”.

1. What is the order of variables in Enum?  
a) Ascending order  
b) Descending order  
c) Random order  
d) depends on the order() method  
View Answer

2. Can we create instance of Enum outside of Enum itself?  
a) True  
b) False  
View Answer

3.

1. **enum** Season {
2. WINTER, SPRING, SUMMER, FALL
3. };
4. System.out.println(Season.WINTER.ordinal());

a) 0  
b) 1  
c) 2  
d) 3  
View Answer

4. If we try to add Enum constants to a TreeSet, what sorting order will it use?  
a) Sorted in the order of declaration of Enums  
b) Sorted in alphabetical order of Enums  
c) Sorted based on order() method  
d) Sorted in descending order of names of Enums  
View Answer

5. What is the output of below code snippet?

1. **class** A
2. {
4. }
6. **enum** Enums **extends** A
7. {
8. ABC, BCD, CDE, DEF;
9. }

a) Runtime Error  
b) Compilation Error  
c) It runs successfully  
d) EnumNotDefined Exception  
View Answer

6. What is the output of below code snippet?

1. **enum** Levels
2. {
3. **private** TOP,
5. **public** MEDIUM,
7. **protected** BOTTOM;
8. }

a) Runtime Error  
b) EnumNotDefined Exception  
c) It runs successfully  
d) Compilation Error  
View Answer

Answer: d  
Explanation: Enum cannot have any modifiers. They are public, static and final by default.

7. What is the output of below code snippet?

1. **enum** Enums
2. {
3. A, B, C;
5. **private** Enums()
6. {
7. System.out.println(10);
8. }
9. }
11. **public** **class** MainClass
12. {
13. **public** **static** **void** main(String[] args)
14. {
15. **Enum** en = Enums.B;
16. }
17. }

a) 10  
10  
10  
b) Compilation Error  
c) 10  
10  
d) Runtime Exception  
View Answer

Answer: a  
Explanation: The constructor of Enums is called which prints 10.

8. Which method returns the elements of Enum class?  
a) getEnums()  
b) getEnumConstants()  
c) getEnumList()  
d) getEnum()  
View Answer

Answer: b  
Explanation: getEnumConstants() returns the elements of this enum class or null if this Class object does not represent an enum type.

9. Which class does all the Enums extend?  
a) Object  
b) Enums  
c) Enum  
d) EnumClass  
View Answer

Answer: c  
Explanation: All enums implicitly extend java.lang.Enum. Since Java does not support multiple inheritance, an enum cannot extend anything else.

10. Are enums are type-safe?  
a) True  
b) False  
View Answer

Answer: a  
Explanation: Enums are type-safe as they have own name-space.

# Java Questions & Answers – Integer and Floating Data Types

This Section of our 1000+ Java MCQs focuses on Integer and Floating Datatypes of Java Programming Language.

1. What is the range of short data type in Java?  
a) -128 to 127  
b) -32768 to 32767  
c) -2147483648 to 2147483647  
d) None of the mentioned  
View Answer

2. What is the range of byte data type in Java?  
a) -128 to 127  
b) -32768 to 32767  
c) -2147483648 to 2147483647  
d) None of the mentioned  
View Answer

3. Which of the following are legal lines of Java code?  
1. int w = (int)888.8;  
2. byte x = (byte)100L;  
3. long y = (byte)100;  
4. byte z = (byte)100L;  
a) 1 and 2  
b) 2 and 3  
c) 3 and 4  
d) All statements are correct.  
View Answer

4. An expression involving byte, int, and literal numbers is promoted to which of these?  
a) int  
b) long  
c) byte  
d) float  
View Answer

5. Which of these literals can be contained in float data type variable?  
a) -1.7e+308  
b) -3.4e+038  
c) +1.7e+308  
d) -3.4e+050  
View Answer

6. Which data type value is returned by all transcendental math functions?  
a) int  
b) float  
c) double  
d) long  
View Answer

7. What is the output of this program?

1. **class** average {
2. **public** **static** **void** main(String args[])
3. {
4. **double** num[] = {5.5, 10.1, 11, 12.8, 56.9, 2.5};
5. **double** result;
6. result = 0;
7. **for** (**int** i = 0; i < 6; ++i)
8. result = result + num[i];
9. System.out.print(result/6);
11. }
12. }

a) 16.34  
b) 16.566666644  
c) 16.46666666666667  
d) 16.46666666666666  
View Answer

8. What will be the output of these statement?

1. **class** output {
2. **public** **static** **void** main(String args[])
3. {
4. **double** a, b,c;
5. a = 3.0/0;
6. b = 0/4.0;
7. c=0/0.0;
9. System.out.println(a);
10. System.out.println(b);
11. System.out.println(c);
12. }
13. }

a) Infinity  
b) 0.0  
c) NaN  
d) all of the mentioned  
View Answer

9. What is the output of this program?

1. **class** increment {
2. **public** **static** **void** main(String args[])
3. {
4. **int** g = 3;
5. System.out.print(++g \* 8);
6. }
7. }

a) 25  
b) 24  
c) 32  
d) 33  
View Answer

10. What is the output of this program?

1. **class** area {
2. **public** **static** **void** main(String args[])
3. {
4. **double** r, pi, a;
5. r = 9.8;
6. pi = 3.14;
7. a = pi \* r \* r;
8. System.out.println(a);
9. }
10. }

a) 301.5656  
b) 301  
c) 301.56  
d) 301.56560000  
View Answer

Java Questions & Answers – Data Type-BigDecimal

This set of Tough Java Questions and Answers focuses on “Data Type-BigDecimal”.

1. Which of the following is the advantage of BigDecimal over double?  
a) Syntax  
b) Memory usage  
c) Garbage creation  
d) Precision  
View Answer

Answer: d  
Explanation: BigDecimal has unnatural syntax, needs more memory and creates great amount of garbage. But it has high precision which is useful for some calculations like money.

2. Which of the below datatype doesnt support overloaded methods for +,-,\* and /?  
a) int  
b) float  
c) double  
d) BigDecimal  
View Answer

Answer: d  
Explanation: int, float, double provide overloaded methods for +,-,\* and /. BigDecimal does not provide these overloaded methods.

3. What is the output of below code snippet?

1. **double** a = 0.02;
2. **double** b = 0.03;
3. **double** c = b - a;
4. System.out.println(c);
6. BigDecimal \_a = **new** BigDecimal("0.02");
7. BigDecimal \_b = **new** BigDecimal("0.03");
8. BigDecimal \_c = b.subtract(\_a);
9. System.out.println(\_c);

a) 0.009999999999999998  
0.01  
b) 0.01  
0.009999999999999998  
c) 0.01  
0.01  
d) 0.009999999999999998  
0.009999999999999998  
View Answer

Answer: a  
Explanation: BigDecimal provides more precision as compared to double. Double is faster in terms of performance as compared to BigDecimal.

4. What is the base of BigDecimal data type?  
a) Base 2  
b) Base 8  
c) Base 10  
d) Base e  
View Answer

Answer: c  
Explanation: A BigDecimal is n\*10^scale where n is an arbitrary large signed integer. Scale can be thought of as the number of digits to move the decimal point to left or right.

5. What is the limitation of toString() method of BigDecimal?  
a) There is no limitation  
b) toString returns null  
c) toString returns the number in expanded form  
d) toString uses scientific notation  
View Answer

Answer: d  
Explanation: toString() of BigDecimal uses scientific notation to represent numbers known as canonical representation. We must use toPlainString() to avoid scientific notation.

6. Which of the following is not provided by BigDecimal?  
a) scale manipulation  
b) + operator  
c) rounding  
d) hashing  
View Answer

Answer: b  
Explanation: toBigInteger() converts BigDecimal to a BigInteger.toBigIntegerExact() converts this BigDecimal to a BigInteger by checking for lost information.

7. BigDecimal is a part of which package?  
a) java.lang  
b) java.math  
c) java.util  
d) java.io  
View Answer

Answer: b  
Explanation: BigDecimal is a part of java.math. This package provides various classes for storing numbers and mathematical operations.

8. What is BigDecimal.ONE?  
a) wrong statement  
b) custom defined statement  
c) static variable with value 1 on scale 10  
d) static variable with value 1 on scale 0  
View Answer

Answer: d  
Explanation: BigDecimal.ONE is a static variable of BigDecimal class with value 1 on scale 0.

9. Which class is a library of functions to perform arithmetic operations of BigInteger and BigDecimal?  
a) MathContext  
b) MathLib  
c) BigLib  
d) BigContext  
View Answer

Answer: a  
Explanation: MathContext class is a library of functions to perform arithmetic operations of BigInteger and BigDecimal.

10. What is the output of below code snippet?

1. **public** **class** AddDemo
2. {
3. **public** **static** **void** main(String args[])
4. {
5. BigDecimal b = **new** BigDecimal("23.43");
6. BigDecimal br = **new** BigDecimal("24");
7. BigDecimal bres = b.add(**new** BigDecimal("450.23"));
8. System.out.println("Add: "+bres);
10. MathContext mc = **new** MathContext(2, RoundingMode.DOWN);
11. BigDecimal bdecMath = b.add(**new** BigDecimal("450.23"), mc);
12. System.out.println("Add using MathContext: "+bdecMath);
13. }
14. }

a) Compilature failure  
b) Add: 684.66  
Add using MathContext: 6.8E+2  
c) Runtime exception  
d) Add 6.8E+2  
Add using MathContext: 684.66  
View Answer

Answer: b  
Explanation: add() adds the two numbers, MathContext provides library for carrying out various arithmetic operations.

# Java Questions & Answers – Data Type-Date, TimeZone

This set of Java Multiple Choice Questions & Answers (MCQs) focuses on “Data Type – Date and TimeZone”.

1. How to format date from one form to another?  
a) SimpleDateFormat  
b) DateFormat  
c) SimpleFormat  
d) DateConverter  
View Answer

Answer: a  
Explanation: SimpleDateFormat can be used as  
Date now = new Date();  
SimpleDateFormat sdf = new SimpleDateFormat (“yyyy-mm-dd’T’hh:MM:ss”);  
String nowStr = sdf.format(now);  
System.out.println(“Current Date: ” + );

2. How to convert Date object to String?  
a) SimpleDateFormat sdf = new SimpleDateFormat(“yyyy-mm-dd”);  
sdf.parse(new Date());  
b) SimpleDateFormat sdf = new SimpleDateFormat(“yyyy-mm-dd”);  
sdf.format(new Date());  
c)SimpleDateFormat sdf = new SimpleDateFormat(“yyyy-mm-dd”);  
new Date().parse();  
d) SimpleDateFormat sdf = new SimpleDateFormat(“yyyy-mm-dd”);  
new Date().format();  
View Answer

Answer: b  
Explanation: SimpleDateFormat takes a string containing pattern. sdf.format converts the Date object to String.

3. How to convert Date object to String?  
a) SimpleDateFormat sdf = new SimpleDateFormat(“yyyy-mm-dd”);  
sdf.parse(new Date());  
b) SimpleDateFormat sdf = new SimpleDateFormat(“yyyy-mm-dd”);  
sdf.format(new Date());  
c)SimpleDateFormat sdf = new SimpleDateFormat(“yyyy-mm-dd”);  
new Date().parse();  
d) SimpleDateFormat sdf = new SimpleDateFormat(“yyyy-mm-dd”);  
new Date().format();  
View Answer

Answer: a  
Explanation: SimpleDateFormat takes a string containing pattern. sdf.parse converts the String to Date object.

4. Is SimpleDateFormat thread safe?  
a) True  
b) False  
View Answer

Answer: b  
Explanation: SimpleDateFormat is not thread safe. In multithreaded environment we need to manage threads explicitly.

5. How to identify if a timezone is eligible for DayLight Saving?  
a) useDaylightTime() of Time class  
b) useDaylightTime() of Date class  
c) useDaylightTime() of TimeZone class  
d) useDaylightTime() of DateTime class  
View Answer

Answer: c  
Explanation: public abstract boolean useDaylightTime() is provided in TimeZone class.

6. What is the replacement of joda time library in java 8?  
a) java.time (JSR-310)  
b) java.date (JSR-310)  
c) java.joda  
d) java.jodaTime  
View Answer

Answer: a  
Explanation: In java 8,we are asked to migrate to java.time (JSR-310) which is a core part of the JDK which replaces joda library project.

7. How is Date stored in database?  
a) java.sql.Date  
b) java.util.Date  
c) java.sql.DateTime  
d) java.util.DateTime  
View Answer

Answer: a  
Explanation: java.sql.Date is the datatype of Date stored in database.

8. What does LocalTime represent?  
a) Date without time  
b) Time without Date  
c) Date and Time  
d) Date and Time with timezone  
View Answer

Answer: b  
Explanation: LocalTime of joda library represents time without date.

9. How to get difference between two dates?  
a) long diffInMilli = java.time.Duration.between(dateTime1, dateTime2).toMillis();  
b) long diffInMilli = java.time.difference(dateTime1, dateTime2).toMillis();  
c) Date diffInMilli = java.time.Duration.between(dateTime1, dateTime2).toMillis();  
d) Time diffInMilli = java.time.Duration.between(dateTime1, dateTime2).toMillis();  
View Answer

10. How to get UTC time?  
a) Time.getUTC();  
b) Date.getUTC();  
c) Instant.now();  
d) TimeZone.getUTC();  
View Answer

Answer: c  
Explanation: In java 8, Instant.now() provides current time in UTC/GMT.

Java Questions & Answers – Literals & Variables

This section of our 1000+ Java MCQs focuses on literals & variables of Java Programming Language.

1. Which of these is long data type literal?  
a) 0x99fffL  
b) ABCDEFG  
c) 0x99fffa  
d) 99671246  
View Answer

Answer: a  
Explanation: Data type long literals are appended by an upper or lowercase L. 0x99fffL is hexadecimal long literal.

2. Which of these can be returned by the operator & ?  
a) Integer  
b) Boolean  
c) Character  
d) Integer or Boolean  
View Answer

Answer: d  
Explanation: We can use binary ampersand operator on integers/chars (and it returns an integer) or on booleans (and it returns a boolean).

3. Literals in java must be appended by which of these?  
a) L  
b) l  
c) D  
d) L and I  
View Answer

Answer: d  
Explanation: Data type long literals are appended by an upper or lowercase L.

4. Literal can be of which of these data types?  
a) integer  
b) float  
c) boolean  
d) all of the mentioned  
View Answer

Answer: d  
Explanation: None

5. Which of these can not be used for a variable name in Java?  
a) identifier  
b) keyword  
c) identifier & keyword  
d) none of the mentioned  
View Answer

Answer: b  
Explanation: Keywords are specially reserved words which can not be used for naming a user defined variable, example : class, int, for etc.

6. What is the output of this program?

1. **class** evaluate
2. {
3. **public** **static** **void** main(String args[])
4. {
5. **int** a[] = {1,2,3,4,5};
6. **int** d[] = a;
7. **int** sum = 0;
8. **for** (**int** j = 0; j < 3; ++j)
9. sum += (a[j] \* d[j + 1]) + (a[j + 1] \* d[j]);
10. System.out.println(sum);
11. }
12. }

a) 38  
b) 39  
c) 40  
d) 41  
View Answer

Answer: c  
Explanation: None  
output:  
$ javac evaluate.java  
$ java evaluate  
40

7. What is the output of this program?

1. **class** array\_output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. **int** array\_variable [] = **new** **int**[10];
6. **for** (**int** i = 0; i < 10; ++i) {
7. array\_variable[i] = i/2;
8. array\_variable[i]++;
9. System.out.print(array\_variable[i] + " ");
10. i++;
11. }
13. }
14. }

a) 0 2 4 6 8  
b) 1 2 3 4 5  
c) 0 1 2 3 4 5 6 7 8 9  
d) 1 2 3 4 5 6 7 8 9 10  
View Answer

Answer: b  
Explanation: When an array is declared using new operator then all of its elements are initialized to 0 automatically. for loop body is executed 5 times as whenever controls comes in the loop i value is incremented twice, first by i++ in body of loop then by ++i in increment condition of for loop.  
output:  
$ javac array\_output.java  
$ java array\_output  
1 2 3 4 5

8. What is the output of this program?

1. **class** variable\_scope
2. {
3. **public** **static** **void** main(String args[])
4. {
5. **int** x;
6. x = 5;
7. {
8. **int** y = 6;
9. System.out.print(x + " " + y);
10. }
11. System.out.println(x + " " + y);
12. }
13. }

a) 5 6 5 6  
b) 5 6 5  
c) Runtime error  
d) Compilation error  
View Answer

Answer: d  
Explanation: Second print statement doesn’t have access to y , scope y was limited to the block defined after initialization of x.  
output:  
$ javac variable\_scope.java  
Exception in thread “main” java.lang.Error: Unresolved compilation problem: y cannot be resolved to a variable

9. Which of these is incorrect string literal?  
a) “Hello World”  
b) “Hello\nWorld”  
c) “\”Hello World\””  
d) “Hello  
world”  
View Answer

Answer: d  
Explanation: all string literals must begin and end in same line.

10. What is the output of this program?

1. **class** dynamic\_initialization
2. {
3. **public** **static** **void** main(String args[])
4. {
5. **double** a, b;
6. a = 3.0;
7. b = 4.0;
8. **double** c = Math.sqrt(a \* a + b \* b);
9. System.out.println(c);
10. }
11. }

a) 5.0  
b) 25.0  
c) 7.0  
d) Compilation Error  
View Answer

Answer: a  
Explanation: Variable c has been dynamically initialized to square root of a \* a + b \* b, during run time.  
output:  
$ javac dynamic\_initialization.java  
$ java dynamic\_initialization  
5.0

Java Questions & Answers – Type Conversions, Promotions and Castings

This section of our 1000+ Java MCQs focuses on Type conversions, promotions and castings of Java Programming Language.

1. Which of these is necessary condition for automatic type conversion in Java?  
a) The destination type is smaller than source type  
b) The destination type is larger than source type  
c) The destination type can be larger or smaller than source type  
d) None of the mentioned  
View Answer

Answer: b  
Explanation: None.

2. What is the prototype of the default constructor of this class?  
public class prototype { }  
a) prototype( )  
b) prototype(void)  
c) public prototype(void)  
d) public prototype( )  
View Answer

Answer: d  
Explanation: None.

3. What is the error in this code?  
byte b = 50;  
b = b \* 50;  
a) b can not contain value 100, limited by its range.  
b) \* operator has converted b \* 50 into int, which can not be converted to byte without casting.  
c) b can not contain value 50.  
d) No error in this code  
View Answer

Answer: b  
Explanation: While evaluating an expression containing int, bytes or shorts , the whole expression is converted to int then evaluated and result is also of type int.

4. If an expression contains double, int, float, long, then whole expression will promoted into which of these data types?  
a) long  
b) int  
c) double  
d) float  
View Answer

Answer: c  
Explanation: If any operand is double the result of expression is double.

5. What is Truncation is Java?  
a) Floating-point value assigned to an integer type  
b) Integer value assigned to floating type  
c) Floating-point value assigned to an Floating type  
d) Integer value assigned to floating type  
View Answer

Answer: a  
Explanation: None.

6. What is the output of this program?

1. **class** char\_increment
2. {
3. **public** **static** **void** main(String args[])
4. {
5. **char** c1 = 'D';
6. **char** c2 = 84;
7. c2++;
8. c1++;
9. System.out.println(c1 + " " + c2);
10. }
11. }

a) E U  
b) U E  
c) V E  
d) U F  
View Answer

Answer: a  
Explanation: Operator ++ increments the value of character by 1. c1 and c2 are given values D and 84, when we use ++ operator their values increments by 1, c1 and c2 becomes E and U respectively.  
output:  
$ javac char\_increment.java  
$ java char\_increment  
E U

7. What is the output of this program?

1. **class** conversion
2. {
3. **public** **static** **void** main(String args[])
4. {
5. **double** a = 295.04;
6. **int** b = 300;
7. **byte** c = (**byte**) a;
8. **byte** d = (**byte**) b;
9. System.out.println(c + " " + d);
10. }
11. }

a) 38 43  
b) 39 44  
c) 295 300  
d) 295.04 300  
View Answer

Answer: b  
Explanation: Type casting a larger variable into a smaller variable results in modulo of larger variable by range of smaller variable. b contains 300 which is larger than byte’s range i:e -128 to 127 hence d contains 300 modulo 256 i:e 44.  
output:  
$ javac conversion.java  
$ java conversion  
39 44

8. What is the output of this program?

1. **class** A
2. {
3. **final** **public** **int** calculate(**int** a, **int** b) { **return** 1; }
4. }
5. **class** B **extends** A
6. {
7. **public** **int** calculate(**int** a, **int** b) { **return** 2; }
8. }
9. **public** **class** output
10. {
11. **public** **static** **void** main(String args[])
12. {
13. B object = **new** B();
14. System.out.print("b is " + b.calculate(0, 1));
15. }
16. }

a) b is : 2  
b) b is : 1  
c) Compilation Error.  
d) An exception is thrown at runtime.  
View Answer

Answer: c  
Explanation: The code does not compile because the method calculate() in class A is final and so cannot be overridden by method of class b.

9. What is the output of this program, if we run as “java main\_arguments 1 2 3”?

1. **class** main\_arguments
2. {
3. **public** **static** **void** main(String [] args)
4. {
5. String [][] argument = **new** String[2][2];
6. **int** x;
7. argument[0] = args;
8. x = argument[0].length;
9. **for** (**int** y = 0; y < x; y++)
10. System.out.print(" " + argument[0][y]);
11. }
12. }

a) 1 1  
b) 1 0  
c) 1 0 3  
d) 1 2 3  
View Answer

Answer: d  
Explanation: In argument[0] = args;, the reference variable arg[0], which was referring to an array with two elements, is reassigned to an array (args) with three elements.  
Output:  
$ javac main\_arguments.java  
$ java main\_arguments  
1 2 3

10. What is the output of this program?

1. **class** c
2. {
3. **public** **void** main( String[] args )
4. {
5. System.out.println( "Hello" + args[0] );
6. }
7. }

a) Hello c  
b) Hello  
c) Hello world  
d) Runtime Error  
View Answer

Answer: d  
Explanation: A runtime error will occur owning to the main method of the code fragment not being declared static.  
Output:  
$ javac c.java  
Exception in thread “main” java.lang.NoSuchMethodError: main

Java Questions & Answers – Arrays

This section of our 1000+ Java MCQs focuses on Array Data Structure of Java Programming Language.

1. Which of these operators is used to allocate memory to array variable in Java?  
a) malloc  
b) alloc  
c) new  
d) new malloc  
View Answer

Answer: c  
Explanation: Operator new allocates block of memory specified by the size of array, and gives the reference of memory allocated to the array variable.

2. Which of these is an incorrect array declaration?

a) **int** arr[] = **new** **int**[5]

b) **int** [] arr = **new** **int**[5]

c) **int** arr[] = **new** **int**[5]

d) **int** arr[] = **int** [5] **new**

View Answer

Answer: d  
Explanation: Operator new must be succeeded by array type and array size.

3. What will this code print?  
int arr[] = new int [5];  
System.out.print(arr);  
a) 0  
b) value stored in arr[0].  
c) 00000  
d) Class name@ hashcode in hexadecimal form  
View Answer

Answer: d  
Explanation: If we trying to print any reference variable internally, toString() will be called which is implemented to return the String in following form:  
classname@hashcode in hexadecimal form

4. Which of these is an incorrect Statement?  
a) It is necessary to use new operator to initialize an array.  
b) Array can be initialized using comma separated expressions surrounded by curly braces.  
c) Array can be initialized when they are declared.  
d) None of the mentioned  
View Answer

Answer: a  
Explanation: Array can be initialized using both new and comma separated expressions surrounded by curly braces example : int arr[5] = new int[5]; and int arr[] = { 0, 1, 2, 3, 4};

5. Which of these is necessary to specify at time of array initialization?  
a) Row  
b) Column  
c) Both Row and Column  
d) None of the mentioned  
View Answer

Answer: a  
Explanation: None.

6. What is the output of this program?

1. **class** array\_output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. **int** array\_variable [] = **new** **int**[10];
6. **for** (**int** i = 0; i < 10; ++i)
7. {
8. array\_variable[i] = i;
9. System.out.print(array\_variable[i] + " ");
10. i++;
11. }
12. }
13. }

a) 0 2 4 6 8  
b) 1 3 5 7 9  
c) 0 1 2 3 4 5 6 7 8 9  
d) 1 2 3 4 5 6 7 8 9 10  
View Answer

Answer: a  
Explanation: When an array is declared using new operator then all of its elements are initialized to 0 automatically. for loop body is executed 5 times as whenever controls comes in the loop i value is incremented twice, first by i++ in body of loop then by ++i in increment condition of for loop.  
output:  
$ javac array\_output.java  
$ java array\_output  
0 2 4 6 8

7. What is the output of this program?

1. **class** multidimention\_array
2. {
3. **public** **static** **void** main(String args[])
4. {
5. **int** arr[][] = **new** **int**[3][];
6. arr[0] = **new** **int**[1];
7. arr[1] = **new** **int**[2];
8. arr[2] = **new** **int**[3];
9. **int** sum = 0;
10. **for** (**int** i = 0; i < 3; ++i)
11. **for** (**int** j = 0; j < i + 1; ++j)
12. arr[i][j] = j + 1;
13. **for** (**int** i = 0; i < 3; ++i)
14. **for** (**int** j = 0; j < i + 1; ++j)
15. sum + = arr[i][j];
16. System.out.print(sum);
17. }
18. }

a) 11  
b) 10  
c) 13  
d) 14  
View Answer

Answer: b  
Explanation: arr[][] is a 2D array, array has been allotted memory in parts. 1st row contains 1 element, 2nd row contains 2 elements and 3rd row contains 3 elements. each element of array is given i + j value in loop. sum contains addition of all the elements of the array.  
output:  
$ javac multidimention\_array.java  
$ java multidimention\_array  
10

8. What is the output of this program?

1. **class** evaluate
2. {
3. **public** **static** **void** main(String args[])
4. {
5. **int** arr[] = **new** **int**[] {0 , 1, 2, 3, 4, 5, 6, 7, 8, 9};
6. **int** n = 6;
7. n = arr[arr[n] / 2];
8. System.out.println(arr[n] / 2);
9. }
10. }

a) 3  
b) 0  
c) 6  
d) 1  
View Answer

Answer: d  
Explanation: Array arr contains 10 elements. n contains 6 thus in next line n is given value 2 printing arr[2]/2 i:e 2/2 = 1.  
output:  
$ javac evaluate.java  
$ java evaluate  
1

9. What is the output of this program?

1. **class** array\_output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. **char** array\_variable [] = **new** **char**[10];
6. **for** (**int** i = 0; i < 10; ++i)
7. {
8. array\_variable[i] = 'i';
9. System.out.print(array\_variable[i] + "");
10. }
11. }
12. }

a) 1 2 3 4 5 6 7 8 9 10  
b) 0 1 2 3 4 5 6 7 8 9 10  
c) i j k l m n o p q r  
d) i i i i i i i i i i  
View Answer

Answer: d  
Explanation: None.  
output:  
$ javac array\_output.java  
$ java array\_output  
i i i i i i i i i i

10. What is the output of this program?

1. **class** array\_output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. **int** array\_variable[][] = {{ 1, 2, 3}, { 4 , 5, 6}, { 7, 8, 9}};
6. **int** sum = 0;
7. **for** (**int** i = 0; i < 3; ++i)
8. **for** (**int** j = 0; j < 3 ; ++j)
9. sum = sum + array\_variable[i][j];
10. System.out.print(sum / 5);
11. }
12. }

a) 8  
b) 9  
c) 10  
d) 11  
View Answer

Answer: b  
Explanation: None.  
output:  
$ javac array\_output.java  
$ java array\_output  
9

Java Questions & Answers – Data Structures-Arrays

This set of Tricky Java Questions and Answers focuses on “Data Structures-Arrays”.

1. What is the type of variable ‘b’ and ‘d’ in the below snippet?  
int a[], b;  
int []c, d;  
a) ‘b’ and ‘d’ are int  
b) ‘b’ and ‘d’ are arrays of type int  
c) ‘b’ is int variable; ‘d’ is int array  
d) ‘d’ is int variable; ‘b’ is int array  
View Answer

Answer: c  
Explanation: If [] is declared after variable it is applicable only to one variable. If [] is declared before variable it is applicable to all the variables.

2. Which of these is an incorrect array declaration?  
a) int arr[] = new int[5] ;  
b) int [] arr = new int[5] ;  
c) int arr[];  
arr = new int[5];  
d) int arr[] = int [5] new;  
View Answer

Answer: d  
Explanation: Operator new must be succeeded by array type and array size. The order is important and determines the type of variable.

3. What will this code print?

1. **int** arr[] = **new** **int** [5];
2. System.out.print(arr);

a) 0  
b) value stored in arr[0].  
c) 00000  
d) Garbage value  
View Answer

Answer: d  
Explanation: arr is an array variable, it is pointing to array of integers. Printing arr will print garbage value. It is not same as printing arr[0].

4. What is the output of below snippet?

1. Object[] names = **new** String[3];
2. names[0] = **new** Integer(0);

a) ArrayIndexOutOfBoundsException  
b) ArrayStoreException  
c) Compilation Error  
d) Code runs successfully  
View Answer

Answer: b  
Explanation: ArrayIndexOutOfBoundsException comes when code tries to access an invalid index for a given array. ArrayStoreException comes when you have stored an element of type other than the type of array.

5. Generics does not work with?  
a) Set  
b) List  
c) Tree  
d) Array  
View Answer

Answer: d  
Explanation: Generics gives the flexibility to strongly typecast collections. Generics is applicable to Set, List and Tree. It is not applicable to Array.

6. How to sort an array?  
a) Array.sort()  
b) Arrays.sort()  
c) Collection.sort()  
d) System.sort()  
View Answer

Answer: b  
Explanation: Arrays class contains various methods for manipulating arrays (such as sorting and searching). Array is not a valid class.

7. How to copy contents of array?  
a) System.arrayCopy()  
b) Array.copy()  
c) Arrays.copy()  
d) Collection.copy()  
View Answer

Answer: a  
Explanation: Arrays class contains various methods for manipulating arrays (such as sorting and searching). Array is not a valid class.

8. Can you make an array volatile?  
a) True  
b) False  
View Answer

Answer: a  
Explanation: You can only make variable pointing to array volatile. If array is changed by replacing individual elements then guarantee provided by volatile variable will not be held.

9. Where is array stored in memory?  
a) heap space  
b) stack space  
c) heap space and stack space  
d) first generation memory  
View Answer

Answer: a  
Explanation: Array is stored in heap space.Whenever an object is created, it’s always stored in the Heap space and stack memory contains the reference to it.

10. An array elements are always stored in \_\_\_\_\_\_\_\_ memory locations?  
a) Sequential  
b) Random  
c) Sequential and Random  
d) Binary search  
View Answer

Answer: a  
Explanation: Array elements are stored in contiguous memory.Linked List is stored in random memory locations.

Java Questions & Answers – Arithmetic Operators

This section of our 1000+ Java MCQs focuses on Arithmetic Operators of Java Programming Language.

1. Which of the following can be operands of arithmetic operators?  
a) Numeric  
b) Boolean  
c) Characters  
d) Both Numeric & Characters  
View Answer

Answer:d  
Explanation: The operand of arithmetic operators can be any of numeric or character type, But not boolean.

2. Modulus operator, %, can be applied to which of these?  
a) Integers  
b) Floating – point numbers  
c) Both Integers and floating – point numbers.  
d) None of the mentioned  
View Answer

Answer:c  
Explanation: Modulus operator can be applied to both integers and floating point numbers. .

3. With x = 0, which of the following are legal lines of Java code for changing the value of x to 1?  
1. x++;  
2. x = x + 1;  
3. x += 1;  
4. x =+ 1;  
a) 1, 2 & 3  
b) 1 & 4  
c) 1, 2, 3 & 4  
d) 3 & 2  
View Answer

Answer: c  
Explanation: Operator ++ increases value of variable by 1. x = x + 1 can also be written in shorthand form as x += 1. Also x =+ 1 will set the value of x to 1.

4. Decrement operator, −−, decreases value of variable by what number?  
a) 1  
b) 2  
c) 3  
d) 4  
View Answer

Answer: a  
Explanation: None.

5. Which of these statements are incorrect?  
a) Assignment operators are more efficiently implemented by Java run-time system than their equivalent long forms.  
b) Assignment operators run faster than their equivalent long forms.  
c) Assignment operators can be used only with numeric and character data type.  
d) None  
View Answer

Answer: d  
Explanation: None.

6. What is the output of this program?

1. **class** increment
2. {
3. **public** **static** **void** main(String args[])
4. {
5. **double** var1 = 1 + 5;
6. **double** var2 = var1 / 4;
7. **int** var3 = 1 + 5;
8. **int** var4 = var3 / 4;
9. System.out.print(var2 + " " + var4);
11. }
12. }

a) 1 1  
b) 0 1  
c) 1.5 1  
d) 1.5 1.0  
View Answer

Answer:c  
Explanation: None  
output:  
$ javac increment.java  
$ java increment  
1.5 1

7. What is the output of this program?

1. **class** Modulus
2. {
3. **public** **static** **void** main(String args[])
4. {
5. **double** a = 25.64;
6. **int** b = 25;
7. a = a % 10;
8. b = b % 10;
9. System.out.println(a + " " + b);
10. }
11. }

a) 5.640000000000001 5  
b) 5.640000000000001 5.0  
c) 5 5  
d) 5 5.640000000000001  
View Answer

Answer: a  
Explanation: Modulus operator returns the remainder of a division operation on the operand. a = a % 10 returns 25.64 % 10 i:e 5.640000000000001. Similarly b = b % 10 returns 5.  
output:  
$ javac Modulus.java  
$ java Modulus  
5.640000000000001 5

8. What is the output of this program?

1. **class** increment
2. {
3. **public** **static** **void** main(String args[])
4. {
5. **int** g = 3;
6. System.out.print(++g \* 8);
7. }
8. }

a) 25  
b) 24  
c) 32  
d) 33  
View Answer

Answer:c  
Explanation: Operator ++ has more preference than \*, thus g becomes 4 and when multiplied by 8 gives 32.  
output:  
$ javac increment.java  
$ java increment  
32

9. Can 8 byte long data type be automatically type cast to 4 byte float data type?  
a) True  
b) False  
View Answer

Answer: a  
Explanation: Both data types have different memory representation that’s why 8-byte integral data type can be stored to 4-byte floating point data type.

10. What is the output of this program?

1. **class** Output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. **int** a = 1;
6. **int** b = 2;
7. **int** c;
8. **int** d;
9. c = ++b;
10. d = a++;
11. c++;
12. b++;
13. ++a;
14. System.out.println(a + " " + b + " " + c);
15. }
16. }

a) 3 2 4  
b) 3 2 3  
c) 2 3 4  
d) 3 4 4  
View Answer

Answer: d  
Explanation: None.  
output:  
$ javac Output.java  
$ java Output  
3 4 4

Java Questions & Answers – Bitwise Operators

This section of our 1000+ Java MCQs focuses on Bitwise operators of Java Programming Language.

1. Which of these is not a bitwise operator?  
a) &  
b) &=  
c) |=  
d) <=  
View Answer

Answer:d  
Explanation: <= is a relational operator.

2. Which operator is used to invert all the digits in binary representation of a number?  
a) ~  
b) <<<  
c) >>>  
d) ^  
View Answer

Answer:a  
Explanation: Unary not operator, ~, inverts all of the bits of its operand in binary representation.

3. On applying Left shift operator, <<, on an integer bits are lost one they are shifted past which position bit?  
a) 1  
b) 32  
c) 33  
d) 31  
View Answer

Answer: d  
Explanation: The left shift operator shifts all of the bits in a value to the left specified number of times. For each shift left, the high order bit is shifted out and lost, zero is brought in from right. When a left shift is applied to an integer operand, bits are lost once they are shifted past the bit position 31.

4. Which right shift operator preserves the sign of the value?  
a) <<  
b) >>  
c) <<=  
d) >>=  
View Answer

Answer: b  
Explanation: None.

5. Which of these statements are incorrect?  
a) The left shift operator, <<, shifts all of the bits in a value to the left specified number of times  
b) The right shift operator, >>, shifts all of the bits in a value to the right specified number of times  
c) The left shift operator can be used as an alternative to multiplying by 2  
d) The right shift operator automatically fills the higher order bits with 0  
View Answer

6. What is the output of this program?

1. **class** bitwise\_operator
2. {
3. **public** **static** **void** main(String args[])
4. {
5. **int** var1 = 42;
6. **int** var2 = ~var1;
7. System.out.print(var1 + " " + var2);
8. }
9. }

a) 42 42  
b) 43 43  
c) 42 -43  
d) 42 43  
View Answer

Answer:c  
Explanation: Unary not operator, ~, inverts all of the bits of its operand. 42 in binary is 00101010 in using ~ operator on var1 and assigning it to var2 we get inverted value of 42 i:e 11010101 which is -43 in decimal.  
output:  
$ javac bitwise\_operator.java  
$ java bitwise\_operator  
42 -43

7. What is the output of this program?

1. **class** bitwise\_operator
2. {
3. **public** **static** **void** main(String args[])
4. {
5. **int** a = 3;
6. **int** b = 6;
7. **int** c = a | b;
8. **int** d = a & b;
9. System.out.println(c + " " + d);
10. }
11. }

a) 7 2  
b) 7 7  
c) 7 5  
d) 5 2  
View Answer

Answer: a  
Explanation: And operator produces 1 bit if both operand are 1. Or operator produces 1 bit if any bit of the two operands in 1.  
output:  
$ javac bitwise\_operator.java  
$ java bitwise\_operator  
7 2

8. What is the output of this program?

1. **class** leftshift\_operator
2. {
3. **public** **static** **void** main(String args[])
4. {
5. **byte** x = 64;
6. **int** i;
7. **byte** y;
8. i = x << 2;
9. y = (**byte**) (x << 2)
10. System.out.print(i + " " + y);
11. }
12. }

a) 0 64  
b) 64 0  
c) 0 256  
d) 256 0  
View Answer

Answer:d  
Explanation: None.  
output:  
$ javac leftshift\_operator.java  
$ java leftshift\_operator  
256 0

9. What is the output of this program?

1. **class** rightshift\_operator
2. {
3. **public** **static** **void** main(String args[])
4. {
5. **int** x;
6. x = 10;
7. x = x >> 1;
8. System.out.println(x);
9. }
10. }

a) 10  
b) 5  
c) 2  
d) 20  
View Answer

Answer: b  
Explanation: Right shift operator, >>, devides the value by 2.  
output:  
$ javac rightshift\_operator.java  
$ java rightshift\_operator  
5

10. What is the output of this program?

1. **class** Output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. **int** a = 1;
6. **int** b = 2;
7. **int** c = 3;
8. a |= 4;
9. b >>= 1;
10. c <<= 1;
11. a ^= c;
12. System.out.println(a + " " + b + " " + c);
13. }
14. }

a) 3 1 6  
b) 2 2 3  
c) 2 3 4  
d) 3 3 6  
View Answer

Answer: a  
Explanation: None.  
output:  
$ javac Output.java  
$ java Output  
3 1 6

Java Questions & Answers – Relational Operators and Boolean Logic Operators

This section of our 1000+ Java MCQs focuses on relational operators and boolean logic operators of Java Programming Language.

1. What is the output of relational operators?  
a) Integer  
b) Boolean  
c) Characters  
d) Double  
View Answer

Answer: b  
Explanation: None.

2. Which of these is returned by “greater than”, “less than” and “equal to” operators?  
a) Integers  
b) Floating – point numbers  
c) Boolean  
d) None of the mentioned  
View Answer

Answer:c  
Explanation: All relational operators return a boolean value ie. true and false.

3. Which of the following operators can operate on a boolean variable?

1. &&

2. ==

3. ?:

4. +=

a) 3 & 2  
b) 1 & 4  
c) 1, 2 & 4  
d) 1, 2 & 3  
View Answer

Answer: d  
Explanation: Operator Short circuit AND, &&, equal to, == , ternary if-then-else, ?:, are boolean logical operators. += is an arithmetic operator it can operate only on numeric values.

4. Which of these operators can skip evaluating right hand operand?  
a) !  
b) |  
c) &  
d) &&  
View Answer

Answer: d  
Explanation: Operator short circuit and, &&, and short circuit or, ||, skip evaluating right hand operand when output can be determined by left operand alone.

5. Which of these statement is correct?  
a) true and false are numeric values 1 and 0  
b) true and false are numeric values 0 and 1  
c) true is any non zero value and false is 0  
d) true and false are non numeric values  
View Answer

6. What is the output of this program?

1. **class** Relational\_operator
2. {
3. **public** **static** **void** main(String args[])
4. {
5. **int** var1 = 5;
6. **int** var2 = 6;
7. System.out.print(var1 > var2);
8. }
9. }

a) 1  
b) 0  
c) true  
d) false  
View Answer

Answer:d  
Explanation: Operator > returns a boolean value. 5 is not greater than 6 therefore false is returned.  
output:  
$ javac Relational\_operator.java  
$ java Relational\_operator  
false

7. What is the output of this program?

1. **class** bool\_operator
2. {
3. **public** **static** **void** main(String args[])
4. {
5. **boolean** a = **true**;
6. **boolean** b = !**true**;
7. **boolean** c = a | b;
8. **boolean** d = a & b;
9. **boolean** e = d ? b : c;
10. System.out.println(d + " " + e);
11. }
12. }

a) false false  
b) true ture  
c) true false  
d) false true  
View Answer

Answer: d  
Explanation: Operator | returns true if any one operand is true, thus ‘c = true | false’ is true. Operator & returns a true if both of the operand is true thus d is false. Ternary operator ?: assigns left of ‘:’ if condition is true and right hand of ‘:’ if condition is false. d is false thus e = d ? b : c , assigns c to e , e contains true.  
output:  
$ javac bool\_operator.java  
$ java bool\_operator  
false true

8. What is the output of this program?

1. **class** ternary\_operator
2. {
3. **public** **static** **void** main(String args[])
4. {
5. **int** x = 3;
6. **int** y = ~ x;
7. **int** z;
8. z = x > y ? x : y;
9. System.out.print(z);
10. }
11. }

a) 0  
b) 1  
c) 3  
d) -4  
View Answer

Answer:c  
Explanation: None.  
output:  
$ javac ternary\_operator.java  
$ java ternary\_operator  
3

9. What is the output of this program?

1. **class** Output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. **int** x , y = 1;
6. x = 10;
7. **if** (x != 10 && x / 0 == 0)
8. System.out.println(y);
9. **else**
10. System.out.println(++y);
11. }
12. }

a) 1  
b) 2  
c) Runtime error owing to division by zero in if condition  
d) Unpredictable behavior of program  
View Answer

Answer: b  
Explanation: Operator short circuit and, &&, skips evaluating right hand operand if left hand operand is false thus division by zero in if condition does not give an error.  
output:  
$ javac Output.java  
$ java Output  
2

10. What is the output of this program?

1. **class** Output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. **boolean** a = **true**;
6. **boolean** b = **false**;
7. **boolean** c = a ^ b;
8. System.out.println(!c);
9. }
10. }

a) 0  
b) 1  
c) false  
d) true  
View Answer

Answer: c  
Explanation: None.  
output:  
$ javac Output.java  
$ java Output  
false

Java Questions & Answers – Assignment Operators and Operator Precedence

This section of our 1000+ Java MCQs focuses on assignment operators and operator precedence in Java Programming Language.

1. Which of these have highest precedence?  
a) ()  
b) ++  
c) \*  
d) >>  
View Answer

Answer: a  
Explanation: Order of precedence is (highest to lowest) a -> b -> c -> d.

2. What should be expression1 evaluate to in using ternary operator as in this line?

expression1 ? expression2 : expression3

a) Integer  
b) Floating – point numbers  
c) Boolean  
d) None of the mentioned  
View Answer

Answer:c  
Explanation: The controlling condition of ternary operator must evaluate to boolean.

3. What is the value stored in x in following lines of code?

int x, y, z;

x = 0;

y = 1;

x = y = z = 8;

a) 0  
b) 1  
c) 9  
d) 8  
View Answer

Answer: d  
Explanation: None.

4. What is the order of precedence (highest to lowest) of following operators?

1. &

2. ^

3. ?:

a) 1 -> 2 -> 3  
b) 2 -> 1 -> 3  
c) 3 -> 2 -> 1  
d) 2 -> 3 -> 1  
View Answer

Answer: a  
Explanation: None.

5. Which of these statements are incorrect?  
a) Equal to operator has least precedence  
b) Brackets () have highest precedence  
c) Division operator, /, has higher precedence than multiplication operator  
d) Addition operator, +, and subtraction operator have equal precedence  
View Answer

Answer: c  
Explanation: Division operator, /, has equal precedence as of multiplication operator. In expression involving multiplication and division evaluation of expression will begin from right side when no brackets are used.

6. What is the output of this program?

1. **class** operators
2. {
3. **public** **static** **void** main(String args[])
4. {
5. **int** var1 = 5;
6. **int** var2 = 6;
7. **int** var3;
8. var3 = ++ var2 \* var1 / var2 + var2;
9. System.out.print(var3);
10. }
11. }

a) 10  
b) 11  
c) 12  
d) 56  
View Answer

Answer: c  
Explanation: Operator ++ has the highest precedence than / , \* and +. var2 is incremented to 7 and then used in expression, var3 = 7 \* 5 / 7 + 7, gives 12.  
output:  
$ javac operators.java  
$ java operators  
12

7. What is the output of this program?

1. **class** operators
2. {
3. **public** **static** **void** main(String args[])
4. {
5. **int** x = 8;
6. System.out.println(++x \* 3 + " " + x);
7. }
8. }

a) 24 8  
b) 24 9  
c) 27 8  
d) 27 9  
View Answer

Answer: d  
Explanation: Operator ++ has higher precedence than multiplication operator, \*, x is incremented to 9 than multiplied with 3 giving 27.  
output:  
$ javac operators.java  
$ java operators  
27 9

8. What is the output of this program?

1. **class** Output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. **int** x=y=z=20;
7. }
8. }

a) compile and runs fine  
b) 20  
c) run time error  
d) compile time error  
View Answer

Answer:d  
Explanation: None.

9. Which of these lines of code will give better performance?

1. a | 4 + c >> b & 7;

2. (a | ((( 4 \* c ) >> b ) & 7 ))

a) 1 will give better performance as it has no parentheses.  
b) 2 will give better performance as it has parentheses.  
c) Both 1 & 2 will give equal performance.  
d) Dependent on the computer system.  
View Answer

Answer: c  
Explanation: Parentheses do not degrade the performance of the program. Adding parentheses to reduce ambiguity does not negatively affect your system.

10. What is the output of this program?

1. **class** Output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. **int** a,b,c,d;
6. a=b=c=d=20
7. a+=b-=c\*=d/=20
8. System.out.println(a+" "+b+" "+c+" "+d);
10. }
11. }

a) compile time error  
b) runtime error  
c) a=20 b=0 c=20 d=1  
d) none of the mentioned  
View Answer

Answer: c  
Explanation: Expression will evaluate from right to left.  
output:  
$ javac Output.java  
$ java Output  
20 0 20 1

Java Questions & Answers – Control Statements – 1

This section of our 1000+ Java MCQs focuses on control statements of Java Programming Language.

1. Which of these selection statements test only for equality?  
a) if  
b) switch  
c) if & switch  
d) none of the mentioned  
View Answer

Answer: b  
Explanation: Switch statements checks for equality between the controlling variable and its constant cases.

2. Which of these are selection statements in Java?  
a) if()  
b) for()  
c) continue  
d) break  
View Answer

Answer:a  
Explanation: Continue and break are jump statements, and for is an looping statement.

3. Which of the following loops will execute the body of loop even when condition controlling the loop is initially false?  
a) do-while  
b) while  
c) for  
d) none of the mentioned  
View Answer

Answer: a  
Explanation: None.

4. Which of these jump statements can skip processing remainder of code in its body for a particular iteration?  
a) break  
b) return  
c) exit  
d) continue  
View Answer

Answer: d  
Explanation: None.

5. Which of these statement is incorrect?  
a) switch statement is more efficient than a set of nested ifs  
b) two case constants in the same switch can have identical values  
c) switch statement can only test for equality, whereas if statement can evaluate any type of boolean expression  
d) it is possible to create a nested switch statements  
View Answer

Answer: b  
Explanation: No two case constants in the same switch can have identical values.

6. What is the output of this program?

1. **class** selection\_statements
2. {
3. **public** **static** **void** main(String args[])
4. {
5. **int** var1 = 5;
6. **int** var2 = 6;
7. **if** ((var2 = 1) == var1)
8. System.out.print(var2);
9. **else**
10. System.out.print(++var2);
11. }
12. }

a) 1  
b) 2  
c) 3  
d) 4  
View Answer

Answer:b  
Explanation: var2 is initialised to 1. The conditional statement returns false and the else part gets executed.  
output:  
$ javac selection\_statements.java  
$ java selection\_statements  
2

7. What is the output of this program?

1. **class** comma\_operator
2. {
3. **public** **static** **void** main(String args[])
4. {
5. **int** sum = 0;
6. **for** (**int** i = 0, j = 0; i < 5 & j < 5; ++i, j = i + 1)
7. sum += i;
8. System.out.println(sum);
9. }
10. }

a) 5  
b) 6  
c) 14  
d) compilation error  
View Answer

Answer: b  
Explanation: Using comma operator , we can include more than one statement in the initialization and iteration portion of the for loop. Therefore both ++i and j = i + 1 is executed i gets the value – 0,1,2,3,4 & j gets the values -0,1,2,3,4,5.  
output:  
$ javac comma\_operator.java  
$ java comma\_operator  
6

8. What is the output of this program?

1. **class** jump\_statments
2. {
3. **public** **static** **void** main(String args[])
4. {
5. **int** x = 2;
6. **int** y = 0;
7. **for** ( ; y < 10; ++y)
8. {
9. **if** (y % x == 0)
10. **continue**;
11. **else** **if** (y == 8)
12. **break**;
13. **else**
14. System.out.print(y + " ");
15. }
16. }
17. }

a) 1 3 5 7  
b) 2 4 6 8  
c) 1 3 5 7 9  
d) 1 2 3 4 5 6 7 8 9  
View Answer

Answer:c  
Explanation: Whenever y is divisible by x remainder body of loop is skipped by continue statement, therefore if condition y == 8 is never true as when y is 8, remainder body of loop is skipped by continue statements of first if. Control comes to print statement only in cases when y is odd.  
output:  
$ javac jump\_statments.java  
$ java jump\_statments  
1 3 5 7 9

9. What is the output of this program?

1. **class** Output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. **final** **int** a=10,b=20;
6. **while**(a<b)
7. {
9. System.out.println("Hello");
10. }
11. System.out.println("World");
13. }
14. }

a) Hello  
b) run time error  
c) Hello world  
d) compile time error  
View Answer

Answer: d  
Explanation: Every final variable is compile time constant.

10. What is the output of this program?

1. **class** Output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. **int** a = 5;
6. **int** b = 10;
7. first:
8. {
9. second:
10. {
11. third:
12. {
13. **if** (a == b >> 1)
14. **break** second;
15. }
16. System.out.println(a);
17. }
18. System.out.println(b);
19. }
20. }
21. }

a) 5 10  
b) 10 5  
c) 5  
d) 10  
View Answer

Answer: d  
Explanation: b >> 1 in if returns 5 which is equal to a i:e 5, therefore body of if is executed and block second is exited. Control goes to end of the block second executing the last print statement, printing 10.  
output:  
$ javac Output.java  
$ java Output  
10

Java Questions & Answers – Control Statements – 2

This set of Java Multiple Choice Questions & Answers (MCQs) focuses on “Control Statements”.

1. What would be the output of the following codesnippet if variable a=10?

1. **if**(a<=0)
2. {
3. **if**(a==0)
4. {
5. System.out.println("1 ");
6. }
7. **else**
8. {
9. System.out.println("2 ");
10. }
11. }
12. System.out.println("3 ");

a) 1 2  
b) 2 3  
c) 1 3  
d) 3  
View Answer

Answer: d  
Explanation: Since the first if condition is not met, control would not go inside if statement and hence only statement after the entire if block will be executed.

2. The while loop repeats a set of code while the condition is not met?  
a) True  
b) False  
View Answer

Answer: b  
Explanation: While loop repeats a set of code only until condition is met.

3. What is true about break?  
a) Break stops the execution of entire program  
b) Break halts the execution and forces the control out of the loop  
c) Break forces the control out of the loop and starts the execution of next iteration.  
d) Break halts the execution of the loop for certain time frame  
View Answer

Answer: b  
Explanation: Break halts the execution and forces the control out of the loop.

4. What is true about do statement?  
a) do statement executes the code of a loop at least once  
b) do statement does not get execute if condition is not matched in the first iteration  
c) do statement checks the condition at the beginning of the loop  
d) do statement executes the code more than once always  
View Answer

Answer: a  
Explanation: Do statement checks the condition at the end of the loop. Hence, code gets executed at least once.

5. Which of the following is used with switch statement?  
a) Continue  
b) Exit  
c) break  
d) do  
View Answer

Answer: c  
Explanation: Break is used with switch statement to shift control out of switch.

6. What is the valid data type for variable “a” to print “Hello World”?

1. **switch**(a)
2. {
3. System.out.println("Hello World");
4. }

a) int and float  
b) byte and short  
c) char and long  
d) byte and char  
View Answer

Answer: d  
Explanation: The switch condition would only meet if variable “a” is of type byte or char.

7. Which of the following is not a decision making statement?  
a) if  
b) if-else  
c) switch  
d) do-while  
View Answer

Answer: d  
Explanation: do-while is an iteration statement. Others are decision making statements.

8. Which of the following is not a valid jump statement?  
a) break  
b) goto  
c) continue  
d) return  
View Answer

Answer: b  
Explanation: break, continue and return transfer control to another part of the program and returns back to caller after execution. However, goto is marked as not used in Java.

9. From where break statement causes an exit?  
a) Only from innermost loop  
b) Terminates a program  
c) Only from innermost switch  
d) From innermost loops or switches  
View Answer

Answer: d  
Explanation: The break statement causes an exit from innermost loop or switch.

10. Which of the following is not a valid flow control statement?  
a) exit()  
b) break  
c) continue  
d) return  
View Answer

Answer: a  
Explanation: exit() is not a flow control statement in Java. exit() terminates the currently running JVM.

# Java Questions & Answers – Concepts of OOPs

This set of Java Multiple Choice Questions & Answers (MCQs) focuses on “Concepts of OOPs”.

1. Which of the following is not OOPS concept in Java?  
a) Inheritance  
b) Encapsulation  
c) Polymorphism  
d) Compilation  
View Answer

Answer: d  
Explanation: There are 4 OOPS concepts in Java. Inheritance, Encapsulation, Polymorphism and Abstraction.

2. Which of the following is a type of polymorphism in Java?  
a) Compile time polymorphism  
b) Execution time polymorphism  
c) Multiple polymorphism  
d) Multilevel polymorphism  
View Answer

Answer: a  
Explanation: There are two type of polymorphism in Java. Compile time polymorphism (overloading) and runtime polymorphism (overriding).

3. When does method overloading is determined?  
a) At run time  
b) At compile time  
c) At coding time  
d) At execution time  
View Answer

Answer: b  
Explanation: Overloading is determined at compile time. Hence, it is also known as compile time polymorphism.

4. When Overloading does not occur?  
a) More than one method with same name but different method signature and different number or type of parameters  
b) More than one method with same name, same signature but different number of signature  
c) More than one method with same name, same signature, same number of parameters but different type  
d) More than one method with same name, same number of parameters and type but different signature  
View Answer

Answer: d  
Explanation: Overloading occurs when more than one method with same name but different constructor and also when same signature but different number of parameters and/or parameter type.

5. Which concept of Java is a way of converting real world objects in terms of class?  
a) Polymorphism  
b) Encapsulation  
c) Abstraction  
d) Inheritance  
View Answer

Answer: c  
Explanation: Abstraction is concept of defining real world objects in terms of classes or interfaces.

6. Which concept of Java is achieved by combining methods and attribute into a class?  
a) Encapsulation  
b) Inheritance  
c) Polymorphism  
d) Abstration  
View Answer

Answer: a  
Explanation: Encapsulation is implemented by combining methods and attribute into a class. The class acts like a container of encapsulating properties.

7. What is it called if an object has its own lifecycle and there is no owner?  
a) Aggregation  
b) Composition  
c) Encapsulation  
d) Association  
View Answer

Answer: d  
Explanation: It is a relationship where all objects have their own lifecycle and there is no owner. This occurs where many to many relationship is available, instead of one to one or one to many.

8. What is it called where child object gets killed if parent object is killed?  
a) Aggregation  
b) Composition  
c) Encapsulation  
d) Association  
View Answer

Answer: b  
Explanation: Composition occurs when child object gets killed if parent object gets killed. Aggregation is also known as strong Aggregation.

9. What is it called where object has its own lifecycle and child object cannot belong to another parent object?  
a) Aggregation  
b) Compostion  
c) Encapsulation  
d) Association  
View Answer

Answer: a  
Explanation: Aggregation occurs when objects have their own life cycle and child object can associate with only one parent object.

10. Method overriding is combination of inheritance and polymorphism?  
a) True  
b) false  
View Answer

Answer: a  
Explanation: In order for method overriding, method with same signation in both superclass and subclass is required with same signature. That satisfies both concepts inheritance and polymorphism.

# Java Questions & Answers – JDK-JRE-JIT-JVM

This set of Java Multiple Choice Questions & Answers (MCQs) focuses on “JDK-JRE-JIT-JVM”.

1. Which component is used to compile, debug and execute java program?  
a) JVM  
b) JDK  
c) JIT  
d) JRE  
View Answer

Answer: b  
Explanation: JDK is core component of Java Environment and provides all the tools, executables and binaries required to compile, debug and execute a Java Program.

2. Which component is responsible for converting byte code into machine specific code?  
a) JVM  
b) JDK  
c) JIT  
d) JRE  
View Answer

Answer: a  
Explanation: JVM is responsible to converting byte code to the machine specific code. JVM is also platform dependent and provides core java functions like garbage collection,memory management, security etc.

3. Which component is responsible to run java program?  
a) JVM  
b) JDK  
c) JIT  
d) JRE  
View Answer

Answer: d  
Explanation: JRE is the implementation of JVM, it provides platform to execute java programs.

4. Which component is responsible to optimise byte code to machine code?  
a) JVM  
b) JDK  
c) JIT  
d) JRE  
View Answer

Answer: c  
Explanation: JIT optimise byte code to machine specific language code by compiling similar byte codes at same time.This reduces overall time taken for compilation of byte code to machine specific language.

5. Which statement is true about java?  
a) Platform independent programming language  
b) Platform dependent programming language  
c) Code dependent programming language  
d) Sequence dependent programming language  
View Answer

6. Which of the below is invalid indentifier with main method?  
a) public  
b) static  
c) private  
d) final  
View Answer

Answer: c  
Explanation: main method cannot be private as it is invoked by external method. Other identifier are valid with main method.

7. What is the extension of java code files?  
a) .class  
b) .java  
c) .txt  
d) .js  
View Answer

Answer: b  
Explanation: Java files have .java extension.

8. What is the extension of compiled java classes?  
a) .class  
b) .java  
c) .txt  
d) .js  
View Answer

Answer: a  
Explanation: The compiled java files have .class extension.

9. How can we identify whether a compilation unit is class or interface from a .class file?  
a) Java source file header  
b) Extension of compilation unit  
c) We cannot differentiate between class and interface  
d) The class or interface name should be postfixed with unit type  
View Answer

Answer: a  
Explanation: The Java source file contains a header that declares the type of class or interface, its visibility with respect to other classes, its name and any superclass it may extend, or interface it implements.

10. What is use of interpreter?  
a) They convert byte code to machine language code  
b) They read high level code and execute them  
c) They are intermediated between JIT and JVM  
d) It is a synonym for JIT  
View Answer

Answer: b  
Explanation: Interpreters read high level language (interprets it) and execute the program. Interpreters are normally not passing through byte-code and jit compilation.

Java Questions & Answers – Class Fundamentals & Declaring objects

This section of our 1000+ Java MCQs focuses on class fundamentals & object declaration in Java Programming Language.

1. What is the stored in the object obj in following lines of code?  
box obj;  
a) Memory address of allocated memory of object  
b) NULL  
c) Any arbitrary pointer  
d) Garbage  
View Answer

Answer: b  
Explanation: Memory is allocated to an object using new operator. box obj; just declares a reference to object, no memory is allocated to it hence it points to NULL.

2. Which of these keywords is used to make a class?  
a) class  
b) struct  
c) int  
d) none of the mentioned  
View Answer

Answer: a  
Explanation: None.

3. Which of the following is a valid declaration of an object of class Box?  
a) Box obj = new Box();  
b) Box obj = new Box;  
c) obj = new Box();  
d) new Box obj;  
View Answer

Answer: a  
Explanation: None.

4. Which of these operators is used to allocate memory for an object?  
a) malloc  
b) alloc  
c) new  
d) give  
View Answer

Answer: c  
Explanation: Operator new dynamically allocates memory for an object and returns a reference to it. This reference is address in memory of the object allocated by new.

5. Which of these statement is incorrect?  
a) Every class must contain a main() method  
b) Applets do not require a main() method at all  
c) There can be only one main() method in a program  
d) main() method must be made public  
View Answer

Answer: a  
Explanation: Every class does not need to have a main() method, there can be only one main() method which is made public.

6. What is the output of this program?

1. **class** main\_class
2. {
3. **public** **static** **void** main(String args[])
4. {
5. **int** x = 9;
6. **if** (x == 9)
7. {
8. **int** x = 8;
9. System.out.println(x);
10. }
11. }
12. }

a) 9  
b) 8  
c) Compilation error  
d) Runtime error  
View Answer

Answer: c  
Explanation: Two variables with the same name can’t be created in a class.  
output:  
$ javac main\_class.java  
Exception in thread “main” java.lang.Error: Unresolved compilation problem:  
Duplicate local variable x

7. Which of the following statements is correct?  
a) Public method is accessible to all other classes in the hierarchy  
b) Public method is accessible only to subclasses of its parent class  
c) Public method can only be called by object of its class  
d) Public method can be accessed by calling object of the public class  
View Answer

Answer: a  
Explanation: None.

8. What is the output of this program?

1. **class** box
2. {
3. **int** width;
4. **int** height;
5. **int** length;
6. }
7. **class** mainclass
8. {
9. **public** **static** **void** main(String args[])
10. {
11. box obj = **new** box();
12. obj.width = 10;
13. obj.height = 2;
14. obj.length = 10;
15. **int** y = obj.width \* obj.height \* obj.length;
16. System.out.print(y);
17. }
18. }

a) 12  
b) 200  
c) 400  
d) 100  
View Answer

Answer: b  
Explanation: None.  
output:  
$ javac mainclass.java  
$ java mainclass  
200

9. What is the output of this program?

1. **class** box
2. {
3. **int** width;
4. **int** height;
5. **int** length;
6. }
7. **class** mainclass
8. {
9. **public** **static** **void** main(String args[])
10. {
11. box obj1 = **new** box();
12. box obj2 = **new** box();
13. obj1.height = 1;
14. obj1.length = 2;
15. obj1.width = 1;
16. obj2 = obj1;
17. System.out.println(obj2.height);
18. }
19. }

a) 1  
b) 2  
c) Runtime error  
d) Garbage value  
View Answer

Answer: a  
Explanation: When we assign an object to another object of same type, all the elements of right side object gets copied to object on left side of equal to, =, operator.  
output:  
$ javac mainclass.java  
$ java mainclass  
1

10. What is the output of this program?

1. **class** box
2. {
3. **int** width;
4. **int** height;
5. **int** length;
6. }
7. **class** mainclass
8. {
9. **public** **static** **void** main(String args[])
10. {
11. box obj = **new** box();
12. System.out.println(obj);
13. }
14. }

a) 0  
b) 1  
c) Runtime error  
d) classname@hashcode in hexadecimal form  
View Answer

Answer: d  
Explanation: When we print object internally toString() will be called to return string into this format classname@hashcode in hexadecimal form.  
output:  
$ javac mainclass.java  
$ java mainclass  
box@130671e

Java Questions & Answers – Introduction To Methods

This section of our 1000+ Java MCQs focuses on Methods of Java Programming Language.

1. What is the return type of a method that does not returns any value?  
a) int  
b) float  
c) void  
d) double  
View Answer

2. What is the process of defining more than one method in a class differentiated by method signature?  
a) Function overriding  
b) Function overloading  
c) Function doubling  
d) None of the mentioned  
View Answer

3. Which of the following is a method having same name as that of it’s class?  
a) finalize  
b) delete  
c) class  
d) constructor  
View Answer

4. Which method can be defined only once in a program?  
a) main method  
b) finalize method  
c) static method  
d) private method  
View Answer

5. Which of these statement is incorrect?  
a) All object of a class are allotted memory for the all the variables defined in the class  
b) If a function is defined public it can be accessed by object of other class by inheritation  
c) main() method must be made public  
d) All object of a class are allotted memory for the methods defined in the class  
View Answer

6. What is the output of this program?

1. **class** box
2. {
3. **int** width;
4. **int** height;
5. **int** length;
6. **int** volume;
7. **void** volume(**int** height, **int** length, **int** width)
8. {
9. volume = width\*height\*length;
10. }
11. }
12. **class** Prameterized\_method
13. {
14. **public** **static** **void** main(String args[])
15. {
16. box obj = **new** box();
17. obj.height = 1;
18. obj.length = 5;
19. obj.width = 5;
20. obj.volume(3,2,1);
21. System.out.println(obj.volume);
22. }
23. }

a) 0  
b) 1  
c) 6  
d) 25  
View Answer

7. What is the output of this program?

1. **class** equality
2. {
3. **int** x;
4. **int** y;
5. **boolean** isequal()
6. {
7. **return**(x == y);
8. }
9. }
10. **class** Output
11. {
12. **public** **static** **void** main(String args[])
13. {
14. equality obj = **new** equality();
15. obj.x = 5;
16. obj.y = 5;
17. System.out.println(obj.isequal());
18. }
19. }

a) false  
b) true  
c) 0  
d) 1  
View Answer

8. What is the output of this program?

1. **class** box
2. {
3. **int** width;
4. **int** height;
5. **int** length;
6. **int** volume;
7. **void** volume()
8. {
9. volume = width\*height\*length;
10. }
11. }
12. **class** Output
13. {
14. **public** **static** **void** main(String args[])
15. {
16. box obj = **new** box();
17. obj.height = 1;
18. obj.length = 5;
19. obj.width = 5;
20. obj.volume();
21. System.out.println(obj.volume);
22. }
23. }

a) 0  
b) 1  
c) 25  
d) 26  
View Answer

9. In the below code, which call to sum() method is appropriate?

1. **class** Output
2. {
4. **public** **static** **int** sum(**int** ...x)
5. {
6. **return**;
7. }
8. **static** **void** main(String args[])
9. {
10. sum(10);
11. sum(10,20);
12. sum(10,20,30);
13. sum(10,20,30,40);
14. }
15. }

a) only sum(10)  
b) only sum(10,20)  
c) only sum(10) & sum(10,20)  
d) all of the mentioned  
View Answer

10. What is the output of this program?

1. **class** area
2. {
3. **int** width;
4. **int** length;
5. **int** volume;
6. area()
7. {
8. width=5;
9. length=6;
10. }
11. **void** volume()
12. {
13. volume = width\*length\*height;
14. }
15. }
16. **class** cons\_method
17. {
18. **public** **static** **void** main(String args[])
19. {
20. area obj = **new** area();
21. obj.volume();
22. System.out.println(obj.volume);
23. }
24. }

a) 0  
b) 1  
c) 30  
d) error  
View Answer

Answer: d  
Explanation: Variable height is not defined.  
output:  
$ javac cons\_method.java  
$ java cons\_method  
error: cannot find symbol hei

Java Questions & Answers – Constructors & Garbage Collection

This section of our 1000+ Java MCQs focuses constructors and garbage collection of Java Programming Language.

1. What is the return type of Constructors?  
a) int  
b) float  
c) void  
d) none of the mentioned  
View Answer

Answer: d  
Explanation: Constructors does not have any return type, not even void.

2. Which keyword is used by method to refer to the object that invoked it?  
a) import  
b) catch  
c) abstract  
d) this  
View Answer

Answer: d  
Explanation: this keyword can be used inside any method to refer to the current object. this is always a reference to the object on which the method was invoked.

3. Which of the following is a method having same name as that of its class?  
a) finalize  
b) delete  
c) class  
d) constructor  
View Answer

Answer: d  
Explanation: A constructor is a method that initializes an object immediately upon creation. It has the same name as that of class in which it resides.

4. Which operator is used by Java run time implementations to free the memory of an object when it is no longer needed?  
a) delete  
b) free  
c) new  
d) none of the mentioned  
View Answer

Answer: d  
Explanation: Java handles deallocation of memory automatically, we do not need to explicitly delete an element. Garbage collection only occurs during execution of the program. When no references to the object exist, that object is assumed to be no longer needed, and the memory occupied by the object can be reclaimed.

5. Which function is used to perform some action when the object is to be destroyed?  
a) finalize()  
b) delete()  
c) main()  
d) none of the mentioned  
View Answer

Answer: a  
Explanation: None.

6. What is the output of this program?

1. **class** box
2. {
3. **int** width;
4. **int** height;
5. **int** length;
6. **int** volume;
7. box()
8. {
9. width = 5;
10. height = 5;
11. length = 6;
12. }
13. **void** volume()
14. {
15. volume = width\*height\*length;
16. }
17. }
18. **class** constructor\_output
19. {
20. **public** **static** **void** main(String args[])
21. {
22. box obj = **new** box();
23. obj.volume();
24. System.out.println(obj.volume);
25. }
26. }

a) 100  
b) 150  
c) 200  
d) 250  
View Answer

Answer: b  
Explanation: None.  
output:  
$ constructor\_output.java  
$ constructor\_output  
150

7. What is the output of this program?

1. **class** San
2. {
3. San()**throws** IOException
4. {
6. }
8. }
9. **class** Foundry **extends** San
10. {
11. Foundry()
12. {
14. }
15. **public** **static** **void** main(String[]args)
16. {
18. }
19. }

a) compile time error  
b) run time error  
c) compile and runs fine  
d) unreported exception java.io.IOException in default constructor  
View Answer

Answer: a  
Explanation: If parent class constructor throws any checked exception, compulsory child class constructor should throw the same checked exception as its parent, otherwise code won’t compile.

8. What is the output of this program?

1. **class** box
2. {
3. **int** width;
4. **int** height;
5. **int** length;
6. **int** volume;
7. **void** finalize()
8. {
9. volume = width\*height\*length;
10. System.out.println(volume);
11. }
12. **protected** **void** volume()
13. {
14. volume = width\*height\*length;
15. System.out.println(volume);
16. }
17. }
18. **class** Output
19. {
20. **public** **static** **void** main(String args[])
21. {
22. box obj = **new** box();
23. obj.width=5;
24. obj.height=5;
25. obj.length=6;
26. obj.volume();
27. }
28. }

a) 150  
b) 200  
c) Run time error  
d) Compilation error  
View Answer

Answer: a  
Explanation: None.  
output:  
$ javac Output.java  
$ java Output  
150

9. Which of the following statements are incorrect?  
a) default constructor is called at the time of object declaration  
b) Constructor can be parameterized  
c) finalize() method is called when a object goes out of scope and is no longer needed  
d) finalize() method must be declared protected  
View Answer

Answer: c  
Explanation: finalize() method is called just prior to garbage collection. it is not called when object goes out of scope.

10. What is the output of this program?

1. **class** area
2. {
3. **int** width;
4. **int** length;
5. **int** area;
6. **void** area(**int** width, **int** length)
7. {
8. **this**.width = width;
9. **this**.length = length;
10. }
12. }
13. **class** Output
14. {
15. **public** **static** **void** main(String args[])
16. {
17. area obj = **new** area();
18. obj.area(5 , 6);
19. System.out.println(obj.length + " " + obj.width);
20. }
21. }

a) 0 0  
b) 5 6  
c) 6 5  
d) 5 5  
View Answer

Answer: c  
Explanation: this keyword can be used inside any method to refer to the current object. this is always a reference to the object on which the method was invoked.  
output:  
$ javac Output.java  
$ java Output  
6 5

# Java Questions & Answers – Constructor

This set of Java Multiple Choice Questions & Answers (MCQs) focuses on “Constructor”.

1. What is true about private constructor?  
a) Private constructor ensures only one instance of a class exist at any point of time  
b) Private constructor ensures multiple instances of a class exist at any point of time  
c) Private constructor eases the instantiation of a class  
d) Private constructor allows creating objects in other classes  
View Answer

Answer: a  
Explanation: Object of private constructor can only be createed withing class. Private constructor is used in singleton pattern.

2. What would be the behaviour if this() and super() used in a method?  
a) Runtime error  
b) Throws exception  
c) compile time error  
d) Runs successfully  
View Answer

Answer: c  
Explanation: this() and super() cannot be used in a method. This throws compile time error.

3. What is false about constructor?  
a) Constructors cannot be synchronized in Java  
b) Java does not provide default copy constructor  
c) Constructor can be overloaded  
d) “this” and “super” can be used in a constructor  
View Answer

Answer: c  
Explanation: Default, parameterised constructors can be defined.

4. What is true about Class.getInstance()?  
a) Class.getInstance calls the constructor  
b) Class.getInstance is same as new operator  
c) Class.getInstance needs to have matching constructor  
d) Class.getInstance creates object if class does not have any constructor  
View Answer

Answer: d  
Explanation: Class class provides list of methods for use like getInstance().

5. What is true about constructor?  
a) It can contain return type  
b) It can take any number of parameters  
c) It can have any non access modifiers  
d) Constructor cannot throw exception  
View Answer

Answer: b  
Explanation: Constructor returns a new object with variables defined as in the class. Instance variables are newly created and only one copy of static variables are created.

6. Abstract class cannot have a constructor.  
a) True  
b) False  
View Answer

Answer: b  
Explanation: No instance can be created of abstract class. Only pointer can hold instance of object.

7. What is true about protected constructor?  
a) Protected constructor can be called directly  
b) Protected constructor can only be called using super()  
c) Protected constructor can be used outside package  
d) protected constructor can be instantiated even if child is in a differnt package  
View Answer

Answer: b  
Explanation: Protected access modifier means that constructor can be accessed by child classes of the parent class and classes in the same package.

8. What is not the use of “this” keyword in Java?  
a) Passing itself to another method  
b) Calling another constructor in constructor chaining  
c) Referring to the instance variable when local variable has the same name  
d) Passing itself to method of the same class  
View Answer

Answer: d  
Explanation: “this” is an important keyword in java. It helps to distinguish between local variable and variables passed in the method as parameters.

9. What would be the behaviour if one parameterized constructor is explicitly defined?  
a) Compilation error  
b) Compilation succeeds  
c) Runtime error  
d) Compilation succeeds but at the time of creating object using default constructor, it throws compilation error  
View Answer

Answer: d  
Explanation: The class compiles successfully. But the object creation of that class gives compilation error.

10. What would be behaviour if constructor has a return type?  
a) Compilation error  
b) Runtime error  
c) Compilation and runs successfully  
d) Only String return type is allowed  
View Answer

Answer: a  
Explanation:The constructor cannot have a return type. It should create and return new object. Hence it would give compilation er

# Java Questions & Answers – Heap and Garbage Collection

This set of Java Multiple Choice Questions & Answers (MCQs) focuses on “Heap and Garbage Collection”.

1. Which of the following has highest memory requirement?  
a) Heap  
b) Stack  
c) JVM  
d) Class  
View Answer

Answer: c  
Explanation: JVM is the super set which contains heap, stack, objects, pointers, etc.

2. Where is a new object allotted memory?  
a) Young space  
b) Old space  
c) Young or Old space depending on space availability  
d) JVM  
View Answer

Answer: a  
Explanation: A new object is always created in young space.Once young space is full, a special young collection is run where objects which have lived long enough are moved to old space and memory is freed up in young space for new objects.

3. Which of the following is a garbage collection technique?  
a) Cleanup model  
b) Mark and sweep model  
c) Space management model  
d) Sweep model  
View Answer

Answer: b  
Explanation: A mark and sweep garbage collection consists of two phases, the mark phase and the sweep phase.I mark phase all the objects reachable by java threads, native handles and other root sources are marked alive and others are garbage. In sweep phase, the heap is traversed to find gaps between live objects and the gaps are marked free list used for allocating memory to new objects.

4. What is -Xms and -Xmx while starting jvm?  
a) Initial; Maximum memory  
b) Maximum; Initial memory  
c) Maximum memory  
d) Initial memory  
View Answer

Answer: a  
Explanation: JVM will be started with Xms amount of memory and will be able to use a maximum of Xmx amount of memory. java -Xmx2048m -Xms256m.

5. Which exception is thrown when java is out of memory?  
a) MemoryFullException  
b) MemoryOutOfBoundsException  
c) OutOfMemoryError  
d) MemoryError  
View Answer

Answer: c  
Explanation: The Xms flag has no default value, and Xmx typically has a default value of 256MB. A common use for these flags is when you encounter a java.lang.OutOfMemoryError.

6. How to get prints of shared object memory maps or heap memory maps for a given process?  
a) jmap  
b) memorymap  
c) memorypath  
d) jvmmap  
View Answer

Answer: a  
Explanation: We can use jmap as jmap -J-d64 -heap pid .

7. What happens to thread when garbage collection kicks off?  
a) The thread continues its operation  
b) Garbage collection cannot happen until the thread is running  
c) The thread is paused while garbage collection runs  
d) The thread and garbage collection do not interfere with each other  
View Answer

Answer: c  
Explanation: The thread is paused when garbage collection runs which slows the application performance.

8. Which of the below is not a Java Profiler?  
a) JVM  
b) JConsole  
c) JProfiler  
d) Eclipse Profiler  
View Answer

Answer: a  
Explanation: Memory leak is like holding a strong reference to an object although it would never be needed anymore. Objects that are reachable but not live are considered memory leaks. Various tools help us to identify memory leaks.

9. Which of the below is not a memory leak solution?  
a) Code changes  
b) JVM parameter tuning  
c) Process restart  
d) GC parameter tuning  
View Answer

Answer: c  
Explanation: Process restart is not a permanent fix to memory leak problem. The problem will resurge again.

10. Garbage Collection can be controlled by program?  
a) True  
b) False  
View Answer

Answer: b  
Explanation: Garbage Collection cannot be controlled by program.

Java Questions & Answers – Overloading Methods & Argument Passing

This section of our 1000+ Java MCQs focuses on overloading methods & argument passing in Java Programming Language.

1. What is process of defining two or more methods within same class that have same name but different parameters declaration?  
a) method overloading  
b) method overriding  
c) method hiding  
d) none of the mentioned  
View Answer

Answer: a  
Explanation: Two or more methods can have same name as long as their parameters declaration is different, the methods are said to be overloaded and process is called method overloading. Method overloading is a way by which Java implements polymorphism.

2. Which of these can be overloaded?  
a) Methods  
b) Constructors  
c) All of the mentioned  
d) None of the mentioned  
View Answer

Answer: c  
Explanation: None.

3. Which of these is correct about passing an argument by call-by-value process?  
a) Copy of argument is made into the formal parameter of the subroutine  
b) Reference to original argument is passed to formal parameter of the subroutine  
c) Copy of argument is made into the formal parameter of the subroutine and changes made on parameters of subroutine have effect on original argument  
d) Reference to original argument is passed to formal parameter of the subroutine and changes made on parameters of subroutine have effect on original argument  
View Answer

Answer: a  
Explanation: When we pass an argument by call-by-value a copy of argument is made into the formal parameter of the subroutine and changes made on parameters of subroutine have no effect on original argument, they remain the same.

4. What is the process of defining a method in terms of itself, that is a method that calls itself?  
a) Polymorphism  
b) Abstraction  
c) Encapsulation  
d) Recursion  
View Answer

Answer: d  
Explanation: None.

5. What is the output of the following code?

1. **class** San
2. {
3. **public** **void** m1 (**int** i,**float** f)
4. {
5. System.out.println(" int float method");
6. }
8. **public** **void** m1(**float** f,**int** i);
9. {
10. System.out.println("float int method");
11. }
13. **public** **static** **void** main(String[]args)
14. {
15. San s=**new** San();
16. s.m1(20,20);
17. }
18. }

a) int float method  
b) float int method  
c) compile time error  
d) run time error  
View Answer

Answer: c  
Explanation: While resolving overloaded method, compiler automatically promotes if exact match is not found. But in this case, which one to promote is an ambiguity.

6. What is the output of this program?

1. **class** overload
2. {
3. **int** x;
4. **int** y;
5. **void** add(**int** a)
6. {
7. x = a + 1;
8. }
9. **void** add(**int** a, **int** b)
10. {
11. x = a + 2;
12. }
13. }
14. **class** Overload\_methods
15. {
16. **public** **static** **void** main(String args[])
17. {
18. overload obj = **new** overload();
19. **int** a = 0;
20. obj.add(6);
21. System.out.println(obj.x);
22. }
23. }

a) 5  
b) 6  
c) 7  
d) 8  
View Answer

Answer: c  
Explanation: None.  
output:  
$ javac Overload\_methods.java  
$ java Overload\_methods  
7

7. What is the output of this program?

1. **class** overload
2. {
3. **int** x;
4. **int** y;
5. **void** add(**int** a)
6. {
7. x = a + 1;
8. }
9. **void** add(**int** a , **int** b)
10. {
11. x = a + 2;
12. }
13. }
14. **class** Overload\_methods
15. {
16. **public** **static** **void** main(String args[])
17. {
18. overload obj = **new** overload();
19. **int** a = 0;
20. obj.add(6, 7);
21. System.out.println(obj.x);
22. }
23. }

a) 6  
b) 7  
c) 8  
d) 9  
View Answer

Answer: c  
Explanation: None.  
output:  
$ javac Overload\_methods.java  
$ java Overload\_methods  
8

8. What is the output of this program?

1. **class** overload
2. {
3. **int** x;
4. **double** y;
5. **void** add(**int** a , **int** b)
6. {
7. x = a + b;
8. }
9. **void** add(**double** c , **double** d)
10. {
11. y = c + d;
12. }
13. overload()
14. {
15. **this**.x = 0;
16. **this**.y = 0;
17. }
18. }
19. **class** Overload\_methods
20. {
21. **public** **static** **void** main(String args[])
22. {
23. overload obj = **new** overload();
24. **int** a = 2;
25. **double** b = 3.2;
26. obj.add(a, a);
27. obj.add(b, b);
28. System.out.println(obj.x + " " + obj.y);
29. }
30. }

a) 6 6  
b) 6.4 6.4  
c) 6.4 6  
d) 4 6.4  
View Answer

Answer: d  
Explanation: For obj.add(a,a); ,the function in line number 4 gets executed and value of x is 4. For the next function call, the function in line number 7 gets executed and value of y is 6.4  
output:  
$ javac Overload\_methods.java  
$ java Overload\_methods  
4 6.4

9. What is the output of this program?

1. **class** test
2. {
3. **int** a;
4. **int** b;
5. **void** meth(**int** i , **int** j)
6. {
7. i \*= 2;
8. j /= 2;
9. }
10. }
11. **class** Output
12. {
13. **public** **static** **void** main(String args[])
14. {
15. test obj = **new** test();
16. **int** a = 10;
17. **int** b = 20;
18. obj.meth(a , b);
19. System.out.println(a + " " + b);
20. }
21. }

a) 10 20  
b) 20 10  
c) 20 40  
d) 40 20  
View Answer

Answer: a  
Explanation: Variables a & b are passed by value, copy of their values are made on formal parameters of function meth() that is i & j. Therefore changes done on i & j are not reflected back on original arguments. a & b remain 10 & 20 respectively.  
output:  
$ javac Output.java  
$ java Output  
10 20

10. What is the output of this program?

1. **class** test
2. {
3. **int** a;
4. **int** b;
5. test(**int** i, **int** j)
6. {
7. a = i;
8. b = j;
9. }
10. **void** meth(test o)
11. {
12. o.a \*= 2;
13. O.b /= 2;
14. }
15. }
16. **class** Output
17. {
18. **public** **static** **void** main(String args[])
19. {
20. test obj = **new** test(10 , 20);
21. obj.meth(obj);
22. System.out.println(obj.a + " " + obj.b);
23. }
24. }

a) 10 20  
b) 20 10  
c) 20 40  
d) 40 20  
View Answer

Answer: b  
Explanation: Class objects are always passed by reference, therefore changes done are reflected back on original arguments. obj.meth(obj) sends object obj as parameter whose variables a & b are multiplied and divided by 2 respectively by meth() function of class test. a & b becomes 20 & 10 respectively.  
output:  
$ javac Output.java  
$ java Output  
20 10

Java Questions & Answers – Access Control – 1

This section of our 1000+ Java MCQs focuses on access control of Java Programming Language.

1. Which of these access specifiers must be used for main() method?  
a) private  
b) public  
c) protected  
d) none of the mentioned  
View Answer

Answer: b  
Explanation: main() method must be specified public as it called by Java run time system, outside of the program. If no access specifier is used then by default member is public within its own package & cannot be accessed by Java run time system.

2. Which of these is used to access member of class before object of that class is created?  
a) public  
b) private  
c) static  
d) protected  
View Answer

Answer: c  
Explanation: None.

3. Which of these is used as default for a member of a class if no access specifier is used for it?  
a) private  
b) public  
c) public, within its own package  
d) protected  
View Answer

Answer: a  
Explanation: When we pass an argument by call-by-value a copy of argument is made into the formal parameter of the subroutine and changes made on parameters of subroutine have no effect on original argument, they remain the same.

4. What is the process by which we can control what parts of a program can access the members of a class?  
a) Polymorphism  
b) Abstraction  
c) Encapsulation  
d) Recursion  
View Answer

Answer: c  
Explanation: None.

5. Which of the following statements are incorrect?  
a) public members of class can be accessed by any code in the program  
b) private members of class can only be accessed by other members of the class  
c) private members of class can be inherited by a sub class, and become protected members in sub class  
d) protected members of a class can be inherited by a sub class, and become private members of the sub class  
View Answer

6. What is the output of this program?

1. **class** access
2. {
3. **public** **int** x;
4. **private** **int** y;
5. **void** cal(**int** a, **int** b)
6. {
7. x = a + 1;
8. y = b;
9. }
10. }
11. **class** access\_specifier
12. {
13. **public** **static** **void** main(String args[])
14. {
15. access obj = **new** access();
16. obj.cal(2, 3);
17. System.out.println(obj.x + " " + obj.y);
18. }
19. }

a) 3 3  
b) 2 3  
c) Runtime Error  
d) Compilation Error  
View Answer

Answer: c  
Explanation: None.  
output:  
$ javac access\_specifier.java  
Exception in thread “main” java.lang.Error: Unresolved compilation problem:  
The field access.y is not visible

7. What is the output of this program?

1. **class** access
2. {
3. **public** **int** x;
4. **private** **int** y;
5. **void** cal(**int** a, **int** b)
6. {
7. x = a + 1;
8. y = b;
9. }
10. **void** print()
11. {
12. system.out.println(" " + y);
13. }
14. }
15. **class** access\_specifier
16. {
17. **public** **static** **void** main(String args[])
18. {
19. access obj = **new** access();
20. obj.cal(2, 3);
21. System.out.println(obj.x);
22. obj.print();
23. }
24. }

a) 2 3  
b) 3 3  
c) Runtime Error  
d) Compilation Error  
View Answer

Answer: b  
Explanation: None.  
output:  
$ javac access\_specifier.java  
$ java access\_specifier  
3 3

8. What is the output of this program?

1. **class** static\_out
2. {
3. **static** **int** x;
4. **static** **int** y;
5. **void** add(**int** a, **int** b)
6. {
7. x = a + b;
8. y = x + b;
9. }
10. }
11. **class** static\_use
12. {
13. **public** **static** **void** main(String args[])
14. {
15. static\_out obj1 = **new** static\_out();
16. static\_out obj2 = **new** static\_out();
17. **int** a = 2;
18. obj1.add(a, a + 1);
19. obj2.add(5, a);
20. System.out.println(obj1.x + " " + obj2.y);
21. }
22. }

a) 7 7  
b) 6 6  
c) 7 9  
d) 9 7  
View Answer

Answer: c  
Explanation: None.  
output:  
$ javac static\_use.java  
$ java static\_use  
6 6.4

9. Which of these access specifier must be used for class so that it can be inherited by another sub class?  
a) public  
b) private  
c) protected  
d) none of the mentioned  
View Answer

Answer: a  
Explanation: None.

10. What is the output of this program?

1. **class** test
2. {
3. **int** a;
4. **int** b;
5. test(**int** i, **int** j)
6. {
7. a = i;
8. b = j;
9. }
10. **void** meth(test o)
11. {
12. o.a \*= 2;
13. O.b /= 2;
14. }
15. }
16. **class** Output
17. {
18. **public** **static** **void** main(String args[])
19. {
20. test obj = **new** test(10 , 20);
21. obj.meth(obj);
22. System.out.println(obj.a + " " + obj.b);
23. }
24. }

a) 10 20  
b) 20 10  
c) 20 40  
d) 40 20  
View Answer

Answer: b  
Explanation: Class objects are always passed by reference, therefore changes done are reflected back on original arguments. obj.meth(obj) sends object obj as parameter whose variables a & b are multiplied and divided by 2 respectively by meth() function of class test. a & b becomes 20 & 10 respectively.  
output:  
$ javac Output.java  
$ java Output  
20 10

# Java Questions & Answers – Access Control – 2

This set of Java Interview Questions and Answers focuses on “Access Control – 2”.

1. Which one of the following is not an access modifier?  
a) Public  
b) Private  
c) Protected  
d) Void  
View Answer

Answer: d  
Explanation: Public, private, protected and default are the access modifiers.

2. All the variables of class should be ideally declared as ?  
a) private  
b) public  
c) protected  
d) default  
View Answer

Answer: a  
Explanation: The variables should be private and should be accessed with get and set methods.

3. Which of the following modifier means a particular variable cannot be accessed within the package?  
a) private  
b) public  
c) protected  
d) default  
View Answer

Answer: a  
Explanation: Private variables are accessible only within the class.

4. How can a protected modifier be accessed?  
a) accessible only within the class  
b) accessible only within package  
c) accessible within package and outside the package but through inheritance only  
d) accessible by all  
View Answer

Answer: c  
Explanation: The protected access modifier is accessible within package and outside the package but only through inheritance. The protected access modifier can be used with data member, method and constructor. It cannot be applied on the class.

5. What happens if constructor of class A is made private?  
a) Any class can instantiate objects of class A  
b) Objects of class A can be instantiated only within the class where it is declared  
c) Inherited class can instantiate objects of class A  
d) classes within the same package as class A can instantiate objects of class A  
View Answer

Answer: b  
Explanation: If we make any class constructor private, we cannot create the instance of that class from outside the class.

6. All the variables of interface should be ?  
a) default and final  
b) default and static  
c) public,static and final  
d) protect, static and final  
View Answer

Answer: c  
Explanation: Variables of an interface are public, static and final by default because the interfaces cannot be instantiated, final ensures the value assigned cannot be changed with the implementing class and public for it to be accessible by all the implementing classes.

7. What is true of final class?  
a) Final class cause compilation failure  
b) Final class cannot be instantiated  
c) Final class cause runtime failure  
d) Final class cannot be inherited  
View Answer

Answer: d  
Explanation: Final class cannot be inherited. This helps when we do not want classes to provide extension to these classes.

8. How many copies of static and class variables are created when 10 objects are created of a class?  
a) 1, 10  
b) 10, 10  
c) 10, 1  
d) 1, 1  
View Answer

Answer: a  
Explanation: Only one copy of static variables is created when a class is loaded. Each object instantiated has its own copy of instance variables.

9. Can a class be declared with protected modifier?  
a) True  
b) False  
View Answer

Answer: b  
Explanation: Protected class member (method or variable) is like package-private (default visibility), except that it also can be accessed from subclasses. Since there is no such concept as ‘subpackage’ or ‘package-inheritance’ in Java, declaring class protected or package-private would be the same thing.

10. Which is the modifier when there is none mentioned explicitly?  
a) protected  
b) private  
c) public  
d) default  
View Answer

Answer: d  
Explanation: Default is the access modifier when none is defined explicitly. It means the member (method or variable) can be accessed within the same package.

Java Questions & Answers – Arrays Revisited & Keyword static

This set of Java Question Bank focuses on “Arrays Revisited & Keyword static”.

1. Arrays in Java are implemented as?  
a) class  
b) object  
c) variable  
d) none of the mentioned  
View Answer

Answer: b  
Explanation: None.

2. Which of these keywords is used to prevent content of a variable from being modified?  
a) final  
b) last  
c) constant  
d) static  
View Answer

Answer: a  
Explanation: A variable can be declared final, doing so prevents its content from being modified. Final variables must be initialized when it is declared.

3. Which of these cannot be declared static?  
a) class  
b) object  
c) variable  
d) method  
View Answer

Answer: b  
Explanation: static statements are run as soon as class containing then is loaded, prior to any object declaration.

4. Which of the following statements are incorrect?  
a) static methods can call other static methods only  
b) static methods must only access static data  
c) static methods can not refer to this or super in any way  
d) when object of class is declared, each object contains its own copy of static variables  
View Answer

Answer: d  
Explanation: All objects of class share same static variable, when object of a class are declared, all the objects share same copy of static members, no copy of static variables are made.

5. Which of the following statements are incorrect?  
a) Variables declared as final occupy memory  
b) final variable must be initialized at the time of declaration  
c) Arrays in java are implemented as an object  
d) All arrays contain an attribute-length which contains the number of elements stored in the array  
View Answer

Answer: a  
Explanation: None.

6. Which of these methods must be made static?  
a) main()  
b) delete()  
c) run()  
d) finalize()  
View Answer

Answer: a  
Explanation: main() method must be declared static, main() method is called by Java’s run time system before any object of any class exists.

7. What is the output of this program?

1. **class** access
2. {
3. **public** **int** x;
4. **static** **int** y;
5. **void** cal(**int** a, **int** b)
6. {
7. x += a ;
8. y += b;
9. }
10. }
11. **class** static\_specifier
12. {
13. **public** **static** **void** main(String args[])
14. {
15. access obj1 = **new** access();
16. access obj2 = **new** access();
17. obj1.x = 0;
18. obj1.y = 0;
19. obj1.cal(1, 2);
20. obj2.x = 0;
21. obj2.cal(2, 3);
22. System.out.println(obj1.x + " " + obj2.y);
23. }
24. }

a) 1 2  
b) 2 3  
c) 3 2  
d) 1 5  
View Answer

Answer: d  
Explanation: None.  
output:  
$ javac static\_specifier.java  
$ java static\_specifier  
1 5

8. What is the output of this program?

1. **class** access
2. {
3. **static** **int** x;
4. **void** increment()
5. {
6. x++;
7. }
8. }
9. **class** static\_use
10. {
11. **public** **static** **void** main(String args[])
12. {
13. access obj1 = **new** access();
14. access obj2 = **new** access();
15. obj1.x = 0;
16. obj1.increment();
17. obj2.increment();
18. System.out.println(obj1.x + " " + obj2.x);
19. }
20. }

a) 1 2  
b) 1 1  
c) 2 2  
d) Compilation Error  
View Answer

Answer: c  
Explanation: All objects of class share same static variable, all the objects share same copy of static members, obj1.x and obj2.x refer to same element of class which has been incremented twice and its value is 2.  
output:  
$ javac static\_use.java  
$ java static\_use  
2 2

9. What is the output of this program?

1. **class** static\_out
2. {
3. **static** **int** x;
4. **static** **int** y;
5. **void** add(**int** a , **int** b)
6. {
7. x = a + b;
8. y = x + b;
9. }
10. }
11. **class** static\_use
12. {
13. **public** **static** **void** main(String args[])
14. {
15. static\_out obj1 = **new** static\_out();
16. static\_out obj2 = **new** static\_out();
17. **int** a = 2;
18. obj1.add(a, a + 1);
19. obj2.add(5, a);
20. System.out.println(obj1.x + " " + obj2.y);
21. }
22. }

a) 7 7  
b) 6 6  
c) 7 9  
d) 9 7  
View Answer

Answer: c  
Explanation: None.  
output:  
$ javac static\_use.java  
$ java static\_use  
7 9

10. What is the output of this program?

1. **class** Output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. **int** arr[] = {1, 2, 3, 4, 5};
6. **for** ( **int** i = 0; i < arr.length - 2; ++i)
7. System.out.println(arr[i] + " ");
8. }
9. }

a) 1 2  
b) 1 2 3  
c) 1 2 3 4  
d) 1 2 3 4 5  
View Answer

Answer: b  
Explanation: arr.length() is 5, so the loop is executed for three times.  
output:  
$ javac Output.java  
$ java Output  
1 2 3

11. What is the output of this program?

1. **class** Output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. **int** a1[] = **new** **int**[10];
6. **int** a2[] = {1, 2, 3, 4, 5};
7. System.out.println(a1.length + " " + a2.length);
8. }
9. }

a) 10 5  
b) 5 10  
c) 0 10  
d) 0 5  
View Answer

Answer: a  
Explanation: Arrays in java are implemented as objects, they contain an attribute that is length which contains the number of elements that can be stored in the array. Hence a1.length gives 10 and a2.length gives 5.  
output:  
$ javac Output.java  
$ java Output  
10

Java Questions & Answers – String Class

This section of our 1000+ Java MCQs focuses on String class of Java Programming Language.

1. String in Java is a?  
a) class  
b) object  
c) variable  
d) character array  
View Answer

Answer: a  
Explanation: None.

2. Which of these method of String class is used to obtain character at specified index?  
a) char()  
b) Charat()  
c) charat()  
d) charAt()  
View Answer

Answer: d  
Explanation: None.

3. Which of these keywords is used to refer to member of base class from a sub class?  
a) upper  
b) super  
c) this  
d) none of the mentioned  
View Answer

Answer: b  
Explanation: Whenever a subclass needs to refer to its immediate superclass, it can do so by use of the keyword super.

4. Which of these method of String class can be used to test to strings for equality?  
a) isequal()  
b) isequals()  
c) equal()  
d) equals()  
View Answer

Answer: d  
Explanation: None.

5. Which of the following statements are incorrect?  
a) String is a class  
b) Strings in java are mutable  
c) Every string is an object of class String  
d) Java defines a peer class of String, called StringBuffer, which allows string to be altered  
View Answer

Answer: b  
Explanation: Strings in Java are immutable that is they can not be modified.

6. What is the output of this program?

1. **class** string\_demo
2. {
3. **public** **static** **void** main(String args[])
4. {
5. String obj = "I" + "like" + "Java";
6. System.out.println(obj);
7. }
8. }

a) I  
b) like  
c) Java  
d) IlikeJava  
View Answer

Answer: d  
Explanation: Java defines an operator +, it is used to concatenate strings.  
output:  
$ javac string\_demo.java  
$ java string\_demo  
IlikeJava

7. What is the output of this program?

1. **class** string\_class
2. {
3. **public** **static** **void** main(String args[])
4. {
5. String obj = "I LIKE JAVA";
6. System.out.println(obj.charAt(3));
7. }
8. }

a) I  
b) L  
c) K  
d) E  
View Answer

Answer: a  
Explanation: charAt() is a method of class String which gives the character specified by the index. obj.charAt(3) gives 4th character i:e I.  
output:  
$ javac string\_class.java  
$ java string\_class  
I

8. What is the output of this program?

1. **class** string\_class
2. {
3. **public** **static** **void** main(String args[])
4. {
5. String obj = "I LIKE JAVA";
6. System.out.println(obj.length());
7. }
8. }

a) 9  
b) 10  
c) 11  
d) 12  
View Answer

Answer: c  
Explanation: None.  
output:  
$ javac string\_class.java  
$ java string\_class  
11

9. What is the output of this program?

1. **class** string\_class
2. {
3. **public** **static** **void** main(String args[])
4. {
5. String obj = "hello";
6. String obj1 = "world";
7. String obj2 = obj;
8. obj2 = " world";
9. System.out.println(obj + " " + obj2);
10. }
11. }

a) hello hello  
b) world world  
c) hello world  
d) world hello  
View Answer

Answer: c  
Explanation: None.  
output:  
$ javac string\_class.java  
$ java string\_class  
hello world

10. What is the output of this program?

1. **class** string\_class
2. {
3. **public** **static** **void** main(String args[])
4. {
5. String obj = "hello";
6. String obj1 = "world";
7. String obj2 = "hello";
8. System.out.println(obj.equals(obj1) + " " + obj.equals(obj2));
9. }
10. }

a) false false  
b) true true  
c) true false  
d) false true  
View Answer

Answer: d  
Explanation: equals() is method of class String, it is used to check equality of two String objects, if they are equal, true is retuned else false.  
output:  
$ javac string\_class.java  
$ java string\_class  
false true

Java Questions & Answers – Methods Taking Parameters

This set of Java Questions and Answers for Entrance exams focuses on “Methods Taking Parameters”.

1. Which of these is the method which is executed first before execution of any other thing takes place in a program?  
a) main method  
b) finalize method  
c) static method  
d) private method  
View Answer

Answer: c  
Explanation: If a static method is present in the program then it will be executed first, then main will be executed.

2. What is the process of defining more than one method in a class differentiated by parameters?  
a) Function overriding  
b) Function overloading  
c) Function doubling  
d) None of the mentioned  
View Answer

Answer:b  
Explanation: Function overloading is a process of defining more than one method in a class with same name differentiated by function signature i:e return type or parameters type and number. Example – int volume(int length, int width) & int volume(int length , int width , int height) can be used to calculate volume.

3. Which of these can be used to differentiate two or more methods having same name?  
a) Parameters data type  
b) Number of parameters  
c) Return type of method  
d) All of the mentioned  
View Answer

Answer: d  
Explanation: None.

4. Which of these data type can be used for a method having a return statement in it?  
a) void  
b) int  
c) float  
d) both int and float  
View Answer

Answer: d  
Explanation: None.

5. Which of these statement is incorrect?  
a) Two or more methods with same name can be differentiated on the basis of their parameters data type  
b) Two or more method having same name can be differentiated on basis of number of parameters  
c) Any already defined method in java’s library can be defined again in the program with different data type of parameters  
d) If a method is returning a value the calling statement must have a variable to store that value  
View Answer

Answer: d  
Explanation: Even if a method is returning a value, it is not necessary to store that value.

6. What is the output of this program?

1. **class** box
2. {
3. **int** width;
4. **int** height;
5. **int** length;
6. **int** volume;
7. **void** volume(**int** height, **int** length, **int** width)
8. {
9. volume = width \* height \* length;
10. }
11. }
12. **class** Prameterized\_method{
13. **public** **static** **void** main(String args[])
14. {
15. box obj = **new** box();
16. obj.height = 1;
17. obj.length = 5;
18. obj.width = 5;
19. obj.volume(3, 2, 1);
20. System.out.println(obj.volume);
21. }
22. }

a) 0  
b) 1  
c) 6  
d) 25  
View Answer

Answer: c  
Explanation: None  
output:  
$ Prameterized\_method.java  
$ Prameterized\_method  
6

7. What is the output of this program?

1. **class** equality
2. {
3. **int** x;
4. **int** y;
5. **boolean** isequal()
6. {
7. **return**(x == y);
8. }
9. }
10. **class** Output
11. {
12. **public** **static** **void** main(String args[])
13. {
14. equality obj = **new** equality();
15. obj.x = 5;
16. obj.y = 5;
17. System.out.println(obj.isequal);
18. }
19. }

a) false  
b) true  
c) 0  
d) 1  
View Answer

Answer: b  
Explanation: None  
output:  
$ javac Output.java  
$ java Output  
true

8. What is the output of this program?

1. **class** box
2. {
3. **int** width;
4. **int** height;
5. **int** length;
6. **int** volume;
7. **void** volume()
8. {
9. volume = width \* height \* length;
10. }
11. **void** volume(**int** x)
12. {
13. volume = x;
14. }
15. }
16. **class** Output
17. {
18. **public** **static** **void** main(String args[])
19. {
20. box obj = **new** box();
21. obj.height = 1;
22. obj.length = 5;
23. obj.width = 5;
24. obj.volume(5);
25. System.out.println(obj.volume);
26. }
27. }

a) 0  
b) 5  
c) 25  
d) 26  
View Answer

Answer:b  
Explanation: None.  
output:  
$ javac Output.java  
$ java Output  
5

9. What is the output of this program?

1. **class** Output
2. {
3. **static** **void** main(String args[])
4. {
5. **int** x , y = 1;
6. x = 10;
7. **if**(x != 10 && x / 0 == 0)
8. System.out.println(y);
9. **else**
10. System.out.println(++y);
11. }
12. }

a) 1  
b) 2  
c) Runtime Error  
d) Compilation Error  
View Answer

Answer: d  
Explanation: main() method must be made public. Without main() being public java run time system will not be able to access main() and will not be able to execute the code.  
output:  
$ javac Output.java  
Error: Main method not found in class Output, please define the main method as:  
public static void main(String[] args)

10. What is the output of this program?

1. **class** area
2. {
3. **int** width;
4. **int** length;
5. **int** height;
6. area()
7. {
8. width = 5;
9. length = 6;
10. height = 1;
11. }
12. **void** volume()
13. {
14. volume = width \* height \* length;
15. }
16. }
17. **class** cons\_method
18. {
19. **public** **static** **void** main(String args[])
20. {
21. area obj = **new** area();
22. obj.volume();
23. System.out.println(obj.volume);
24. }
25. }

a) 0  
b) 1  
c) 25  
d) 30  
View Answer

Answer: d  
Explanation: None.  
output:  
$ javac cons\_method.java  
$ java cons\_method  
30

Java Questions & Answers – Command Line Arguments – 1

This section of our 1000+ Java MCQs focuses on Command Line Arguments in Java Programming Language.

1. Which of these method is given parameter via command line arguments?  
a) main()  
b) recursive() method  
c) Any method  
d) System defined methods  
View Answer

Answer: a  
Explanation: Only main() method can be given parameters via using command line arguments.

2. Which of these data types is used to store command line arguments?  
a) Array  
b) Stack  
c) String  
d) Integer  
View Answer

Answer: c  
Explanation: None.

3. How many arguments can be passed to main()?  
a) Infinite  
b) Only 1  
c) System Dependent  
d) None of the mentioned  
View Answer

Answer: a  
Explanation: None.

4. Which of these is a correct statement about args in this line of code?  
public static viod main(String args[])  
a) args is a String  
b) args is a Character  
c) args is an array of String  
d) args in an array of Character  
View Answer

Answer: c  
Explanation: args in an array of String.

5. Can command line arguments be converted into int automatically if required?  
a) Yes  
b) No  
c) Compiler Dependent  
d) Only ASCII characters can be converted  
View Answer

Answer: b  
Explanation: All command Line arguments are passed as a string. We must convert numerical value to their internal forms manually.

6. What is the output of this program, Command line execution is done as – “java Output This is a command Line”?

1. **class** Output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. System.out.print("args[0]");
6. }
7. }

a) java  
b) Oupput  
c) This  
d) is  
View Answer

Answer: c  
Explanation: None.  
Output:  
$ javac Output.javac  
java Output This is a command Line  
This

7. What is the output of this program, Command line exceution is done as – “java Output This is a command Line”?

1. **class** Output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. System.out.print("args[3]");
6. }
7. }

a) java  
b) is  
c) This  
d) command  
View Answer

Answer: d  
Explanation: None.  
Output:  
$ javac Output.javac  
java Output This is a command Line  
command

8. What is the output of this program, Command line execution is done as – “java Output This is a command Line”?

1. **class** Output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. System.out.print("args");
6. }
7. }

a) This  
b) java Output This is a command Line  
c) This is a command Line  
d) Compilation Error  
View Answer

Answer: c  
Explanation: None.  
Output:  
$ javac Output.javac  
java Output This is a command Line  
This is a command Line

9. What is the output of this program, Command line execution is done as – “java Output command Line 10 A b 4 N”?

1. **class** Output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. System.out.print("(int)args[2] \* 2");
6. }
7. }

a) java  
b) 10  
c) 20  
d) b  
View Answer

Answer: c  
Explanation: None.  
Output:  
$ javac Output.javac  
java Output command Line 10 A b 4 N  
20

10. What is the output of this program, Command line execution is done as – “java Output command Line 10 A b 4 N”?

1. **class** Output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. System.out.print("args[6]");
6. }
7. }

a) java  
b) 10  
c) b  
d) N  
View Answer

Answer: d  
Explanation: None.  
Output:  
$ javac Output.javac  
java Output command Line 10 A b 4 N  
N

Java Questions & Answers – Command Line Arguments – 2

This set of Java Questions and Answers for Freshers focuses on “Command Line Arguments – 2”.

1. What would be the output of following snippet, if attempted to compile and run this code with command line argument “java abc Rakesh Sharma”?

1. **public** **class** abc
2. {
3. **int** a=2000;
4. **public** **static** **void** main(String argv[])
5. {
6. System.out.println(argv[1]+" :-Please pay Rs."+a);
7. }
8. }

a) Compile time error  
b) Compilation but runtime error  
c) Compilation and output Rakesh :-Please pay Rs.2000  
d) Compilation and output Sharma :-Please pay Rs.2000  
View Answer

Answer: a  
Explanation: Main method is static and cannot access non static variable a.

2. What would be the output of following snippet, if attempted to compile and execute?

1. **class** abc
2. {
3. **public** **static** **void** main(String args[])
4. {
5. **if**(args.length>0)
6. System.out.println(args.length);
7. }
8. }

a) The snippet compiles, runs and prints 0  
b) The snippet compiles, runs and prints 1  
c) The snippet does not compile  
d) The snippet compiles and runs but does not print anything  
View Answer

Answer: d  
Explanation: As no argument is passed to the code, the length of args is 0. So the code will not print.

3. What would be the output of following snippet, if compiled and executed with command line argument “java abc 1 2 3”?

1. **public** **class** abc
2. {
3. **static** **public** **void** main(String [] xyz)
4. {
5. **for**(**int** n=1;n<xyz.length; n++)
6. {
7. System.out.println(xyz[n]+"");
8. }
9. }
10. }

a) 1 2  
b) 2 3  
c) 1 2 3  
d) Compilation error  
View Answer

Answer: b  
Explantion: The index of array starts with 0. Since the loop is starting with 1 it will print 2 3.

4. What is the output of the following snippet running with “java demo I write java code”?

1. **public** **class** demo
2. {
3. **public** **static** **void** main(String args[])
4. {
5. System.out.println(args[0]+""+args[args.length-1]);
6. }
7. }

a) The snippet compiles, runs and prints “java demo”  
b) The snippet compiles, runs and prints “java code”  
c) The snippet compiles, runs and prints “demo code”  
d) The snippet compiles, runs and prints “I code”  
View Answer

Answer: d  
Explanation: The index of array starts with 0 till length – 1. Hence it would print “I code”.

5. What would be the output of the following snippet, if compiled and executed with command line “hello there”?

1. **public** **class** abc
2. {
3. String[] xyz;
5. **public** **static** **void** main(String argv[])
6. {
7. xyz=argv;
8. }
10. **public** **void** runMethod()
11. {
12. System.out.println(argv[1]);
13. }
14. }

a) Compile time error  
b) Output would be “hello”  
c) Output would be “there”  
d) Output would be “hello there”  
View Answer

Answer:a  
Explanation: Error would be “Cannot make static reference to a non static variable”. Even if main method was not static, the array argv is local to the main method and would not be visible within runMethod.

6. How do we pass command line argument in Eclipse?  
a) Arguments tab  
b) Variable tab  
c) Cannot pass command line argument in eclipse  
d) Environment variable tab  
View Answer

Answer: a  
Explanation: Arguments tab is used to pass command line argument in eclipse.

7. Which class allows parsing of command line arguments?  
a) Args  
b) JCommander  
c) CommandLine  
d) Input  
View Answer

Answer: b  
Explanation: JCommander is a very small Java framework that makes it trivial to parse command line parameters.

8. Which annotation is used to represent command line input and assigned to correct data type?  
a) @Input  
b) @Variable  
c) @CommandLine  
d) @Parameter  
View Answer

Answer: d  
Explanation: @Parameter, @Parameter(names = { “-log”, “-verbose” }, description = “Level of verbosity”), etc are various forms of using @Parameter

9. What is the output of below snippet run as $ java Demo –length 512 –breadth 2 -h 3 ?

1. **class** Demo {
2. @Parameter(names={"--length"})
3. **int** length;
5. @Parameter(names={"--breadth"})
6. **int** breadth;
8. @Parameter(names={"--height","-h"})
9. **int** height;
11. **public** **static** **void** main(String args[])
12. {
13. Demo demo = **new** Demo();
14. **new** JCommander(demo, args);
15. demo.run();
16. }
18. **public** **void** run()
19. {
20. System.out.println(length+" "+ breadth+" "+height);
21. }
22. }

a) 2 512 3  
b) 2 2 3  
c) 512 2 3  
d) 512 512 3  
View Answer

Answer: c  
Explanation: JCommander helps easily pass commandline arguments. @Parameter assigns input to desired parameter.

10. What is the use of @syntax?  
a) Allows multiple parameters to be passed  
b) Allows one to put all your options into a file and pass this file as parameter  
c) Allows one to pass only one parameter  
d) Allows one to pass one file containing only one parameter  
View Answer

Answer: b  
Explanation: JCommander supports the @syntax, which allows us to put all our options into a file and pass this file as parameter.  
/tmp/parameters  
-verbose  
file1  
file2  
$ java Main @/tmp/parameters

Java Questions & Answers – Recursion

This section of our 1000+ Java MCQs focuses on recursion of Java Programming Language.

1. What is Recursion in Java?  
a) Recursion is a class  
b) Recursion is a process of defining a method that calls other methods repeatedly  
c) Recursion is a process of defining a method that calls itself repeatedly  
d) Recursion is a process of defining a method that calls other methods which in turn call again this method  
View Answer

Answer: b  
Explanation: Recursion is the process of defining something in terms of itself. It allows us to define method that calls itself.

2. Which of these data types is used by operating system to manage the Recursion in Java?  
a) Array  
b) Stack  
c) Queue  
d) Tree  
View Answer

Answer: b  
Explanation: Recursions are always managed by using stack.

3. Which of these will happen if recursive method does not have a base case?  
a) An infinite loop occurs  
b) System stops the program after some time  
c) After 1000000 calls it will be automatically stopped  
d) None of the mentioned  
View Answer

Answer: a  
Explanation: If a recursive method does not have a base case then an infinite loop occurs which results in stackoverflow.

4. Which of these is not a correct statement?  
a) A recursive method must have a base case  
b) Recursion always uses stack  
c) Recursive methods are faster that programmers written loop to call the function repeatedly using a stack  
d) Recursion is managed by Java’s Run – Time environment  
View Answer

Answer: d  
Explanation: Recursion is always managed by operating system.

5. Which of these packages contains the exception Stackoverflow in Java?  
a) java.lang  
b) java.util  
c) java.io  
d) java.system  
View Answer

Answer: a  
Explanation: None.

6. What is the output of this program?

1. **class** recursion
2. {
3. **int** func (**int** n)
4. {
5. **int** result;
6. result = func (n - 1);
7. **return** result;
8. }
9. }
10. **class** Output
11. {
12. **public** **static** **void** main(String args[])
13. {
14. recursion obj = **new** recursion() ;
15. System.out.print(obj.func(12));
16. }
17. }

a) 0  
b) 1  
c) Compilation Error  
d) Runtime Error  
View Answer

Answer: d  
Explanation: Since the base case of the recursive function func() is not defined hence infinite loop occurs and results in stackoverflow.  
Output:  
$ javac Output.javac  
$ java Output  
Exception in thread “main” java.lang.StackOverflowError

7. What is the output of this program?

1. **class** recursion
2. {
3. **int** func (**int** n)
4. {
5. **int** result;
6. **if** (n == 1)
7. **return** 1;
8. result = func (n - 1);
9. **return** result;
10. }
11. }
12. **class** Output
13. {
14. **public** **static** **void** main(String args[])
15. {
16. recursion obj = **new** recursion() ;
17. System.out.print(obj.func(5));
18. }
19. }

a) 0  
b) 1  
c) 120  
d) None of the mentioned  
View Answer

Answer: b  
Explanation: None.  
Output:  
$ javac Output.javac  
$ java Output  
1

8. What is the output of this program?

1. **class** recursion
2. {
3. **int** fact(**int** n)
4. {
5. **int** result;
6. **if** (n == 1)
7. **return** 1;
8. result = fact(n - 1) \* n;
9. **return** result;
10. }
11. }
12. **class** Output
13. {
14. **public** **static** **void** main(String args[])
15. {
16. recursion obj = **new** recursion() ;
17. System.out.print(obj.fact(5));
18. }
19. }

a) 24  
b) 30  
c) 120  
d) 720  
View Answer

Answer: c  
Explanation: fact() method recursively calculates factorial of a number, when value of n reaches 1, base case is excuted and 1 is returned.  
Output:  
$ javac Output.javac  
$ java Output  
120

9. What is the output of this program?

1. **class** recursion
2. {
3. **int** fact(**int** n)
4. {
5. **int** result;
6. **if** (n == 1)
7. **return** 1;
8. result = fact(n - 1) \* n;
9. **return** result;
10. }
11. }
12. **class** Output
13. {
14. **public** **static** **void** main(String args[])
15. {
16. recursion obj = **new** recursion() ;
17. System.out.print(obj.fact(1));
18. }
19. }

a) 1  
b) 30  
c) 120  
d) Runtime Error  
View Answer

Answer: a  
Explanation: fact() method recursively calculates factorial of a number, when value of n reaches 1, base case is excuted and 1 is returned.  
Output:  
$ javac Output.javac  
$ java Output  
1

10. What is the output of this program?

1. **class** recursion
2. {
3. **int** fact(**int** n)
4. {
5. **int** result;
6. **if** (n == 1)
7. **return** 1;
8. result = fact(n - 1) \* n;
9. **return** result;
10. }
11. }
12. **class** Output
13. {
14. **public** **static** **void** main(String args[])
15. {
16. recursion obj = **new** recursion() ;
17. System.out.print(obj.fact(6));
18. }
19. }

a) 1  
b) 30  
c) 120  
d) 720  
View Answer

Answer: d  
Explanation: None.  
Output:  
$ javac Output.javac  
$ java Output  
720

Java Questions & Answers – Method overriding

This section of our 1000+ Java MCQs focuses on method overriding in Java Programming Language.

1. Which of these keyword can be used in subclass to call the constructor of superclass?  
a) super  
b) this  
c) extent  
d) extends  
View Answer

Answer: a  
Explanation: None.

2. What is the process of defining a method in subclass having same name & type signature as a method in its superclass?  
a) Method overloading  
b) Method overriding  
c) Method hiding  
d) None of the mentioned  
View Answer

Answer: b  
Explanation: None.

3. Which of these keywords can be used to prevent Method overriding?  
a) static  
b) constant  
c) protected  
d) final  
View Answer

Answer: d  
Explanation: To disallow a method from being overridden, specify final as a modifier at the start of its declaration. Methods declared as final cannot be overridden.

4. Which of these is correct way of calling a constructor having no parameters, of superclass A by subclass B?  
a) super(void);  
b) superclass.();  
c) super.A();  
d) super();  
View Answer

Answer: d  
Explanation: None.

5. At line number 2 below, choose 3 valid data-type attributes/qualifiers among “final, static, native, public, private, abstract, protected”

1. **public** **interface** Status
2. {
3. */\* insert qualifier here \*/* **int** MY\_VALUE = 10;
4. }

a) final, native, private  
b) final, static, protected  
c) final, private, abstract  
d) final, static, public  
View Answer

Answer: d  
Explanation: Every interface variable is implicitly public static and final.

6. Which of these is supported by method overriding in Java?  
a) Abstraction  
b) Encapsulation  
c) Polymorphism  
d) None of the mentioned  
View Answer

Answer: c  
Explanation: None.

7. What is the output of this program?

1. **class** Alligator
2. {
3. **public** **static** **void** main(String[] args)
4. {
5. **int** []x[] = {{1,2}, {3,4,5}, {6,7,8,9}};
6. **int** [][]y = x;
7. System.out.println(y[2][1]);
8. }
9. }

a) 2  
b) 3  
c) 7  
d) Compilation Error  
View Answer

Answer: c  
Explanation: Both x,and y are pointing to the same array.

8. What is the output of this program?

1. **final** **class** A
2. {
3. **int** i;
4. }
5. **class** B **extends** A
6. {
7. **int** j;
8. System.out.println(j + " " + i);
9. }
10. **class** inheritance
11. {
12. **public** **static** **void** main(String args[])
13. {
14. B obj = **new** B();
15. obj.display();
16. }
17. }

a) 2 2  
b) 3 3  
c) Runtime Error  
d) Compilation Error  
View Answer

Answer: d  
Explanation: class A has been declared final hence it cannot be inherited by any other class. Hence class B does not have member i, giving compilation error.  
output:  
$ javac inheritance.java  
Exception in thread “main” java.lang.Error: Unresolved compilation problem:  
i cannot be resolved or is not a field

9. What is the output of this program?

1. **class** Abc
2. {
3. **public** **static** **void** main(String[]args)
4. {
5. String[] elements = { "for", "tea", "too" };
6. String first = (elements.length > 0) ? elements[0]: **null**;
7. }
8. }

a) Compilation error  
b) An exception is thrown at run time  
c) The variable first is set to null  
d) The variable first is set to elements[0].  
View Answer

Answer: d  
Explanation: The value at the 0th position will be assigned to the variable first.

10. What is the output of this program?

1. **class** A
2. {
3. **int** i;
4. **public** **void** display()
5. {
6. System.out.println(i);
7. }
8. }
9. **class** B **extends** A
10. {
11. **int** j;
12. **public** **void** display()
13. {
14. System.out.println(j);
15. }
16. }
17. **class** Dynamic\_dispatch
18. {
19. **public** **static** **void** main(String args[])
20. {
21. B obj2 = **new** B();
22. obj2.i = 1;
23. obj2.j = 2;
24. A r;
25. r = obj2;
26. r.display();
27. }
28. }

a) 1  
b) 2  
c) 3  
d) 4  
View Answer

Answer: b  
Explanation: r is reference of type A, the program assigns a reference of object obj2 to r and uses that reference to call function display() of class B.  
output:  
$ javac Dynamic\_dispatch.java  
$ java Dynamic\_dispatch  
2

Java Questions & Answers – The Object Class

This section of our 1000+ Java MCQs focuses on Object class of Java Programming Language.

1. Which of these class is superclass of every class in Java?  
a) String class  
b) Object class  
c) Abstract class  
d) ArrayList class  
View Answer

Answer: b  
Explanation: Object class is superclass of every class in Java.

2. Which of these method of Object class can clone an object?  
a) Objectcopy()  
b) copy()  
c) Object clone()  
d) clone()  
View Answer

Answer: c  
Explanation: None.

3. Which of these method of Object class is used to obtain class of an object at run time?  
a) get()  
b) void getclass()  
c) Class getclass()  
d) None of the mentioned  
View Answer

Answer: c  
Explanation: None.

4. Which of these keywords can be used to prevent inheritance of a class?  
a) super  
b) constant  
c) class  
d) final  
View Answer

Answer: d  
Explanation: Declaring a class final implicitly declares all of its methods final, and makes the class inheritable.

5. Which of these keywords cannot be used for a class which has been declared final?  
a) abstract  
b) extends  
c) abstract and extends  
d) none of the mentioned  
View Answer

Answer: a  
Explanation: A abstract class is incomplete by itself and relies upon its subclasses to provide complete implementation. If we declare a class final then no class can inherit that class, an abstract class needs its subclasses hence both final and abstract cannot be used for a same class.

6. Which of these class relies upon its subclasses for complete implementation of its methods?  
a) Object class  
b) abstract class  
c) ArrayList class  
d) None of the mentioned  
View Answer

Answer: b  
Explanation: None.

7. What is the output of this program?

1. **abstract** **class** A
2. {
3. **int** i;
4. **abstract** **void** display();
5. }
6. **class** B **extends** A
7. {
8. **int** j;
9. **void** display()
10. {
11. System.out.println(j);
12. }
13. }
14. **class** Abstract\_demo
15. {
16. **public** **static** **void** main(String args[])
17. {
18. B obj = **new** B();
19. obj.j=2;
20. obj.display();
21. }
22. }

a) 0  
b) 2  
c) Runtime Error  
d) Compilation Error  
View Answer

Answer: b  
Explanation: class A is an abstract class, it contains a abstract function display(), the full implementation of display() method is given in its subclass B, Both the display functions are the same. Prototype of display() is defined in class A and its implementation is given in class B.  
output:  
$ javac Abstract\_demo.java  
$ java Abstract\_demo  
2

8. What is the output of this program?

1. **class** A
2. {
3. **int** i;
4. **int** j;
5. A()
6. {
7. i = 1;
8. j = 2;
9. }
10. }
11. **class** Output
12. {
13. **public** **static** **void** main(String args[])
14. {
15. A obj1 = **new** A();
16. A obj2 = **new** A();
17. System.out.print(obj1.equals(obj2));
18. }
19. }

a) false  
b) true  
c) 1  
d) Compilation Error  
View Answer

Answer: a  
Explanation: obj1 and obj2 are two different objects. equals() is a method of Object class, Since Object class is superclass of every class it is available to every object.  
output:  
$ javac Output.java  
$ java Output  
false

9. What is the output of this program?

1. **class** Output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. Object obj = **new** Object();
6. System.out.print(obj.getclass());
7. }
8. }

a) Object  
b) class Object  
c) class java.lang.Object  
d) Compilation Error  
View Answer

Answer: c  
Explanation: None.  
output:  
$ javac Output.java  
$ java Output  
class java.lang.Object

10. What is the output of this program?

1. **class** A
2. {
3. **int** i;
4. **int** j;
5. A()
6. {
7. i = 1;
8. j = 2;
9. }
10. }
11. **class** Output
12. {
13. **public** **static** **void** main(String args[])
14. {
15. A obj1 = **new** A();
16. System.out.print(obj1.toString());
17. }
18. }

a) true  
b) false  
c) String associated with obj1  
d) Compilation Error  
View Answer

Answer: c  
Explanation: toString() is method of class Object, since it is superclass of every class, every object has this method. toString() returns the string associated with the calling object.  
output:  
$ javac Output.java  
$ java Output  
A@1cd2e5f

Java Questions & Answers – Inheritance – Abstract Class and Super

This section of our 1000+ Java MCQs focuses on Abstract class in Java Programming Language.

1. Which of these keywords are used to define an abstract class?  
a) abst  
b) abstract  
c) Abstract  
d) abstract class  
View Answer

Answer: b  
Explanation: None.

2. Which of these is not abstract?  
a) Thread  
b) AbstractList  
c) List  
d) None of the Mentioned  
View Answer

Answer: a  
Explanation: Thread is not an abstract class.

3. If a class inheriting an abstract class does not define all of its function then it will be known as?  
a) Abstract  
b) A simple class  
c) Static class  
d) None of the mentioned  
View Answer

Answer: a  
Explanation: Any subclass of an abstract class must either implement all of the abstract method in the superclass or be itself declared abstract.

4. Which of these is not a correct statement?  
a) Every class containing abstract method must be declared abstract  
b) Abstract class defines only the structure of the class not its implementation  
c) Abstract class can be initiated by new operator  
d) Abstract class can be inherited  
View Answer

Answer: c  
Explanation: Abstract class cannot be directly initiated with new operator, Since abstract class does not contain any definition of implementation it is not possible to create an abstract object.

5. Which of these packages contains abstract keyword?  
a) java.lang  
b) java.util  
c) java.io  
d) java.system  
View Answer

Answer: a  
Explanation: None.

6. What is the output of this program?

1. **class** A
2. {
3. **public** **int** i;
4. **private** **int** j;
5. }
6. **class** B **extends** A
7. {
8. **void** display()
9. {
10. **super**.j = **super**.i + 1;
11. System.out.println(**super**.i + " " + **super**.j);
12. }
13. }
14. **class** inheritance
15. {
16. **public** **static** **void** main(String args[])
17. {
18. B obj = **new** B();
19. obj.i=1;
20. obj.j=2;
21. obj.display();
22. }
23. }

a) 2 2  
b) 3 3  
c) Runtime Error  
d) Compilation Error  
View Answer

Answer: d  
Explanation: Class contains a private member variable j, this cannot be inherited by subclass B and does not have access to it.  
output:  
$ javac inheritance.java  
Exception in thread “main” java.lang.Error: Unresolved compilation problem:  
The field A.j is not visible

7. What is the output of this program?

1. **class** A
2. {
3. **public** **int** i;
4. **public** **int** j;
5. A()
6. {
7. i = 1;
8. j = 2;
9. }
10. }
11. **class** B **extends** A
12. {
13. **int** a;
14. B()
15. {
16. **super**();
17. }
18. }
19. **class** super\_use
20. {
21. **public** **static** **void** main(String args[])
22. {
23. B obj = **new** B();
24. System.out.println(obj.i + " " + obj.j)
25. }
26. }

a) 1 2  
b) 2 1  
c) Runtime Error  
d) Compilation Error  
View Answer

Answer: a  
Explanation: Keyword super is used to call constructor of class A by constructor of class B. Constructor of a initializes i & j to 1 & 2 respectively.  
output:  
$ javac super\_use.java  
$ java super\_use  
1 2

8. What is the output of this program?

1. **abstract** **class** A
2. {
3. **int** i;
4. **abstract** **void** display();
5. }
6. **class** B **extends** A
7. {
8. **int** j;
9. **void** display()
10. {
11. System.out.println(j);
12. }
13. }
14. **class** Abstract\_demo
15. {
16. **public** **static** **void** main(String args[])
17. {
18. B obj = **new** B();
19. obj.j=2;
20. obj.display();
21. }
22. }

a) 0  
b) 2  
c) Runtime Error  
d) Compilation Error  
View Answer

Answer: b  
Explanation: class A is an abstract class, it contains a abstract function display(), the full implementation of display() method is given in its subclass B, Both the display functions are the same. Prototype of display() is defined in class A and its implementation is given in class B.  
output:  
$ javac Abstract\_demo.java  
$ java Abstract\_demo  
2

9. What is the output of this program?

1. **class** A
2. {
3. **int** i;
4. **void** display()
5. {
6. System.out.println(i);
7. }
8. }
9. **class** B **extends** A
10. {
11. **int** j;
12. **void** display()
13. {
14. System.out.println(j);
15. }
16. }
17. **class** method\_overriding
18. {
19. **public** **static** **void** main(String args[])
20. {
21. B obj = **new** B();
22. obj.i=1;
23. obj.j=2;
24. obj.display();
25. }
26. }

a) 0  
b) 1  
c) 2  
d) Compilation Error  
View Answer

Answer: c  
Explanation: class A & class B both contain display() method, class B inherits class A, when display() method is called by object of class B, display() method of class B is executed rather than that of Class A.  
output:  
$ javac method\_overriding.java  
$ java method\_overriding  
2

10. What is the output of this program?

1. **class** A
2. {
3. **public** **int** i;
4. **protected** **int** j;
5. }
6. **class** B **extends** A
7. {
8. **int** j;
9. **void** display()
10. {
11. **super**.j = 3;
12. System.out.println(i + " " + j);
13. }
14. }
15. **class** Output
16. {
17. **public** **static** **void** main(String args[])
18. {
19. B obj = **new** B();
20. obj.i=1;
21. obj.j=2;
22. obj.display();
23. }
24. }

a) 1 2  
b) 2 1  
c) 1 3  
d) 3 1  
View Answer

Answer: a  
Explanation: Both class A & B have member with same name that is j, member of class B will be called by default if no specifier is used. I contains 1 & j contains 2, printing 1 2.  
output:  
$ javac Output.java  
$ java Output  
1 2

Java Questions & Answers – Inheritance – 1

This section of our 1000+ Java MCQs focuses on Inheritance of Java Programming Language.

1. Which of these keyword must be used to inherit a class?  
a) super  
b) this  
c) extent  
d) extends  
View Answer

Answer: d  
Explanation: None.

2. Which of these keywords is used to refer to member of base class from a sub class?  
a) upper  
b) super  
c) this  
d) none of the mentioned  
View Answer

Answer: b  
Explanation: Whenever a subclass needs to refer to its immediate superclass, it can do so by use of the keyword super.

3. A class member declared protected becomes member of subclass of which type?  
a) public member  
b) private member  
c) protected member  
d) static member  
View Answer

Answer: b  
Explanation: A class member declared protected becomes private member of subclass.

4. Which of these is correct way of inheriting class A by class B?  
a) class B + class A {}  
b) class B inherits class A {}  
c) class B extends A {}  
d) class B extends class A {}  
View Answer

Answer: c  
Explanation: None.

5. Which two classes use the Shape class correctly?

A. **public** **class** Circle **implements** Shape

{

**private** **int** radius;

}

B. **public** **abstract** **class** Circle **extends** Shape

{

**private** **int** radius;

}

C. **public** **class** Circle **extends** Shape

{

**private** **int** radius;

**public** **void** draw();

}

D. **public** **abstract** **class** Circle **implements** Shape

{

**private** **int** radius;

**public** **void** draw();

}

E. **public** **class** Circle **extends** Shape

{

**private** **int** radius;

**public** **void** draw()

{

*/\* code here \*/*

}

}

F. **public** **abstract** **class** Circle **implements** Shape

{

**private** **int** radius;

**public** **void** draw()

{

*/\* code here \*/*

}

}

a) B,E  
b) A,C  
c) C,E  
d) T,H  
View Answer

Answer: a  
Explanation: If one is extending any class, then they should use extends keyword not implements.

6. What is the output of this program?

1. **class** A
2. {
3. **int** i;
4. **void** display()
5. {
6. System.out.println(i);
7. }
8. }
9. **class** B **extends** A
10. {
11. **int** j;
12. **void** display()
13. {
14. System.out.println(j);
15. }
16. }
17. **class** inheritance\_demo
18. {
19. **public** **static** **void** main(String args[])
20. {
21. B obj = **new** B();
22. obj.i=1;
23. obj.j=2;
24. obj.display();
25. }
26. }

a) 0  
b) 1  
c) 2  
d) Compilation Error  
View Answer

Answer: c  
Explanation: Class A & class B both contain display() method, class B inherits class A, when display() method is called by object of class B, display() method of class B is executed rather than that of Class A.  
output:  
$ javac inheritance\_demo.java  
$ java inheritance\_demo  
2

7. What is the output of this program?

1. **class** A
2. {
3. **int** i;
4. }
5. **class** B **extends** A
6. {
7. **int** j;
8. **void** display()
9. {
10. **super**.i = j + 1;
11. System.out.println(j + " " + i);
12. }
13. }
14. **class** inheritance
15. {
16. **public** **static** **void** main(String args[])
17. {
18. B obj = **new** B();
19. obj.i=1;
20. obj.j=2;
21. obj.display();
22. }
23. }

a) 2 2  
b) 3 3  
c) 2 3  
d) 3 2  
View Answer

Answer: c  
Explanation: None  
output:  
$ javac inheritance.java  
$ java inheritance  
2 3

8. What is the output of this program?

1. **class** A
2. {
3. **public** **int** i;
4. **private** **int** j;
5. }
6. **class** B **extends** A
7. {
8. **void** display()
9. {
10. **super**.j = **super**.i + 1;
11. System.out.println(**super**.i + " " + **super**.j);
12. }
13. }
14. **class** inheritance
15. {
16. **public** **static** **void** main(String args[])
17. {
18. B obj = **new** B();
19. obj.i=1;
20. obj.j=2;
21. obj.display();
22. }
23. }

a) 2 2  
b) 3 3  
c) Runtime Error  
d) Compilation Error  
View Answer

Answer: d  
Explanation: Class contains a private member variable j, this cannot be inherited by subclass B and does not have access to it.  
output:  
$ javac inheritance.java  
Exception in thread “main” java.lang.Error: Unresolved compilation problem:  
The field A.j is not visible

9. What is the output of this program?

1. **class** A
2. {
3. **public** **int** i;
4. **public** **int** j;
5. A()
6. {
7. i = 1;
8. j = 2;
9. }
10. }
11. **class** B **extends** A
12. {
13. **int** a;
14. B()
15. {
16. **super**();
17. }
18. }
19. **class** super\_use
20. {
21. **public** **static** **void** main(String args[])
22. {
23. B obj = **new** B();
24. System.out.println(obj.i + " " + obj.j)
25. }
26. }

a) 1 2  
b) 2 1  
c) Runtime Error  
d) Compilation Error  
View Answer

Answer: a  
Explanation: Keyword super is used to call constructor of class A by constructor of class B. Constructor of a initializes i & j to 1 & 2 respectively.  
output:  
$ javac super\_use.java  
$ java super\_use  
1 2

10. What is the output of this program?

1. **class** A
2. {
3. **public** **int** i;
4. **protected** **int** j;
5. }
6. **class** B **extends** A
7. {
8. **int** j;
9. **void** display()
10. {
11. **super**.j = 3;
12. System.out.println(i + " " + j);
13. }
14. }
15. **class** Output
16. {
17. **public** **static** **void** main(String args[])
18. {
19. B obj = **new** B();
20. obj.i=1;
21. obj.j=2;
22. obj.display();
23. }
24. }

a) 1 2  
b) 2 1  
c) 1 3  
d) 3 1  
View Answer

Answer: a  
Explanation: Both class A & B have member with same name that is j, member of class B will be called by default if no specifier is used. I contains 1 & j contains 2, printing 1 2.  
output:  
$ javac Output.java  
$ java Output  
1 2

# Java Questions & Answers – Inheritance – 2

This set of Java Interview Questions and Answers for freshers focuses on “Inheritance – 2”.

1. What is not type of inheritance?  
a) Single inheritance  
b) Double inheritance  
c) Hierarchical inheritance  
d) Multiple inheritance  
View Answer

Answer: b  
Explanation: Inheritance is way of acquiring attributes and methods of parent class. Java supports hierarchical inheritance directly.

2. Using which of the following, multiple inheritance in Java can be implemented?  
a) Interfaces  
b) Multithreading  
c) Protected methods  
d) Private methods  
View Answer

Answer: a  
Explanation: Multiple inheritance in java is implemented using interfaces. Multiple interfaces can be implemented by a class.

3. All classes in Java are inherited from which class?  
a) java.lang.class  
b) java.class.inherited  
c) java.class.object  
d) java.lang.Object  
View Answer

Answer: d  
Explanation: All classes in java are inherited from Object class. Interfaces are not inherited from Object Class.

4. In order to restrict a variable of a class from inheriting to sub class, how variable should be declared?  
a) Protected  
b) Private  
c) Public  
d) Static  
View Answer

Answer: b  
Explanation: By declaring variable private, the variable will not be available in inherited to sub class.

5. If super class and sub class have same variable name, which keyword should be used to use super class?  
a) super  
b) this  
c) upper  
d) classname  
View Answer

Answer: a  
Explanation: Super keyword is used to access hidden super class variable in sub class.

6. Static members are not inherited to sub class.  
a) True  
b) False  
View Answer

Answer: b  
Explanation: Static members are also inherited to sub classes.

7. Which of the following is used for implementing inheritance through interface?  
a) inherited  
b) using  
c) extends  
d) implements  
View Answer

Answer: d  
Explanation: Interface is implemented using implements keyword. A concrete class must implement all the methods of an iterface, else it must be declared abstract.

8. Which of the following is used for implementing inheritance through class?  
a) inherited  
b) using  
c) extends  
d) implements  
View Answer

Answer: c  
Explanation: Class can be extended using extends keyword. One class can extend only one class. A final class cannot be extended.

9. What would be the result if class extends two interfaces and both have method with same name and signature?  
a) Runtime error  
b) Compile time error  
c) Code runs successfully  
d) First called method is executed successfully  
View Answer

Answer: b  
Explanation: In case of such conflict, compiler will not be able to link a method call due to ambiguity. It will throw compile time error.

10. Does Java support multiple level inheritance?  
a) True  
b) False  
View Answer

Answer: a  
Explanation: Java supports multiple level inheritance through implementing multiple interfaces.

Java Questions & Answers – String Handling Basics

This section of our 1000+ Java MCQs focuses on string handling in Java Programming Language.

1. Which of these class is superclass of String and StringBuffer class?  
a) java.util  
b) java.lang  
c) ArrayList  
d) None of the mentioned  
View Answer

Answer: b  
Explanation: None.

2. Which of these operators can be used to concatenate two or more String objects?  
a) +  
b) +=  
c) &  
d) ||  
View Answer

Answer: a  
Explanation: Operator + is used to concatenate strings, Example String s = “i ” + “like ” + “java”; String s contains “I like java”.

3. Which of these method of class String is used to obtain length of String object?  
a) get()  
b) Sizeof()  
c) lengthof()  
d) length()  
View Answer

Answer: d  
Explanation: Method length() of string class is used to get the length of the object which invoked method length().

4. Which of these method of class String is used to extract a single character from a String object?  
a) CHARAT()  
b) chatat()  
c) charAt()  
d) ChatAt()  
View Answer

Answer: c  
Explanation: None.

5. Which of these constructors is used to create an empty String object?  
a) String()  
b) String(void)  
c) String(0)  
d) None of the mentioned  
View Answer

Answer: a  
Explanation: None.

6. Which of these is an oncorrect statement?  
a) String objects are immutable, they cannot be changed  
b) String object can point to some other reference of String variable  
c) StringBuffer class is used to store string in a buffer for later use  
d) None of the mentioned  
View Answer

Answer: c  
Explanation: StringBuffer class is used to create strings that can be modified after they are created.

7. What is the output of this program?

1. **class** String\_demo
2. {
3. **public** **static** **void** main(String args[])
4. {
5. **char** chars[] = {'a', 'b', 'c'};
6. String s = **new** String(chars);
7. System.out.println(s);
8. }
9. }

a) a  
b) b  
c) c  
d) abc  
View Answer

Answer: d  
Explanation: String(chars) is a constructor of class string, it initializes string s with the values stored in character array chars, therefore s contains “abc”.  
output:  
$ javac String\_demo.java  
$ java String\_demo  
abc

8. What is the output of this program?

1. **class** String\_demo
2. {
3. **public** **static** **void** main(String args[])
4. {
5. **int** ascii[] = { 65, 66, 67, 68};
6. String s = **new** String(ascii, 1, 3);
7. System.out.println(s);
8. }
9. }

a) ABC  
b) BCD  
c) CDA  
d) ABCD  
View Answer

Answer: b  
Explanation: ascii is an array of integers which contains ascii codes of Characters A, B, C, D. String(ascii, 1, 3) is an constructor which initializes s with Characters corresponding to ascii codes stored in array ascii, starting position being given by 1 & ending position by 3, Thus s stores BCD.  
output:  
$ javac String\_demo.java  
$ java String\_demo  
BCD

9. What is the output of this program?

1. **class** String\_demo
2. {
3. **public** **static** **void** main(String args[])
4. {
5. **char** chars[] = {'a', 'b', 'c'};
6. String s = **new** String(chars);
7. String s1 = "abcd";
8. **int** len1 = s1.length();
9. **int** len2 = s.length();
10. System.out.println(len1 + " " + len2);
11. }
12. }

a) 3 0  
b) 0 3  
c) 3 4  
d) 4 3  
View Answer

10. What is the output of this program?

1. **class** A
2. {
3. **int** i;
4. **int** j;
5. A()
6. {
7. i = 1;
8. j = 2;
9. }
10. }
11. **class** Output
12. {
13. **public** **static** **void** main(String args[])
14. {
15. A obj1 = **new** A();
16. System.out.print(obj1.toString());
17. }
18. }

a) True  
b) False  
c) String associated with obj1  
d) Compilation Error  
View Answer

Answer: c  
Explanation: toString() is method of class Object, since it is superclass of every class, every object has this method. toString() returns the string associated with the calling object.  
output:  
$ javac Output.java  
$ java Output  
A@1cd2e5f

Java Questions & Answers – Character Extraction

This section of our 1000+ Java MCQs focuses on character extraction of Java Programming Language.

1. Which of these method of class String is used to extract more than one character at a time a String object?  
a) getchars()  
b) GetChars()  
c) Getchars()  
d) getChars()  
View Answer

Answer: d  
Explanation: None.

2. Which of these methods is an alternative to getChars() that stores the characters in an array of bytes?  
a) getBytes()  
b) GetByte()  
c) giveByte()  
d) Give Bytes()  
View Answer

Answer: a  
Explanation: getBytes() stores the character in an array of bytes. It uses default character to byte conversions provided by platform.

3. In below code, what can directly access and change the value of the variable name?

1. **package** test;
2. **class** Target
3. {
4. **public** String name = "hello";
5. }

a) any class  
b) only the Target class  
c) any class in the test package  
d) any class that extends Target  
View Answer

Answer: c  
Explanation: Any class in the test package can access and change name.

4. What will be output of the following code?

1. **public** **class** Boxer1
2. {
3. Integer i;
4. **int** x;
5. **public** Boxer1(**int** y)
6. {
7. x = i+y;
8. System.out.println(x);
9. }
10. **public** **static** **void** main(String[] args)
11. {
12. **new** Boxer1 (**new** Integer(4));
13. }
14. }

a) The value “4” is printed at the command line  
b) Compilation fails because of an error in line  
c) A NullPointerException occurs at runtime  
d) An IllegalStateException occurs at runtime  
View Answer

Answer: d  
Explanation: Because we are performing operation on reference variable which is null.

5. Which of these methods can be used to convert all characters in a String into a character array?  
a) charAt()  
b) both getChars() & charAt()  
c) both toCharArray() & getChars()  
d) all of the mentioned  
View Answer

Answer: c  
Explanation: charAt() return one character only not array of character.

6. What is the output of this program?

1. **class** output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. String c = "Hello i love java";
6. **int** start = 2;
7. **int** end = 9;
8. **char** s[]=**new** **char**[end-start];
9. c.getChars(start,end,s,0);
10. System.out.println(s);
11. }
12. }

a) Hello, i love java  
b) i love ja  
c) lo i lo  
d) llo i l  
View Answer

Answer: d  
Explanation: getChars(start,end,s,0) returns an array from the string c, starting index of array is pointed by start and ending index is pointed by end. s is the target character array where the new string of letters is going to be stored and the new string will be stored from 0th position in s.  
Output:  
$ javac output.java  
$ java output  
llo i l

7. What is the output of this program?

1. **class** output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. String a = "hello i love java";
6. System.out.println(a.indexOf('i')+" "+a.indexOf('o') +" "+a.lastIndexOf('i')+" "+a.lastIndexOf('o'));
7. }
8. }

a) 6 4 6 9  
b) 5 4 5 9  
c) 7 8 8 9  
d) 4 3 6 9  
View Answer

8. What is the output of this program?

1. **class** output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. **char** c[]={'a', '1', 'b' ,' ' ,'A' , '0'};
6. **for** (**int** i = 0; i < 5; ++i)
7. {
8. **if**(Character.isDigit(c[i]))
9. System.out.println(c[i]+" is a digit");
10. **if**(Character.isWhitespace(c[i]))
11. System.out.println(c[i]+" is a Whitespace character");
12. **if**(Character.isUpperCase(c[i]))
13. System.out.println(c[i]+" is an Upper case Letter");
14. **if**(Character.isLowerCase(c[i]))
15. System.out.println(c[i]+" is a lower case Letter");
16. i=i+3;
17. }
18. }
19. }

a) a is a lower case Letter  
is White space character  
b) b is a lower case Letter  
is White space character  
c) a is a lower case Letter  
A is a upper case Letter  
d) a is a lower case Letter  
0 is a digit  
View Answer

9. What is the output of this program?

1. **class** String\_demo
2. {
3. **public** **static** **void** main(String args[])
4. {
5. **char** chars[] = {'a', 'b', 'c'};
6. String s = **new** String(chars);
7. System.out.println(s);
8. }
9. }

a) a  
b) b  
c) c  
d) abc  
View Answer

10. What is the output of this program?

1. **class** output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. **char** ch;
6. ch = "hello".charAt(1);
7. System.out.println(ch);
8. }
9. }

a) h  
b) e  
c) l  
d) o  
View Answer

**Sanfoundry Glob**

Java Questions & Answers – String Comparison

This section of our 1000+ Java MCQs focuses on String comparision in Java Programming Language.

1. Which of these method of class String is used to compare two String objects for their equality?  
a) equals()  
b) Equals()  
c) isequal()  
d) Isequal()  
View Answer

Answer: a  
Explanation: None.

2. Which of these methods is used to compare a specific region inside a string with another specific region in another string?  
a) regionMatch()  
b) match()  
c) RegionMatches()  
d) regionMatches()  
View Answer

Answer: d  
Explanation: None.

3. Which of these method of class String is used to check weather a given object starts with a particular string literal?  
a) startsWith()  
b) endsWith()  
c) Starts()  
d) ends()  
View Answer

Answer: a  
Explanation: Method startsWith() of string class is used to check whether the String in question starts with a specified string. It is specialized form of method regionMatches().

4. What is the value returned by unction compareTo() if the invoking string is less than the string compared?  
a) zero  
b) value less than zero  
c) value greater than zero  
d) none of the mentioned  
View Answer

Answer: b  
Explanation: compareTo() function returns zero when both the strings are equal, it returns a value less than zero if the invoking string is less than the other string being compared and value greater than zero when invoking string is greater than the string compared to.

5. Which of these data type value is returned by equals() method of String class?  
a) char  
b) int  
c) boolean  
d) all of the mentioned  
View Answer

Answer: c  
Explanation: equals() method of string class returns boolean value true if both the string are equal and false if they are unequal.

6. What is the output of this program?

1. **class** output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. String c = "Hello i love java";
6. **boolean** var;
7. var = c.startsWith("hello");
8. System.out.println(var);
9. }
10. }

a) true  
b) false  
c) 0  
d) 1  
View Answer

Answer: b  
Explanation: startsWith() method is case sensitive “hello” and “Hello” are treated differently, hence false is stored in var.  
Output:  
$ javac output.java  
$ java output  
false

7. What is the output of this program?

1. **class** output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. String s1 = "Hello i love java";
6. String s2 = **new** String(s1);
7. System.out.println((s1 == s2) + " " + s1.equals(s2));
8. }
9. }

a) true true  
b) false false  
c) true false  
d) false true  
View Answer

Answer: d  
Explanation: The == operator compares two object references to see whether they refer to the same instance, where as equals() compares the content of the two objects.  
Output:  
$ javac output.java  
$ java output  
false true

8. What is the output of this program?

1. **class** output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. String s1 = "Hello";
6. String s2 = **new** String(s1);
7. String s3 = "HELLO";
8. System.out.println(s1.equals(s2) + " " + s2.equals(s3));
9. }
10. }

a) true true  
b) false false  
c) true false  
d) false true  
View Answer

Answer: c  
Explanation: None.  
Output:  
$ javac output.java  
$ java output  
true false

9. In the below code, which code fragment should be inserted at line 3 so that the output will be: “123abc 123abc”?

1 StringBuilder sb1 = **new** StringBuilder("123");

2 String s1 = "123";

3 *// insert code here*

4 System.out.println(sb1 + " " + s1);

a) sb1.append(“abc”); s1.append(“abc”);  
b) sb1.append(“abc”); s1.concat(“abc”);  
c) sb1.concat(“abc”); s1.append(“abc”);  
d) sb1.append(“abc”); s1 = s1.concat(“abc”);  
View Answer

Answer: d  
Explanation: append() is stringbuffer method and concat is String class method.  
append() is stringbuffer method and concat is String class method.

10. What is the output of this program?

1. **class** output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. String chars[] = {"a", "b", "c", "a", "c"};
6. **for** (**int** i = 0; i < chars.length; ++i)
7. **for** (**int** j = i + 1; j < chars.length; ++j)
8. **if**(chars[i].compareTo(chars[j]) == 0)
9. System.out.print(chars[j]);
10. }
11. }

a) ab  
b) bc  
c) ca  
d) ac  
View Answer

Answer: d  
Explanation: compareTo() function returns zero when both the strings are equal, it returns a value less than zero if the invoking string is less than the other string being compared and value greater than zero when invoking string is greater than the string compared to.  
output:  
$ javac output.java  
$ java output  
ac

Java Questions & Answers – Searching & Modifying a String

This section of our 1000+ Java MCQs focuses on searching and modifying a string of Java Programming Language.

1. Which of these method of class String is used to extract a substring from a String object?  
a) substring()  
b) Substring()  
c) SubString()  
d) None of the mentioned  
View Answer

Answer: a  
Explanation: None.

2. What will s2 contain after following lines of code?

String s1 = "one";

String s2 = s1.concat("two")

a) one  
b) two  
c) onetwo  
d) twoone  
View Answer

Answer: c  
Explanation: Two strings can be concatenated by using concat() method.

3. Which of these method of class String is used to remove leading and trailing whitespaces?  
a) startsWith()  
b) trim()  
c) Trim()  
d) doTrim()  
View Answer

4. What is the value returned by function compareTo() if the invoking string is greater than the string compared?  
a) zero  
b) value less than zero  
c) value greater than zero  
d) none of the mentioned  
View Answer

Answer: c  
Explanation: if (s1 == s2) then 0, if(s1 > s2) > 0, if (s1 < s2) then < 0.

5. Which of the following statement is correct?  
a) replace() method replaces all occurrences of one character in invoking string with another character  
b) replace() method replaces only first occurances of a character in invoking string with another character  
c) replace() method replaces all the characters in invoking string with another character  
d) replace() replace() method replaces last occurrence of a character in invoking string with another character  
View Answer

6. What is the output of this program?

1. **class** output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. String c = " Hello World ";
6. String s = c.trim();
7. System.out.println("**\"**"+s+"**\"**");
8. }
9. }

a) “”Hello World””  
b) “”Hello World”  
c) “Hello World”  
d) Hello world  
View Answer

Answer: c  
Explanation: trim() method is used to remove leading and trailing whitespaces in a string.  
Output:  
$ javac output.java  
$ java output  
“Hello World”

7. What is the output of this program?

1. **class** output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. String s1 = "one";
6. String s2 = s1 + " two";
7. System.out.println(s2);
8. }
9. }

a) one  
b) two  
c) one two  
d) compilation error  
View Answer

Answer: c  
Explanation: None.  
Output:  
$ javac output.java  
$ java output  
one two

8. What is the output of this program?

1. **class** output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. String s1 = "Hello";
6. String s2 = s1.replace('l','w');
7. System.out.println(s2);
8. }
9. }

a) hello  
b) helwo  
c) hewlo  
d) hewwo  
View Answer

Answer: d  
Explanation: replace() method replaces all occurrences of one character in invoking string with another character. s1.replace(‘l’,’w’) replaces every occurrence of ‘l’ in hello by ‘w’, giving hewwo.  
Output:  
$ javac output.java  
$ java output  
hewwo

9. What is the output of this program?

1. **class** output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. String s1 = "Hello World";
6. String s2 = s1.substring(0 , 4);
7. System.out.println(s2);
8. }
9. }

a) Hell  
b) Hello  
c) Worl  
d) World  
View Answer

Answer: a  
Explanation: substring(0,4) returns the character from 0 th position to 3 rd position.  
output:  
$ javac output.java  
$ java output  
Hell

10. What is the output of this program?

1. **class** output
2. {
3. **public** **static** **void** main(String args[])
4. { String s = "Hello World";
5. **int** i = s.indexOf('o');
6. **int** j = s.lastIndexOf('l');
7. System.out.print(i + " " + j);
9. }
10. }

a) 4 8  
b) 5 9  
c) 4 9  
d) 5 8  
View Answer

Answer: c  
Explanation: indexOf() method returns the index of first occurrence of the character where as lastIndexOf() returns the index of last occurrence of the character.  
output:  
$ javac output.java  
$ java output  
4 9

Java Questions & Answers – StringBuffer Class

This section of our 1000+ Java MCQs focuses on StringBuffer class of Java Programming Language.

1. Which of these class is used to create an object whose character sequence is mutable?  
a) String()  
b) StringBuffer()  
c) Both of the mentioned  
d) None of the mentioned  
View Answer

Answer: b  
Explanation: StringBuffer represents growable and writeable character sequence.

2. Which of these method of class StringBuffer is used to concatenate the string representation to the end of invoking string?  
a) concat()  
b) append()  
c) join()  
d) concatenate()  
View Answer

Answer: b  
Explanation: None.

3. Which of these method of class StringBuffer is used to find the length of current character sequence?  
a) length()  
b) Length()  
c) capacity()  
d) Capacity()  
View Answer

Answer: a  
Explanation: None.

4. What is the string contained in s after following lines of code?  
StringBuffer s new StringBuffer(“Hello”);  
s.deleteCharAt(0);  
a) Hell  
b) ello  
c) Hel  
d) llo  
View Answer

Answer: b  
Explanation: deleteCharAt() method deletes the character at the specified index location and returns the resulting StringBuffer object.

5. Which of the following statement is correct?  
a) reverse() method reverses all characters  
b) reverseall() method reverses all characters  
c) replace() method replaces first occurrence of a character in invoking string with another character  
d) replace() method replaces last occurrence of a character in invoking string with another character  
View Answer

Answer: a  
Explanation: reverse() method reverses all characters. It returns the reversed object on which it was called.

6. What is the output of this program?

1. **class** output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. String a = "hello i love java";
6. System.out.println(a.indexOf('e')+" "+a.indexOf('a')+" "+a.lastIndexOf('l')+" "+a.lastIndexOf('v'));
7. }
8. }

a) 6 4 6 9  
b) 5 4 5 9  
c) 7 8 8 9  
d) 1 14 8 15  
View Answer

Answer: a  
Explantion: indexof(‘c’) and lastIndexof(‘c’) are pre defined function which are used to get the index of first and last occurrence of  
the character pointed by c in the given array.  
Output:  
$ javac output.java  
$ java output  
1 14 8 15

7. What is the output of this program?

1. **class** output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. StringBuffer c = **new** StringBuffer("Hello");
6. c.delete(0,2);
7. System.out.println(c);
8. }
9. }

a) He  
b) Hel  
c) lo  
d) llo  
View Answer

Answer: d  
Explanation: delete(0,2) is used to delete the characters from 0 th position to 1 st position.  
Output:  
$ javac output.java  
$ java output  
llo

8. What is the output of this program?

1. **class** output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. StringBuffer c = **new** StringBuffer("Hello");
6. StringBuffer c1 = **new** StringBuffer(" World");
7. c.append(c1);
8. System.out.println(c);
9. }
10. }

a) Hello  
b) World  
c) Helloworld  
d) Hello World  
View Answer

Answer: d  
Explanation: append() method of class StringBuffer is used to concatenate the string representation to the end of invoking string.  
Output:  
$ javac output.java  
$ java output  
Hello World

9. What is the output of this program?

1. **class** output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. StringBuffer s1 = **new** StringBuffer("Hello");
6. StringBuffer s2 = s1.reverse();
7. System.out.println(s2);
8. }
9. }

a) Hello  
b) olleH  
c) HelloolleH  
d) olleHHello  
View Answer

Answer: b  
Explanation: reverse() method reverses all characters. It returns the reversed object on which it was called.  
Output:  
$ javac output.java  
$ java output  
olleH

10. What is the output of this program?

1. **class** output
2. {
3. **class** output
4. {
5. **public** **static** **void** main(String args[])
6. {
7. **char** c[]={'A', '1', 'b' ,' ' ,'a' , '0'};
8. **for** (**int** i = 0; i < 5; ++i)
9. {
10. i++;
11. **if**(Character.isDigit(c[i]))
12. System.out.println(c[i]+" is a digit");
13. **if**(Character.isWhitespace(c[i]))
14. System.out.println(c[i]+" is a Whitespace character");
15. **if**(Character.isUpperCase(c[i]))
16. System.out.println(c[i]+" is an Upper case Letter");
17. **if**(Character.isLowerCase(c[i]))
18. System.out.println(c[i]+" is a lower case Letter");
19. i++;
20. }
21. }
22. }

a) a is a lower case Letter  
is White space character  
b) b is a lower case Letter  
is White space character  
c) 1 is a digit  
a is a lower case Letter  
d) a is a lower case Letter  
0 is a digit  
View Answer

Answer: c  
Explanation: Character.isDigit(c[i]),Character.isUpperCase(c[i]),Character.isWhitespace(c[i]) are the function of library java.lang they are used to find whether the given character is of specified type or not. They return true or false i:e Boolean variable.  
Output:  
$ javac output.java  
$ java output  
1 is a digit  
a is a lower

Java Questions & Answers – StringBuffer Methods

This section of our 1000+ Java MCQs focuses on StringBuffer class’s methods of Java Programming Language.

1. 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  
View Answer

Answer: a  
Explanation: None.

2. What will s2 contain after following lines of code?

StringBuffer s1 = "one";

StringBuffer s2 = s1.append("two")

a) one  
b) two  
c) onetwo  
d) twoone  
View Answer

Answer: c  
Explanation: Two strings can be concatenated by using append() method.

3. Which of these method of class StringBuffer is used to reverse sequence of characters?  
a) reverse()  
b) reverseall()  
c) Reverse()  
d) reverseAll()  
View Answer

Answer: a  
Explanation: reverse() method reverses all characters. It returns the reversed object on which it was called.

4. Which of these method of class StringBuffer is used to get the length of sequence of characters?  
a) length()  
b) capacity()  
c) Length()  
d) Capacity()  
View Answer

Answer: a  
Explanation: length()- returns the length of String the StringBuffer would create whereas capacity() returns total number of characters that can be supported before it is grown.

5. Which of the following are incorrect form of StringBuffer class constructor?  
a) StringBuffer()  
b) StringBuffer(int size)  
c) StringBuffer(String str)  
d) StringBuffer(int size , String str)  
View Answer

Answer: d  
Explanation: None.

6. What is the output of this program?

1. **class** output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. StringBuffer c = **new** StringBuffer("Hello");
6. System.out.println(c.length());
7. }
8. }

a) 4  
b) 5  
c) 6  
d) 7  
View Answer

Answer: b  
Explanation: length() method is used to obtain length of StringBuffer object, length of “Hello” is 5.  
Output:  
$ javac output.java  
$ java output  
5

7. What is the output of this program?

1. **class** output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. StringBuffer sb=**new** StringBuffer("Hello");
6. sb.replace(1,3,"Java");
7. System.out.println(sb);
8. }
9. }

a) Hello java  
b) Hellojava  
c) HJavalo  
d) Hjava  
View Answer

Answer: c  
Explanation: The replace() method replaces the given string from the specified beginIndex and endIndex.  
$ javac output.java  
$ java output  
HJavalo

8. What is the output of this program?

1. **class** output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. StringBuffer s1 = **new** StringBuffer("Hello");
6. s1.setCharAt(1,'x');
7. System.out.println(s1);
8. }
9. }

a) xello  
b) xxxxx  
c) Hxllo  
d) Hexlo  
View Answer

Answer: c  
Explanation: None.  
Output:  
$ javac output.java  
$ java output  
Hxllo

9. What is the output of this program?

1. **class** output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. StringBuffer s1 = **new** StringBuffer("Hello World");
6. s1.insert(6 , "Good ");
7. System.out.println(s1);
8. }
9. }

a) HelloGoodWorld  
b) HellGoodoWorld  
c) HellGood oWorld  
d) Hello Good World  
View Answer

Answer: d  
Explanation: The insert() method inserts one string into another. It is overloaded to accept values of all simple types, plus String and Objects. Sting is inserted into invoking object at specified position. “Good ” is inserted in “Hello World” T index 6 giving “Hello Good World”.  
output:  
$ javac output.java  
$ java output  
Hello Good World

10. What is the output of this program?

1. **class** output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. StringBuffer s1 = **new** StringBuffer("Hello");
6. s1.insert(1,"Java");
7. System.out.println(s1);
8. }
9. }

a) hello  
b) java  
c) Hello Java  
d) HelloJava  
View Answer

Answer: d  
Explanation: Insert method will insert string at a specified position  
Output:  
$ javac output.java  
$ java output  
HelloJava

Java Questions & Answers – Java.lang Introduction

This section of our 1000+ Java MCQs focuses on java.lang library of Java Programming Language.

1. Which of these classes is not included in java.lang?  
a) Byte  
b) Integer  
c) Array  
d) Class  
View Answer

Answer: c  
Explanation: Array class is a member of java.util.

2. Which of these is a process of converting a simple data type into a class?  
a) type wrapping  
b) type conversion  
c) type casting  
d) none of the Mentioned.  
View Answer

Answer: a  
Explanation: None.

3. Which of these is a super class of wrappers Double & Integer?  
a) Long  
b) Digits  
c) Float  
d) Number  
View Answer

4. Which of these is wrapper for simple data type float?  
a) float  
b) double  
c) Float  
d) Double  
View Answer

Answer: c  
Explanation: None.

5. Which of the following is method of wrapper Float for converting the value of an object into byte?  
a) bytevalue()  
b) byte byteValue()  
c) Bytevalue()  
d) Byte Bytevalue().  
View Answer

Answer: b  
Explanation: None.

6. Which of these methods is used to check for infinitely large and small values?  
a) isInfinite()  
b) isNaN()  
c) Isinfinite()  
d) IsNaN()  
View Answer

Answer: a  
Explanation: isinfinite() method returns true is the value being tested is infinitely large or small in magnitude.

7. Which of the following package stores all the simple data types in java?  
a) lang  
b) java  
c) util  
d) java.packages  
View Answer

Answer: a  
Explanation: None.

8. What is the output of this program?

1. **class** isinfinite\_output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. Double d = **new** Double(1 / 0.);
6. **boolean** x = d.isInfinite();
7. System.out.print(x);
8. }
9. }

a) 0  
b) 1  
c) true  
d) false  
View Answer

Answer: c  
Explanation: isInfinite() method returns true is the value being tested is infinitely large or small in magnitude. 1/0. is infinitely large in magnitude hence true is stored in x.  
Output:  
$ javac isinfinite\_output.java  
$ java isinfinite\_output  
true

9. What is the output of this program?

1. **class** isNaN\_output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. Double d = **new** Double(1 / 0.);
6. **boolean** x = d.isNaN();
7. System.out.print(x);
8. }
9. }

a) 0  
b) 1  
c) true  
d) false  
View Answer

Answer: d  
Explanation: isisNaN() method returns true is the value being tested is a number. 1/0. is infinitely large in magnitude, which cant not be defined as a number hence false is stored in x.  
Output:  
$ javac isNaN\_output.java  
$ java isNaN\_output  
false

10. What is the output of this program?

1. **class** binary
2. {
3. **public** **static** **void** main(String args[])
4. {
5. **int** num = 17;
6. System.out.print(Integer.toBinaryString(num));
7. }
8. }

a) 1001  
b) 10011  
c) 11011  
d) 10001  
View Answer

Answer: d  
Explanation: None.  
output:  
$ javac binary.java  
$ java binary  
10001

Java Questions & Answers – Java.lang – Integer, Long & Character Wrappers

This section of our 1000+ Java MCQs focuses on Integer, Long & Character wrappers of Java Programming Language.

1. Which of these is a wrapper for data type int?  
a) Integer  
b) Long  
c) Byte  
d) Double  
View Answer

Answer: a  
Explanation: None.

2. Which of the following methods is a method of wrapper Integer for obtaining hash code for the invoking object?  
a) int hash()  
b) int hashcode()  
c) int hashCode()  
d) Integer hashcode()  
View Answer

Answer: c  
Explanation: None.

3. Which of these is a super class of wrappers Long, Character & Integer?  
a) Long  
b) Digits  
c) Float  
d) Number  
View Answer

Answer: d  
Explanation: Number is an abstract class containing subclasses Double, Float, Byte, Short, Integer and Long.

4. Which of these is wrapper for simple data type char?  
a) Float  
b) Character  
c) String  
d) Integer  
View Answer

Answer: b  
Explanation: None.

5. Which of the following is method of wrapper Integer for converting the value of an object into int?  
a) bytevalue()  
b) int intValue();  
c) Bytevalue()  
d) Byte Bytevalue()  
View Answer

Answer: b  
Explanation: None.

6. Which of these methods is used to obtain value of invoking object as a long?  
a) long value()  
b) long longValue()  
c) Long longvalue()  
d) Long Longvalue()  
View Answer

Answer: b  
Explanation: long longValue() is used to obtain value of invoking object as a long.

7. What is the output of this program?

1. **class** Output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. **char** a[] = {'a', '5', 'A', ' '};
6. System.out.print(Character.isDigit(a[0]) + " ");
7. System.out.print(Character.isWhitespace(a[3]) + " ");
8. System.out.print(Character.isUpperCase(a[2]));
9. }
10. }

a) true false true  
b) false true true  
c) true true false  
d) false false false  
View Answer

Answer: b  
Explanation: Character.isDigit(a[0]) checks for a[0], whether it is a digit or not, since a[0] i:e ‘a’ is a character false is returned. a[3] is a whitespace hence Character.isWhitespace(a[3]) returns a true. a[2] is an upper case letter i:e ‘A’ hence Character.isUpperCase(a[2]) returns true.  
Output:  
$ javac Output.java  
$ java Output  
false true true

8. What is the output of this program?

1. **class** Output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. Integer i = **new** Integer(257);
6. **byte** x = i.byteValue();
7. System.out.print(x);
8. }
9. }

a) 0  
b) 1  
c) 256  
d) 257  
View Answer

Answer: b  
Explanation: i.byteValue() method returns the value of wrapper i as a byte value. i is 257, range of byte is 256 therefore i value exceeds byte range by 1 hence 1 is returned and stored in x.  
Output:  
$ javac Output.java  
$ java Output  
1

9. What is the output of this program?

1. **class** Output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. Integer i = **new** Integer(257);
6. **float** x = i.floatValue();
7. System.out.print(x);
8. }
9. }

a) 0  
b) 1  
c) 257  
d) 257.0  
View Answer

Answer: d  
Explanation: None.  
Output:  
$ javac Output.java  
$ java Output  
257.0

10. What is the output of this program?

1. **class** Output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. Long i = **new** Long(256);
6. System.out.print(i.hashCode());
7. }
8. }

a) 256  
b) 256.0  
c) 256.00  
d) 257.00  
View Answer

Answer: a  
Explanation: None.  
Output:  
$ javac Output.java  
$ java Output  
256

Java Questions & Answers – Java.lang – Void, Process & System Class

This section of our 1000+ Java MCQs focuses on Void, Process & System classes of Java Programming Language.

1. Which of these class have only one field ‘TYPE’?  
a) Void  
b) Process  
c) System  
d) Runtime  
View Answer

Answer: a  
Explanation: The Void class has one field, TYPE, which holds a reference to the Class object for the type void.

2. Which of the following method of Process class can terminate a process?  
a) void kill()  
b) void destroy()  
c) void terminate()  
d) void exit()  
View Answer

Answer: b  
Explanation: Kills the subprocess. The subprocess represented by this Process object is forcibly terminated.

3. Standard output variable ‘out’ is defined in which class?  
a) Void  
b) Process  
c) Runtime  
d) System  
View Answer

Answer: d  
Explanation: Standard output variable ‘out’ is defined in System class. out is usually used in print statement i:e System.out.print().

4. Which of these class can encapsulate an entire executing program?  
a) Void  
b) Process  
c) Runtime  
d) System  
View Answer

Answer: b  
Explanation: None.

5. Which of the following is method of System class is used to find how long a program takes to execute?  
a) currenttime()  
b) currentTime()  
c) currentTimeMillis()  
d) currenttimeMillis()  
View Answer

Answer: c  
Explanation: None.

6. Which of these class holds a collection of static methods and variables?  
a) Void  
b) Process  
c) Runtime  
d) System  
View Answer

Answer: d  
Explanation: System class holds a collection of static methods and variables. The standard input, output and error output of java run time are stored in the in, out and err variables of System class.

7. What is the output of this program?

1. **class** Output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. **long** start, end;
6. start = System.currentTimeMillis();
7. **for** (**int** i = 0; i < 10000000; i++);
8. end = System.currentTimeMillis();
9. System.out.print(end - start);
10. }
11. }

a) 0  
b) 1  
c) 1000  
d) System Dependent  
View Answer

Answer: d  
Explanation: end time is the time taken by loop to execute it can be any non zero value depending on the System.  
Output:  
$ javac Output.java  
$ java Output  
78

8. What is the output of this program?

1. **class** Output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. **byte** a[] = { 65, 66, 67, 68, 69, 70 };
6. **byte** b[] = { 71, 72, 73, 74, 75, 76 };
7. System.arraycopy(a , 0, b, 0, a.length);
8. System.out.print(**new** String(a) + " " + **new** String(b));
9. }
10. }

a) ABCDEF ABCDEF  
b) ABCDEF GHIJKL  
c) GHIJKL ABCDEF  
d) GHIJKL GHIJKL  
View Answer

Answer: a  
Explanation: System.arraycopy() is a method of class System which is used to copy a string into another string.  
Output:  
$ javac Output.java  
$ java Output  
ABCDEF ABCDEF

9. What is the output of this program?

1. **class** Output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. **byte** a[] = { 65, 66, 67, 68, 69, 70 };
6. **byte** b[] = { 71, 72, 73, 74, 75, 76 };
7. System.arraycopy(a, 2, b, 1, a.length-2);
8. System.out.print(**new** String(a) + " " + **new** String(b));
9. }
10. }

a) ABCDEF GHIJKL  
b) ABCDEF GCDEFL  
c) GHIJKL ABCDEF  
d) GCDEFL GHIJKL  
View Answer

Answer: b  
Explanation: None.  
Output:  
$ javac Output.java  
$ java Output  
ABCDEF GCDEFL

10. What is the output of this program?

1. **class** Output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. **byte** a[] = { 65, 66, 67, 68, 69, 70 };
6. **byte** b[] = { 71, 72, 73, 74, 75, 76 };
7. System.arraycopy(a, 1, b, 3, 0);
8. System.out.print(**new** String(a) + " " + **new** String(b));
9. }
10. }

a) ABCDEF GHIJKL  
b) ABCDEF GCDEFL  
c) GHIJKL ABCDEF  
d) GCDEFL GHIJKL  
View Answer

Answer: a  
Explanation: Since last parameter of System.arraycopy(a,1,b,3,0) is 0 nothing is copied from array a to array b, hence b remains as it is.  
Output:  
$ javac Output.java  
$ java Output  
ABCDEF GHIJK

Java Questions & Answers – Java.lang – Object & Math Class

This section of our 1000+ Java MCQs focuses on Object & Math classes of Java Programming Language.

1. Which of these class is superclass of all other classes?  
a) Math  
b) Process  
c) System  
d) Object  
View Answer

Answer: d  
Explanation: The object class class is superclass of all other classes.

2. Which of these method of Object class can generate duplicate copy of the object on which it is called?  
a) clone()  
b) copy()  
c) dublicate()  
d) dito()  
View Answer

Answer: a  
Explanation: None.

3. What is the value of double constant ‘E’ defined in Math class?  
a) approximately 3  
b) approximately 3.14  
c) approximately 2.72  
d) approximately 0  
View Answer

4. Which of these method is a rounding function of Math class?  
a) max()  
b) min()  
c) abs()  
d) all of the mentioned  
View Answer

Answer: d  
Explanation: max(), min() and abs() are all rounding functions.

5. Which of these class contains only floating point functions?  
a) Math  
b) Process  
c) System  
d) Object  
View Answer

6. Which of these class encapsulate the run time state of an object or an interface?  
a) Class  
b) Object  
c) Runtime  
d) System  
View Answer

7. What is the value of “d” after this line of code has been executed?

**double** d = Math.round ( 2.5 + Math.random() );

a) 2  
b) 3  
c) 4  
d) 2.5  
View Answer

Answer: b  
Explanation: The Math.random() method returns a number greater than or equal to 0 and less than 1. so 2.5 will be greater than or equal to 2.5 and less than 3.5, we can be sure that Math.round() will round that number to 3.

8. What is the output of this program?

1. **class** Output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. **int** x = 3.14;
6. **int** y = (**int**) Math.abs(x);
7. System.out.print(y);
8. }
9. }

a) 0  
b) 3  
c) 3.0  
d) 3.1  
View Answer

Answer: b  
Explanation: None.  
Output:  
$ javac Output.java  
$ java Output  
3

9. What is the output of this program?

1. **class** Output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. **double** x = 3.1;
6. **double** y = 4.5;
7. **double** z = Math.max( x, y );
8. System.out.print(z);
9. }
10. }

a) true  
b) flase  
c) 3.1  
d) 4.5  
View Answer

10. What is the output of this program?

1. **class** Output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. **double** x = 2.0;
6. **double** y = 3.0;
7. **double** z = Math.pow( x, y );
8. System.out.print(z);
9. }
10. }

a) 2.0  
b) 4.0  
c) 8.0  
d) 9.0  
View Answer

Answer: c  
Explanation: Math.pow(x, y) methods returns value of y to the power x, i:e x ^ y, 2.0 ^ 3.0 = 8.0.  
Output:  
$ javac Output.java  
$ java Output  
8.0

Java Questions & Answers – Java.lang – System Class Advance

This set of Java Multiple Choice Questions & Answers (MCQs) focuses on “System Class Advance”.

1. Which of these exceptions is thrown by methods of System class?  
a) IOException  
b) SystemException  
c) SecurityException  
d) InputOutputException  
View Answer

Answer: c  
Explanation: System class methods throw SecurityException.

2. Which of these methods initiates garbage collection?  
a) gc()  
b) garbage()  
c) garbagecollection()  
d) Systemgarbagecollection()  
View Answer

Answer: a  
Explanation: None.

3. Which of these methods loads the specified dynamic library?  
a) load()  
b) library()  
c) loadlib()  
d) loadlibrary()  
View Answer

Answer: a  
Explanation: load() methods loads the dynamic library whose name is specified.

4. Which of these method can set the out stream to OutputStream?  
a) setStream()  
b) setosteam()  
c) setOut()  
d) streamtoOstream()  
View Answer

Answer: c  
Explanation: None.

5. Which of these values are returns under the case of normal termination of a program?  
a) 0  
b) 1  
c) 2  
d) 3  
View Answer

Answer: a  
Explanation: None.

6. What is the output of this program?

1. **import** java.lang.System;
2. **class** Output
3. {
4. **public** **static** **void** main(String args[])
5. {
6. **long** start, end;
7. start = System.currentTimeMillis();
8. **for** (**int** i = 0; i < 10000000; i++);
9. end = System.currentTimeMillis();
10. System.out.print(end - start);
11. }
12. }

a) 0  
b) 1  
c) 1000  
d) System Dependent  
View Answer

Answer: d  
Explanation: End time is the time taken by loop to execute it can be any non zero value depending on the System.  
Output:  
$ javac Output.java  
$ java Output  
78

7. What is the output of this program?

1. **import** java.lang.System;
2. **class** Output
3. {
4. **public** **static** **void** main(String args[])
5. {
6. **byte** a[] = { 65, 66, 67, 68, 69, 70 };
7. **byte** b[] = { 71, 72, 73, 74, 75, 76 };
8. System.arraycopy(a, 0, b, 0, a.length);
9. System.out.print(**new** String(a) + " " + **new** String(b));
10. }
11. }

a) ABCDEF ABCDEF  
b) ABCDEF GHIJKL  
c) GHIJKL ABCDEF  
d) GHIJKL GHIJKL  
View Answer

8. What is the output of this program?

1. **import** java.lang.System;
2. **class** Output
3. {
4. **public** **static** **void** main(String args[])
5. {
6. **byte** a[] = { 65, 66, 67, 68, 69, 70 };
7. **byte** b[] = { 71, 72, 73, 74, 75, 76 };
8. System.arraycopy(a, 0, b, 3, a.length - 3);
9. System.out.print(**new** String(a) + " " + **new** String(b));
10. }
11. }

a) ABCDEF ABCDEF  
b) ABCDEF GHIJKL  
c) ABCDEF GHIABC  
d) GHIJKL GHIJKL  
View Answer

Answer: c  
Explanation: System.arraycopy() is a method of class System which is used to copy a string into another string.  
Output:  
$ javac Output.java  
$ java Output  
ABCDEF GHIABC

9. What is the output of this program?

1. **import** java.lang.System;
2. **class** Output
3. {
4. **public** **static** **void** main(String args[])
5. {
6. **byte** a[] = { 65, 66, 67, 68, 69, 70 };
7. **byte** b[] = { 71, 72, 73, 74, 75, 76 };
8. System.arraycopy(a, 2, b, 3, a.length - 4);
9. System.out.print(**new** String(a) + " " + **new** String(b));
10. }
11. }

a) ABCDEF ABCDEF  
b) ABCDEF GHIJKL  
c) ABCDEF GHIABC  
d) ABCDEF GHICDL  
View Answer

Answer: d  
Explanation: System.arraycopy() is a method of class System which is used to copy a string into another string.  
Output:  
$ javac Output.java  
$ java Output  
ABCDEF GHICDL

10. What value will this program return to Java run-time system?

1. **import** java.lang.System;
2. **class** Output
3. {
4. **public** **static** **void** main(String args[])
5. {
6. System.exit(5);
7. }
8. }

a) 0  
b) 1  
c) 4  
d) 5  
View Answer

Answer: d  
Explanation: None.

Java Questions & Answers – Java.lang – Double & Float Wrappers

This set of Java Multiple Choice Questions & Answers (MCQs) focuses on “Double & Float Wrappers”.

1. Which of these is a super class of wrappers Double and Float?  
a) Long  
b) Digits  
c) Float  
d) Number  
View Answer

Answer: d  
Explanation: Number is an abstract class containing subclasses Double, Float, Byte, Short, Integer and Long.

2. Which of the following methods return the value as a double?  
a) doubleValue()  
b) converDouble()  
c) getDouble()  
d) getDoubleValue()  
View Answer

Answer: a  
Explanation: None.

3. Which of these methods can be used to check whether the given value is a number or not?  
a) isNaN()  
b) isNumber()  
c) checkNaN()  
d) checkNumber()  
View Answer

Answer: a  
Explanation: isNaN() methods returns true if num specified is not a number, otherwise it returns false.

4. Which of these method of Double wrapper can be used to check weather a given value is infinite or not?  
a) Infinite()  
b) isInfinite()  
c) checkInfinite()  
d) None of the mentioned  
View Answer

Answer: b  
Explanation: isInfinite() methods returns true if specified value is an infinite value otherwise it returns false.

5. Which of these exceptions is thrown by compareTo() method defined in double wrapper?  
a) IOException  
b) SystemException  
c) CastException  
d) ClassCastException  
View Answer

Answer: d  
Explanation: compareTo() methods compare the specified object to be double, if it is not then ClassCastException is thrown.

6. What is the output of this program?

1. **class** Output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. Double i = **new** Double(257.5);
6. **boolean** x = i.isNaN();
7. System.out.print(x);
8. }
9. }

a) true  
b) false  
c) 0  
d) 1  
View Answer

Answer: b  
Explanation: i.isNaN() method returns returns true if i is not a number and false when i is a number. Here false is returned because i is a number i:e 257.5.  
Output:  
$ javac Output.java  
$ java Output  
false

7. What is the output of this program?

1. **class** Output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. Integer i = **new** Integer(257);
6. **byte** x = i.byteValue();
7. System.out.print(x);
8. }
9. }

a) 0  
b) 1  
c) 256  
d) 257  
View Answer

Answer: b  
Explanation: i.byteValue() method returns the value of wrapper i as a byte value. i is 257, range of byte is 256 therefore i value exceeds byte range by 1 hence 1 is returned and stored in x.  
Output:  
$ javac Output.java  
$ java Output  
1

8. What is the output of this program?

1. **class** Output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. Double i = **new** Double(257.578);
6. **int** x = i.intValue();
7. System.out.print(x);
8. }
9. }

a) 0  
b) 1  
c) 256  
d) 257  
View Answer

Answer: d  
Explanation: i.intValue() method returns the value of wrapper i as a Integer. i is 257.578 is double number when converted to an integer data type its value is 257.  
Output:  
$ javac Output.java  
$ java Output  
257

9. What is the output of this program?

1. **class** Output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. Double i = **new** Double(257.578123456789);
6. **float** x = i.floatValue();
7. System.out.print(x);
8. }
9. }

a) 0  
b) 257.0  
c) 257.57812  
d) 257.578123456789  
View Answer

Answer: c  
Explanation: floatValue() converts the value of wrapper i into float, since float can measure till 5 places after decimal hence 257.57812 is stored in floating point variable x.  
Output:  
$ javac Output.java  
$ java Output  
257.57812

10. What is the output of this program?

1. **class** Output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. Double y = **new** Double(257.57812);
6. Double i = **new** Double(257.578123456789);
7. **try**
8. {
9. **int** x = i.compareTo(y);
10. System.out.print(x);
11. }
12. **catch**(ClassCastException e)
13. {
14. System.out.print("Exception");
15. }
16. }
17. }

a) 0  
b) 1  
c) Exception  
d) None of the mentioned  
View Answer

Answer: b  
Explanation: i.compareTo() methods two double values, if they are equal then 0 is returned and if not equal then 1 is returned, here 257.57812 and 257.578123456789 are not equal hence 1 is returned and stored in x.  
Output:  
$ javac Output.java  
$ java Output  
1

Java Questions & Answers – Java.io Introduction

This section of our 1000+ Java MCQs focuses on java.io library of Java Programming Language.

1. Which of these packages contain classes and interfaces used for input & output operations of a program?  
a) java.util  
b) java.lang  
c) java.io  
d) all of the mentioned  
View Answer

Answer: c  
Explanation: java.io provides support for input and output operations.

2. Which of these class is not a member class of java.io package?  
a) String  
b) StringReader  
c) Writer  
d) File  
View Answer

Answer: a  
Explanation: None.

3. Which of these interface is not a member of java.io package?  
a) DataInput  
b) ObjectInput  
c) ObjectFilter  
d) FileFilter  
View Answer

Answer: c  
Explanation: None.

4. Which of these class is not related to input and output stream in terms of functioning?  
a) File  
b) Writer  
c) InputStream  
d) Reader  
View Answer

Answer: a  
Explanation: A File describes properties of a file, a File object is used to obtain or manipulate the information associated with a disk file, such as the permissions, time date, and directories path, and to navigate subdirectories.

5. Which of these is specified by a File object?  
a) a file in disk  
b) directory path  
c) directory in disk  
d) none of the mentioned  
View Answer

Answer: c  
Explanation: None.

6. Which of these is method for testing whether the specified element is a file or a directory?  
a) IsFile()  
b) isFile()  
c) Isfile()  
d) isfile()  
View Answer

Answer: b  
Explanation: isFile() returns true if called on a file and returns false when called on a directory.

7. What is the output of this program?

1. **import** java.io.\*;
2. **class** files
3. {
4. **public** **static** **void** main(String args[])
5. {
6. File obj = **new** File("/java/system");
7. System.out.print(obj.getName());
8. }
9. }

a) java  
b) system  
c) java/system  
d) /java/system  
View Answer

Answer: b  
Explanation: obj.getName() returns the name of the file.  
Output:  
$ javac files.java  
$ java files  
system

8. What is the output of this program?

1. **import** java.io.\*;
2. **class** files
3. {
4. **public** **static** **void** main(String args[])
5. {
6. File obj = **new** File("/java/system");
7. System.out.print(obj.getAbsolutePath());
8. }
9. }

Note: file is made in c drive.  
a) java  
b) system  
c) java/system  
d) /java/system  
View Answer

Answer: d  
Explanation: None.  
Output:  
$ javac files.java  
$ java files  
\java\system

9. What is the output of this program?

1. **import** java.io.\*;
2. **class** files
3. {
4. **public** **static** **void** main(String args[])
5. {
6. File obj = **new** File("/java/system");
7. System.out.print(obj.canWrite());
8. System.out.print(" " + obj.canRead());
9. }
10. }

Note: file is made in c drive.  
a) true false  
b) false true  
c) true true  
d) false false  
View Answer

Answer: d  
Explanation: None.  
Output:  
$ javac files.java  
$ java files  
false false

10. What is the output of this program?

1. **import** java.io.\*;
2. **class** files
3. {
4. **public** **static** **void** main(String args[])
5. {
6. File obj = **new** File("/java/system");
7. System.out.print(obj.getParent());
8. System.out.print(" " + obj.isFile());
9. }
10. }

Note: file is made in c drive.  
a) java true  
b) java false  
c) \java false  
d) \java true  
View Answer

Answer: c  
Explanation: getparent() giver the parent directory of the file and isfile() checks weather the present file is a directory or a file in the disk  
Output:  
$ javac files.java  
$ java files  
\java false

Java Questions & Answers – Java.io Byte Streams

This section of our 1000+ Java MCQs focuses on java.io byte streams of Java Programming Language.

1. Which of these classes is used for input and output operation when working with bytes?  
a) InputStream  
b) Reader  
c) Writer  
d) All of the mentioned  
View Answer

Answer: a  
Explanation: InputStream & OutputStream are designed for byte stream. Reader and writer are designed for character stream.

2. Which of these class is used to read and write bytes in a file?  
a) FileReader  
b) FileWriter  
c) FileInputStream  
d) InputStreamReader  
View Answer

Answer: c  
Explanation: None.

3. Which of these method of InputStream is used to read integer representation of next available byte input?  
a) read()  
b) scanf()  
c) get()  
d) getInteger()  
View Answer

Answer: a  
Explanation: None.

4. Which of these data type is returned by every method of OutputStream?  
a) int  
b) float  
c) byte  
d) none of the mentioned  
View Answer

Answer: d  
Explanation: Every method of OutputStream returns void and throws an IOExeption in case of errors.

5. Which of these is a method to clear all the data present in output buffers?  
a) clear()  
b) flush()  
c) fflush()  
d) close()  
View Answer

Answer: b  
Explanation: None.

6. Which of these method(s) is/are used for writing bytes to an outputstream?  
a) put()  
b) print() and write()  
c) printf()  
d) write() and read()  
View Answer

Answer: b  
Explanation: write() and print() are the two methods of OutputStream that are used for printing the byte data.

7. What is the output of this program?

1. **import** java.io.\*;
2. **class** filesinputoutput
3. {
4. **public** **static** **void** main(String args[])
5. {
6. InputStream obj = **new** FileInputStream("inputoutput.java");
7. System.out.print(obj.available());
8. }
9. }

Note: inputoutput.java is stored in the disk.  
a) true  
b) false  
c) prints number of bytes in file  
d) prints number of characters in the file  
View Answer

Answer: c  
Explanation: obj.available() returns the number of bytes.  
Output:  
$ javac filesinputoutput.java  
$ java filesinputoutput  
1422  
(Output will be different in your case)

8. What is the output of this program?

1. **import** java.io.\*;
2. **public** **class** filesinputoutput
3. {
4. **public** **static** **void** main(String[] args)
5. {
6. String obj = "abc";
7. **byte** b[] = obj.getBytes();
8. ByteArrayInputStream obj1 = **new** ByteArrayInputStream(b);
9. **for** (**int** i = 0; i < 2; ++ i)
10. {
11. **int** c;
12. **while** ((c = obj1.read()) != -1)
13. {
14. **if**(i == 0)
15. {
16. System.out.print((**char**)c);
17. }
18. }
19. }
20. }
21. }

a) abc  
b) ABC  
c) ab  
d) AB  
View Answer

Answer: a  
Explanation: None.  
Output:  
$ javac filesinputoutput.java  
$ java filesinputoutput  
abc

9. What is the output of this program?

1. **import** java.io.\*;
2. **public** **class** filesinputoutput
3. {
4. **public** **static** **void** main(String[] args)
5. {
6. String obj = "abc";
7. **byte** b[] = obj.getBytes();
8. ByteArrayInputStream obj1 = **new** ByteArrayInputStream(b);
9. **for** (**int** i = 0; i < 2; ++ i)
10. {
11. **int** c;
12. **while** ((c = obj1.read()) != -1)
13. {
14. **if** (i == 0)
15. {
16. System.out.print(Character.toUpperCase((**char**)c));
17. }
18. }
19. }
20. }
21. }

a) abc  
b) ABC  
c) ab  
d) AB  
View Answer

Answer: b  
Explanation: None.  
Output:  
$ javac filesinputoutput.java  
$ java filesinputoutput  
ABC

10. What is the output of this program?

1. **import** java.io.\*;
2. **public** **class** filesinputoutput
3. {
4. **public** **static** **void** main(String[] args)
5. {
6. String obj = "abc";
7. **byte** b[] = obj.getBytes();
8. ByteArrayInputStream obj1 = **new** ByteArrayInputStream(b);
9. **for** (**int** i = 0; i < 2; ++ i)
10. {
11. **int** c;
12. **while** ((c = obj1.read()) != -1)
13. {
14. **if** (i == 0)
15. {
16. System.out.print(Character.toUpperCase((**char**)c));
17. obj2.write(1);
18. }
19. }
20. System.out.print(obj2);
21. }
22. }
23. }

a) AaBaCa  
b) ABCaaa  
c) AaaBaaCaa  
d) AaBaaCaaa  
View Answer

Answer: d  
Explanation: None.  
Output:  
$ javac filesinputoutput.java  
$ java filesinputoutput  
AaBaaCaaa

} Java Questions & Answers – Java.io Character Streams

This section of our 1000+ Java MCQs focuses on character streams of Java Programming Language.

1. Which of these stream contains the classes which can work on character stream?  
a) InputStream  
b) OutputStream  
c) Character Stream  
d) All of the mentioned  
View Answer

Answer: c  
Explanation: InputStream & OutputStream classes under byte stream they are not streams. Character Stream contains all the classes which can work with Unicode.

2. Which of these class is used to read characters in a file?  
a) FileReader  
b) FileWriter  
c) FileInputStream  
d) InputStreamReader  
View Answer

Answer: a  
Explanation: None.

3. Which of these method of FileReader class is used to read characters from a file?  
a) read()  
b) scanf()  
c) get()  
d) getInteger()  
View Answer

Answer: a  
Explanation: None.

4. Which of these class can be used to implement input stream that uses a character array as the source?  
a) BufferedReader  
b) FileReader  
c) CharArrayReader  
d) FileArrayReader  
View Answer

Answer: c  
Explanation: CharArrayReader is an implementation of an input stream that uses character array as a source. Here array is the input source.

5. Which of these is a method to clear all the data present in output buffers?  
a) clear()  
b) flush()  
c) fflush()  
d) close()  
View Answer

Answer: b  
Explanation: None.

6. Which of these classes can return more than one character to be returned to input stream?  
a) BufferedReader  
b) Bufferedwriter  
c) PushbachReader  
d) CharArrayReader  
View Answer

Answer: c  
Explanation: PushbackReader class allows one or more characters to be returned to the input stream. This allows looking ahead in input stream and performing action accordingly.

7. What is the output of this program?

1. **import** java.io.\*;
2. **class** filesinputoutput
3. {
4. **public** **static** **void** main(String args[])
5. {
6. InputStream obj = **new** FileInputStream("inputoutput.java");
7. System.out.print(obj.available());
8. }
9. }

Note: inputoutput.java is stored in the disk.  
a) true  
b) false  
c) prints number of bytes in file  
d) prints number of characters in the file  
View Answer

Answer: c  
Explanation: obj.available() returns the number of bytes.  
Output:  
$ javac filesinputoutput.java  
$ java filesinputoutput  
1422  
(Output will be different in your case)

8. What is the output of this program?

1. **import** java.io.\*;
2. **class** Chararrayinput
3. {
4. **public** **static** **void** main(String[] args)
5. {
6. String obj = "abcdef";
7. **int** length = obj.length();
8. **char** c[] = **new** **char**[length];
9. obj.getChars(0,length,c,0);
10. CharArrayReader input1 = **new** CharArrayReader(c);
11. CharArrayReader input2 = **new** CharArrayReader(c, 0, 3);
12. **int** i;
13. **try**
14. {
15. **while** ((i = input1.read()) != -1)
16. {
17. System.out.print((**char**)i);
18. }
19. }
20. **catch** (IOException e)
21. {
22. *// TODO Auto-generated catch block*
23. e.printStackTrace();
24. }
25. }
26. }

a) abc  
b) abcd  
c) abcde  
d) abcdef  
View Answer

Answer: d  
Explanation: None.  
Output:  
$ javac Chararrayinput.java  
$ java Chararrayinput  
abcdef

9. What is the output of this program?

1. **import** java.io.\*;
2. **class** Chararrayinput
3. {
4. **public** **static** **void** main(String[] args)
5. {
6. String obj = "abcdef";
7. **int** length = obj.length();
8. **char** c[] = **new** **char**[length];
9. obj.getChars(0, length, c, 0);
10. CharArrayReader input1 = **new** CharArrayReader(c);
11. CharArrayReader input2 = **new** CharArrayReader(c, 0, 3);
12. **int** i;
13. **try**
14. {
15. **while** ((i = input2.read()) != -1)
16. {
17. System.out.print((**char**)i);
18. }
19. }
20. **catch** (IOException e)
21. {
22. *// TODO Auto-generated catch block*
23. e.printStackTrace();
24. }
25. }
26. }

a) abc  
b) abcd  
c) abcde  
d) abcdef  
View Answer

Answer: a  
Explanation: None.  
Output:  
$ javac Chararrayinput.java  
$ java Chararrayinput  
abc

10. What is the output of this program?

1. **import** java.io.\*;
2. **class** Chararrayinput
3. {
4. **public** **static** **void** main(String[] args)
5. {
6. String obj = "abcdefgh";
7. **int** length = obj.length();
8. **char** c[] = **new** **char**[length];
9. obj.getChars(0, length, c, 0);
10. CharArrayReader input1 = **new** CharArrayReader(c);
11. CharArrayReader input2 = **new** CharArrayReader(c, 1, 4);
12. **int** i;
13. **int** j;
14. **try**
15. {
16. **while** ((i = input1.read()) == (j = input2.read()))
17. {
18. System.out.print((**char**)i);
19. }
20. }
21. **catch** (IOException e)
22. {
23. *// TODO Auto-generated catch block*
24. e.printStackTrace();
25. }
26. }

a) abc  
b) abcd  
c) abcde  
d) None of the mentioned  
View Answer

Answer: d  
Explanation: No output is printed. CharArrayReader object input1 contains string “abcdefgh” whereas object input2 contains string “bcde”, when while((i=input1.read())==(j=input2.read())) is executed the starting character of each object is compared since they are unequal control comes out of loop and nothing is printed on the screen.  
Output:  
$ javac Chararrayinput.java  
$ java Chararrayinput

# Java Questions & Answers – Memory Management

This set of Java Multiple Choice Questions & Answers (MCQs) focuses on “Memory Management”.

1. Which of the following is not a segment of memory in java?  
a) Stack Segment  
b) Heap Segment  
c) Code Segment  
d) Register Segment  
View Answer

Answer: d  
Explanation: There are only 3 type of memory segment. Stack Segment, Heap Segment and Code Segment.

2. Does code Segment loads the java code?  
a) True  
b) False  
View Answer

Answer: a  
Explanation: Code Segment loads compiled java bytecode.Byte code is platform independent.

3. What is JVM?  
a) Bootstrap  
b) Interpretor  
c) Extension  
d) Compiler  
View Answer

Answer: b  
Explanation: JVM is interpretor. It reads .class files which is the byte code generated by compiler line by line and converts it into native OS code.

4. Which one of the following is a class loader?  
a) Bootstrap  
b) Compiler  
c) Heap  
d) Interpretor  
View Answer

Answer: a  
Explanation: Bootstrap is a class loader. It loads the classes into memory.

5. Which class loader loads jar files from JDK directory?  
a) Bootstrap  
b) Extension  
c) System  
d) Heap  
View Answer

Answer: b  
Explanation: Extension loads jar files from lib/ext directory of the JRE.This gives the basic functionality available.

6. Which of the following is not a memory classification in java?  
a) Young  
b) Old  
c) Permanent  
d) Temporary  
View Answer

Answer: d  
Explanation: Young generation is further classified into Eden space and Survivor space. Old generation is also the tenured space. Permanent generation is the non heap space.

7. What is the java 8 update of PermGen?  
a) Code Cache  
b) Tenured Space  
c) Metaspace  
d) Eden space  
View Answer

Answer: c  
Explanation: Metaspace is the replacement of PermGen in java 8. It is very similar to PermGen except that it resizes itself dynamically. Thus, it is unbounded.

8. Classes and Methods are stored in which space?  
a) Eden space  
b) Survivor space  
c) Tenured space  
d) Permanent space  
View Answer

Answer: d  
Explanation: The permanent generation holds objects which JVM finds convenient to have the garbage collector. Objects describing classes and methods, as well as the classes and methods themselves are a part of Permanent generation.

9. Where is String Pool stored?  
a) Java Stack  
b) Java Heap  
c) Permanent Generation  
d) Metaspace  
View Answer

Answer: b  
Explanation: When a string is created ; if the string already exists in the pool, the reference of the existing string will be returned, else a new object is created and its reference is returned.

10. The same import package/class be called twice in java?  
a) True  
b) False  
View Answer

Answer: a  
Explanation: We can import the same package or same class multiple times. Neither compiler nor JVM complains will complain about it. JVM will internally load the class only once no matter how many times we import the same class or package.

Java Questions & Answers – Java’s Built in Exceptions

This section of our 1000+ Java MCQs focuses on Java’s built in exceptions of Java Programming Language.

1. Which of these exceptions handles the situations when illegal argument is used to invoke a method?  
a) IllegalException  
b) Argument Exception  
c) IllegalArgumentException  
d) IllegalMethodArgumentExcepetion  
View Answer

Answer: c  
Explanation: None.

2. Which of these exceptions will be thrown if we declare an array with negative size?  
a) IllegalArrayException  
b) IllegalArraySizeExeption  
c) NegativeArrayException  
d) NegativeArraySizeExeption  
View Answer

Answer: d  
Explanation: Array size must always be positive, if we declare an array with negative size then built in exception “NegativeArraySizeException” is thrown by the java’s run time system.

3. Which of these packages contain all the Java’s built in exceptions?  
a) java.io  
b) java.util  
c) java.lang  
d) java.net  
View Answer

Answer: c  
Explanation: None.

4. Which of these exceptions will be thrown if we use null reference for an arithmetic operation?  
a) ArithmeticException  
b) NullPointerException  
c) IllegalAccessException  
d) IllegalOperationException  
View Answer

Answer: b  
Explanation: If we use null reference anywhere in the code where the value stored in that reference is used then NullPointerException occurs.

5. Which of these class is used to create user defined exception?  
a) java.lang  
b) Exception  
c) RunTime  
d) System  
View Answer

Answer: b  
Explanation: Exception class contains all the methods necessary for defining an exception. The class contains the Throwable class.

6. What is the output of this program?

1. **class** exception\_handling
2. {
3. **public** **static** **void** main(String args[])
4. {
5. **try**
6. {
7. **int** a[] = {1, 2,3 , 4, 5};
8. **for** (**int** i = 0; i < 7; ++i)
9. System.out.print(a[i]);
10. }
11. **catch**(ArrayIndexOutOfBoundsException e)
12. {
13. System.out.print("0");
14. }
15. }
16. }

a) 12345  
b) 123450  
c) 1234500  
d) Compilation Error  
View Answer

Answer: b  
Explanation: When array index goes out of bound then ArrayIndexOutOfBoundsException exception is thrown by the system.  
Output:  
$ javac exception\_handling.java  
$ java exception\_handling  
123450

7. What is the output of this program?

1. **class** exception\_handling
2. {
3. **public** **static** **void** main(String args[])
4. {
5. **try**
6. {
7. **int** a[] = {1, 2,3 , 4, 5};
8. **for** (**int** i = 0; i < 5; ++i)
9. System.out.print(a[i]);
10. **int** x = 1/0;
11. }
12. **catch**(ArrayIndexOutOfBoundsException e)
13. {
14. System.out.print("A");
15. }
16. **catch**(ArithmeticException e)
17. {
18. System.out.print("B");
19. }
20. }
21. }

a) 12345  
b) 12345A  
c) 12345B  
d) Comiplation Error  
View Answer

Answer: d  
Explanation: There can be more than one catch of a single try block. Here Arithmetic exception occurs instead of Array index out of bound exception hence B is printed after 12345  
Output:  
$ javac exception\_handling.java  
$ java exception\_handling  
12345B

8. What is the output of this program?

1. **class** exception\_handling
2. {
3. **static** **void** throwexception() **throws** ArithmeticException
4. {
5. System.out.print("0");
6. **throw** **new** ArithmeticException ("Exception");
7. }
8. **public** **static** **void** main(String args[])
9. {
10. **try**
11. {
12. throwexception();
13. }
14. **catch** (ArithmeticException e)
15. {
16. System.out.println("A");
17. }
18. }
19. }

a) A  
b) 0  
c) 0A  
d) Exception  
View Answer

Answer: c  
Explanation: None.  
Output:  
$ javac exception\_handling.java  
$ java exception\_handling  
0A

9. What is the output of this program?

1. **class** exception\_handling
2. {
3. **public** **static** **void** main(String args[])
4. {
5. **try**
6. {
7. **int** a = 1;
8. **int** b = 10 / a;
9. **try**
10. {
11. **if** (a == 1)
12. a = a / a - a;
13. **if** (a == 2)
14. {
15. **int** c[] = {1};
16. c[8] = 9;
17. }
18. **finally**
19. {
20. System.out.print("A");
21. }
23. }
24. **catch** (NullPointerException e)
25. {
26. System.out.println("B");
27. }
28. }
29. }

a) A  
b) B  
c) AB  
d) BA  
View Answer

Answer: c  
Explanation: The inner try block does not have a catch which can tackle ArrayIndexOutOfBoundException hence finally is executed which prints ‘A’ the outer try block does have catch for NullPointerException exception but no such exception occurs in it hence its catch is never executed and only ‘A’ is printed.  
Output:  
$ javac exception\_handling.java  
$ java exception\_handling  
A

10. What is the output of this program?

1. **class** exception\_handling
2. {
3. **public** **static** **void** main(String args[])
4. {
5. **try**
6. {
7. **int** a = args.length;
8. **int** b = 10 / a;
9. System.out.print(a);
10. **try**
11. {
12. **if** (a == 1)
13. a = a / a - a;
14. **if** (a == 2)
15. {
16. **int** c = {1};
17. c[8] = 9;
18. }
19. }
20. **catch** (ArrayIndexOutOfBoundException e)
21. {
22. System.out.println("TypeA");
23. }
24. **catch** (ArithmeticException e)
25. {
26. System.out.println("TypeB");
27. }
28. }
29. }

a) TypeA  
b) TypeB  
c) 0TypeA  
d) 0TypeB  
Note: Execution command line: $ java exception\_handling one two  
View Answer

Answer: c  
Explanation: Execution command line is “$ java exception\_ handling one two” hence there are two input making args.length = 2, hence “c[8] = 9” in second try block is executing which throws ArrayIndexOutOfBoundException which is caught by catch of nested try block. Hence 0TypeB is printed  
Output:  
$ javac exception\_handling.java  
$ java exception\_handling  
0TypeB

Java Questions & Answers – Java.lang – Rounding Functions

This section of our 1000+ Java MCQs focuses on rounding functions in Java Programming Language.

1. Which of these class provides various types of rounding functions?  
a) Math  
b) Process  
c) System  
d) Object  
View Answer

Answer: a  
Explanation: None.

2. Which of these method return a smallest whole number greater than or equal to variable X?  
a) double ceil(double X)  
b) double floor(double X)  
c) double max(double X)  
d) double min(double X)  
View Answer

Answer: a  
Explanation: ceil(double X) returns the smallest whole number greater than or equal to variable X.

3. Which of these method return a largest whole number less than or equal to variable X?  
a) double ceil(double X)  
b) double floor(double X)  
c) double max(double X)  
d) double min(double X)  
View Answer

Answer: b  
Explanation: double floor(double X) returns a largest whole number less than or equal to variable X.

4. Which of these method is a rounding function of Math class?  
a) max()  
b) min()  
c) abs()  
d) rint()  
View Answer

Answer: d  
Explanation: rint() rounds up a variable to nearest integer.

5. Which of these class contains only floating point functions?  
a) Math  
b) Process  
c) System  
d) Object  
View Answer

Answer: a  
Explanation: Math class contains all the floating point functions that are used for geometry, trigonometry, as well as several general purpose methods. Example : sin(), cos(), exp(), sqrt() etc.

6. Which of function return absolute value of a variable?  
a) abs()  
b) absolute()  
c) absolutevariable()  
d) none of the mentioned  
View Answer

Answer: a  
Explanation: abs() returns the absolute value of a variable.

7. What is the output of this program?

1. **class** A
2. {
3. **int** x;
4. **int** y;
5. **void** display()
6. {
7. System.out.print(x + " " + y);
8. }
9. }
10. **class** Output
11. {
12. **public** **static** **void** main(String args[])
13. {
14. A obj1 = **new** A();
15. A obj2 = **new** A();
16. obj1.x = 1;
17. obj1.y = 2;
18. obj2 = obj1.clone();
19. obj1.display();
20. obj2.display();
21. }
22. }

a) 1 2 0 0  
b) 1 2 1 2  
c) 0 0 0 0  
d) System Dependent  
View Answer

Answer: b  
Explanation: clone() method of object class is used to generate duplicate copy of the object on which it is called. Copy of obj1 is generated and stored in obj2.  
Output:  
$ javac Output.java  
$ java Output  
1 2 1 2

8. What is the output of this program?

1. **class** Output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. **double** x = 3.14;
6. **int** y = (**int**) Math.abs(x);
7. System.out.print(y);
8. }
9. }

a) 0  
b) 3  
c) 3.0  
d) 3.1  
View Answer

Answer: b  
Explanation: None.  
Output:  
$ javac Output.java  
$ java Output  
3

9. What is the output of this program?

1. **class** Output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. **double** x = 3.14;
6. **int** y = (**int**) Math.ceil(x);
7. System.out.print(y);
8. }
9. }

a) 0  
b) 3  
c) 3.0  
d) 4  
View Answer

Answer: d  
Explanation: ciel(double X) returns the smallest whole number greater than or equal to variable x.  
Output:  
$ javac Output.java  
$ java Output  
4

10. What is the output of this program?

1. **class** Output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. **double** x = 3.14;
6. **int** y = (**int**) Math.floor(x);
7. System.out.print(y);
8. }
9. }

a) 0  
b) 3  
c) 3.0  
d) 4  
View Answer

Answer: d  
Explanation: double floor(double X) returns a largest whole number less than or equal to variable X. Here the smallest whole number less than 3.14 is 3.  
Output:  
$ javac Output.java  
$ java Output  
3

Java Questions & Answers – Java.lang – Byte & Short Wrappers

This set of Java Multiple Choice Questions & Answers (MCQs) focuses on “Java.lang – Byte & Short Wrappers”.

1. Which of these methods of Byte wrapper can be used to obtain Byte object from a string?  
a) toString()  
b) getString()  
c) decode()  
d) encode()  
View Answer

Answer: c  
Explanation: decode() methods returns a Byte object that contains the value specified by string.

2. Which of the following methods Byte wrapper return the value as a double?  
a) doubleValue()  
b) converDouble()  
c) getDouble()  
d) getDoubleValue()  
View Answer

Answer: a  
Explanation: doubleValue() returns the value of invoking object as double.

3. Which of these is a super class of wrappers Byte and short wrappers?  
a) Long  
b) Digits  
c) Float  
d) Number  
View Answer

Answer: d  
Explanation: Number is an abstract class containing subclasses Double, Float, Byte, Short, Integer and Long.

4. Which of these methods is not defined in both Byte and Short wrappers?  
a) intValue()  
b) isInfinite()  
c) toString()  
d) hashCode()  
View Answer

Answer: b  
Explanation: isInfinite() methods is defined in Integer and Long Wrappers, returns true if specified value is an infinite value otherwise it returns false.

5. Which of these exceptions is thrown by compareTo() method defined in double wrapper?  
a) IOException  
b) SystemException  
c) CastException  
d) ClassCastException  
View Answer

Answer: d  
Explanation: compareTo() methods compare the specified object to be double, if it is not then ClassCastException is thrown.

6. What is the output of this program?

1. **class** Output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. Double i = **new** Double(257.5);
6. Double x = i.MAX\_VALUE;
7. System.out.print(x);
8. }
9. }

a) 0  
b) 1.7976931348623157E308  
c) 1.7976931348623157E30  
d) None of the mentioned  
View Answer

Answer: b  
Explanation: The super class of Double class defines a constant MAX\_VALUE above which a number is considered to be infinity. MAX\_VALUE is 1.7976931348623157E308.  
Output:  
$ javac Output.java  
$ java Output  
1.7976931348623157E308

7. What is the output of this program?

1. **class** Output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. Double i = **new** Double(257.5);
6. Double x = i.MIN\_VALUE;
7. System.out.print(x);
8. }
9. }

a) 0  
b) 4.9E-324  
c) 1.7976931348623157E308  
d) None of the mentioned  
View Answer

Answer: b  
Explanation: The super class of Byte class defines a constant MIN\_VALUE below which a number is considered to be negative infinity. MIN\_VALUE is 4.9E-324.  
Output:  
$ javac Output.java  
$ java Output  
4.9E-324

8. What is the output of this program?

1. **class** Output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. Integer i = **new** Integer(257);
6. **byte** x = i.byteValue();
7. System.out.print(x);
8. }
9. }

a) 0  
b) 1  
c) 256  
d) 257  
View Answer

Answer: b  
Explanation: i.byteValue() method returns the value of wrapper i as a byte value. i is 257, range of byte is 256 therefore i value exceeds byte range by 1 hence 1 is returned and stored in x.  
Output:  
$ javac Output.java  
$ java Output  
1

9. What is the output of this program?

1. **class** Output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. Double i = **new** Double(257.578123456789);
6. **float** x = i.floatValue();
7. System.out.print(x);
8. }
9. }

a) 0  
b) 257.0  
c) 257.57812  
d) 257.578123456789  
View Answer

Answer: c  
Explanation: floatValue() converts the value of wrapper i into float, since float can measure till 5 places after decimal hence 257.57812 is stored in floating point variable x.  
Output:  
$ javac Output.java  
$ java Output  
257.57812

10. What is the output of this program?

1. **class** Output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. Double y = **new** Double(257.57812);
6. Double i = **new** Double(257.578123456789);
7. **try**
8. {
9. **int** x = i.compareTo(y);
10. System.out.print(x);
11. }
12. **catch**(ClassCastException e)
13. {
14. System.out.print("Exception");
15. }
16. }
17. }

a) 0  
b) 1  
c) Exception  
d) None of the mentioned  
View Answer

Answer: b  
Explanation: i.compareTo() methods two double values, if they are equal then 0 is returned and if not equal then 1 is returned, here 257.57812 and 257.578123456789 are not equal hence 1 is returned and stored in x.  
Output:  
$ javac Output.java  
$ java Output  
1

Java Questions & Answers – Java.lang – Byte & Short Wrappers

This set of Java Multiple Choice Questions & Answers (MCQs) focuses on “Java.lang – Byte & Short Wrappers”.

1. Which of these methods of Byte wrapper can be used to obtain Byte object from a string?  
a) toString()  
b) getString()  
c) decode()  
d) encode()  
View Answer

Answer: c  
Explanation: decode() methods returns a Byte object that contains the value specified by string.

2. Which of the following methods Byte wrapper return the value as a double?  
a) doubleValue()  
b) converDouble()  
c) getDouble()  
d) getDoubleValue()  
View Answer

Answer: a  
Explanation: doubleValue() returns the value of invoking object as double.

3. Which of these is a super class of wrappers Byte and short wrappers?  
a) Long  
b) Digits  
c) Float  
d) Number  
View Answer

Answer: d  
Explanation: Number is an abstract class containing subclasses Double, Float, Byte, Short, Integer and Long.

4. Which of these methods is not defined in both Byte and Short wrappers?  
a) intValue()  
b) isInfinite()  
c) toString()  
d) hashCode()  
View Answer

Answer: b  
Explanation: isInfinite() methods is defined in Integer and Long Wrappers, returns true if specified value is an infinite value otherwise it returns false.

5. Which of these exceptions is thrown by compareTo() method defined in double wrapper?  
a) IOException  
b) SystemException  
c) CastException  
d) ClassCastException  
View Answer

Answer: d  
Explanation: compareTo() methods compare the specified object to be double, if it is not then ClassCastException is thrown.

6. What is the output of this program?

1. **class** Output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. Double i = **new** Double(257.5);
6. Double x = i.MAX\_VALUE;
7. System.out.print(x);
8. }
9. }

a) 0  
b) 1.7976931348623157E308  
c) 1.7976931348623157E30  
d) None of the mentioned  
View Answer

Answer: b  
Explanation: The super class of Double class defines a constant MAX\_VALUE above which a number is considered to be infinity. MAX\_VALUE is 1.7976931348623157E308.  
Output:  
$ javac Output.java  
$ java Output  
1.7976931348623157E308

7. What is the output of this program?

1. **class** Output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. Double i = **new** Double(257.5);
6. Double x = i.MIN\_VALUE;
7. System.out.print(x);
8. }
9. }

a) 0  
b) 4.9E-324  
c) 1.7976931348623157E308  
d) None of the mentioned  
View Answer

Answer: b  
Explanation: The super class of Byte class defines a constant MIN\_VALUE below which a number is considered to be negative infinity. MIN\_VALUE is 4.9E-324.  
Output:  
$ javac Output.java  
$ java Output  
4.9E-324

8. What is the output of this program?

1. **class** Output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. Integer i = **new** Integer(257);
6. **byte** x = i.byteValue();
7. System.out.print(x);
8. }
9. }

a) 0  
b) 1  
c) 256  
d) 257  
View Answer

Answer: b  
Explanation: i.byteValue() method returns the value of wrapper i as a byte value. i is 257, range of byte is 256 therefore i value exceeds byte range by 1 hence 1 is returned and stored in x.  
Output:  
$ javac Output.java  
$ java Output  
1

9. What is the output of this program?

1. **class** Output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. Double i = **new** Double(257.578123456789);
6. **float** x = i.floatValue();
7. System.out.print(x);
8. }
9. }

a) 0  
b) 257.0  
c) 257.57812  
d) 257.578123456789  
View Answer

Answer: c  
Explanation: floatValue() converts the value of wrapper i into float, since float can measure till 5 places after decimal hence 257.57812 is stored in floating point variable x.  
Output:  
$ javac Output.java  
$ java Output  
257.57812

10. What is the output of this program?

1. **class** Output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. Double y = **new** Double(257.57812);
6. Double i = **new** Double(257.578123456789);
7. **try**
8. {
9. **int** x = i.compareTo(y);
10. System.out.print(x);
11. }
12. **catch**(ClassCastException e)
13. {
14. System.out.print("Exception");
15. }
16. }
17. }

a) 0  
b) 1  
c) Exception  
d) None of the mentioned  
View Answer

Answer: b  
Explanation: i.compareTo() methods two double values, if they are equal then 0 is returned and if not equal then 1 is returned, here 257.57812 and 257.578123456789 are not equal hence 1 is returned and stored in x.  
Output:  
$ javac Output.java  
$ java Output  
1

Java Questions & Answers – Java.lang – Boolean Wrapper Advance

This set of Java Multiple Choice Questions & Answers (MCQs) focuses on “Java.lang – Boolean Wrapper Advance”.

1. Which of these methods of Boolean wrapper returns boolean equivalent of an object.  
a) getBool()  
b) booleanValue()  
c) getbooleanValue()  
d) getboolValue()  
View Answer

Answer: b  
Explanation: None.

2. Which of the following constant are defined in Boolean wrapper?  
a) TRUE  
b) FALSE  
c) TYPE  
d) All of the mentioned  
View Answer

Answer: d  
Explanation: Boolean wrapper defines 3 constants – TRUE, FALSE & TYPE.

3. Which of these methods return string equivalent of Boolean object?  
a) getString()  
b) toString()  
c) converString()  
d) getStringObject()  
View Answer

Answer: b  
Explanation: None.

4. Which of these methods is used to know whether a string contains “true”?  
a) valueOf()  
b) valueOfString()  
c) getString()  
d) none of the mentioned  
View Answer

Answer: a  
Explanation: valueOf() returns true if the specified string contains “true” in lower or uppercase and false otherwise.

5. Which of these class have only one field?  
a) Character  
b) Boolean  
c) Byte  
d) void  
View Answer

Answer: d  
Explanation: Void class has only one field – TYPE, ehich holds a reference to the Class object for type void. We do not create instance of this class.

6. What is the output of this program?

1. **class** Output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. String str = "true";
6. **boolean** x = Boolean.valueOf(str);
7. System.out.print(x);
8. }
9. }

a) True  
b) False  
c) Compilation Error  
d) Runtime Error  
View Answer

Answer: a  
Explanation: valueOf() returns true if the specified string contains “true” in lower or uppercase and false otherwise.  
Output:  
$ javac Output.java  
$ java Output  
true

7. What is the output of this program?

1. **class** Output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. String str = "true false true";
6. **boolean** x = Boolean.valueOf(str);
7. System.out.print(x);
8. }
9. }

a) True  
b) False  
c) Compilation Error  
d) Runtime Error  
View Answer

Answer: b  
Explanation: valueOf() returns true if the specified string contains “true” in lower or uppercase and false otherwise.  
Output:  
$ javac Output.java  
$ java Output  
false

8. What is the output of this program?

1. **class** Output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. String str = "TRUE";
6. **boolean** x = Boolean.valueOf(str);
7. System.out.print(x);
8. }
9. }

a) True  
b) False  
c) Compilation Error  
d) Runtime Error  
View Answer

Answer: a  
Explanation: valueOf() returns a Boolean instance representing the specified boolean value. If the specified boolean value is true, this method returns Boolean.TRUE; if it is false, this method returns Boolean.FALSE. If a new Boolean instance is not required, this method should generally be used in preference to the constructor Boolean(boolean), as this method is likely to yield significantly better space and time.  
Output:  
$ javac Output.java  
$ java Output  
true

9. What is the output of this program?

1. **class** Output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. String str = "true false";
6. **boolean** x = Boolean.parseBoolean(str);
7. System.out.print(x);
8. }
9. }

a) True  
b) False  
c) System Dependent  
d) Compilation Error  
View Answer

Answer: b  
Explanation: parseBoolean() Parses the string argument as a boolean. The boolean returned represents the value true if the string argument is not null and is equal, ignoring case, to the string “true”.  
Example: Boolean.parseBoolean(“True”) returns true.  
Example: Boolean.parseBoolean(“yes”) returns false.  
Output:  
$ javac Output.java  
$ java Output  
false

10. What is the output of this program?

1. **class** Output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. String x = Boolean.toString(**false**);
6. }
7. }

a) True  
b) False  
c) System Dependent  
d) Compilation Error  
View Answer

Answer: b  
Explanation: toString() Returns a String object representing the specified boolean. If the specified boolean is true, then the string “true” will be returned, otherwise the string “false” will be returned  
Output:  
$ javac Output.java  
$ java Output  
false

Java Questions & Answers – Java.lang – Miscellaneous Math Methods & StrictMath Class

This section of our 1000+ Java MCQs focuses on miscellaneous Math methods & StrictMath class of Java Programming Language.

1. Which of these class contains all the methods present in Math class?  
a) SystemMath  
b) StrictMath  
c) Compiler  
d) ClassLoader  
View Answer

Answer: b  
Explanation: SystemMath class defines complete set of mathematical methods that are parallel those in Math class. The difference is that the StrictMath version is guaranteed to generate precisely identical results across all Java implementations.

2. Which of these method return a pseudorandom number?  
a) rand()  
b) random()  
c) randomNumber()  
d) randGenerator()  
View Answer

Answer: b  
Explanation: None.

3. Which of these method returns the remainder of dividend / devisor?  
a) remainder()  
b) getRemainder()  
c) CSIremainder()  
d) IEEEremainder()  
View Answer

Answer: d  
Explanation: IEEEremainder() returns the remainder of dividend / devisor.

4. Which of these method converts radians to degrees?  
a) toRadian()  
b) toDegree()  
c) convertRadian()  
d) converDegree()  
View Answer

Answer: b  
Explanation: None.

5. toRadian() and toDegree() methods were added by which version of Java?  
a) Java 1.0  
b) Java 1.5  
c) Java 2.0  
d) Java 3.0  
View Answer

Answer: c  
Explanation: toRadian() and toDegree() methods were added by Java 2.0 before that there was no method which could directly convert degree into radians and vice versa.

6. Which of these method return a smallest whole number greater than or equal to variable X?  
a) double ciel(double X)  
b) double floor(double X)  
c) double max(double X)  
d) double min(double X)  
View Answer

Answer: a  
Explanation: ciel(double X) returns the smallest whole number greater than or equal to variable X.

7. What is the output of this program?

1. **class** Output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. **double** x = 3.14;
6. **int** y = (**int**) Math.toDegrees(x);
7. System.out.print(y);
8. }
9. }

a) 0  
b) 179  
c) 180  
d) 360  
View Answer

Answer: b  
Explanation: 3.14 in degree 179.9087. We usually take it to be 180. Buts here we have type casted it to integer data type hence 179.  
Output:  
$ javac Output.java  
$ java Output  
179

8. What is the output of this program?

1. **class** Output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. **double** x = 3.14;
6. **int** y = (**int**) Math.toRadians(x);
7. System.out.print(y);
8. }
9. }

a) 0  
b) 3  
c) 3.0  
d) 3.1  
View Answer

Answer: a  
Explanation: None.  
Output:  
$ javac Output.java  
$ java Output  
0

9. What is the output of this program?

1. **class** Output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. **double** x = 102;
6. **double** y = 5;
7. **double** z = Math.IEEEremainder(x, y);
8. System.out.print(z);}
9. }
10. }

a) 0  
b) 1  
c) 2  
d) 3  
View Answer

Answer: b  
Explanation: IEEEremainder() returns the remainder of dividend / devisor. Here dividend is 102 and devisor is 5 therefore remainder is 2. It is similar to modulus – ‘%’ operator of C/C++ language.  
Output:  
$ javac Output.java  
$ java Output  
2

10. Will this program generate same output is executed again?

1. **class** Output
2. {
3. **public** **static** **void** main(String args[])
4. {
5. **int** y = **double** z = Math.random();
6. System.out.print(y);
7. }
8. }

a) Yes  
b) No  
c) Compiler Dependent  
d) Operating System Dependent  
View Answer

Answer: b  
Explanation: There is no relation between random numbers generated previously in Java

Java Questions & Answers – Java.lang – Runtime & ClassLoader Classes

This section of our 1000+ Java MCQs focuses Runtime & ClassLoader classes of Java Programming Language.

1. Which of these classes encapsulate runtime enviroment?  
a) Class  
b) System  
c) Runtime  
d) ClassLoader  
View Answer

Answer: c  
Explanation: None.

2. Which of the following exceptions is thrown by every method of Runtime class?  
a) IOException  
b) SystemException  
c) SecurityException  
d) RuntimeException  
View Answer

Answer: c  
Explanation: Every method of Runtime class throws SecurityException.

3. Which of these methods returns the total number of bytes of memory available to the program?  
a) getMemory()  
b) TotalMemory()  
c) SystemMemory()  
d) getProcessMemory()  
View Answer

Answer: b  
Explanation: TotalMemory() returns the total number of bytes available to the program.

4. Which of these class defines how the classes are loaded?  
a) Class  
b) System  
c) Runtime  
d) ClassLoader  
View Answer

Answer: d  
Explanation: None.

5. Which of these methods return a class object given its name?  
a) getClass()  
b) findClass()  
c) getSystemClass()  
d) findSystemClass()  
View Answer

Answer: d  
Explanation: findSystemClass() returns a class object given its name.

6. Which of these Exceptions is thrown by loadClass() method of ClassLoader class?  
a) IOException  
b) SystemException  
c) ClassFormatError  
d) ClassNotFoundException  
View Answer

Answer: d  
Explanation: None.

7. What is the output of this program?

1. **class** X
2. {
3. **int** a;
4. **double** b;
5. }
6. **class** Y **extends** X
7. {
8. **int** c;
9. }
10. **class** Output
11. {
12. **public** **static** **void** main(String args[])
13. {
14. X a = **new** X();
15. Y b = **new** Y();
16. **Class** obj;
17. obj = b.getClass();
18. System.out.print(obj.getSuperclass());
19. }
20. }

a) X  
b) Y  
c) class X  
d) class Y  
View Answer

Answer: c  
Explanation: getSuperClass() returns the super class of an object. b is an object of class Y which extends class X , Hence Super class of b is X. therefore class X is printed.  
Output:  
$ javac Output.java  
$ java Output  
class X

8. What is the output of this program?

1. **class** X
2. {
3. **int** a;
4. **double** b;
5. }
6. **class** Y **extends** X
7. {
8. **int** c;
9. }
10. **class** Output
11. {
12. **public** **static** **void** main(String args[])
13. {
14. X a = **new** X();
15. Y b = **new** Y();
16. **Class** obj;
17. obj = b.getClass();
18. System.out.print(b.equals(a));
19. }
20. }

a) 0  
b) 1  
c) true  
d) false  
View Answer

Answer: d  
Explanation: None.  
Output:  
$ javac Output.java  
$ java Output  
false

9. What is the output of this program?

1. **class** X
2. {
3. **int** a;
4. **double** b;
5. }
6. **class** Y **extends** X
7. {
8. **int** c;
9. }
10. **class** Output
11. {
12. **public** **static** **void** main(String args[])
13. {
14. X a = **new** X();
15. Y b = **new** Y();
16. **Class** obj;
17. obj = b.getClass();
18. System.out.print(obj.isInstance(a));
19. }
20. }

a) 0  
b) 1  
c) true  
d) false  
View Answer

Answer: d  
Explanation: Although class Y extends class X but still a is not considered related to Y. hence isInstance() returns false.  
Output:  
$ javac Output.java  
$ java Output  
false

10. What is the output of this program?

1. **class** X
2. {
3. **int** a;
4. **double** b;
5. }
6. **class** Y **extends** X
7. {
8. **int** c;
9. }
10. **class** Output
11. {
12. **public** **static** **void** main(String args[])
13. {
14. X a = **new** X();
15. Y b = **new** Y();
16. **Class** obj;
17. obj = b.getClass();
18. System.out.print(obj.isLocalClass());
19. }
20. }

a) 0  
b) 1  
c) true  
d) false  
View Answer

Answer: d  
Explanation: None.  
Output:  
$ javac Output.java  
$ java Output  
false

Java Questions & Answers – Java.lang – Class

This section of our 1000+ Java MCQs focuses Class of java.lang library of Java Programming Language.

1. Which of these classes encapsulate run-time state of an object?  
a) Class  
b) System  
c) Runtime  
d) Catche  
View Answer

Answer: a  
Explanation: None.

2. Which of the following constant are defined in Boolean wrapper?  
a) TRUE  
b) FLASE  
c) TYPE  
d) All of the mentioned  
View Answer

Answer: d  
Explanation: Boolean wrapper defines 3 constants – TRUE, FLASE & TYPE.

3. Which of these methods returns the class of an object?  
a) getClass()  
b) Class()  
c) WhoseClass()  
d) WhoseObject()  
View Answer

Answer: a  
Explanation: None.

4. Which of these methods is used to know whether a string contains “true”?  
a) valueOf()  
b) valueOfString()  
c) getString()  
d) None of the mentioned  
View Answer

Answer: a  
Explanation: valueOf() returns true if the specified string contains “true” in lower or uppercase and false otherwise.

5. Which of these class have only one field?  
a) Character  
b) Boolean  
c) Byte  
d) void  
View Answer

Answer: d  
Explanation: Void class has only one field – TYPE, which holds a reference to the Class object for type void. We do not create instance of this class.

6. What is the output of this program?

1. **class** X
2. {
3. **int** a;
4. **double** b;
5. }
6. **class** Y **extends** X
7. {
8. **int** c;
9. }
10. **class** Output
11. {
12. **public** **static** **void** main(String args[])
13. {
14. X a = **new** X();
15. Y b = **new** Y();
16. **Class** obj;
17. obj = a.getClass();
18. System.out.print(obj.getName());
19. }
20. }

a) X  
b) Y  
c) a  
d) b  
View Answer

Answer: a  
Explanation: getClass() is used to obtain the class of an object, here ‘a’ is an object of class ‘X’. hence a.getClass() returns ‘X’ which is stored in class Class object obj.  
Output:  
$ javac Output.java  
$ java Output  
X

7. What is the output of this program?

1. **class** X
2. {
3. **int** a;
4. **double** b;
5. }
6. **class** Y **extends** X
7. {
8. **int** c;
9. }
10. **class** Output
11. {
12. **public** **static** **void** main(String args[])
13. {
14. X a = **new** X();
15. Y b = **new** Y();
16. **Class** obj;
17. obj = b.getClass();
18. System.out.print(obj.getSuperclass());
19. }
20. }

a) X  
b) Y  
c) class X  
d) class Y  
View Answer

Answer: c  
Explanation: getSuperClass() returns the super class of an object. b is an object of class Y which extends class X , Hence Super class of b is X. therefore class X is printed.  
Output:  
$ javac Output.java  
$ java Output  
class X

8. What is the output of this program?

1. **class** X
2. {
3. **int** a;
4. **double** b;
5. }
6. **class** Y **extends** X
7. {
8. **int** c;
9. }
10. **class** Output
11. {
12. **public** **static** **void** main(String args[])
13. {
14. X a = **new** X();
15. Y b = **new** Y();
16. **Class** obj;
17. obj = b.getClass();
18. System.out.print(b.equals(a));
19. }
20. }

a) 0  
b) 1  
c) true  
d) false  
View Answer

Answer: d  
Explanation: None.  
Output:  
$ javac Output.java  
$ java Output  
false

9. What is the output of this program?

1. **class** X
2. {
3. **int** a;
4. **double** b;
5. }
6. **class** Y **extends** X
7. {
8. **int** c;
9. }
10. **class** Output
11. {
12. **public** **static** **void** main(String args[])
13. {
14. X a = **new** X();
15. Y b = **new** Y();
16. **Class** obj;
17. obj = b.getClass();
18. System.out.print(obj.isInstance(a));
19. }
20. }

a) 0  
b) 1  
c) true  
d) false  
View Answer

Answer: d  
Explanation: Although class Y extends class X but still a is not considered related to Y. hence isInstance() returns false.  
Output:  
$ javac Output.java  
$ java Output  
false

10. What is the output of this program?

1. **class** X
2. {
3. **int** a;
4. **double** b;
5. }
6. **class** Y **extends** X
7. {
8. **int** c;
9. }
10. **class** Output
11. {
12. **public** **static** **void** main(String args[])
13. {
14. X a = **new** X();
15. Y b = **new** Y();
16. **Class** obj;
17. obj = b.getClass();
18. System.out.print(obj.isLocalClass());
19. }
20. }

a) 0  
b) 1  
c) true  
d) false  
View Answer

Answer: d  
Explanation: None.  
Output:  
$ javac Output.java  
$ java Output  
false

Java Questions & Answers – Java.lang – ThreadGroup class & Runnable Interface

This set of Java online test focuses on “Java.lang – ThreadGroup class & Runnable Interface”.

1. Which of interface contains all the methods used for handling thread related operations in Java?  
a) Runnable interface  
b) Math interface  
c) System interface  
d) ThreadHandling interface  
View Answer

Answer: a  
Explanation: Runnable interface defines all the methods for handling thread operations in Java.

2.Which of these class is used to make a thread?

a) String  
b) System  
c) Thread  
d) Runnable  
View Answer

Answer: c  
Explanation: Thread class is used to make threads in java, Thread encapsulates a thread of execution. To create a new thread the program will either extend Thread or implement the Runnable interface.

3. Which of these interface is implemented by Thread class?  
a) Runnable  
b) Connections  
c) Set  
d) MapConnections  
View Answer

Answer: a  
Explanation: None.

4. Which of these method of Thread class is used to suspend a thread for a period of time?  
a) sleep()  
b) terminate()  
c) suspend()  
d) stop()  
View Answer

Answer: a  
Explanation: None.

5. toRadian() and toDegree() methods were added by which version of Java?  
a) Java 1.0  
b) Java 1.5  
c) Java 2.0  
d) Java 3.0  
View Answer

Answer: c  
Explanation: toRadian() and toDegree() methods were added by Java 2.0 before that there was no method which could directly convert degree into radians and vice versa.

6. What is the output of this program?

1. **class** newthread **implements** Runnable
2. {
3. Thread t1,t2;
4. newthread()
5. {
6. t1 = **new** Thread(**this**,"Thread\_1");
7. t2 = **new** Thread(**this**,"Thread\_2");
8. t1.start();
9. t2.start();
10. }
11. **public** **void** run()
12. {
13. t2.setPriority(Thread.MAX\_PRIORITY);
14. System.out.print(t1.equals(t2));
15. }
16. }
17. **class** multithreaded\_programing
18. {
19. **public** **static** **void** main(String args[])
20. {
21. **new** newthread();
22. }
23. }

a) true  
b) false  
c) truetrue  
d) falsefalse  
View Answer

Answer: d  
Explanation: Threads t1 & t2 are created by class newthread that is implementing runnable interface, hence both the threads are provided their own run() method specifying the actions to be taken. When constructor of newthread class is called first the run() method of t1 executes than the run method of t2 printing 2 times “false” as both the threads are not equal one is having different priority than other, hence falsefalse is printed.  
Output:  
$ javac multithreaded\_programing.java  
$ java multithreaded\_programing  
falsefalse

7. What is the output of this program?

1. **class** newthread **implements** Runnable
2. {
3. Thread t;
4. newthread()
5. {
6. t = **new** Thread(**this**,"New Thread");
7. t.start();
8. }
9. **public** **void** run()
10. {
11. t.setPriority(Thread.MAX\_PRIORITY);
12. System.out.println(t);
13. }
14. }
15. **class** multithreaded\_programing
16. {
17. **public** **static** **void** main(String args[])
18. {
19. **new** newthread();
20. }
21. }

a) Thread[New Thread,0,main].  
b) Thread[New Thread,1,main].  
c) Thread[New Thread,5,main].  
d) Thread[New Thread,10,main].  
View Answer

Answer: d  
Explanation: Thread t has been made with default priority value 5 but in run method the priority has been explicitly changed to MAX\_PRIORITY of class thread, that is 10 by code ‘t.setPriority(Thread.MAX\_PRIORITY);’ using the setPriority function of thread t.  
Output:  
$ javac multithreaded\_programing.java  
$ java multithreaded\_programing  
Thread[New Thread,10,main]

8. What is the output of this program?

1. **class** newthread **implements** Runnable
2. {
3. Thread t;
4. newthread()
5. {
6. t = **new** Thread(**this**,"My Thread");
7. t.start();
8. }
9. }
10. **class** multithreaded\_programing
11. {
12. **public** **static** **void** main(String args[])
13. {
14. **new** newthread();
15. }
16. }

a) My Thread  
b) Thread[My Thread,5,main].  
c) Compilation Error  
d) Runtime Error  
View Answer

Answer: c  
Explanation: Thread t has been made by using Runnable interface, hence it is necessary to use inherited abstract method run() method to specify instructions to be implemented on the thread, since no run() method is used it gives a compilation error.  
Output:  
$ javac multithreaded\_programing.java  
The type newthread must implement the inherited abstract method Runnable.run()

9. What is the output of this program?

1. **class** newthread **implements** Runnable
2. {
3. Thread t;
4. newthread()
5. {
6. t = **new** Thread(**this**,"My Thread");
7. t.start();
8. }
9. **public** **void** run()
10. {
11. System.out.println(t.getName());
12. }
13. }
14. **class** multithreaded\_programing
15. {
16. **public** **static** **void** main(String args[])
17. {
18. **new** newthread();
19. }
20. }

a) My Thread  
b) Thread[My Thread,5,main].  
c) Compilation Error  
d) Runtime Error  
View Answer

Answer: a  
Explanation: None.  
Output:  
$ javac multithreaded\_programing.java  
$ java multithreaded\_programing  
My Thread

10. What is the output of this program?

1. **class** newthread **implements** Runnable
2. {
3. Thread t;
4. newthread()
5. {
6. t = **new** Thread(**this**,"My Thread");
7. t.start();
8. }
9. **public** **void** run()
10. {
11. System.out.println(t);
12. }
13. }
14. **class** multithreaded\_programing
15. {
16. **public** **static** **void** main(String args[])
17. {
18. **new** newthread();
19. }
20. }

a) My Thread  
b) Thread[My Thread,5,main].  
c) Compilation Error  
d) Runtime Error  
View Answer

Answer: b  
Explanation: None.  
Output:  
$ javac multithreaded\_programing.java  
$ java multithreaded\_programing  
Thread[My Thread,5,main]

# Java Questions & Answers – Environment Properties

This set of Java Multiple Choice Questions & Answers (MCQs) focuses on “Environment Properties”.

1. Which object Java application uses to create a new process?  
a) Process  
b) Builder  
c) ProcessBuilder  
d) CreateBuilder  
View Answer

Answer: c  
Explanation: Java application uses ProcessBuilder object to create a new process. By default, same set of environment variables passed which are set in application’s virtual machine process.

2. Which of the following is true about Java system properties?  
a) Java system properties are accessible by any process  
b) Java system properties are accessible by processes they are added to  
c) Java system properties are retrieved by System.getenv()  
d) Java system prooerties are set by System.setenv()  
View Answer

Answer: b  
Explanation: Java system properties are only used and accessible by the processes they are added.

3. Java system properties can be set at runtime.  
a) True  
b) False  
View Answer

Answer: a  
Explanation: Java system properties can be set at runtime using System.setProperty(name, value) or using System.getProperties().load() methods.

4. Which system property stores installation directory of JRE?  
a) user.home  
b) java.class.path  
c) java.home  
d) user.dir  
View Answer

Answer: c  
Explanation: java.home is the installation directory of Java Runtime Environment.

5. What does System.getProperty(“variable”) return?  
a) compilation error  
b) value stored in variable  
c) runtime error  
d) null  
View Answer

Answer: d  
Explanation: System.getProperty(“variable”) returns null value. Because, variable is not a property and if property does not exist, this method returns null value.

6. What is true about setProperties method?  
a) setProperties method changes the set of Java Properties which are persistent  
b) Changing the system properties within an application will affect future invocations  
c) setProperties method changes the set of Java Properties which are not persistent  
d) setProperties writes the values directly into the file which stores all the properties  
View Answer

Answer: c  
Explanation: The changes made by setProperties method are not persistent. Hence, it does not affect future invocation.

7. How to use environment properties in the class?  
a) @Environment  
b) @Variable  
c) @Property  
d) @Autowired  
View Answer

Answer: d  
Explanation: @Autowired  
private Environment env;  
This is how environment variables are injected in the class where they can be used.

8. How to assign values to variable using property?  
a) @Value(“${my.property}”)  
private String prop;  
b) @Property(“${my.property}”)  
private String prop;  
c) @Environment(“${my.property}”)  
private String prop;  
d) @Env(“${my.property}”)  
private String prop;  
View Answer

Answer: a  
Explanation: @Value are used to inject the properties and assign them to variables.

9. Which environment variable is used to set java path?  
a) JAVA  
b) JAVA\_HOME  
c) CLASSPATH  
d) MAVEN\_HOME  
View Answer

Answer: b  
Explanation: JAVA\_HOME is used to store path to the java installation.

10. How to read a classpath file?  
a) InputStream in =this.getClass().getResource(“SomeTextFile.txt”);  
b) InputStream in =this.getClass().getResourceClasspath(“SomeTextFile.txt”);  
c) InputStream in =this.getClass().getResourceAsStream(“SomeTextFile.txt”);  
d) InputStream in =this.getClass().getResource(“classpath:/SomeTextFile.txt”);  
View Answer

Answer: c  
Explanation: This method can be used to load files using relative path to the package of the class.

Java Questions & Answers – Serialization – 1

This set of Java Multiple Choice Questions & Answers (MCQs) focuses on “Serialization – 1”.

1. Which of these is a process of writing the state of an object to a byte stream?  
a) Serialization  
b) Externalization  
c) File Filtering  
d) All of the mentioned  
View Answer

Answer: a  
Explanation: Serialization is the process of writing the state of an object to a byte stream. This is used when you want to save the state of your program to persistent storage area.

2. Which of these process occur automatically by java run time system?  
a) Serialization  
b) Garbage collection  
c) File Filtering  
d) All of the mentioned  
View Answer

Answer: a  
Explanation: Serialization and deserialization occur automatically by java run time system, Garbage collection also occur automatically but is done by CPU or the operating system not by the java run time system.

3. Which of these is an interface for control over serialization and deserialization?  
a) Serializable  
b) Externalization  
c) FileFilter  
d) ObjectInput  
View Answer

Answer: b  
Explanation: None.

4. Which of these interface extends DataOutput interface?  
a) Serializable  
b) Externalization  
c) ObjectOutput  
d) ObjectInput  
View Answer

Answer: c  
Explanation: ObjectOutput interface extends the DataOutput interface and supports object serialization.

5. Which of these is a method of ObjectOutput interface used to finalize the output state so that any buffers are cleared?  
a) clear()  
b) flush()  
c) fflush()  
d) close()  
View Answer

Answer: b  
Explanation: None.

6. Which of these is method of ObjectOutput interface used to write the object to input or output stream as required?  
a) write()  
b) Write()  
c) StreamWrite()  
d) writeObject()  
View Answer

Answer: d  
Explanation: writeObject() is used to write an object into invoking stream, it can be input stream or output stream.

7. What is the output of this program?

1. **import** java.io.\*;
2. **class** serialization
3. {
4. **public** **static** **void** main(String[] args)
5. {
6. **try**
7. {
8. Myclass object1 = **new** Myclass("Hello", -7, 2.1e10);
9. FileOutputStream fos = **new** FileOutputStream("serial");
10. ObjectOutputStream oos = **new** ObjectOutputStream(fos);
11. oos.writeObject(object1);
12. oos.flush();
13. oos.close();
14. }
15. **catch**(Exception e)
16. {
17. System.out.println("Serialization" + e);
18. System.exit(0);
19. }
20. **try**
21. {
22. Myclass object2;
23. FileInputStream fis = **new** FileInputStream("serial");
24. ObjectInputStream ois = **new** ObjectInputStream(fis);
25. object2 = (Myclass)ois.readObject();
26. ois.close();
27. System.out.println(object2);
28. }
29. **catch** (Exception e)
30. {
31. System.out.print("deserialization" + e);
32. System.exit(0);
33. }
34. }
35. }
36. **class** Myclass **implements** Serializable
37. {
38. String s;
39. **int** i;
40. **double** d;
41. Myclass (String s, **int** i, **double** d)
42. {
43. **this**.d = d;
44. **this**.i = i;
45. **this**.s = s;
46. }
47. }

a) s=Hello; i=-7; d=2.1E10  
b) Hello; -7; 2.1E10  
c) s; i; 2.1E10  
d) Serialization  
View Answer

Answer: a  
Explanation: None.  
Output:  
$ javac serialization.java  
$ java serialization  
s=Hello; i=-7; d=2.1E10

8. What is the output of this program?

1. **import** java.io.\*;
2. **class** serialization
3. {
4. **public** **static** **void** main(String[] args)
5. {
6. **try**
7. {
8. Myclass object1 = **new** Myclass("Hello", -7, 2.1e10);
9. FileOutputStream fos = **new** FileOutputStream("serial");
10. ObjectOutputStream oos = **new** ObjectOutputStream(fos);
11. oos.writeObject(object1);
12. oos.flush();
13. oos.close();
14. }
15. **catch**(Exception e)
16. {
17. System.out.println("Serialization" + e);
18. System.exit(0);
19. }
20. **try**
21. {
22. **int** x;
23. FileInputStream fis = **new** FileInputStream("serial");
24. ObjectInputStream ois = **new** ObjectInputStream(fis);
25. x = ois.readInt();
26. ois.close();
27. System.out.println(x);
28. }
29. **catch** (Exception e)
30. {
31. System.out.print("deserialization");
32. System.exit(0);
33. }
34. }
35. }
36. **class** Myclass **implements** Serializable
37. {
38. String s;
39. **int** i;
40. **double** d;
41. Myclass(String s, **int** i, **double** d)
42. {
43. **this**.d = d;
44. **this**.i = i;
45. **this**.s = s;
46. }
47. }

a) -7  
b) Hello  
c) 2.1E10  
d) deserialization  
View Answer

Answer: d  
Explanation: x = ois.readInt(); will try to read an integer value from the stream ‘serial’ created before, since stream contains an object of Myclass hence error will occur and it will be catched by catch printing deserialization.  
Output:  
$ javac serialization.java  
$ java serialization  
deserialization

9. What is the output of this program?

1. **import** java.io.\*;
2. **class** Chararrayinput
3. {
4. **public** **static** **void** main(String[] args)
5. {
6. String obj = "abcdefgh";
7. **int** length = obj.length();
8. **char** c[] = **new** **char**[length];
9. obj.getChars(0, length, c, 0);
10. CharArrayReader input1 = **new** CharArrayReader(c);
11. CharArrayReader input2 = **new** CharArrayReader(c, 1, 4);
12. **int** i;
13. **int** j;
14. **try**
15. {
16. **while** ((i = input1.read()) == (j = input2.read()))
17. {
18. System.out.print((**char**)i);
19. }
20. }
21. **catch** (IOException e)
22. {
23. e.printStackTrace();
24. }
25. }
26. }

a) abc  
b) abcd  
c) abcde  
d) None of the mentioned  
View Answer

Answer: d  
Explanation: No output is printed. CharArrayReader object input1 contains string “abcdefgh” whereas object input2 contains string “bcde”, when while((i=input1.read())==(j=input2.read())) is executed the starting character of each object is compared since they are unequal control comes out of loop and nothing is printed on the screen.  
Output:  
$ javac Chararrayinput.java  
$ java Chararrayinput

10. What is the output of this program?

1. **import** java.io.\*;
2. **class** streams
3. {
4. **public** **static** **void** main(String[] args)
5. {
6. **try**
7. {
8. FileOutputStream fos = **new** FileOutputStream("serial");
9. ObjectOutputStream oos = **new** ObjectOutputStream(fos);
10. oos.writeFloat(3.5);
11. oos.flush();
12. oos.close();
13. }
14. **catch**(Exception e)
15. {
16. System.out.println("Serialization" + e);
17. System.exit(0);
18. }
19. **try**
20. {
21. **float** x;
22. FileInputStream fis = **new** FileInputStream("serial");
23. ObjectInputStream ois = **new** ObjectInputStream(fis);
24. x = ois.readInt();
25. ois.close();
26. System.out.println(x);
27. }
28. **catch** (Exception e)
29. {
30. System.out.print("deserialization");
31. System.exit(0);
32. }
33. }
34. }

a) 3  
b) 3.5  
c) serialization  
d) deserialization  
View Answer

Answer: b  
Explanation: oos.writeFloat(3.5); writes in output stream which is extracted by x = ois.readInt(); and stored in x hence x contains 3.5.  
Output:  
$ javac streams.java  
$ java streams  
3.5

# Java Questions & Answers – Serialization – 2

This set of Java Multiple Choice Questions & Answers (MCQs) focuses on “Serialization – 2”.

1. How an object can become serializable?  
a) If class implements java.io.Serializable class  
b) If class or any superclass implements java.io.Serializable interface  
c) Any object is serializable  
d) No object is serializable  
View Answer

Answer: b  
Explanation: A Java object is serializable if class or any its superclass implements java.io.Serializable or its subinterface java.io.Externalizable.

2. What is serialization?  
a) Turning object in memory into stream of bytes  
b) Turning stream of bytes into an object in memory  
c) Turning object in memory into stream of bits  
d) Turning stream of bits into an object in memory  
View Answer

Answer: a  
Explanation: Serialization in Java is the process of turning object in memory into stream of bytes.

3. What is deserialization?  
a) Turning object in memory into stream of bytes  
b) Turning stream of bytes into an object in memory  
c) Turning object in memory into stream of bits  
d) Turning stream of bits into an object in memory  
View Answer

4. How many methods Serializable has?  
a) 1  
b) 2  
c) 3  
d) 0  
View Answer

Answer: d  
Explanation: Serializable interface does not have any method. It is also called as marker interface.

5. What type of members are not serialized?  
a) Private  
b) Protected  
c) Static  
d) Throwable  
View Answer

Answer: c  
Explanation: All static and transient variables are not serialized.

6. If member does not implement serialization, which exception would be thrown?  
a) RuntimeException  
b) SerializableException  
c) NotSerializableException  
d) UnSerializedException  
View Answer

Answer: c  
Explanation: If member of a class does not implement serialization, NotSerializationException will be thrown.

7. Default Serialization process cannot be overridden.  
a) True  
b) False  
View Answer

Answer: b  
Explanation: Default serialization process can be overridden.

8. Which of the following methods is used to avoid serialization of new class whose super class already implements Serialization?  
a) writeObject()  
b) readWriteObject()  
c) writeReadObject()  
d) unSerializaedObject()  
View Answer

Answer: a  
Explanation: writeObject() and readObject() methods should be implemented to avoid Java serialization.

9. Which of the following methods is not used while Serialization and DeSerialization?  
a) readObject()  
b) readExternal()  
c) readWriteObject()  
d) writeObject()  
View Answer

Answer: c  
Explanation: Using readObject(), writeObject(), readExternal() and writeExternal() methods Serialization and DeSerialization are implemented.

10. Serializaed object can be transfered via network.  
a) True  
b) False  
View Answer

Answer: a  
Explanation: Serialized object can be transfered via network because Java serialized object remains in form of bytes which can be transmitted over network.

Java Questions & Answers – Serialization & Deserialization

This set of Java Multiple Choice Questions & Answers (MCQs) focuses on “Serialization & Deserialization”.

1. Which of these is a process of extracting/removing the state of an object from a stream?  
a) Serialization  
b) Externalization  
c) File Filtering  
d) Deserialization  
View Answer

Answer: d  
Explanation: Deserialization is a process by which the data written in the stream can be extracted out from the stream.

2. Which of these process occur automatically by java run time system?  
a) Serialization  
b) Memory allocation  
c) Deserialization  
d) All of the mentioned  
View Answer

Answer: d  
Explanation: Serialization, deserialization and Memory allocation occur automatically by java run time system.

3. Which of these is an interface for control over serialization and deserialization?  
a) Serializable  
b) Externalization  
c) FileFilter  
d) ObjectInput  
View Answer

Answer: b  
Explanation: None.

4. Which of these interface extends DataInput interface?  
a) Serializable  
b) Externalization  
c) ObjectOutput  
d) ObjectInput  
View Answer

Answer: d  
Explanation: ObjectInput interface extends the DataInput interface and supports object serialization.

5. Which of these is a method of ObjectInput interface used to deserialize an object from a stream?  
a) int read()  
b) void close()  
c) Object readObject()  
d) Object WriteObject()  
View Answer

Answer: c  
Explanation: None.

6. Which of these class extend InputStream class?  
a) ObjectStream  
b) ObjectInputStream  
c) ObjectOutput  
d) ObjectInput  
View Answer

Answer: b  
Explanation: ObjectInputStream class extends the InputStream class and implements the ObjectInput interface.

7. What is the output of this program?

1. **import** java.io.\*;
2. **class** streams
3. {
4. **public** **static** **void** main(String[] args)
5. {
6. **try**
7. {
8. FileOutputStream fos = **new** FileOutputStream("serial");
9. ObjectOutputStream oos = **new** ObjectOutputStream(fos);
10. oos.writeInt(5);
11. oos.flush();
12. oos.close();
13. }
14. **catch**(Exception e)
15. {
16. System.out.println("Serialization" + e);
17. System.exit(0);
18. }
19. **try**
20. {
21. **int** z;
22. FileInputStream fis = **new** FileInputStream("serial");
23. ObjectInputStream ois = **new** ObjectInputStream(fis);
24. z = ois.readInt();
25. ois.close();
26. System.out.println(x);
27. }
28. **catch** (Exception e)
29. {
30. System.out.print("deserialization");
31. System.exit(0);
32. }
33. }
34. }

a) 5  
b) void  
c) serialization  
d) deserialization  
View Answer

Answer: a  
Explanation: oos.writeInt(5); writes integer 5 in the Output stream which is extracted by z = ois.readInt(); and stored in z hence z contains 5.  
Output:  
$ javac streams.java  
$ java streams  
5

8. What is the output of this program?

1. **import** java.io.\*;
2. **class** serialization
3. {
4. **public** **static** **void** main(String[] args)
5. {
6. **try**
7. {
8. Myclass object1 = **new** Myclass("Hello", -7, 2.1e10);
9. FileOutputStream fos = **new** FileOutputStream("serial");
10. ObjectOutputStream oos = **new** ObjectOutputStream(fos);
11. oos.writeObject(object1);
12. oos.flush();
13. oos.close();
14. }
15. **catch**(Exception e)
16. {
17. System.out.println("Serialization" + e);
18. System.exit(0);
19. }
20. **try**
21. {
22. **int** x;
23. FileInputStream fis = **new** FileInputStream("serial");
24. ObjectInputStream ois = **new** ObjectInputStream(fis);
25. x = ois.readInt();
26. ois.close();
27. System.out.println(x);
28. }
29. **catch** (Exception e)
30. {
31. System.out.print("deserialization");
32. System.exit(0);
33. }
34. }
35. }
36. **class** Myclass **implements** Serializable
37. {
38. String s;
39. **int** i;
40. **double** d;
41. Myclass(String s, **int** i, **double** d)
42. {
43. **this**.d = d;
44. **this**.i = i;
45. **this**.s = s;
46. }
47. }

a) -7  
b) Hello  
c) 2.1E10  
d) deserialization  
View Answer

Answer: d  
Explanation: x = ois.readInt(); will try to read an integer value from the stream ‘serial’ created before, since stream contains an object of Myclass hence error will occur and it will be catched by catch printing deserialization.  
Output:  
$ javac serialization.java  
$ java serialization  
deserialization

9. What is the output of this program?

1. **import** java.io.\*;
2. **class** streams
3. {
4. **public** **static** **void** main(String[] args)
5. {
6. **try**
7. {
8. FileOutputStream fos = **new** FileOutputStream("serial");
9. ObjectOutputStream oos = **new** ObjectOutputStream(fos);
10. oos.writeFloat(3.5);
11. oos.flush();
12. oos.close();
13. }
14. **catch**(Exception e)
15. {
16. System.out.println("Serialization" + e);
17. System.exit(0);
18. }
19. **try**
20. {
21. FileInputStream fis = **new** FileInputStream("serial");
22. ObjectInputStream ois = **new** ObjectInputStream(fis);
23. ois.close();
24. System.out.println(ois.available());
25. }
26. **catch** (Exception e)
27. {
28. System.out.print("deserialization");
29. System.exit(0);
30. }
31. }
32. }

a) 1  
b) 2  
c) 3  
d) 0  
View Answer

Answer: d  
Explanation: New input stream is linked to streal ‘serials’, an object ‘ois’ of ObjectInputStream is used to access this newly created stream, ois.close(); closes the stream hence we can’t access the stream and ois.available() returns 0.  
Output:  
$ javac streams.java  
$ java streams  
0

10. What is the output of this program?

1. **import** java.io.\*;
2. **class** streams
3. {
4. **public** **static** **void** main(String[] args)
5. {
6. **try**
7. {
8. FileOutputStream fos = **new** FileOutputStream("serial");
9. ObjectOutputStream oos = **new** ObjectOutputStream(fos);
10. oos.writeFloat(3.5);
11. oos.flush();
12. oos.close();
13. }
14. **catch**(Exception e)
15. {
16. System.out.println("Serialization" + e);
17. System.exit(0);
18. }
19. **try**
20. {
21. FileInputStream fis = **new** FileInputStream("serial");
22. ObjectInputStream ois = **new** ObjectInputStream(fis);
23. System.out.println(ois.available());
24. }
25. **catch** (Exception e)
26. {
27. System.out.print("deserialization");
28. System.exit(0);
29. }
30. }
31. }

a) 1  
b) 2  
c) 3  
d) 4  
View Answer

Answer: d  
Explanation: oos.writeFloat(3.5); writes 3.5 in output stream. A new input stream is linked to stream ‘serials’, an object ‘ois’ of ObjectInputStream is used to access this newly created stream, ois.available() gives the total number of byte in the input stream since a float was written in the stream thus the stream contains 4 byte, hence 4 is returned and printed.  
Output:  
$ javac streams.java  
$ java streams  
4

11. What will be printed to the output and written to the file, in the below program?

1. **import** java.io.FileOutputStream;
2. **public** **class** FileOutputStreamExample
3. {
4. **public** **static** **void** main(String args[])
5. {
6. **try**
7. {
8. FileOutputStream fout=**new** FileOutputStream("D:**\\**sanfoundry.txt");
9. String s="Welcome to Sanfoundry.";
10. **byte** b[]=s.getBytes();*//converting string into byte array*
11. fout.write(b);
12. fout.close();
13. System.out.println("Success");
14. } **catch**(Exception e){System.out.println(e);}
15. }
16. }

a) “Success” to the output and “Welcome to Sanfoundry” to the file  
b) only “Welcome to Sanfoundry” to the file  
c) compile time error  
d) No Output  
View Answer

Answer: a  
Explanation: First, it will print “Success” and besides that it will write “Welcome to Sanfoundry” to the file sanfoundry.txt.

Java Questions & Answers – Networking Basics

This section of our 1000+ Java MCQs focuses on networking basics of Java Programming Language.

1. Which of these package contains classes and interfaces for networking?  
a) java.io  
b) java.util  
c) java.net  
d) java.network  
View Answer

Answer: c  
Explanation: None.

2. Which of these is a protocol for breaking and sending packets to an address across a network?  
a) TCIP/IP  
b) DNS  
c) Socket  
d) Proxy Server  
View Answer

Answer: a  
Explanation: TCP/IP – Transfer control protocol/Internet Protocol is used to break data into small packets an send them to an address across a network.

3. How many ports of TCP/IP are reserved for specific protocols?  
a) 10  
b) 1024  
c) 2048  
d) 512  
View Answer

Answer: b  
Explanation: None.

4. How many bits are in a single IP address?  
a) 8  
b) 16  
c) 32  
d) 64  
View Answer

Answer: c  
Explanation: None.

5. Which of these is a full form of DNS?  
a) Data Network Service  
b) Data Name Service  
c) Domain Network Service  
d) Domain Name Service  
View Answer

Answer: d  
Explanation: None.

6. Which of these class is used to encapsulate IP address and DNS?  
a) DatagramPacket  
b) URL  
c) InetAddress  
d) ContentHandler  
View Answer

Answer: c  
Explanation: InetAddress class encapsulate both IP address and DNS, we can interact with this class by using name of an IP host.

7. What is the output of this program?

1. **import** java.net.\*;
2. **class** networking
3. {
4. **public** **static** **void** main(String[] args) **throws** UnknownHostException
5. {
6. InetAddress obj1 = InetAddress.getByName("sanfoundary.com");
7. InetAddress obj2 = InetAddress.getByName("sanfoundary.com");
8. **boolean** x = obj1.equals(obj2);
9. System.out.print(x);
10. }
11. }

a) 0  
b) 1  
c) true  
d) false  
View Answer

Answer: c  
Explanation: None.  
Output:  
$ javac networking.java  
$ java networking  
true

8. What is the output of this program?

1. **import** java.net.\*;
2. **class** networking
3. {
4. **public** **static** **void** main(String[] args) **throws** UnknownHostException
5. {
6. InetAddress obj1 = InetAddress.getByName("cisco.com");
7. InetAddress obj2 = InetAddress.getByName("sanfoundary.com");
8. **boolean** x = obj1.equals(obj2);
9. System.out.print(x);
10. }
11. }

a) 0  
b) 1  
c) true  
d) false  
View Answer

Answer: d  
Explanation: InetAddress obj1 = InetAddress.getByName(“cisco.com”); creates object obj1 having DNS and IP address of cisco.com, InetAddress obj2 = InetAddress.getByName(“sanfoundry.com”); creates obj2 having DNS and IP address of sanfoundry.com , since both these address point to two different locations false is returned by obj1.equals(obj2);.  
Output:  
$ javac networking.java  
$ java networking  
true

9. What is the output of this program?

1. **import** java.io.\*;
2. **import** java.net.\*;
3. **public** **class** URLDemo
4. {
5. **public** **static** **void** main(String[] args)
6. {
7. **try**
8. {
9. URL url=**new** URL("https://www.sanfoundry.com/java-mcq");
10. System.out.println("Protocol: "+url.getProtocol());
11. System.out.println("Host Name: "+url.getHost());
12. System.out.println("Port Number: "+url.getPort());
13. } **catch**(Exception e){System.out.println(e);}
14. }
15. }

a) Protocol: http  
b) Host Name: www.sanfoundry.com  
c) Port Number: -1  
d) all above mentioned  
View Answer

Answer: d  
Explanation: getProtocol() give protocol which is http  
getUrl() give name domain name  
getPort() Since we have not explicitly set the port, default value that is -1 is printed.

10. What is the output of this program?

1. **import** java.net.\*;
2. **class** networking
3. {
4. **public** **static** **void** main(String[] args) **throws** UnknownHostException
5. {
6. InetAddress obj1 = InetAddress.getByName("cisco.com");
7. System.out.print(obj1.getHostName());
8. }
9. }

a) cisco  
b) cisco.com  
c) www.cisco.com  
d) none of the mentioned  
View Answer

Answer: b  
Explanation: None.  
Output:  
$ javac networking.java  
$ java networking  
cisco.com

Java Questions & Answers – Networking – Server, Sockets & httpd Class

This set of Basic Java Questions and Answers focuses on “Nnetworking – Server, Sockets & httpd Class”.

1. Which of these interface abstractes the output of messages from httpd?  
a) LogMessage  
b) LogResponse  
c) Httpdserver  
d) httpdResponse  
View Answer

Answer: a  
Explanation: LogMessage is a simple interface that is used to abstract the output of messages from the httpd.

2. Which of these class is used to create servers that listen for either local or remote client programs?  
a) httpServer  
b) ServerSockets  
c) MimeHeader  
d) HttpResponse  
View Answer

Answer: b  
Explanation: None.

3. Which of these is a standard for communicating multimedia content over email?  
a) http  
b) https  
c) Mime  
d) httpd  
View Answer

Answer: c  
Explanation: MIME is an internet standard for communicating multimedia content over email. The HTTP protocol uses and extends the notion of MIME headers to pass attribute pairs between HTTP client and server.

4. Which of these methods is used to make raw MIME formatted string?  
a) parse()  
b) toString()  
c) getString()  
d) parseString()  
View Answer

Answer: a  
Explanation: None.

5. Which of these class is used for operating on request from the client to the server?  
a) http  
b) httpDecoder  
c) httpConnection  
d) httpd  
View Answer

Answer: d  
Explanation: None.

6. Which of these method of MimeHeader is used to return the string equivalent of the values stores on MimeHeader?  
a) string()  
b) toString()  
c) convertString()  
d) getString()  
View Answer

Answer:b  
Explanation: toString() does the reverse of parse() method, it is used to return the string equivalent of the values stores on MimeHeader.

7. What is the output of this program?

1. **import** java.net.\*;
2. **class** networking
3. {
4. **public** **static** **void** main(String[] args) **throws** Exception
5. {
6. URL obj = **new** URL("https://www.sanfoundry.com/javamcq");
7. URLConnection obj1 = obj.openConnection();
8. System.out.print(obj1.getContentType());
9. }
10. }

Note: Host URL is written in html and simple text.  
a) html  
b) text  
c) html/text  
d) text/html  
View Answer

Answer: d  
Explanation: None.  
Output:  
$ javac networking.java  
$ java networking  
text/html

8. Which of these is an instance variable of class httpd?  
a) port  
b) cache  
c) log  
d) All of the mentioned  
View Answer

Answer: d  
Explanation: There are 5 instance variables : port, docRoot, log, cache and stopFlag. All of them are private.

9. What is the output of this program?

1. **import** java.net.\*;
2. **class** networking
3. {
4. **public** **static** **void** main(String[] args) **throws** MalformedURLException
5. {
6. URL obj = **new** URL("https://www.sanfoundry.com/javamcq");
7. System.out.print(obj.toExternalForm());
8. }
9. }

a) sanfoundry  
b) sanfoundry.com  
c) www.sanfoundry.com  
d) https://www.sanfoundry.com/javamcq  
View Answer

Answer: d  
Explanation: toExternalForm() is used to know the full URL of an URL object.  
Output:  
$ javac networking.java  
$ java networking  
https://www.sanfoundry.com/javamcq

Java Questions & Answers – Networking – httpd.java Class

This section of our 1000+ Java MCQs focuses on httpd.java of Java Programming Language.

1. Which of these methods of httpd class is used to read data from the stream?  
a) getDta()  
b) GetResponse()  
c) getStream()  
d) getRawRequest()  
View Answer

Answer: d  
Explanation: The getRawRequest() method reads data from a stream until it gets two consecutive newline characters.

2. Which of these method of httpd class is used to get report on each hit to HTTP server?  
a) log()  
b) logEntry()  
c) logHttpd()  
d) logResponse()  
View Answer

Answer: b  
Explanation: None.

3. Which of these method is used to find a URL from the cache of httpd?  
a) findfromCache()  
b) findFromCache()  
c) serveFromCache()  
d) getFromCache()  
View Answer

Answer: c  
Explanation: serveFromCatche() is a boolean method that attempts to find a particular URL in the cache. If it is successful then the content of that cache entry are written to the client, otherwise it returns false.

4. Which of these variables stores the number of hits that are successfully served out of cache?  
a) hits  
b) hitstocache  
c) hits\_to\_cache  
d) hits.to.cache  
View Answer

Answer: d  
Explanation: None.

5. Which of these class is used for operating on request from the client to the server?  
a) http  
b) httpDecoder  
c) httpConnection  
d) httpd  
View Answer

Answer: d  
Explanation: None.

6. Which of these method of httpd class is used to write UrlCacheEntry object into local disk?  
a) writeDiskCache()  
b) writetoDisk()  
c) writeCache()  
d) writeDiskEntry()  
View Answer

Answer: a  
Explanation: The writeDiskCache() method takes an UrlCacheEntry object and writes it persistently into the local disk. It constructs directory names out of URL, making sure to replace the slash(/) characters with system dependent seperatorChar.

7. What is the output of this program?

1. **import** java.net.\*;
2. **class** networking
3. {
4. **public** **static** **void** main(String[] args) **throws** Exception
5. {
6. URL obj = **new** URL("https://www.sanfoundry.com/javamcq");
7. URLConnection obj1 = obj.openConnection();
8. **int** len = obj1.getContentLength();
9. System.out.print(len);
10. }
11. }

Note: Host URL is having length of content 127.  
a) 126  
b) 127  
c) Compilation Error  
d) Runtime Error  
View Answer

Answer: b  
Explanation: None.  
Output:  
$ javac networking.java  
$ java networking  
127

8. Which of these method is used to start a server thread?  
a) run()  
b) start()  
c) runThread()  
d) startThread()  
View Answer

Answer: a  
Explanation: run() method is caleed when the server thread is started.

9. Which of these method is called when http daemon is acting like a normal web server?  
a) Handle()  
b) HandleGet()  
c) handleGet()  
d) Handleget()  
View Answer

Answer: c  
Explanation: None.

10. What is the output of this program?

1. **import** java.net.\*;
2. **class** networking
3. {
4. **public** **static** **void** main(String[] args) **throws** UnknownHostException
5. {
6. InetAddress obj1 = InetAddress.getByName("cisco.com");
7. System.out.print(obj1.getHostName());
8. }
9. }

a) cisco  
b) cisco.com  
c) www.cisco.com  
d) none of the mentioned  
View Answer

Answer: b  
Explanation: None.  
Output:  
$ javac networking.java  
$ java networking  
cisco.com

# Advanced Java Questions & Answers – JDBC

This set of Advanced Java Multiple Choice Questions & Answers (MCQs) focuses on “JDBC”.

1. Which of the following contains both date and time?  
a) java.io.date  
b) java.sql.date  
c) java.util.date  
d) java.util.dateTime  
View Answer

Answer: d  
Explanation: java.util.date contains both date and time. Whereas, java.sql.date contains only date.

2. Which of the following is advantage of using JDBC connection pool?  
a) Slow performance  
b) Using more memory  
c) Using less memory  
d) Better performance  
View Answer

Answer: d  
Explanation: Since the JDBC connection takes time to establish. Creating connection at the application start-up and reusing at the time of requirement, helps performance of the application.

3. Which of the following is advantage of using PreparedStatement in Java?  
a) Slow performance  
b) Encourages SQL injection  
c) Prevents SQL injection  
d) More memory usage  
View Answer

Answer: c  
Explanation: PreparedStatement in Java improves performance and also prevents from SQL injection.

4. Which one of the following contains date information?  
a) java.sql.TimeStamp  
b) java.sql.Time  
c) java.io.Time  
d) java.io.TimeStamp  
View Answer

Answer: a  
Explanation: java.sql.Time contains only time. Whereas, java.sql.TimeStamp contains both time and date.

5. What does setAutoCommit(false) do?  
a) commits transaction after each query  
b) explicitly commits transaction  
c) does not commit transaction automatically after each query  
d) never commits transaction  
View Answer

Answer: c  
Explanation: setAutoCommit(false) does not commit transaction automatically after each query. That saves lot of time of the execution and hence improves performance.

6. Which of the following is used to call stored procedure?  
a) Statement  
b) PreparedStatement  
c) CallableStatment  
d) CalledStatement  
View Answer

Answer: c  
Explanation: CallableStatement is used in JDBC to call stored procedure from Java program.

7. Which of the following is used to limit the number of rows returned?  
a) setMaxRows(int i)  
b) setMinRows(int i)  
c) getMaxrows(int i)  
d) getMinRows(int i)  
View Answer

Answer: a  
Explanation: setMaxRows(int i) method is used to limit the number of rows that the database returns from the query.

8. Which of the following is method of JDBC batch process?  
a) setBatch()  
b) deleteBatch()  
c) removeBatch()  
d) addBatch()  
View Answer

Answer: d  
Explanation: addBatch() is a method of JDBC batch process. It is faster in processing than executing one statement at a time.

9. Which of the following is used to rollback a JDBC transaction?  
a) rollback()  
b) rollforward()  
c) deleteTransaction()  
d) RemoveTransaction()  
View Answer

Answer: a  
Explanation: rollback() method is used to rollback the transaction. It will rollback all the changes made by the transaction.

10. Which of the following is not a JDBC connection isolation levels?  
a) TRANSACTION\_NONE  
b) TRANSACTION\_READ\_COMMITTED  
c) TRANSACTION\_REPEATABLE\_READ  
d) TRANSACTION\_NONREPEATABLE\_READ  
View Answer

Answer: d  
Explanation: TRANSACTION\_NONREPEATABLE\_READ is not a JDBC connection isolation level.

# Advanced Java Questions & Answers – Design Patterns

This set of Advanced Java Multiple Choice Questions & Answers (MCQs) focuses on “Design Patterns”.

1. Which of the below is not a valid design pattern?  
a) Singleton  
b) Factory  
c) Command  
d) Java  
View Answer

Answer: d  
Explanation: Design pattern is a general repeatable solution to a commonly occurring problem in software design. There are various patterns available for use in day to day coding problems.

2. Which of the below author is not a part of GOF (Gang of Four)?  
a) Erich Gamma  
b) Gang Pattern  
c) Richard Helm  
d) Ralph Johnson  
View Answer

Answer: b  
Explanation: Four authors named Richard Helm, Erich Gamma, Ralph Johnson and John Vlissides published a book on design patterns. This book initiated the concept of Design Pattern in Software development. They are known as Gang of Four (GOF).

3. Which of the below is not a valid classification of design pattern?  
a) Creational patterns  
b) Structural patterns  
c) Behavioural patterns  
d) Java patterns  
View Answer

Answer: d  
Explanation: Java patterns is not a valid classification of design patterns. The correct one is J2EE patterns.

4. Which design pattern provides a single class which provides simplified methods required by client and delagates call to those methods?  
a) Adapter pattern  
b) Builder pattern  
c) Facade pattern  
d) Prototype pattern  
View Answer

Answer: c  
Explanation: Facade pattern hides the complexities of the system and provides an interface to the client using which client can access the system.

5. Which design pattern ensures that only one object of particular class gets created?  
a) Singleton pattern  
b) Filter pattern  
c) State pattern  
d) Bridge pattern  
View Answer

Answer: a  
Explanation: Singleton pattern involves a single class which is responsible to create an object while making sure that only one object gets created. This class provides a way to access the only object which can be accessed directly without need to instantiate another object of the same class.

6. Which design pattern suggest multiple classes through which request is passed and multiple but only relevant classes carry out operations on the request?  
a) Singleton pattern  
b) Chain of responsibility pattern  
c) State pattern  
d) Bridge pattern  
View Answer

Answer: b  
Explanation: Chain of responsibility pattern creates a chain of receiver objects for a particular request. The sender and receiver of a request are decoupled based on the type of request. This pattern is one of the behavioral patterns.

7. Which design pattern represents a way to access all the objects in a collection?  
a) Iterator pattern  
b) Facade pattern  
c) Builder pattern  
d) Bridge pattern  
View Answer

Answer: a  
Explanation: Iterator pattern represents a way to access the elements of a collection object in sequential manner without the need to know its underlying representation.

8. What does MVC pattern stands for ?  
a) Mock View Control  
b) Model view Controller  
c) Mock View Class  
d) Model View Class  
View Answer

Answer: b  
Explanation: Model represents an object or JAVA POJO carrying data.View represents the visualization of the data that model contains.Controller acts on both model and view.It is usually used in web development.

9. Is design pattern a logical concept?  
a) True  
b) False  
View Answer

Answer: a  
Explanation: Design pattern is a logical concept. Various classes and frameworks are provided to enable users to implement these design patterns.

10. Which design pattern works on data and action taken based on data provided?  
a) Command pattern  
b) Singleton pattern  
c) MVC pattern  
d) Facade pattern  
View Answer

Answer: a  
Explanation: Command pattern is a data driven design pattern. It is a behavioral pattern. A request is wrapped under an object as command and passed to the invoker object.The invoker object looks for the appropriate object which can handle this command and passes this command to the corresponding object which executes the comm

# Advanced Java Questions & Answers – JDBC

This set of Advanced Java Multiple Choice Questions & Answers (MCQs) focuses on “JDBC”.

1. Which of the following contains both date and time?  
a) java.io.date  
b) java.sql.date  
c) java.util.date  
d) java.util.dateTime  
View Answer

Answer: d  
Explanation: java.util.date contains both date and time. Whereas, java.sql.date contains only date.

2. Which of the following is advantage of using JDBC connection pool?  
a) Slow performance  
b) Using more memory  
c) Using less memory  
d) Better performance  
View Answer

Answer: d  
Explanation: Since the JDBC connection takes time to establish. Creating connection at the application start-up and reusing at the time of requirement, helps performance of the application.

3. Which of the following is advantage of using PreparedStatement in Java?  
a) Slow performance  
b) Encourages SQL injection  
c) Prevents SQL injection  
d) More memory usage  
View Answer

Answer: c  
Explanation: PreparedStatement in Java improves performance and also prevents from SQL injection.

4. Which one of the following contains date information?  
a) java.sql.TimeStamp  
b) java.sql.Time  
c) java.io.Time  
d) java.io.TimeStamp  
View Answer

Answer: a  
Explanation: java.sql.Time contains only time. Whereas, java.sql.TimeStamp contains both time and date.

5. What does setAutoCommit(false) do?  
a) commits transaction after each query  
b) explicitly commits transaction  
c) does not commit transaction automatically after each query  
d) never commits transaction  
View Answer

Answer: c  
Explanation: setAutoCommit(false) does not commit transaction automatically after each query. That saves lot of time of the execution and hence improves performance.

6. Which of the following is used to call stored procedure?  
a) Statement  
b) PreparedStatement  
c) CallableStatment  
d) CalledStatement  
View Answer

Answer: c  
Explanation: CallableStatement is used in JDBC to call stored procedure from Java program.

7. Which of the following is used to limit the number of rows returned?  
a) setMaxRows(int i)  
b) setMinRows(int i)  
c) getMaxrows(int i)  
d) getMinRows(int i)  
View Answer

Answer: a  
Explanation: setMaxRows(int i) method is used to limit the number of rows that the database returns from the query.

8. Which of the following is method of JDBC batch process?  
a) setBatch()  
b) deleteBatch()  
c) removeBatch()  
d) addBatch()  
View Answer

Answer: d  
Explanation: addBatch() is a method of JDBC batch process. It is faster in processing than executing one statement at a time.

9. Which of the following is used to rollback a JDBC transaction?  
a) rollback()  
b) rollforward()  
c) deleteTransaction()  
d) RemoveTransaction()  
View Answer

Answer: a  
Explanation: rollback() method is used to rollback the transaction. It will rollback all the changes made by the transaction.

10. Which of the following is not a JDBC connection isolation levels?  
a) TRANSACTION\_NONE  
b) TRANSACTION\_READ\_COMMITTED  
c) TRANSACTION\_REPEATABLE\_READ  
d) TRANSACTION\_NONREPEATABLE\_READ  
View Answer

Answer: d  
Explanation: TRANSACTION\_NONREPEATABLE\_READ is not a JDBC connection isolation level.

# Advanced Java Questions & Answers – Design Patterns

This set of Advanced Java Multiple Choice Questions & Answers (MCQs) focuses on “Design Patterns”.

1. Which of the below is not a valid design pattern?  
a) Singleton  
b) Factory  
c) Command  
d) Java  
View Answer

Answer: d  
Explanation: Design pattern is a general repeatable solution to a commonly occurring problem in software design. There are various patterns available for use in day to day coding problems.

2. Which of the below author is not a part of GOF (Gang of Four)?  
a) Erich Gamma  
b) Gang Pattern  
c) Richard Helm  
d) Ralph Johnson  
View Answer

Answer: b  
Explanation: Four authors named Richard Helm, Erich Gamma, Ralph Johnson and John Vlissides published a book on design patterns. This book initiated the concept of Design Pattern in Software development. They are known as Gang of Four (GOF).

3. Which of the below is not a valid classification of design pattern?  
a) Creational patterns  
b) Structural patterns  
c) Behavioural patterns  
d) Java patterns  
View Answer

Answer: d  
Explanation: Java patterns is not a valid classification of design patterns. The correct one is J2EE patterns.

4. Which design pattern provides a single class which provides simplified methods required by client and delagates call to those methods?  
a) Adapter pattern  
b) Builder pattern  
c) Facade pattern  
d) Prototype pattern  
View Answer

Answer: c  
Explanation: Facade pattern hides the complexities of the system and provides an interface to the client using which client can access the system.

5. Which design pattern ensures that only one object of particular class gets created?  
a) Singleton pattern  
b) Filter pattern  
c) State pattern  
d) Bridge pattern  
View Answer

Answer: a  
Explanation: Singleton pattern involves a single class which is responsible to create an object while making sure that only one object gets created. This class provides a way to access the only object which can be accessed directly without need to instantiate another object of the same class.

6. Which design pattern suggest multiple classes through which request is passed and multiple but only relevant classes carry out operations on the request?  
a) Singleton pattern  
b) Chain of responsibility pattern  
c) State pattern  
d) Bridge pattern  
View Answer

Answer: b  
Explanation: Chain of responsibility pattern creates a chain of receiver objects for a particular request. The sender and receiver of a request are decoupled based on the type of request. This pattern is one of the behavioral patterns.

7. Which design pattern represents a way to access all the objects in a collection?  
a) Iterator pattern  
b) Facade pattern  
c) Builder pattern  
d) Bridge pattern  
View Answer

Answer: a  
Explanation: Iterator pattern represents a way to access the elements of a collection object in sequential manner without the need to know its underlying representation.

8. What does MVC pattern stands for ?  
a) Mock View Control  
b) Model view Controller  
c) Mock View Class  
d) Model View Class  
View Answer

Answer: b  
Explanation: Model represents an object or JAVA POJO carrying data.View represents the visualization of the data that model contains.Controller acts on both model and view.It is usually used in web development.

9. Is design pattern a logical concept?  
a) True  
b) False  
View Answer

Answer: a  
Explanation: Design pattern is a logical concept. Various classes and frameworks are provided to enable users to implement these design patterns.

10. Which design pattern works on data and action taken based on data provided?  
a) Command pattern  
b) Singleton pattern  
c) MVC pattern  
d) Facade pattern  
View Answer

Answer: a  
Explanation: Command pattern is a data driven design pattern. It is a behavioral pattern. A request is wrapped under an object as command and passed to the invoker object.The invoker object looks for the appropriate object which can handle this command and passes this command to the corresponding object w

# Advanced Java Questions & Answers – Debugging in Eclipse

This set of Advanced Java Multiple Choice Questions & Answers (MCQs) focuses on “Debugging in Eclipse”.

1. Which mode allows us to run program interactively while watching source code and variables during execution?  
a) safe mode  
b) debug mode  
c) successfully run mode  
d) exception mode  
View Answer

2. How can we move from one desired step to another step?  
a) breakpoints  
b) System.out.println  
c) logger.log  
d) logger.error  
View Answer

3. Which part stores the program arguments and startup parameters?  
a) debug configuration  
b) run configuration  
c) launch configuration  
d) project configuration  
View Answer

4. How to deep dive into the execution of a method from a method call?  
a) F3  
b) F5  
c) F7  
d) F8  
View Answer

5. Which key helps to step out of the caller of currently executed method?  
a) F3  
b) F5  
c) F7  
d) F8  
View Answer

6. Which view allows us to delete and deactivate breakpoints and watchpoints?  
a) breakpoint view  
b) variable view  
c) debug view  
d) logger view  
View Answer

7. What is debugging an application which runs on another java virtual machine on another machine?  
a) virtual debugging  
b) remote debugging  
c) machine debugging  
d) compiling debugging  
View Answer

8. What happens when the value of variable change?  
a) changed value pop on the screen  
b) variable changes are printed in logs  
c) dump of variable changes are printed on screen on end of execution  
d) variable tab shows variables highlighted when values change  
View Answer

9. Which perspective is used to run a program in debug view?  
a) java perspective  
b) eclipse perspective  
c) debug perspective  
d) jdbc perspective  
View Answer

10. How does eclipse provide capability for debugging browser actions?  
a) internal web browser  
b) chrome web browser  
c) firefox web browser  
d) internet explorer browser  
View Answer

# Advanced Java Questions & Answers – Web application

This set of Advanced Java Multiple Choice Questions & Answers (MCQs) focuses on “Web application”.

1. Servlet are used to program which component in a web application?  
a) client  
b) server  
c) tomcat  
d) applet  
View Answer

Answer: b  
Explanation: A servlet class extends the capabilities of servers that host applications which are accessed by way of a request-response programming model.

2. Which component can be used for sending messages from one application to another?  
a) server  
b) client  
c) mq  
d) webapp  
View Answer

Answer: c  
Explanation: Messaging is a method of communication between software components or applications. MQ can be used for passing message from sender to receiver.

3. How are java web applications packaged?  
a) jar  
b) war  
c) zip  
d) both jar and war  
View Answer

Answer: d  
Explanation: war are deployed on apache servers or tomcat servers. With Spring boot and few other technologies tomcat is brought on the machine by deploying jar.

4. How can we connect to database in a web application?  
a) oracle sql developer  
b) toad  
c) JDBC template  
d) mysql  
View Answer

Answer: c  
Explanation: JDBC template can be used to connect to database and fire queries against it.

5. How can we take input text from user in HTML page?  
a) input tag  
b) inoutBufferedReader tag  
c) meta tag  
d) scanner tag  
View Answer

Answer: a  
Explanation: HTML provides various user input options like input, radio, text, etc.

6. Which of the below is not a javascript framework for UI?  
a) Vaadin  
b) AngularJS  
c) KendoUI  
d) Springcore  
View Answer

Answer: d  
Explanation: Springcore is not a javascript framework. It is a comprehensive programming and configuration model for enterprise applications based on java.

7. Which of the below can be used to debug front end of a web application ?  
a) Junit  
b) Fitnesse  
c) Firebug  
d) Mockito  
View Answer

Answer: c  
Explanation: Firebug integrates with firefox and enables to edit, debug and monitor CSS, HTML and javascript of any web page.

8. What type of protocol is HTTP?  
a) stateless  
b) stateful  
c) transfer protocol  
d) information protocol  
View Answer

Answer: a  
Explanation: HTTP is a stateless protocol. It works on request and response mechanism and each request is an independent transaction.

9. What does MIME stand for?  
a) Multipurpose Internet Messaging Extension  
b) Multipurpose Internet Mail Extension  
c) Multipurpose Internet Media Extension  
d) Multipurpose Internet Mass Extension  
View Answer

Answer: b  
Explanation: MIME is an acronym for Multi-purpose Internet Mail Extensions. It is used for classifying file types over the Internet. It contains type/subtype e.g. application/msword.

10. What is the storage capacity of single cookie?  
a) 2048 MB  
b) 2048 bytes  
c) 4095 bytes  
d) 4095 MB  
View Answer

Answer: c  
Explanation: Storage capacity of cookies is 4095 bytes/cookie.

# Advanced Java Questions & Answers – Client and Server

This set of Advanced Java Multiple Choice Questions & Answers (MCQs) focuses on “Client and Server”.

1. How does applet and servlet communicate?  
a) HTTP  
b) HTTPS  
c) FTP  
d) HTTP Tunneling  
View Answer

Answer: d  
Explanation: Applet and Servlet communicate through HTTP Tunneling.

2. In CGI, process starts with each request and will initiate OS level process.  
a) True  
b) False  
View Answer

Answer: a  
Explanation: A new process is started with each client request and that corresponds to initiate a heavy OS level process for each client request.

3. Which class provides system independent server side implementation?  
a) Socket  
b) ServerSocket  
c) Server  
d) ServerReader  
View Answer

Answer: b  
Explanation: ServerSocket is a java.net class which provides system independent implementation of server side socket connection.

4. What happens is ServerSocket is not able to listen on the specified port?  
a) The system exits gracefully with appropriate message  
b) The system will wait till port is free  
c) IOException is thrwon when opening the socket  
d) PortOccupiedException is thrown  
View Answer

Answer: c  
Explanation: public ServerSocket() creates an unbound server socket.It throws IOException if specified port is busy when opening the socket.

5. What does bind() method of ServerSocket offer?  
a) binds the serversocket to a specific address (IPAddress and port)  
b) binds the server and client browser  
c) binds the server socket to the JVM  
d) binds the port to the JVM  
View Answer

Answer: a  
Explanation: bind() binds the serverver socket to a specific address (IPAddress and port). If address is null, the system will pick a emphemeral port and valid local address to bind socket.

6. Which of the below are common network protocols?  
a) TCP  
b) UDP  
c) TCP and UDP  
d) CNP  
View Answer

Answer: c  
Explanation: Transmission Control Protocol(TCP) and User Datagram Protocol(UDP) are the two common network protocol. TCP/IP allows reliable communication between two applications. UDP is connection less protocol.

7. Which class represents an Internet Protocol address?  
a) InetAddress  
b) Address  
c) IPAddress  
d) TCPAddress  
View Answer

Answer: a  
Explanation: InetAddress represents an Internet Protocol address. It provides static methods like getByAddress(), getByName() and other instance methods like getHostName(), getHostAddress(), getLocalHost().

8. What does local IP address start with?  
a) 10.X.X.X  
b) 172.X.X.X  
c) 192.168.X.X  
d) 10.X.X.X, 172.X.X.X, or 192.168.X.X  
View Answer

Answer: d  
Explanation: Local IP addresses look like 10.X.X.X, 172.X.X.X, or 192.168.X.X.

9. What happens if IPAddress of host cannot be determined?  
a) The system exit with no message  
b) UnknownHostException is thrown  
c) IOException is thrown  
d) Temporary IPAddress is assigned  
View Answer

10. What is the java method for ping?  
a) hostReachable()  
b) ping()  
c) isReachable()  
d) portBusy()  
View Answer

Answer: c  
Explanation: inet.isReachable(5000) is a way to ping a server in java.

# Advanced Java Questions & Answers – Servlet

This set of Advanced Java Multiple Choice Questions & Answers (MCQs) focuses on “Servlet”.

1. How constructor can be used for a servlet?  
a) Initialization  
b) Contructor function  
c) Initialization and Contructor function  
d) Setup() method  
View Answer

Answer: c  
Explanation: We cannot declare constructors for interface in Java.This means we cannot enforce this requirement to any class which implements Servlet interface.  
Also, Servlet require ServletConfig object for initialization which is created by container.

2. Can servlet class declare constructor with ServletConfig object as a argument?  
a) True  
b) False  
View Answer

Answer: b  
Explanation: ServletConfig object is created after the constructor is called and before init() is called. So, servlet init parameters cannot be accessed in the constructor.

3. What is the difference between servlets and applets?  
i.Servlets execute on Server; Applets execute on browser  
ii.Servlets have no GUI; Applet has GUI  
iii.Servlets creates static web pages; Applets creates dynamic web pages  
iv.Servlets can handle only a single request; Applet can handle multiple requests  
a) i,ii,iii are correct  
b) i,ii are correct  
c) i,iii are correct  
d) i,ii,iii,iv are correct  
View Answer

Answer: b  
Explanation: Servlets execute on Server and doesn’t have GUI.Applets execute on browser and has GUI.

4. Which of the following code is used to get an attribute in a HTTP Session object in servlets?  
a) session.getAttribute(String name)  
b) session.alterAttribute(String name)  
c) session.updateAttribute(String name)  
d) session.setAttribute(String name)  
View Answer

Answer: a  
Explanation: session has various methods for use.

5. Which method is used to get three-letter abbreviation for locale’s country in servlets?  
a) Request.getISO3Country()  
b) Locale.getISO3Country()  
c) Response.getISO3Country()  
d) Local.retrieveISO3Country()  
View Answer

Answer: a  
Explanation: Each country is usually denoted by a 3 digit code.ISO3 is the 3 digit country code.

6. Which of the following code retrieves the body of the request as binary data?  
a) DataInputStream data = new InputStream()  
b) DataInputStream data = response.getInputStream()  
c) DataInputStream data = request.getInputStream()  
d) DataInputStream data = request.fetchInputStream()  
View Answer

Answer: c  
Explanation: InputStream is an abstract class. getInputStream() retrieves the request in binary data.

7. When destroy() method of filter is called?  
a) The destroy() method is called only once at the end of the life cycle of a filter  
b) The destroy() method is called after the filter has executed doFilter method  
c) Both of the mentioned  
d) The destroyer() method is called after the filter has executed  
View Answer

Answer: a  
Explanation: destroy() is an end of life cycle method so it is called at the end of life cycle.

8. Which of the following is true about servlets?  
a) Servlets execute within the address space of web server  
b) Servlets are platform-independent because they are written in java  
c) Servlets can use the full functionality of the Java class libraries  
d) Servlets execute within the address space of web server, platform independent and uses the functionality of java class libraries  
View Answer

Answer: d  
Explanation: Servlets execute within the address space of a web server. Since it is written in java it is platform independent.The full functionality is available through libraries.

9. How is dynamic interception of requests and responses to transform the information done?  
a) servlet container  
b) servlet config  
c) servlet context  
d) servlet filter  
View Answer

Answer: d  
Explanation: Servlet has various components like container, config, context, filter. Servlet filter provides the dynamic interception of requests and responses to transform the information.

10. Which are the session tracking techniques?  
i. URL rewriting  
ii. Using session object  
iii.Using response object  
iv. Using hidden fields  
v. Using cookies  
vi. Using servlet object  
a) i, ii, iii, vi  
b) i, ii, iv, v  
c) i, vi, iii, v  
d) i, ii, iii, v  
View Answer

Answer: b  
Explanation: URL rewriting, using session object, using cookies, using hidden fields are session tracking technique

# Advanced Java Questions & Answers – Session Management

This set of Advanced Java Multiple Choice Questions & Answers (MCQs) focuses on “Session Management”.

1. Which of the following is used for session migration?  
a) Persisting the session in database  
b) URL rewritting  
c) Create new database connection  
d) Kill session from multiple sessions  
View Answer

Answer: a  
Explanation: Session migration is done by persisting session in database. It can also be done by storing session in memory on multiple servers.

2. Which of the below is not a session tracking method?  
a) URL rewriting  
b) History  
c) Cookies  
d) SSL sessions  
View Answer

Answer: b  
Explanation: History is not a session tracking type. Cookies, URL rewriting, Hidden form fields and SSL sessions are session tracking methods.

3. Which of the following is stored at client side?  
a) URL rewritting  
b) Hidden form fields  
c) SSL sessions  
d) Cookies  
View Answer

Answer: d  
Explanation: Cookies are stored at client side. Hence, it is advantageous in some cases where clients disable cookies.

4. Which of the following leads to high network traffic?  
a) URL rewritting  
b) Hidden form fields  
c) SSL sessions  
d) Cookies  
View Answer

Answer: a  
Explanation: WRL rewritting requires large data transfer to and from server which leads to network traffic and access may be slow.

5. Which of the following is not true about session?  
a) All users connect to the same session  
b) All users have same session variable  
c) Default time-out value for session variable is 20 minutes  
d) New session cannot be created for a new user  
View Answer

Answer: c  
Explanation: Default time-out value for session variable is 20 minutes. This can be changed as per requirement.

6. SessionIDs are stored in cookies.  
a) True  
b) False  
View Answer

Answer: a  
Explanation: SessionIDs are stored in cookies, URLs and hidden form fields.

7. What is the maximum size of cookie ?  
a) 4 KB  
b) 4 MB  
c) 4 bytes  
d) 40 KB  
View Answer

Answer: a  
Explanation: The 4K is the maximum size for the entire cookie, including name, value, expiry date etc. To support most browsers, it is suggested to keep the name under 4000 bytes, and the overall cookie size under 4093 bytes.

8. How can we invalidate a session?  
a) session.discontinue()  
b) session.invalidate()  
c) session.disconnect()  
d) session.falsify()  
View Answer

Answer: b  
Explanation: We can invalidate session by calling session.invalidate() to destroy the session.

9. Which method creates unique fields in the HTML which are not shown to the user ?  
a) User authentication  
b) URL writing  
c) HTML Hidden field  
d) HTML invisible field  
View Answer

Answer: c  
Explanation: HTML Hidden field is the simplest way to pass information but it is not secure and session can be hacked easily.

10. Which object is used by spring for authentication?  
a) ContextHolder  
b) SecurityHolder  
c) AnonymousHolder  
d) SecurityContextHolder  
View Answer

Answer: d  
Explanation: The SessionManagementFilter checks the contents of the SecurityContextRepository against the current contents of the SecurityContextHolder to determine whether user has been authenticated during the current request by a non-interactive authentication mechanism, like pre authentication or remember me.

# Advanced Java Questions & Answers – JSP

This set of Advanced Java Multiple Choice Questions & Answers (MCQs) focuses on “JSP”.

1. Which page directive should be used in JSP to generate a PDF page?  
a) contentType  
b) generatePdf  
c) typePDF  
d) contentPDF  
View Answer

Answer: a  
Explanation: <%page contentType=”application/pdf”> tag is used in JSP to generate PDF.

2. Which tag should be used to pass information from JSP to included JSP?  
a) Using <%jsp:page> tag  
b) Using <%jsp:param> tag  
c) Using <%jsp:import> tag  
d) Using <%jsp:useBean> tag  
View Answer

Answer: a  
Explanation: <%jsp:param> tag is used to pass information from JSP to included JSP.

3. Application is instance of which class?  
a) javax.servlet.Application  
b) javax.servlet.HttpContext  
c) javax.servlet.Context  
d) javax.servlet.ServletContext  
View Answer

Answer: d  
Explanation: Application object is wrapper around the ServletContext object and it is an instance of a javax.servlet.ServletContext object.

4. \_jspService() method of HttpJspPage class should not be overridden.  
a) True  
b) False  
View Answer

Answer: a  
Explanation: \_jspService() method is created by JSP container. Hence, it should not be overridden.

5. Which option is true about session scope?  
a) Objects are accessible only from the page in which they are created  
b) Objects are accessible only from the pages which are in same session  
c) Objects are accessible only from the pages which are processing the same request  
d) Objects are accessible only from the pages which reside in same application  
View Answer

Answer: b  
Explanation: Object data is available till session is alive.

6. Default value of autoFlush attribute is?  
a) true  
b) false  
View Answer

Answer: a  
Explanation: Default value “true” depicts automatic buffer flushing.

7. Which one is correct order of phases in JSP life cycle?  
a) Initialization, Cleanup, Compilation, Execution  
b) Initialization, Compilation, Cleanup, Execution  
c) Compilation, Initialization, Execution, Cleanup  
d) Cleanup, Compilation, Initialization, Execution  
View Answer

Answer: c  
Explanation: The correct order is Compilation, Initialization, Execution, Cleanup.

8. “request” is instance of which one of the foloowing classes?  
a) Request  
b) HttpRequest  
c) HttpServletRequest  
d) ServletRequest  
View Answer

Answer: c  
Explanation: request is object of HttpServletRequest.

9. Which is not a directive?  
a) include  
b) page  
c) export  
d) useBean  
View Answer

Answer: c  
Explanation: Export is not a directive.

10. Which is mandatory in tag?  
a) id, class  
b) id, type  
c) type, property  
d) type,id  
View Answer

Answer: a  
Explanation: The useBean searches existing object and if not found creates object using class.

# Advanced Java Questions & Answers – JSP Elements

This set of Advanced Java Multiple Choice Questions & Answers (MCQs) focuses on “JSP Elements”.

1. Which one of the following is correct for directive in JSP?  
a) <%@directive%>  
b) <%!directive%>  
c) <%directive%>  
d) <%=directive%>  
View Answer

Answer: a  
Explanation: Directive is declared as <%@directive%>.

2. Which of the following action variable is used to include a file in JSP?  
a) jsp:setProperty  
b) jsp:getProperty  
c) jsp:include  
d) jsp:plugin  
View Answer

Answer: c  
Explanation: jsp:include action variable is used to inclide a file in JSP.

3. Which attribute uniquely identifyaction element?  
a) ID  
b) Class  
c) Name  
d) Scope  
View Answer

Answer: a  
Explanation: ID attribute is used to uniquely identify action element.

4. “out” is implicit object of which class?  
a) javax.servlet.jsp.PrintWriter  
b) javax.servlet.jsp.SessionWriter  
c) javax.servlet.jsp.SessionPrinter  
d) javax.servlet.jsp.JspWriter  
View Answer

Answer: d  
Explanation: JspWriter object is referenced by the implicit variable out which is initialized automatically using methods in the PageContext object.

5. Which object stores references to the request and response objects?  
a) sessionContext  
b) pageContext  
c) HttpSession  
d) sessionAttribute  
View Answer

Answer: b  
Explanation: pageContext object contains information about directives issued to JSP page.

6. What temporarily redirects response to the browser?  
a) <jsp:forward>  
b) <%@directive%>  
c) response.sendRedirect(URL)  
d) response.setRedirect(URL)  
View Answer

Answer: c  
Explanation: response.sendRedirect(URL) directs response to the browser and creates a new request.

7. Which tag is used to set a value of a JavaBean?  
a) <c:set>  
b) <c:param>  
c) <c:choose>  
d) <c:forward>  
View Answer

Answer: a  
Explanation: <c:set> is used to set a value of a java.util.Map object.

8. Can <!–comment–> and <%–comment–%> be used alternatively in JSP?  
a) True  
b) False  
View Answer

Answer: b  
Explanation: <!–comment–> is an HTML comment. <%–comment–%> is JSP comment.

9. Java code is embedded under which tag in JSP?  
a) Declaration  
b) Scriptlet  
c) Expression  
d) Comment  
View Answer

Answer: b  
Explanation: Scriptlet is used to embedd java code in JSP.

10. Which of the following is not a directive in JSP?  
a) page directive  
b) include directive  
c) taglib directive  
d) command directive  
View Answer

Answer: d  
Explanation: command directive is not a directive in JSP.

# Advanced Java Questions & Answers – Reflection API

This set of Advanced Java Multiple Choice Questions & Answers (MCQs) focuses on “Reflection API”.

1. What are the components of marker interface?  
a) Fields and methods  
b) No fields, only methods  
c) Fields, no methods  
d) No fields, No methods  
View Answer

Answer: d  
Explanation: Marker interface in Java is an empty interface in Java.

2. Which of the following is not a marker interface?  
a) Serializable  
b) Cloneable  
c) Remote  
d) Reader  
View Answer

Answer: d  
Explanation: Reader is not a marker interface. Serializable, Cloneable and Remote interfaces are marker interface.

3. What is not the advantage of Reflection?  
a) Examine a class’s field and method at runtime  
b) Construct an object for a class at runtime  
c) Examine a class’s field at compile time  
d) Examine an object’s class at runtime  
View Answer

Answer: c  
Explanation: Reflecation inspects classes, interfaces, fields and methods at a runtime.

4. How private method can be called using reflection?  
a) getDeclaredFields  
b) getDeclaredMethods  
c) getMethods  
d) getFields  
View Answer

Answer: b  
Explanation: getDeclaredMethods gives instance of java.lang.reflect.Method.

5. How private field can be called using reflection?  
a) getDeclaredFields  
b) getDeclaredMethods  
c) getMethods  
d) getFields  
View Answer

Answer: a  
Explanation: getDeclaredFields gives instance of java.lang.reflect.Fields.

6. What is used to get class name in reflection?  
a) getClass().getName()  
b) getClass().getFields()  
c) getClass().getDeclaredFields()  
d) new getClass()  
View Answer

Answer: a  
Explanation: getClass().getName() is used to get a class name from object in reflection.

7. How method can be invoked on unknown object?  
a) obj.getClass().getDeclaredMethod()  
b) obj.getClass().getDeclaredField()  
c) obj.getClass().getMethod()  
d) obj.getClass().getObject()  
View Answer

Answer: c  
Explanation: obj.getClass().getMethod is used to invoke a method on unknown object obj.

8. How to get the class object of associated class using Reflection?  
a) Class.forName(“className”)  
b) Class.name(“className”)  
c) className.getClass()  
d) className.getClassName()  
View Answer

Answer: a  
Explanation: forName(String className) returns the Class object associated with the class or interface with the given string name.

9. What does Class.forName(“myreflection.Foo”).getInstance() return?  
a) An array of Foo objects  
b) class object of Foo  
c) Calls the getInstance() method of Foo class  
d) Foo object  
View Answer

Answer: d  
Explanation: Class.forName(“myreflection.Foo”) returns the class object of Foo and getInstance() would return a new object.

10. What does foo.getClass().getMethod(“doSomething”, null) return?  
a) doSomething method instance  
b) Method is returned and we can call the method as method.invoke(foo,null);  
c) Class object  
d) Exception is thrown  
View Answer

Answer: b  
Explanation: foo.getClass().getMethod() returns a method and we can call the method using method.invoke();

# Advanced Java Questions & Answers – AutoCloseable, Closeable and Flushable Interfaces

This set of Advanced Java Multiple Choice Questions & Answers (MCQs) focuses on “AutoCloseable, Closeable and Flushable Interfaces”.

1. Autocloseable was introduced in which Java version?  
a) java SE 7  
b) java SE 8  
c) java SE 6  
d) java SE 4  
View Answer

2. What is the alternate of using finally to close resource?  
a) catch block  
b) autocloseable interface to be implemented  
c) try block  
d) throw Exception  
View Answer

3. Which of the below is a child interface of Autocloseable?  
a) Closeable  
b) Close  
c) Auto  
d) Cloneable  
View Answer

4. It is a good practise to not throw which exception in close() method of autocloseable?  
a) IOException  
b) CustomException  
c) InterruptedException  
d) CloseException  
View Answer

5. What is the output of below snippet?

1. **try** (InputStream is = ...)
2. {
3. *// do stuff with is...*
4. }
5. **catch** (IOException e)
6. {
7. *// handle exception*
8. }

a) Runtime Error  
b) IOException  
c) Compilation Error  
d) Runs successfully  
View Answer

6. What is the difference between AutoCloseable and Closeable?  
a) Closeable is an iterface and AutoCloseable is a concrete class  
b) Closeable throws IOException; AutoCloseable throws Exception  
c) Closeable is a concept; AutoCloseable is an implementation  
d) Closeable throws Exception; AutoCloseable throws IOException  
View Answer

7. What is the use of Flushable interface?  
a) Flushes this stream by writing any buffered output to the underlying stream  
b) Flushes this stream and starts reading again  
c) Flushes this connection and closes it  
d) Flushes this stream and throws FlushException  
View Answer

8. Which version of java added Flushable interface?  
a) java SE 7  
b) java SE 8  
c) java SE 6  
d) java SE 5  
View Answer

9. Does close() implicitly flush() the stream?  
a) True  
b) False  
View Answer

10. AutoCloseable and Flushable are part of which package?  
a) Autocloseable java.lang; Flushable java.io  
b) Autocloseable java.io; Flushable java.lang  
c) Autocloseable and Flushable java.io  
d) Autocloseable and Flushable java.lang  
View Answer

# Advanced Java Questions & Answers – Application Lifecycle – Ant, Maven and Jenkins

This set of Advanced Java Multiple Choice Questions & Answers (MCQs) focuses on “Application Lifecycle – Ant, Maven and Jenkins”.

1. Which of below is not a dependency management tool?  
a) Ant  
b) Maven  
c) Gradle  
d) Jenkins  
View Answer

Answer: d  
Explanation: Jenkins is continuous integration system. Ant, Maven, Gradle is used for build process.

2. Which of the following is not a maven goal?  
a) clean  
b) package  
c) install  
d) debug  
View Answer

Answer: d  
Explanation: clean, package, install are maven goals. Debug is used finding and resolving of defects.

3. Which file is used to define dependency in maven?  
a) build.xml  
b) pom.xml  
c) dependency.xml  
d) version.xml  
View Answer

Answer: b  
Explanation: pom.xml is used to define dependency which are used to package the jar. POM stands for project object model.

4. Which file is used to specify the packaging cycle?  
a) build.xml  
b) pom.xml  
c) dependency.xml  
d) version.xml  
View Answer

Answer: a  
Explanation: Project structure is specified in build.xml.

5. Which environment variable is used to specify the path to maven?  
a) JAVA\_HOME  
b) PATH  
c) MAVEN\_HOME  
d) CLASSPATH  
View Answer

Answer: c  
Explanation: MAVEN\_HOME should be set to the bin folder of maven installation.

6. Which of the below is a source code management tool?  
a) Jenkins  
b) Maven  
c) Git  
d) Hudson  
View Answer

Answer: c  
Explanation: Source code management tools help is version control, compare different versions of code, crash management, etc. Git, SVN are popular source code management tools.

7. Can we run Junits as a part of Jenkins job?  
a) True  
b) False  
View Answer

8. Which command can be used to check maven version?  
a) mvn -ver  
b) maven -ver  
c) maven -version  
d) mvn -version  
View Answer

Answer: d  
Explanation: mvn -version can be used to check the version of installed maven from command prompt.

9. Which of the following is not true for Ant ?  
a) It is a tool box  
b) It provides lifecycle management  
c) It is procedural  
d) It doesnt have formal conventions  
View Answer

Answer: b  
Explanation: Ant doesnt provide lifecycle management. Maven provides lifecycle.

10. Which maven plugin creates the project structure?  
a) dependency  
b) properties  
c) archetype  
d) execution  
View Answer

Answer: c  
Explanation: Archetype is the maven plugin which creates the project structure.

# Advanced Java Questions & Answers – Annotations

This set of Advanced Java Multiple Choice Questions & Answers (MCQs) focuses on “Annotations”.

1. Which version of Java introduced annotation?  
a) Java 5  
b) Java 6  
c) Java 7  
d) Java 8  
View Answer

Answer: a  
Explanation: Annotation were introduced with Java 5 version.

2. Annotation type definition looks similar to which of the following?  
a) Method  
b) Class  
c) Interface  
d) Field  
View Answer

Answer: c  
Explanation: Annotation type definition is similar to an interface definition in which the keyword interface is preceded by the sign @.

3. Which is the following is not pre defined annotation in Java?  
a) @Deprecated  
b) @Overriden  
c) @SafeVarags  
d) @FunctionInterface  
View Answer

Answer: b  
Explanation: @Overriden is not a pre defined annotation in Java. @Depricated, @Override, @SuppressWarnings, @SafeVarags and @FunctionInterface are the pre defined annotations.

4. Annotations which are applied to other annotations are called meta annotations.  
a) True  
b) False  
View Answer

Answer: a  
Explanation: Annotations which are applied to other annotations are called meta annotations.

5. Which one of the following annotations is not used in Hibernate?  
a) @Entity  
b) @Column  
c) @Basic  
d) @Query  
View Answer

Answer: d  
Explanation: @Query is not an annotation used in Hibernate.

6. Which one of the following is not ID generating stratagy using @GeneratedValue annotation?  
a) Auto  
b) Manual  
c) Identity  
d) Sequence  
View Answer

Answer: b  
Explanation: Auto, Table, Identity and Sequence are the ID generating stratagies using @GeneratedValue annotation.

7. Which one of the following is not an annotation used by Junit with Junit4?  
a) @Test  
b) @BeforeClass  
c) @AfterClass  
d) @Ignored  
View Answer

Answer: d  
Explanation: @Test, @Before, @BeforeClass, @After, @AfterClass and @Ignores are the annotations used by Junit with Junit4.

8. Using which annotation non visible or private method can be tested?  
a) @VisibleForTesting  
b) @NonVisibleForTesting  
c) @Visible  
d) @NonVisible  
View Answer

Answer: a  
Explanation: Using @VisibleForTesting annotation private or non visible method can be tested.

9. Which of the following annotation is used to avoid executation of Junits?  
a) @NoTest  
b) @explicit  
c) @avoid  
d) @ignore  
View Answer

Answer: d  
Explanation: @ignore annotation is used to avoid executation of Junits.

10. Which is the Parent class of annotation class?  
a) Class  
b) Object  
c) Main  
d) Super  
View Answer

Answer: b  
Explanation: Object is the parent class of annotation class.