

Given

```
1. public class Whiz{  
2.     public static void main(String[] args) {  
3.         Comparable com = new Comparable(){  
4.             public int compareTo(Object o){  
5.                 return -1;  
6.             }  
7.         };  
8.     }  
9. }
```

Which of the following can be used to replace from line 3 to line 7?

Please select :

- A. Comparable com = (o) -> return -1;
- B. Comparable com = o -> -1;
- C. Comparable com = () -> return -1;
- D. Comparable com = o -> {return -1};
- E. None of above.

Given

```
1. public class Whiz{
2.     public static void main(String[] args) {
3.         System.out.print(ln2.print());
4.     }
5. }
6.
7. interface ln1{
8.     public static void print(){
9.         System.out.println("ln1");
10.    }
11. }
12.
13. interface ln2 extends ln1{
14.     static String print(){
15.         return "ln2";
16.     }
17. }
```

What is the output?

Please select :

- A. ln1
- B. ln2
- C. An Exception.
- D. Compilation fails due to error at line 14.
- E. Compilation fails due to multiple errors.

Given

```
1. public class Whiz{  
2.  
3.     int x = 10;  
4.  
5.     public static void method(int c,int i){  
6.  
7.         class Test{  
8.             public void in0{  
9.                 // here  
10.            }  
11.        }  
12.        c +=2;  
13.        new Test().in0;  
14.    }  
15.  
16.    public static void main(String p[]){  
17.        Whiz.method(3,4);  
18.    }  
19. }
```

Which of the following variables can be accessed at line 9?

Please select :

- A. Only x.
- B. Only i.
- C. Only c.
- D. Only c and i.
- E. All

Which of the following method can be used to iterate over all instance of an Enum?

Please select :

- A. `forEach(Consumer)`
- B. `set()`
- C. `forEach()`
- D. `values()`
- E. None of above

Which of the following is true about JDBC API? (Choose 2)

Please select :

- A. The JDBC API supports only two-tier processing models for database access.
- B. The JDBC API supports only three-tier processing models for database access.
- C. The JDBC API supports both two-tier and three-tier processing models for database access.
- D. With JDBC it possible to write a single database application that can run on different platforms and interact with different DBMS.
- E. JDBC is language independent.

Which of the following is true about establishing a connection using JDBC?

Please select :

- A. Using JDBC, we can only connect to DBMS.
- B. We can only use `java.sql.DriverManager` class for connecting a JDBC application to a target data source.
- C. We can use `DriverManager` or `DataSource` class to connect a JDBC application to a target data source.
- D. None of above.

Consider following five steps for processing any SQL statement with JDBC.

1. *Create a statement.*
2. *Establishing a connection.*
3. *Process the ResultSet object.*
4. *Execute the query.*
5. *Close the connection.*

Which of the following shows correct order of above five steps?

Please select :

- A. 1, 2, 3, 4, 5.
- B. 4, 3, 2, 1, 5.
- C. 1, 2, 4, 3, 5.
- D. 2, 1, 4, 3, 5
- E. 1, 4, 2, 5, 3.

Given

```
1 import java.sql.*;
2 public class Pro{
3
4     public static void main(String []args)throws SQLException{
5         DBConnect dc = new DBConnect();
6         dc.getUID();
7     }
8 }
9
10 class DBConnect{
11     private Connection con;
12     private Statement stm;
13     private ResultSet rs;
14
15     public DBConnect(){
16         // Assume: A connection object and a Statement object are successfully created in this constructor.
17     }
18
19     public void getUID()throws SQLException{
20         try{
21             String query = "SELECT ID , LName FROM users";
22             rs = stm.executeQuery(query);
23             rs.absolute(3);
24             System.out.print(rs.getInt(1) + "\t" + rs.getString("LName"));
25         }finally{
26             con.close();
27         }
28     }
29 }
```

Note: You can consider given table as the "users" table.

ID	Fname	Lname	Jdate
1	Buddhika	Livera	2012-10-01
2	Piumi	Rekha	2012-10-04
7	Clerk	Kent	2012-10-11
8	Oliver	Queen	2012-10-14

Which of the following is true?

Please select :

- A. "7" Kent" will be printed as the output.
- B. "1" Livera" will be printed as the output.
- C. "3" Queen" will be printed as the output.
- D. Compilation fails due to error at line 23.
- E. An exception will be thrown at runtime.

Given

```
1 import java.sql.*;
2 public class Pro{
3
4     public static void main(String []args)throws SQLException{
5         DBConnect dc = new DBConnect();
6         dc.insert(6,"Martha","Kent","2012-10-10");
7     }
8 }
9
10 class DBConnect{
11     private Connection con;
12     private Statement stm;
13     private ResultSet rs;
14
15     public DBConnect(){
16         // Assume: A connection object and a Statement object are successfully created in this constructor.
17     }
18
19     public void insert(int id, String fname, String lname, String date)throws SQLException{
20         try{
21             String query = "INSERT INTO users VALUES( null , " + fname + ", "+lname+ ", "+date+"")";
22             stm.executeQuery (query);
23         }finally{
24             con.close();
25         }
26     }
27 }
```

Note: You can consider given table as the "users" table. And consider the ID can be auto increment.

ID	Fname	Lname	Jdate
1	Buddhika	Livera	2012-10-01
2	Piumi	Rekha	2012-10-04
7	Clerk	Kent	2012-10-11
8	Oliver	Queen	2012-10-14

Which of the following is true?

Please select :

- A. A new record will be added as follows:

null	Martha	Kent	2012-10-10
------	--------	------	------------

- B. A new record will be added as follows:

12	Martha	Kent	2012-10-10
----	--------	------	------------

- C. Code will run without any exception but no record will be added.
- D. Compilation fails due to error at line 22.
- E. An exception will be thrown at runtime.

Given

```
1.     try{
2.         con.setAutoCommit(false);
3.
4.     Savepoint sp1 = con.setSavepoint("save1");
5.
6.     String query1 = "DELETE FROM allstar WHERE ID = 103";
7.     stm.executeUpdate(query1);
8.
9.     String query2 = "DELETE FROM allstar WHERE ID = 124";
10.    stm.executeUpdate(query2);
11.
12.    con.commit();
13.
14.    con.rollback(sp1);
15.
16. }finally{
17.     con.close();
18. }
```

Note: Assume "con" is active connection and "stm" as a reference to valid Statement object. And consider the following snapshot of the table "allstar".

ID	FName	LName	Birthday	Country
58	Stuart	Broad	1986-06-24	England
89	Herschelle	Gibbs	1974-02-23	South Africa
103	Lou	Vincent	1978-11-11	New Zealand
115	Shahid	Afridi	1980-03-01	Pakistan
124	Sanath	Jayasuriya	1969-06-30	Sri Lanka
136	Lou	Vincent	1978-11-11	New Zealand
178	Piyush	Chawla	1988-12-24	India
248	Ryan	tenDoeschate	1980-06-30	Netherlands

Which of the following is true after executing the above code fragment?

Please select :

- A. There will be no changes to the "allstar" table.
- B. Only the record with "ID" 103 will be deleted.
- C. Only the record with "ID" 124 will be deleted.
- D. Compilation will be failed if we added this code fragment to a program.
- E. An exception will be thrown at runtime.

Which of the following is true about the RowSet? *(Choose 2)*

Please select :

- A. The ResultSet objects are derived from the RowSet interface.
- B. All RowSet objects are JavaBeans components.
- C. There are three kinds of RowSet objects.
- D. A RowSet object has scrollability and updatability by default.

Which of the following is true about JdbcRowSet objects?

- I. JdbcRowSet always maintains a connection with a DBMS via a JDBC driver.
- II. We can create a JdbcRowSet object by passing a ResultSet object to “JdbcRowSetImpl” class’ constructor.
- III. We can create a JdbcRowSet object by passing a Connection object to “JdbcRowSetImpl” class’ constructor.

A JdbcRowSet object is especially well suited for use with a thin client.

Please select :

- A. Only IV.
- B. Only I and II.
- C. Only I, II and III.
- D. Only I, II and IV.
- E. Only II, III and IV.
- F. All.

Given

```
1. class ExTest{
2.     Integer l;
3.     public static void main(String args[]){
4.         String s;
5.         try{
6.             s = new Ex6().l.toString();
7.         }finally{
8.
9.             try{
10.                 int i = Integer.parseInt(args[0]);
11.             }catch(NumberFormatException E){
12.                 System.out.print("NumberFormatException ");
13.             }finally{
14.                 System.out.print("Fin2 ");
15.             }
16.             System.out.print("Fin1 ");
17.         }
18.     }
19. }
```

And the given command line invocation is *java ExTest A*

What is the output?

Please select :

- A. NumberFormat Fin2 Fin1
- B. NumberFormat Fin2
- C. NumberFormat Fin2 followed by uncaught exception
- D. Fin2 Fin1 followed by uncaught exception
- E. NumberFormat Fin2 Fin1 followed by uncaught exception
- F. Fin2 Fin1 followed by uncaught exception

Given

```
1. class TooSmallException extends RuntimeException{  
2. }  
3.  
4. class Divider{  
5.     public double divide(double d1, double d2){  
6.         if(d1<0.01 || d2<0.01){  
7.             throw new TooSmallException();  
8.         }else{  
9.             return (d1/d2);  
10.        }  
11.    }  
12. }  
13.  
14. public class ETest{  
15.     public static void main(String [] args){  
16.         Divider d = new Divider();  
17.  
18.         double i=Double.parseDouble(args[0]);  
19.         double j=Double.parseDouble(args[1]);  
20.  
21.         double ans = d.divide(i,j);  
22.         System.out.print(ans);  
23.     }  
24. }
```

Which is true?

Please select :

- A. Compilation succeeds and if we pass 1 and 2 as “args[0]” and “args[1]” the output will be an uncaught TooSmallException.
- B. Compilation succeeds and if we pass 0.001 and 0.2 as “args[0]” and “args[1]” the output will be an uncaught TooSmallException.
- C. Compilation fails due to error on line 7 as there is no Exception class called as “TooSmallException” in java.
- D. Compilation fails as “TooSmallException” is checked Exception and we didn’t declare it or handle it.
- E. Compilation succeeds and if we pass “A” and “B” as “args[0]” and “args[1]” 0.98484848484849 will be printed as the output.(Note ASCII value of ‘A’ is 65 and ‘B’ is 66 in decimals.)

What are true?

Please select :

- A. We can't create custom exception using subclass of java.lang.RuntimeException class.
- B. We can create custom unchecked exception by extending java.lang.Exception class.
- C. We can create custom checked exception by extending java.lang.RuntimeException class.
- D. We can create custom checked exception using subclass of java.lang.Exception class.
- E. None of Above.

Given

```
1. public class Whiz{  
2.     public static void main(String[] args) {  
3.         try(Resource r = new Resource()){  
4.             System.out.print("0");  
5.             throw new RuntimeException();  
6.         }  
7.     }  
8. }  
9. class Resource implements AutoCloseable{  
10.     @Override  
11.     public void close() {  
12.         System.out.println("Auto Closed.");  
13.     }  
14. }
```

What is the output?

Please select :

- A. 0
- B. Auto Closed
- C. 0Auto Closed
- D. 0Auto Closed followed by an exception
- E. None.

Suppose you want to enable assertions on the Whiz class but not for the Lab class which of the following command you will use to execute the Main class? (Whiz class contains the main method)

Please select :

- A. java -ea -da:Whiz Lab
- B. java -ea Whiz
- C. java -da: Lab.class -ea: Whiz
- D. java -ea Whiz -da: Lab
- E. java -ea -da:Lab Whiz

Given

```
1. import java.util.stream.IntStream;  
2.  
3. public class Whiz{  
4.     public static void main(String[] args) {  
5.         IntStream ints = IntStream.range(1,3);  
6.         ints.peek(s -> System.out.print(s*2)).forEach(System.out::print);  
7.     }  
8. }
```

What is the output?

Please select :

- A. 123246
- B. 2142
- C. 1224
- D. An Exception.
- E. Compilation fails.

Which of the following lambda can be passed as the parameter for the forEach method? (Assume the we are going to invoke forEach method on IntSteam).

Please select :

- A. `x -> x+=x`
- B. `x -> x*2;`
- C. `x -> return x*2`
- D. `System.out::print()`
- E. None of above.

Given

```
1. import java.util.function.Predicate;
2. import java.util.stream.LongStream;
3.
4. public class Whiz{
5.     public static void main(String[] args) {
6.         LongStream dbs = LongStream.range(3, 6);
7.         Predicate<Long> pre = l -> l%3==0;
8.         dbs.filter(pre).forEach(System.out::print);
9.     }
10. }
```

What is the output?

Please select :

- A. 2
- B. 24.
- C. No output.
- D. An Exception.
- E. Compilation fails.

Given

```
1. import java.util.stream.Stream;
2.
3. public class Whiz{
4.     public static void main(String[] args) {
5.         Stream<String> dbs = Stream.of("paypal","pyme","paytoo","payee");
6.         dbs.filter(s -> s.startsWith("pay")).filter(s ->
7.             s.endsWith("e")).forEach(System.out::print);
8.     }
}
```

What is the output?

Please select :

- A. pyme
- B. payee
- C. No output.
- D. An Exception.
- E. Compilation fails.

Given

```
1. import java.util.Map;
2. import java.util.HashMap;
3.
4. public class Whiz{
5.     public static void main(String[] args) {
6.         Map<Integer,Double> map = new HashMap<>();
7.         map.put(1, 1.0);
8.         map.put(2, 2.0);
9.         map.put(3, 3.0);
10.        map.put(4, 4.4);
11.        map.putIfPresent(4, null);
12.        map.keySet().forEach(System.out::print);
13.    }
14. }
```

What is the output?

Please select :

- A. 123
- B. 1234
- C. 123null
- D. An Exception.
- E. Compilation fails.

Given

```
1. import java.util.HashMap;
2. import java.util.Map;
3.
4. public class Whiz {
5.     public static void main(String [ ] args){
6.         Map<Integer,Integer> map = new HashMap<>();
7.         map.put(1, 7);
8.         map.put(2, 8);
9.         map.put(3, 6);
10.        map.put(4, 5);
11.
12.        map.putIfAbsent(4, 61);
13.        map.computeIfPresent(3, (k,v)->null);
14.
15.        map.values().forEach(System.out::print);
16.    }
17. }
```

What is the output?

Please select :

- A. 1234
- B. 7865
- C. 785
- D. An Exception.
- E. Compilation fails.

Given

```
1. import java.util.ArrayList;
2. import java.util.List;
3. public class Whiz{
4.     public static void main(String[] args) {
5.         List list = new ArrayList<>();
6.         list.add(new Integer(10));
7.         list.add(new Integer(11));
8.         list.add(new Integer(13));
9.         list.add(new Integer(14));
10.
11.        list.removeIf(in -> in%2 != 0);
12.
13.        System.out.println(list);
14.    }
15.}
```

What is the result?

Please select :

- A. [10,11,13,14]
- B. [10, 14]
- C. [11, 13]
- D. An exception
- E. Compilation fails.

Given

```
1.     public class Whiz{  
2.         public static void main(String[] args) {  
3.             Func<String> sup = _____  
4.         }  
5.     }  
6.  
7.     interface Func<V>{  
8.         V get();  
9.     }
```

Which of the following can be inserted into the blank to create Func?

Please select :

- A. () -> "A";
- B. () -> return "A";
- C. -> return "A";
- D. () -> {"A"};
- E. None of above.

Consider following.

$(x,y) \rightarrow \text{Integer.compare}(x, y);$

Which of the following functional method can be satisfied by above given lambda?

Please select :

- A. public abstract Double compare(Integer i1, Integer i2);
- B. public abstract Integer compare(Integer i);
- C. public abstract Integer compare(Integer i);
- D. public abstract Integer compare(Integer i1, Integer i2);
- E. None of above.

Given

```
1. interface Walk {  
2.     public default int getSpeed() {  
3.         return 5;  
4.     }  
5. }  
6.  
7. interface Run {  
8.     public default int getSpeed() {  
9.         return 10;  
10.    }  
11. }  
12.  
13. public class Animal implements Walk,Run{  
14.  
15.     public static void main(String args[]){  
16.         Animal an = new Animal();  
17.         System.out.println(an.getSpeed());  
18.     }  
19. }
```

What is the result?

Please select :

- A. 5
- B. 10
- C. An exception is thrown at the runtime.
- D. Compilation fails due to an error on line 13.
- E. Compilation fails due to multiple errors.

Which of the following occasions, you have to choose interface inheritance over class inheritance?

1. You are asked to create two classes which should have some behaviors in common and one class is already subclass of another class.
2. You are asked to create two classes which should have some behaviors in common and one class is already implements another interface.
3. You are asked to create two classes which should have some behaviors in common and both classes are already subclasses of other classes

Please select :

- A. All
- B. Only I
- C. Only II and III
- D. Only I and III
- E. none

Consider the following statement about a thread pool which can create with the Executors class.

A thread pool that creates as many threads it needs to execute the task in parallel.

Which of the following thread pool is described by the above statement?

Please select :

- A. Fixed Thread Pool.
- B. Cached Thread Pool.
- C. Scheduled Thread Pool.
- D. None of above.

Given

```
1. import java.util.concurrent.locks.*;
2. import java.util.concurrent.*;
3. import java.util.*;
4. import java.util.concurrent.TimeUnit;
5. class ETest{
6.
7.     static class Task1 implements Runnable{
8.         public void run(){
9.             System.out.print("Task 1 ");
10.        }
11.    }
12.    static class Task2 implements Runnable{
13.        public void run(){
14.            System.out.print("Task 2 ");
15.            try{
16.                Thread.sleep(4500);
17.            }catch(Exception e){
18.                System.out.print(e);
19.            }
20.        }
21.    }
22.
23.    public static void main(String []args)throws Exception{
24.        final ExecutorService pool = Executors.newFixedThreadPool(2);
25.
26.        Future f1 = pool.submit(new Task1());
27.        Future f2 = pool.submit(new Task2());
28.        pool.awaitTermination(4, TimeUnit.SECONDS);
29.
30.        pool.shutdownNow();
31.    }
32. }
```

Which is the output of the above code?

Please select :

- A. Task 1 Task 2 will be printed and then an exception will be thrown at the runtime.
- B. Task 1 Task 2 will be printed in 4.5 second interval and then an exception will be thrown at the runtime.
- C. Task 1 Task 2 will be printed and then the program will exit after 4 seconds.
- D. Task 1 Task 2 will be printed and then the program will immediately exit.
- E. Compilation fails.

Given

```
1. public class Th{  
2.  
3.     private static class CStore implements Runnable{  
4.  
5.         int cps = 10;  
6.  
7.         public void run(){  
8.             while(cps > 0){  
9.                 try{  
10.                     Thread.sleep(1500);  
11.                 }catch(Exception e){  
12.                     System.out.print(e);  
13.                 }  
14.                 synchronized(this){  
15.                     cps -= 1;  
16.                 }  
17.                 System.out.print(cps);  
18.             }  
19.         }  
20.     }  
21.  
22.  
23.     public static void main(String args[]){  
24.         CStore cs = new CStore();  
25.  
26.         Thread th1 = new Thread(cs);  
27.         Thread th2 = new Thread(cs);  
28.         th1.start();  
29.         th2.start();  
30.     }  
31. }
```

Which will be the output?

Please select :

- A. 9876543210
- B. 9876543210-1
- C. 987654321
- D. Never ending loop.
- E. Compilation fails.

Which of the following is true about thread deadlocking scenarios? (choose 2)

Please select :

- A. A static synchronized method and non-static method will always block each other.
- B. Threads calling static synchronized methods in the same class won't lock each other.
- C. When invoking non-static synchronized methods in the same class by using two different instances.
- D. When two threads use nested synchronized blocks to lock two objects and the blocks lock the same objects in different order.
- E. When two threads tried to lock the same object , if one thread calls join method on another thread.

Consider following statements.

- I. It is more efficient to call fork twice for two subprograms and then call join twice.
- II. The order of calling the “compute()” and “join()”is not important.
- III. In some cases fork-join computation might run slower.

Which is true?

Please select :

- A. Only I.
- B. Only II.
- C. Only III.
- D. Only I and II.
- E. Only I and III.

Given

```
1. import java.util.concurrent.atomic.*;
2.
3. public class ATest2{
4.
5.     private AtomicInteger value = new AtomicInteger(10);
6.
7.     public static void main(String []args){
8.         ATest2 at = new ATest2();
9.         //insert here
10.    }
11. }
```

Which, inserted independently at line 9, will compile and print 15 as the output? (Choose 2)

Please select :

- A. at.value = new Integer(15);  
System.out.print(at.value.intValue());
- B. at.value = 15;  
System.out.print(at.value.intValue());
- C. at.value = at.value.getAndAdd(5);  
System.out.print(at.value.intValue());
- D. at.value.weakCompareAndSet(10,15);  
System.out.print(at.value.intValue());
- E. System.out.print(at.value.addAndGet(5));

Which of the following is true? (choose 2)

Please select :

- A. The ReentrantLock was introduced in Java 1.7.
- B. The ReentrantLock and the synchronization provide same functionality and flexibility.
- C. If we are pass the fairness parameter as “true” when creating a new ReentrantLock object, it gives us the guarantee that the longest waiting thread will get the lock next.
- D. Using ReentrantLock, the thread can temporarily release the locks it has when it is going to a wait state.
- E. The constructor for this class accepts an optional time parameter. When set long value, lock will be released by the thread after the specified time.

Consider the following code fragment.

```
1. String [ ]s = new String[2];
2.         s[1] = "B";
3.         ConcurrentHashMap<String, Integer> cmap = new ConcurrentHashMap<String,
   Integer>();
4.             cmap.put("A", new Integer(1));
5.             cmap.put(s[0], new Integer(2));
6.             cmap.put("C", 3);
7.             cmap.putIfAbsent("D", new Integer(4));
8. System.out.print(cmap);
```

Which of the following is true?

Please select :

- A. The output will be {C=3, D=4, A=1, B=2}.
- B. The output will be {D=4, null=2, A=1, C=3}.
- C. This code fragment won't compile due to error on line 7.
- D. This code fragment won't compile due to error on line 6.
- E. An exception will be thrown at the runtime.

Given :

```
1. import java.util.ArrayList;
2.
3. public class Whiz{
4.     public static void main(String[] args) {
5.
6.         ArrayList<Integer> ints = new ArrayList<>();
7.         ints.add(1);
8.         ints.add(2);
9.         ints.add(2);
10.        ints.parallelStream().distinct().peek(System.out::print).count();
11.
12.    }
13. }
```

Please select :

- A. Always 12
- B. Could be 212
- C. Could be 122
- D. Could be 12
- E. Compilation fails.
- F. Exception

Given

```
1. import java.util.stream.Stream;
2.
3. public class Whiz{
4.     public static void main(String[] args) {
5.         Stream<String> sts = Stream.of("1","3","3","4","4").distinct();
6.         String out = sts.collect(StringBuilder::new, StringBuilder::append,
7.             StringBuilder::append).toString();
8.         System.out.println(out);
9.     }
9. }
```

What is the output?

Please select :

- A. 13344
- B. 134
- C. 1
- D. An Exception.
- E. Compilation fails.

Given

```
1. import java.nio.file.*;
2. import java.io.*;
3. import java.nio.file.attribute.*;
4.
5.
6. class NIO{
7.     public static void main(String [] args) throws IOException{
8.         Path path = Paths.get("..\\myfile.txt");
9.         //
10.    }
11. }
```

Note: assume that the myfile.txt actually exists and code runs on a windows flat form and the " myfile.txt" file is already marked as read only and you have enough permission to do any operation on the file.

Which, inserted independently at line 9, will compile and produce the group owner of the file.of the "myfile.txt" as the output?

I.       PosixFileAttributes pa = Files.readAttributes(path, PosixFileAttributes.class);  
System.out.print(pa.group().getName());

II.       PosixFileAttributes pa = Files.readAttributes(path, PosixFileAttributes.class);  
System.out.print(pa.group());

Please select :

- A. Only I and II will cause a compile time error.
- B. Only II and I will cause a compile time error.
- C. Both will cause compile time error
- D. None of above is true

Which is true?

Please select :

- A. We can use `java.nio.file.Files` class methods, to read the attributes of a file but not for setting
- B. We can use the “`setAttributes()`” method of the `java.nio.file.Files` class to set multiple attributes of a file.
- C. We can use the `java.nio.file.FileStore` class to read information about a file store.
- D. We can read any attribute of a file using the `BasicFileAttributesView`.
- E. The “`getKey()`” method of the `BasicFileAttributesView` returns an object that uniquely identifies the given file, or null if a file key is not available.

```

1. import java.io.IOException;
2. import java.nio.file.*;
3. import java.nio.file.attribute.*;
4.
5. public class Mat{
6.
7.     public static void main(String []args)throws IOException{
8.         Path path = Paths.get("I:\\whizlabs");
9.             FileVisitor<Path> searcher = new Search();
10.            Files.walkFileTree(path, searcher);
11.    }
12.    private static final class Search extends SimpleFileVisitor<Path> {
13.
14.        public FileVisitResult visitFile(Path file, BasicFileAttributes bfa)throws IOException {
15.            PathMatcher pm = FileSystems.getDefault().getPathMatcher("glob:*");
16.                if(pm.matches(file.getFileName())){
17.                    System.out.println(file);
18.                }
19.                return FileVisitResult.CONTINUE;
20.            }
21.
22.        }
23.    }

```

Note: assume that the I:\\whizlabs actually exists and contain files and directories and you have enough permission.

Which is the result?

Please select :

- A. Completion succeeds and prints the paths of all the files and the subdirectories which are located in the “whizlabs” directory.
- B. Completion succeeds and prints only the paths of the files of the “whizlabs” directory and its subdirectories.
- C. Completion succeeds and prints only the paths of all the files which are located in the “whizlabs” directory and its subdirectories.
- D. An exception will be thrown at the runtime.
- E. Compilation fails.

Given

```
1. import java.io.IOException;
2. import java.nio.file.Files;
3. import java.nio.file.Path;
4. import java.nio.file.Paths;
5. import java.util.stream.Stream;
6.
7. public class Whiz{
8.     public static void main(String[] args) throws IOException {
9.         Stream<Path> out = Files.find(Paths.get("I:\\whizlabs"),2,
(p,b) -> p.getFileName().toString().startsWith("d"));
10.        System.out.println(out.count());
11.    }
12. }
```

File Structure of the whizlabs directory: [] – directory / () - file

```
[whizlabs] --- (exam.txt)
|-- [data] ---(login.txt)
|-- [files] --- (detail.xls)
```

What is the output?

Please select :

- A. 1
- B. 2
- C. 3
- D. An Exception.
- E. Compilation fails.

Consider following code fragment

```
try (Stream<Path> stream = Files.walk(Paths.get("E:\\whiz"), 0)) {  
    System.out.println(stream.count());  
}  
}catch(IOException e){  
}
```

Directory 'whiz' has following files;

E:\\whiz  
E:\\whiz\\assignment  
E:\\whiz\\assignment\\how to acces.txt  
E:\\whiz\\new.txt  
E:\\whiz\\nio  
E:\\whiz\\nio1\\empty

What is the output?

Please select :

- A. 6
- B. 0
- C. 1
- D. 5
- E. An Exception.

Which of the following method introduced in the java 8 will result a new BufferedReader with the passed Path? (Assume “file” is valid reference to a of a file instance”)

Please select :

- A. Files.bufferedReaderBuilder(file);
- B. Files.BufferedReader(file);
- C. Files.getBufferedReader(file);
- D. Files.newBufferedReader(file.toPath());
- E. None of above.

Given

```
1. import java.io.FileInputStream;
2. //Codes
3.
4. public class Stream{
5.     public static void main(String[] args) throws IOException {
6.
7.         FileInputStream in = null;
8.         FileOutputStream out = null;
9.
10.        try {
11.            in = new FileInputStream("source.txt");
12.            out = new FileOutputStream("dest.txt");
13.            int c;
14.
15.            while ((c = in.read()) != -1) {
16.                out.write(c);
17.            }
18.        } finally {
19.            if (in != null) {
20.                in.close();
21.            }
22.            if (out != null) {
23.                out.close();
24.            }
25.        }
26.    }
27. }
```

Which are true? (Choose 2)

Please select :

- A. We have to import java.io.FileOutputStream and java.io.IOException Compilation to succeed.
- B. Importing only java.io.FileOutputStream is enough Compilation to succeed.
- C. Importing only java.io.IOException is enough Compilation to succeed.
- D. java.io.FileInputStream is enough Compilation to succeed.
- E. Changing import java.io.FileInputStream; to import java.io.\*; will make this code works.

Given

```
1. import java.io.*;
2.
3. public class Stream{
4.     public static void main(String[] args) throws Exception {
5.
6.         File input = new File("source.txt");
7.         File output = new File("dest.txt");
8.         FileReader instream = null;
9.         FileWriter outstream = null;
10.
11.        try {
12.            instream = new FileReader(input);
13.            outstream = new FileWriter(output);
14.            int c;
15.
16.            while ((c = instream.read()) != -1) {
17.                outstream.write(c);
18.            }
19.        } finally {
20.            if (instream != null) {
21.                instream.close();
22.            }
23.            if (outstream != null) {
24.                outstream.close();
25.            }
26.        }
27.    }
28. }
```

Source.txt file contains some texts and there is no file called dest.txt located in that folder when the code is going to compile.

Which is true?

Please select :

- A. Compilation fails as there is no dest.txt file.
- B. Compilation succeeds but no dest.txt file will be created
- C. dest.txt file will be created and it will be empty after running the code.
- D. dest.txt file will be created and it will contain all the texts contained in the source.txt file.
- E. read() method at line 16 returns char value but we assign that char value to int as then it is easy to write data to files.

What is true?

Please select :

- A. We can use print() , println() and write() methods in java.lang.System class for outputting string.
- B. Java.lang.System class has a static field called as “err” of type InputStream.
- C. Java.lang.System class has a static field called as “in” of type PrintStream.
- D. Java.lang.System class has a static field called as “out” of type PrintStream.
- E. None of above

Given

```
1. import java.io.*;
2.
3. class Co{
4.     public static void main(String args[])throws IOException{
5.         byte []b = new byte[10];
6.         InputStream in = System.in;
7.         int i = in.read(b);
8.
9.         for(byte c : b){
10.             System.out.print((char)c);
11.         }
12.         in.close();
13.     }
14. }
```

Which is true?

Please select :

- A. Compilation succeeds and if we enter “abcdefghijklm” the output will be “abcdefghijklm”.
- B. Compilation succeeds and if we enter “abcdefghijklm” the output will be “abcdefghijklij”.
- C. Compilation fails due to error on line 6 as the field “in” in java.lang.System class is not an InputStream.
- D. Compilation fails due to error on line 7 as there is no “read()”method which takes byte array as parameter in InputStream objects.
- E. Compilation fails as we haven’t used try/catch box for InputStreams’ “read()” and “close()” methods.

Consider following stream

```
Stream<String> strings = Stream.of("1","2","3","4");
```

Which of the following will convert it to Stream of Integer wrappers?

Please select :

- A. Stream<Integer> ints = strings.mapToInt(Integer::parseInt);
- B. Stream<Integer> ints = strings.merge(Integer::parseInt);
- C. Stream<Integer> ints = strings.map(Integer::decode);
- D. Stream<Integer> ints = strings.map(Integer::toString);
- E. None of above.

Given

```
1. import java.util.List;
2. import java.util.Map;
3. import java.util.stream.Collectors;
4. import java.util.stream.DoubleStream;
5.
6. public class Whiz{
7.     public static void main(String[] args){
8.         DoubleStream ins = DoubleStream.of(2.5,3.1,5.0);
9.         _____ result = ins.boxed().collect(Collectors.groupingBy(in -> in >=
10.            3.1));
11.     }
12. }
```

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

Please select :

- A. List<List<Double>>
- B. Map<Boolean, Double >
- C. List<Integer>
- D. Map<Boolean,List<Double>>
- E. None of above.

Given

```
1. import java.util.stream.IntStream;  
2.  
3. public class Whiz{  
4.     public static void main(String[] args){  
5.         IntStream in = IntStream.of(3,8,4,1,0,7,2).sorted();  
6.         System.out.println(in.limit(2).findAny(p -> p>3));  
7.     }  
8. }
```

What is the output?

Please select :

- A. 4
- B. 8
- C. 3
- D. An Exception.
- E. Compilation fails.

Given

```
1. import java.util.stream.Stream;
2.
3. public class Whiz{
4.     public static void main(String[] args){
5.         Stream<Integer> stream = Stream.of(5,8,1,3,5);
6.
7.         stream.peek(System.out::print).sorted().distinct().forEach(System.out::print);
8.     }
}
```

How many times '5' will appear in the output?

Please select :

- A. 1
- B. 2
- C. 3
- D. An Exception.
- E. Compilation fails.

Given

```
1. import java.util.stream.Stream;
2.
3. public class Whiz{
4.     public static void main(String[] args){
5.         Stream<Integer> stream = Stream.of(5,8,1,3,5);
6.         System.out.println(stream.max() + stream.min());
7.     }
8. }
```

What is the output?

Please select :

- A. 6
- B. 5
- C. 1
- D. An Exception.
- E. Compilation fails.

Given

```
1. import java.util.Optional;  
2.  
3. public class Whiz{  
4.     public static void main(String[] args){  
5.         Optional<Integer> ops = Optional.of(12);  
6.         ops.filter(p -> p>12).ifPresent(System.out::print);  
7.     }  
8. }
```

What is the output?

Please select :

- A. 12
- B. Optional.empty
- C. No output
- D. An Exception.
- E. Compilation fails.

Consider following statement.

```
Optional<String> op = Optional.of("10");
```

Which of the following can be used to convert above to an optional of integer?

Please select :

- A. op.flatMap(s -> Optional.of(Integer.decode(s)));
- B. op.map(s -> Optional.of(Integer.decode(s)));
- C. op.flatMap(s -> Integer.decode(s));
- D. op.map(s -> Integer.decode(s));
- E. None of above.

Given

```
1. import java.util.ArrayList;
2. import java.util.Collections;
3. import java.util.List;
4.
5. public class Whiz{
6.         public static void main(String[] args){
7.
8.                 List<String> names = new ArrayList<>();
9.                 names.add("Livera");
10.                names.add("Sanka");
11.                names.add("Sharma");
12.
13.                Collections.sort(names, String.CASE_INSENSITIVE_ORDER.reversed());
14.
15.                System.out.println(names);
16.        }
17. }
```

What is the output?

Please select :

- A. [Sharma, Sanka, Livera]
- B. [Livera, Sanka, Sharma]
- C. []
- D. Compilation fails due to error at line 13.
- E. Compilation fails due to multiple errors.

Given

```
1. import java.util.function.Function;  
2.  
3. public class Whiz {  
4.     public static void main(String [ ] args){  
5.         Function<Integer,Double> fun = (s) -> s.doubleValue()*2;  
6.         System.out.println(fun.compose((Integer d) -> d+2).apply(10));  
7.     }  
8. }
```

What is the output?

Please select :

- A. 28
- B. 22.0
- C. 24.0
- D. An Exception is thrown.
- E. Compilation fails.

Given

```
1. import java.util.ArrayList;
2. import java.util.List;
3. import java.util.function.UnaryOperator;
4.
5. public class Whiz{
6.     public static void main(String[] args) {
7.
8.         List<String> list = new ArrayList<>();
9.         list.add("1");
10.        list.add("2");
11.        list.add("3");
12.        UnaryOperator<String> un = UnaryOperator.identity();
13.        list.forEach(s -> un.apply(s));
14.        System.out.println(list);
15.    }
16. }
```

What is the output?

Please select :

- A. [1, 2, 3]
- B. []
- C. An Exception is thrown.
- D. Compilation fails due to error at line 12.
- E. Compilation fails due to multiple errors

Given

```
1. import java.util.function.UnaryOperator;  
2.  
3. public class Whiz{  
4.     public static void main(String[] args) {  
5.         UnaryOperator<String> s = UnaryOperator.identity();  
6.         System.out.println(s.apply("2"));  
7.     }  
8. }
```

What is the output?

Please select :

- A. null
- B. 2
- C. No output.
- D. Compilation fails due to line 5.
- E. Compilation fails due to line 6

Given

```
1. import java.util.function.Function;  
2.  
3. public class Whiz{  
4.     public static void main(String [ ] args){  
5.         Function<Double,Integer> func = (s) -> s.intValue();  
6.         System.out.println(func.andThen(d -> d*2).apply(12.5));  
7.     }  
8. }
```

What is the output?

Please select :

- A. 24
- B. 25
- C. 25.0
- D. An Exception is thrown.
- E. Compilation fails.

Given

```
1. import java.util.function.IntConsumer;
2. import java.util.stream.IntStream;
3.
4. public class Whiz{
5.     public static void main(String [ ] args){
6.         Consumer<Integer> con = System.out::print;
7.         IntStream ins = IntStream.of(1,3,2,4,3).distinct();
8.         ins.forEach(con);
9.     }
10. }
```

What is the output?

Please select :

- A. 13243
- B. 1234
- C. 3
- D. An Exception.
- E. Compilation fails.

Which of the following is true about the java inbuilt Supplier interfaces?

Please select :

- A. Its' functional method name is supply.
- B. It has two default methods called compose and andThen
- C. It hasn't got any static method.
- D. It has binary versions such as IntSupplier
- E. None of above.

Given

```
1. import java.util.function.Consumer;
2. import java.util.function.Predicate;
3. import java.util.stream.Stream;
4.
5. public class Whiz{
6.     public static void main(String[] args) {
7.         Stream<Integer> stream = Stream.of(2,5,6,8);
8.         Predicate<Integer> lp = (l) -> l>5;
9.         Consumer<Integer> ins = (in) -> {if(lp.negate().test(in))
10.           System.out.print(in);}
11.       }
12. }
```

What is the output?

Please select :

- A. 2
- B. 25
- C. 68
- D. An exception is thrown.
- E. Compilation fails.

Given

```
1. import java.util.function.Predicate;
2. import java.util.stream.IntStream;
3.
4. public class Whiz{
5.     public static void main(String[] args){
6.         IntStream stream = IntStream.of(2,5,6,8);
7.         Predicate<Integer> pre = Predicate.isEqual(6);
8.         stream.boxed().filter(pre).forEach(System.out::print);
9.     }
10. }
```

What is the output?

Please select :

- A. 2568
- B. 258
- C. 6
- D. An Exception.
- E. Compilation fails

Given

```
1. import java.util.function.IntSupplier;  
2.  
3. public class Whiz{  
4.     public static void main(String [ ] args){  
5.         IntSupplier in = () -> Integer.compare(1, 2);  
6.         System.out.println(get(in));  
7.     }  
8.  
9.     public static int get(IntSupplier t){  
10.         return t.get();  
11.     }  
12. }
```

What is the output?

Please select :

- A. 0
- B. -1
- C. 1
- D. Compilation fails due to error at line 5.
- E. Compilation fails due to error at line 10.

Which of the following is valid IntFunction declaration?

Please select :

- A. IntFunction<String> in = Integer::parseInt;
- B. IntFunction<String> in = in -> in\*2;
- C. IntFunction<String> in = () -> 10;
- D. IntFunction<String> in = Integer::toString;
- E. None of above.

Which of the following is not primitive or binary variation of the `java.util.function` interface?

Please select :

- A. UnaryOperator
- B. IntUnaryOperator
- C. ToLongFunction
- D. LongFunction
- E. None of above.

Which can be taken as the correct ways of creating Locale object for the English language? (Choose 3)

*Hint: Code for English Language ("en") .*

Please select :

- A. `Locale loc = Locale.forLanguageTag("en-US ")`
- B. `Locale loc = new Locale.Builder().setLanguage("en");`
- C. `Locale loc = new Locale("en" , "US") ;`
- D. `Locale loc = Locale.English;`
- E. `Locale loc = Locale.ENGLISH ;`

Which of the following is true about the `java.util.ResourceBundle`?

Please select :

- A. We can't instantiate it by using the keyword new as the `ResourceBundle` is an interface.
- B. It is not possible to create applications which can handle multiple locales at ones by using the `java.util.ResourceBundle`
- C. The `ResourceBundle` is a non-abstract class.
- D. For selecting appropriate Locale we should call the `ResourceBundle`'s “`getBundle()`” method.
- E. None of above.

If you are asked to create a Locale object. The country should be United States and Language should be English. Which of the following ways you will choose?

NOTE: Code should be run on both java 1.6 and 1.8 versions.

Please select :

- A. `Locale loc = new Locale("en" , "US");`
- B. `Locale loc = new Locale("US" , "en");`
- C. `Locale loc = new Locale.Builder().setLanguage("en").setRegion("US").build();`
- D. `Locale loc = new Locale.Builder().setLanguage("EN"). build();`
- E. `Locale loc = Locale.forLanguageTag("en-US");`