1. Given:

public class Hello {

String title;

int value;

public Hello() {

title += “ World”;

}

public Hello(int value) {

this.value = value;

title = “Hello”;

Hello();

}

}

and:

Hello c = new Hello(5);

System.out.println(c.title);

A. Hello

B. Hello World

C. Compilation fails.

D. Hello World 5

E. The code runs with no output.

F. An exception is thrown at runtime.

2. What is the output of this program?

class newthread extends Thread {

Thread t;

newthread() {

t = new Thread(this,"My Thread");

t.start();

}

public void run() {

try {

t.join()

System.out.println(t.getName());

}catch(Exception e) {

System.out.print("Exception");

}

}

}

class multithreaded\_programing {

public static void main(String args[]) {

new newthread();

}

}

A. My Thread

B. Thread[My Thread,5,main]

C. Exception

D. Runtime Error

3. Which statement is not true in java language?

(A)   A public member of a class can be accessed in all the packages.

(B)   A private member of a class cannot be accessed by the methods of the same class.

(C)   A private member of a class cannot be accessed from its derived class.

(D)   A protected member of a class can be accessed from its derived class.

(E)   None of the above.

4. Which of the following is true?

(A)   A finally block is executed before the catch block but after the try block.

(B)   A finally block is executed, only after the catch block is executed.

(C)   A finally block is executed whether an exception is thrown or not.

(D)   A finally block is executed, only if an exception occurs.

(E)   None of the above.

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

// INSERT DECLARATION HERE

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

List<Integer> row = new ArrayList<Integer>();

for (int j = 0; j <= 10; j++)

row.add(i \* j);

table.add(row);

}

for (List<Integer> row : table)

System.out.println(row);

}

A. List<List<Integer>> table = new List<List<Integer>>();

B. List<List<Integer>> table = new ArrayList<List<Integer>>();

C. List<List<Integer>> table = new ArrayList<ArrayList<Integer>>();

D. List<List, Integer> table = new List<List, Integer>();

E. List<List, Integer> table = new ArrayList<List, Integer>();

F. List<List, Integer> table = new ArrayList<ArrayList, Integer>();

G. None of the above

6. import java.util.regex.\*;

class Regex2 {

public static void main(String[] args) {

Pattern p = Pattern.compile(args[0]);

Matcher m = p.matcher(args[1]);

boolean b = false;

while(b = m.find()) {

System.out.print(m.start() + m.group());

}

}

}

And the command line:

java Regex2 "\d\*" ab34ef

What is the result?

A. 234

B. 334

C. 2334

D. 0123456

E. 01234456

F. 12334567

G. Compilation fails

7. Given:

class Chicks {

synchronized void yack(long id) {

for(int x = 1; x < 3; x++) {

System.out.print(id + " ");

Thread.yield();

}

}

}

public class ChicksYack implements Runnable {

Chicks c;

public static void main(String[] args) {

new ChicksYack().go();

}

void go() {

c = new Chicks();

new Thread(new ChicksYack()).start();

new Thread(new ChicksYack()).start();

}

public void run() {

c.yack(Thread.currentThread().getId());

}

}

Which are true? (Choose all that apply.)

A. Compilation fails

B. The output could be 4 4 2 3

C. The output could be 4 4 2 2

D. The output could be 4 4 4 2

E. The output could be 2 2 4 4

F. An exception is thrown at runtime

8. What will happen when you compile and run the following code?

public class Main{

private int i = 1;

public static void main(String argv[]){

int i = 2;

Main s = new Main ();

s.someMethod();

}

public static void someMethod(){

System.out.println(i);

}

}

1. 1 will be printed out
2. 2 will be printed out
3. A compile time error will be generated
4. An exception will be thrown

9. Given:

class Mixer {

Mixer() { }

Mixer(Mixer m) { m1 = m; }

Mixer m1;

public static void main(String[] args) {

Mixer m2 = new Mixer();

Mixer m3 = new Mixer(m2); m3.go();

Mixer m4 = m3.m1; m4.go();

Mixer m5 = m2.m1; m5.go();

}

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

}

What is the result?

A. hi

B. hi hi

C. hi hi hi

D. Compilation fails

E. hi, followed by an exception

F. hi hi, followed by an exception

10. Given:

class Emu {

static String s = "-";

public static void main(String[] args) {

try {

throw new Exception();

} catch (Exception e) {

try {

try { throw new Exception();

} catch (Exception ex) { s += "ic "; }

throw new Exception(); }

catch (Exception x) { s += "mc "; }

finally { s += "mf "; }

} finally { s += "of "; }

System.out.println(s);

} }

What is the result?

A. -ic of

B. -mf of

C. -mc mf

D. -ic mf of

E. -ic mc mf of

F. -ic mc of mf

G. Compilation fails

11. Which one of the following options correctly makes use of Callable that will compile without any errors?

A. import java.util.concurrent.Callable;

class CallableTask implements Callable {

public int call() {

System.out.println("In Callable.call()");

return 0;

}

}

B. import java.util.concurrent.Callable;

class CallableTask extends Callable {

public Integer call() {

System.out.println("In Callable.call()");

return 0;

}

}

C. import java.util.concurrent.Callable;

class CallableTask implements Callable<Integer> {

public Integer call() {

System.out.println("In Callable.call()");

return 0;

}

}

D. import java.util.concurrent.Callable;

class CallableTask implements Callable<Integer> {

public void call(Integer i) {

System.out.println("In Callable.call(i)");

}

}

12. Which one of the following methods return a Future object?

A. The overloaded replace() methods declared in the ConcurrentMap interface

B. The newThread() method declared in the ThreadFactory interface

C. The overloaded submit( ) methods declared in the ExecutorService interface

D. The call() method declared in the Callable interface

13. In your application, there is a producer component that keeps adding new items to a

fixed-size queue; the consumer component fetches items from that queue. If the queue is full, the producer has to wait for items to be fetched; if the queue is empty, the consumer has to wait for items to be added.

Which one of the following utilities is suitable for synchronizing the common queue for

concurrent use by a producer and consumer?

A. RecursiveAction

B. ForkJoinPool

C. Future

D. Semaphore

E. TimeUnit

14. Here is a class named PingPong that extends the Thread class. Which of the following PingPong class implementations correctly prints “ping” from the worker thread and then prints “pong” from the main thread?

A. class PingPong extends Thread {

public void run() {

System.out.println("ping ");

}

public static void main(String []args) {

Thread pingPong=new PingPong();

System.out.print("pong");

}

}

B. class PingPong extends Thread {

public void run() {

System.out.println("ping ");

}

public static void main(String []args) {

Thread pingPong=new PingPong();

pingPong.run();

System.out.print("pong");

}

}

C. class PingPong extends Thread {

public void run() {

System.out.println("ping");

}

public static void main(String []args) {

Thread pingPong=new PingPong();

pingPong.start();

System.out.println("pong");

}

}

D. class PingPong extends Thread {

public void run() {

System.out.println("ping");

}

public static void main(String []args) throws InterruptedException{

Thread pingPong=new PingPong();

pingPong.start();

pingPong.join();

System.out.println("pong");

}

}

15. Consider the following code snippet:

if(i == 10.0)

System.out.println("true");

Which one of the following declarations of the variable i will compile without errors and print true

when the program executes?

A) int i = 012;

B) int i = 10.0f;

C) int i = 10L;

D) int i = 10.0;

16. Consider the following code segment:

StringBuffer strBuffer = new StringBuffer("This, that, etc.!");

System.out.println(strBuffer.replace(12, 15, "etcetera"));

Which one of the following options correctly describes the behavior of this code segment?

A) This code segment: This, that, etcetera.!

B) This code segment: This, that, etcetera!

C) This code segment: This, that, etc.

D) This program throws in an ArrayIndexOutOfBoundsException.

17. Consider the following code segment:

MODIFIER class SomeClass { }

Which three of the following modifiers, when replaced instead of MODIFIER, will compile cleanly?

A) public

B) protected

C) private

D) abstract

E) final

F) static

18. Consider the following program:

class Outer {

class Inner {

public void print() {

System.out.println("Inner: print");

}

}

}

class Test {

public static void main(String []args) {

// Stmt#1

inner.print();

}

}

Which one of the following statements will you replace in place of // Stmt#1 to make the program

compile and run successfully to print “Inner: print” in console?

a) Outer.Inner inner = new Outer.Inner();

b) Inner inner = new Outer.Inner();

c) Outer.Inner inner = new Outer().Inner();

d) Outer.Inner inner = new Outer().new Inner();

19. Which of the following statements are true with respect to enums? (Select all that

apply.)

a) An enum can have private constructor.

b) An enum can have public constructor.

c) An enum can have public methods and fields.

d) An enum can implement an interface.

e) An enum can extend a class

20. Consider the following program:

class WildCard {

interface BI {}

interface DI extends BI {}

interface DDI extends DI {}

static class C<T> {}

static void foo(C<? super DI> arg) {}

public static void main(String []args) {

foo(new C<BI>()); // ONE

foo(new C<DI>()); // TWO

foo(new C<DDI>()); // THREE

foo(new C()); // FOUR

}

}

Which of the following options are correct?

a) Line marked with comment ONE will result in a compiler error.

b) Line marked with comment TWO will result in a compiler error.

c) Line marked with comment THREE will result in a compiler error.

d) Line marked with comment FOUR will result in a compiler error

21. Which one of the following class definitions will compile without any errors?

a) class P<T> {

static T s\_mem;

}

b) class Q<T> {

T mem;

public Q(T arg) {

mem = arg;

}

}

c) class R<T> {

T mem;

public R() {

mem = new T();

}

}

d) class S<T> {

T []arr;

public S() {

arr = new T[10];

}

}

22. Which one of the following class/interface supports “retrieval if elements based on the

closest match to a given value or values?”

a) EnumSet

b) HashSet

c) AbstractSet

d) NavigableSet

23. Consider the following program:

import java.util.Comparator;

import java.util.Arrays;

class CountryComparator implements Comparator<String> {

public int compare(String country1, String country2) {

return country2.compareTo(country2); // COMPARE\_TO

}

}

public class Sort {

public static void main(String[] args) {

String[] brics = {"Brazil", "Russia", "India", "China"};

Arrays.sort(brics, null);

for(String country : brics) {

System.out.print(country + " ");

}

}

}

Which one of the following options correctly describes the behavior of this program?

a) The program results in a compiler error in the line marked with the comment

COMPARE\_TO.

b) The program prints the following: Brazil Russia India China.

c) The program prints the following: Brazil China India Russia.

d) The program prints the following: Russia India China Brazil.

e) The program throws the exception InvalidComparatorException.

f ) The program throws the exception InvalidCompareException.

g) The program throws the exception NullPointerException.

24. Consider the following program:

import java.util.Scanner;

class AutoCloseableTest {

public static void main(String []args) {

try (Scanner consoleScanner = new Scanner(System.in)) {

consoleScanner.close(); // CLOSE

consoleScanner.close();

}

}

}

Which one of the following statements is correct?

a) This program terminates normally without throwing any exceptions.

b) This program throws an IllegalStateException.

c) This program throws an IOException.

d) This program throws an AlreadyClosedException.

e) This program results in a compiler error in the line marked with the comment CLOSE.

25. Which one of the following options is NOT correct?

a) A Condition object can be acquired from a Lock object.

b) Executor is an interface that declares only one method, namely

void execute(Runnable).

c) Using a semaphore with one resource is similar to using a lock.

d) CountDownLatch allows each thread to complete its assigned task step by step

26. Which one of the following statements is NOT correct with respect to nested classes?

A) An outer class can access the private members of the nested class without declaring

an object of the nested class.

B) Static nested classes can access the static members of the outer class.

C) Static nested classes can be declared abstract or final.

D) Static nested classes can extend another class or they can be used as a base class.

27. Consider the following program:

class MyThread extends Thread {

public void run() {

System.out.println("In run method; thread name is: " + Thread.

currentThread().getName());

}

public static void main(String args[]) {

Thread myThread = new MyThread();

myThread.start();

myThread.start(); //#1

}

}

A) The program results in a compiler error at statement #1.

B) The program results in throwing an IllegalThreadStateException.

C) The program prints the following:

In the run method; thread name is: thread-0

In the main method; thread name is: thread-0

D) The program prints the following:

In the run method; thread name is: thread-0

28. Consider the following program:

class Base {

public void print() {

System.out.println("Base");

}

}

class Derived extends Base {

public void print() {

System.out.println("Derived");

}

}

class Test {

public static void main(String args[]) {

Base obj1 = new Derived();

Base obj2 = (Base)obj1;

obj1.print();

obj2.print();

}

}

A) Derived

Derived

B) Base

Derived

C) Derived

Base

D) Base

Base

29. Which of the following is not true about indexes?

A) Indexes are created to enforce uniqueness on columns.

B) Indexes are created to enable fast retrieval by column values.

C) Columns that are frequently used with equal conditions in WHERE clauses are good candidates for indexes.

D) Indexes are created with the ALTER TABLE command.

30. Which of the following is not true about modifying table columns?

A) You can drop a column at any time.

B) You can add a column at any time as long as it is a NULL column.

C) You can increase the number of characters in character columns or the number of digits in numeric columns

D) You cannot increase or decrease the number of decimal places.

31. What is true about views among all the given below statements:

A) View never references actual table for which it is created.

B) View can’t use JOIN in its query.

C) The performance of the view degrades if they are based on other views.

D) Only option to safeguard data integrity.

32. What is the significance of the statement “GROUP BY d.name” in the given query?

SELECT d.name, count (emp\_id) emp\_no

FROM department d INNER JOIN Employee e

ON d.dept\_id=e.emp\_id

GROUP BY d.name

A) Aggregation of the field “name” of both table

B) Aggregation of the field “name” of table “department”

C) Sorting of the field “name”

D) None of these