2. Given:

```
package test.t1;
public class A {
    public int x = 42;
    protected A() {}                    // line 1
}
```

and

```
package test.t2;
import test.t1.*;
public class B extends A {
    int x = 17;                         // line 2
    public B() { super(); }             // line 3
}
```

and

```
package test;
import test.t1.*;
import test.t2.*;
public class Tester {
    public static void main(String[] args) {
        A obj = new B();                // line 4
        System.out.println(obj.x); // line 5
    }
}
```

What is the result?

- The compilation fails due to an error in line 3.
- The compilation fails due to an error in line 5.
- The compilation fails due to an error in line 2.
- 17
- 42
- The compilation fails due to an error in line 4.

1. Given:

```
public class Person {
    private String name = "Green";
    public void setName(String name) {
        String title = "Mr. ";
        this.name = title + name;
    }
    public String toString() {
        return name;
    }
}
```

and

```
public class Test {
    public static void main(String args[]) {
        Person p = new Person();
        p.setName("Blue");
        System.out.println(p);
    }
}
```

What is the result?

- Mr. Green
- Green
- Mr. Blue
- An exception is thrown at runtime.

2. Given:

```
package test.t1;
public class A {
    public int x = 42;
    protected A() {}                    // line 1
}
```

and

```
package test.t2;
import test.t1.*;
public class B extends A {
    int x = 17;                         // line 2
    public B() { super(); }             // line 3
}
```

and

```
package test;
import test.t1.*;
import test.t2.*;
public class Tester {
    public static void main(String[] args) {
        A obj = new B();                // line 4
        System.out.println(obj.x); // line 5
    }
}
```

What is the result?

- The compilation fails due to an error in line 3.
- The compilation fails due to an error in line 5.
- The compilation fails due to an error in line 2.
- 17
- 42
- The compilation fails due to an error in line 4.

Mark for Review

3. Given the content from the courses.txt file:

```
123:Java:1
124:MySQL:2
125:Java Server Pages: 3
```

Given the code fragment:

```
Path filePath = Paths.get("course.txt");
try {
/* line 1 */
} catch (IOException ex) {
System.out.format("File IO Exception is thrown.", ex);
}
```

Which code fragment at line 1 prints the lines that contain Java from the course.txt file?

- Files.lines(filePath).map(s ->
  s.contains("Java")).forEach(System.out::println);

  return boolean true,false

- List<String> lines2 = Files.readAllLines(filePath).filter(s ->
  s.contains("Java"));
  for (String line : lines2) {
  System.out.println(line);
  }

  b can't be true coz readAllLines will return list so filter methpd won't work

  b and c will return boolean

- List<String> lines1 =
  Files.readAllLines(filePath).contains("Java");
  for (String line : lines2) {
  System.out.println(line); }

- Files.lines(filePath).filter(s ->
  s.contains("Java")).forEach(System.out::println);

- System.out.println(Files.readString(filePath).contains("Java"));

  return one true

1. Given:

```
public class Person {
    private String name = "Green";
    public void setName(String name) {
        String title = "Mr. ";
        this.name = title + name;
    }
    public String toString() {
        return name;
    }
}
```

and

```
public class Test {
    public static void main(String args[]) {
        Person p = new Person();
        p.setName("Blue");
        System.out.println(p);
    }
}
```

What is the result?

- Mr. Green
- Green
- Mr. Blue
- An exception is thrown at runtime.

Time Remaining 01:26:19

5. Given the code fragment:

```
int x = 0;
while(x < 10){
    System.out.print(x++);
}
```

Which "for" loop produces the same output?

- for(int c = 0; ; c++){
  System.out.print(c);
  if(c == 10){
  break;
  }
  }
- for(a; a < 10; a++){
  System.out.print(a);
  }
- int b = 0;
  for(; b < 10;){
  System.out.print(++b);
  }
- for(int d = 0; d < 10;){
  System.out.print(d);
  ++d;
  }

Previous Page

Answer the question(s) on this page, and click Next to go to the next test page. Click Summary to see which questions you ne
answer before submitting the test. Click Finish Test if you are ready to submit your test.

Time Remaining 01:25:22

7. A company has an existing Java app that includes two Java 8 jar files, `sales-8.10.jar` and `clients-10.2.jar`.

Mark

The jar file, `sales-8.10.jar`, references packages in `clients-10.2.jar`, but `clients-10.2.jar` does not reference packages in `sales-8.10.jar`.

They have decided to modularize `clients-10.2.jar`.

Which `module-info.java` file would work for the new library version `clients-10.3.jar`?

```
module com.company.clients {
    exports com.company.clients;
}
```

```
module com.company.clients{
    requires com.company.clients;
}
```

so in export we give package name, not absolute class or interface name

```
module com.company.clients{
    uses com.company.clients;
}
```

```
module com.company.clients {
    exports com.company.clients.Client;
}
```

Test: 819 - Java SE 11 Developer

Answer the question(s) on this page, and click Next to g
answer before submitting the test. Click Finish Test if yo

Time Remaining 01:27:38

4. Given:

```
public class A {
    int a = 0;
    int b = 0;
    int c = 0;
    public void foo(int i) {
        a += b * i;
        c -= b * i;
    }
    public void setB(int i) {
        b = i;
    }
}
```

Which makes `class A` thread safe?

- Make `foo` and `setB` synchronized.
- Class `A` is thread safe.
- Make `A` synchronized.
- Make `foo` synchronized.
- Make `setB` synchronized.

Answer the question(s) on this page, and click Next to go to the next test page. Click Summary to see which questions yo
answer before submitting the test. Click Finish Test if you are ready to submit your test.

Time Remaining 01:18:40

11. Given:

```
5. IntStream str = IntStream.of(2, 3, 4);
6. IntFunction<Integer> func = x -> y -> x * y;
7. str.map(func.apply(10)).forEach(System.out::println);
```

Which action will enable the code to compile?

- Replace line 6 with `Function<UnaryOperator> func = x -> y -> x * y;`
- Replace line 6 with `IntFunction<UnaryOperator> func = x -> y -> x * y;`
- Replace line 6 with `IntFunction<IntUnaryOperator> func = x -> y -> x * y;`
- Replace line 6 with `BiFunction<Integer> func = x -> y -> x * y;`

45. Given:

```
public class Main {
   public static void main(String[] args) {
     List<Player> players = List.of(new Player("Scott", 115), new Player("John", 70), new
Player("Jelly", 105));
     double average = // line 1
     System.out.println("The average is: " + average);
   }
}

class Player {
  public String name;
  public int score;
  public Player(String name, int score) {
    this.name = name;
    this.score = score;
  }
}
```

You want to calculate the average of the Player's score.

Which statement inserted on line 1 will accomplish this?

- players.stream().mapToInt(a -> a.score).average().orElse(0.0);
- players.stream().mapToDouble(a -> a.score).average();
- players.stream().map(a -> a.score).average();
- players.stream().average().orElse(0.0);

Answer the question(s) on this page, and click Next to go to the next test page
answer before submitting the test. Click Finish Test if you are ready to submit

Time Remaining 01:25:05

8. Given:

```
class MyType<T> {
  private T value;
  public T getValue() {
    return value;
  }
  public void setValue(T value) {
    this.value = value;
  }
}
```

and

```
public class Test {
  public static void main(String... args) {
    MyType<String> strType = new MyType<>();
    MyType<? extends Number> type = new MyType<>();
    strType.setValue("test");
    type.setValue(null);
    System.out.println(strType.getValue() + ":" + type.getValue
  }
}
```

What is the result ?

- `test:null`
- `test:0`
- An `Exception` is thrown at runtime.
- The compilation fails.
- `null:null`

Test: 819 - Java SE 11 Developer

Answer the question(s) on this page, and click Next to go to the next test page
answer before submitting the test. Click Finish Test if you are ready to submit

Time Remaining 01:17:24

49. Given:

```
1. interface Pastry {
2.   void getIngredients();
3. }
4. abstract class Cookie implements Pastry {}
5.
6. class ChocolateCookie implements Cookie {
7.   public void getIngredients() {}
8. }
9. class CoconutChocolateCookie extends ChocolateCookie {
10.   void getIngredients(int x) {}
11. }
```

Which is true?

- The compilation fails due to an error in line 4.
- The compilation fails due to an error in line 10.
- The compilation fails due to an error in line 9.
- The compilation fails due to an error in line 6.
- The compilation succeeds.
- The compilation fails due to an error in line 7.
- The compilation fails due to an error in line 2.

50. Given:

```
public enum Status {
   BRONZE(5),SILVER(10),GOLD(15);
   private int rate;
   private Status(int rate) {
      this.rate = rate;
   }
   public int getRate() { return rate; }
   public Status addStatus(int rate) {
       return new Status(20);
   }
}
```

and

```
public class Test {
   public static void main(String[] args) {
      Status silver = Status.SILVER;
      System.out.println(silver+silver.getRate());
      Status platinum = Status.addStatus(20);
      System.out.println(platinum+platinum.getRate());
   }
}
```

What is the result?

- SILVER10
  platinum20
- SILVER10
  PLATINUM20
- The compilation fails.
- An exception is thrown at runtime.
- SILVER10
  20

Answer the question(s) on this page, and click Next to go to the next test page. Click Summary to see your answer before submitting the test. Click Finish Test if you are ready to submit your test.

Time Remaining 01:17:38

47. Given the code fragment:

```
Integer i = 11;
```

Which two statements compile?

- [ ] `Double b = Double.valueOf(i);`
- [ ] `double e = Double.parseDouble(i);`
- [ ] `Double a = i;`
- [ ] `double d = i;`
- [ ] `Double c = (Double) i;`

Answer the question(s) on this page, and click Next to go to th
answer before submitting the test. Click Finish Test if you are r

Time Remaining 01:14:32

48. Given the code fragment:

```
8. public class Test {
9.   private final int x = 1;
10.   static final int y;
11.   public Test() {
12.      System.out.print(x);
13.      System.out.print(y);
14.   }
15.   public static void main(String args[]) {
16.      new Test();
17.   }
18. }
```

What is the result?

- The compilation fails at line 16.
- 10
- The compilation fails at line 9.
- The compilation fails at line 13.
- 1

Answer the question(s) on this page, and click Next to
answer before submitting the test. Click Finish Test if y

Time Remaining 01:14:23

47. Given the code fragment:

```
Integer i = 11;
```

Which two statements compile?

- [ ] `Double b = Double.valueOf(i);`
- [ ] `double e = Double.parseDouble(i);`
- [ ] `Double a = i;`
- [ ] `double d = i;`
- [ ] `Double c = (Double) i;`

Previous

Answer the question(s) on this page, and click Next to go to the next test page. Click Summary to see which questions you need answer before submitting the test. Click Finish Test if you are ready to submit your test.

Time Remaining 01:21:23

Mark for

9. Your organization provides a cloud server to your customer to run their Java code. You are reviewing the changes for the next release and you see this change in one of the config files:

```
old: JAVA_OPTS="$JAVA_OPTS -Xms8g -Xmx8g"
new: JAVA_OPTS="$JAVA_OPTS -Xms8g -Xmx8g -noverify"
```

Which is correct?

- You reject the change because `-Xms8g -Xmx8g` uses too much system memory.
- You accept the change because `-noverify` is a standard option that has been supported since Java 1.0.
- You reject the change because `-noverify` is a critical security risk.
- You accept the change because `-noverify` is necessary for your code to run with the latest version of Java.

```
interface MyInterface1 {
    public int method() throws Exception;
    private void pMethod() { /* an implementation of pMethod */ }
}
interface MyInterface2 {
    public static void sMethod() { /* an implementation of sMethod */ }
    public boolean equals();
}
interface MyInterface3 {
    public void method();
    public void method(String str);
}
interface MyInterface4 {
    public void dMethod() { /* an implementation of dMethod */ }
    public void method();
}
interface MyInterface5 {
    public static void sMethod();
    public void method(String str);
}
```

Which two interfaces can be used in lambda expressions?

- [ ] MyInterface2
- [ ] MyInterface1
- [ ] MyInterface3
- [ ] MyInterface4
- [ ] MyInterface5

45. Given:

Mark for Review

```
public class Main {
  public static void main(String[] args) {
    List<Player> players = List.of(new Player("Scott", 115), new Player("John", 70), new
Player("Jelly", 105));
    double average = // line 1
    System.out.println("The average is: " + average);
  }
}

class Player {
  public String name;
  public int score;
  public Player(String name, int score) {
    this.name = name;
    this.score = score;
  }
}
```

You want to calculate the average of the Player's score.

Which statement inserted on line 1 will accomplish this?

- `players.stream().mapToInt(a -> a.score).average().orElse(0.0);`
- `players.stream().mapToDouble(a -> a.score).average();`
- `players.stream().map(a -> a.score).average();`
- `players.stream().average().orElse(0.0);`

Answer the question(s) on this page, and click Next to g
answer before submitting the test. Click Finish Test if yo

Time Remaining 01:14:16

46. Given:

```
public class Tester {
  public static void main(String[] args) {
    float x = 2, y = 4, z = 4;
    float a = y / x, b = y / z;
    if (a > b) {
      System.out.println(a + b);
    }
  }
}
```

What is the result?

- The program prints nothing.
- An exception is thrown at runtime.
- 3.0
- 2.0
- 1.0

Previous

Time Remaining 01:07:29

17. Given:

```
public interface Converter {
    public static final double POUNDS_PER_KILOGRAM = 2.20462; //
line 1
    public double tare();
    public double net();
    public default double gross() {                              //
line 2
        return tare() + net();
    }
    public default double tare(String units) {
        return toUnit(tare(), units);
    }
    public default double net(String units) {
        return toUnit(net(), units);
    }
    public default double gross(String units) {
        return toUnit(gross(), units);
    }
    private static double toUnit(double kilograms, String unit)
{  // line 3
        switch (unit) {
            case "KILO": return kilograms;
            case "POUND": return kilograms * POUNDS_PER_KILOGRAM;
            default: throw new IllegalArgumentException();
        }
    }
}
```

Which is true?

- Line 3 is the first line to cause a compilation error.
- Line 2 is the first line to cause a compilation error.
- Line 1 is the first line to cause a compilation error.
- It compiles without errors.

Answer the question(s) on this page.
answer before submitting the test. Click Finish Test when you are done.

Time Remaining 01:05:51

43. Given:

```
Path p1 = Paths.get("/scratch/exam/topsecret/answers");
Path p2 = Paths.get("/scratch/exam/answers/temp.txt");
Path p3 = Paths.get("/scratch/answers/topsecret");
```

Which two statements print ..\..\..\answers\topsecret?

- [ ] `System.out.print(p1.relativize(p3));`
- [ ] `System.out.print(p3.relativize(p1));`
- [ ] `System.out.print(p2.relativize(p3));`
- [ ] `System.out.print(p1.relativize(p2));`
- [ ] `System.out.print(p2.relativize(p1));`
- [ ] `System.out.print(p3.relativize(p2));`

Answer the question(s) on this page, and click Next to go to the next test page. Click Summ
answer before submitting the test. Click Finish Test if you are ready to submit your test.

Time Remaining 01:08:34

44. Given the code fragment:

```
public class Main {
   public static void main(String[] args) {
      List<String> fruits = List.of("banana", "orange", "apple", "lemon");
      Stream<String> s1 = fruits.stream();
      Stream<String> s2 = s1.peek(i ->  System.out.print(i + " "));
      System.out.println("-----");
      Stream<String> s3 = s2.sorted();
      Stream s4 = s3.peek(i ->  System.out.print(i + " "));
      System.out.println("-----");
      String strFruits = s4.collect(Collectors.joining(","));
   }
}
```

What is the output?

- `-----`
  `-----`
- `banana orange apple lemon`
  `-----`
- `apple banana lemon orange`
  `-----`
- `-----`
  `-----`
  `banana orange apple lemon apple banana lemon orange`
- `-----`
  `banana orange apple lemon`
  `-----` apple banana lemon orange
- `banana orange apple lemon apple banana lemon orange`
  `-----`
  `-----`

27. Given:

```
<ListInteger> numbers = List.of(2, 3, 0, 8, 1, 9, 5, 7, 6, 4);
int sum = numbers.stream().reduce(0, (n, m) -> n + m);  // line 1
```

You want to make the reduction operation parallelized.

Which two modifications will accomplish this?

- [ ] Replace line 1 with `int sum = numbers.parallel().stream().reduce(0, (n, m) -> n + m);`.
- [ ] Replace line 1 with `int sum = numbers.parallelStream().reduce(0, (n, m) -> n + m);`.
- [ ] Replace line 1 with `int sum = numbers.stream().iterate(0, a -> a+1).reduce(0, (n, m) -> n + m);`.
- [ ] Replace line 1 with `int sum = numbers.stream().parallel().reduce(0, (n, m) -> n + m);`.
- [ ] Replace line 1 with `int sum = numbers.stream().flatMap(a -> a).reduce(0, (n, m) -> n + m);`.

48. Given the code fragment:

```
8. public class Test {
9.    private final int x = 1;
10.    static final int y;
11.    public Test() {
12.       System.out.print(x);
13.       System.out.print(y);
14.    }
15.    public static void main(String args[]) {
16.       new Test();
17.    }
18. }
```

What is the result?

- The compilation fails at line 16.
- 10
- The compilation fails at line 9.
- The compilation fails at line 13.
- 1

```
public class Color {
  String hue;
  int value;
  public Color(String hue, int value) {
    this.hue = hue;
    this.value = value;
  }
  public String toString() {
    return this.hue + ":" + this.value;
  }
  public static void main(String[] args) {
    List clrs = List.of(new Color("Red", 100),
                        new Color("Yellow", 50),
                        new Color("Red", 75),
                        new Color("Yellow", 75));
    Comparator hueSrtr = (h1, h2) -> h1.hue.compareTo(h2.hue);
    Comparator valueSrtr = (h1, h2) -> { if (h1.value >= h2.value) {
                                           return 1;
                                         } else {
                                           return -1;
                                         }
                                       };
    clrs.sort(hueSrtr.thenComparing(valueSrtr));
    System.out.println(clrs);
  }
}
```

we can't sort List.of directly but we can sort in scenerio
new ArrayList<>(List.of(1,2,3));

What is the result?

- [Red:75, Red:100, Yellow:50, Yellow:75]
- [Yellow:75, Yellow:50,Red:100, Red:75]
- [Yellow:50, Yellow:75,Red:75, Red:100]
- An Exception is thrown at runtime.
- [Red:100, Red:75, Yellow:75, Yellow:50]

24. Given:

```
public class Foo {
    public void foo(Collection arg) {
        System.out.println("Bonjour le monde!");
    }
}
```

and

```
public class Bar extends Foo {
    public void foo(List arg) {
        System.out.println("Hello world!");
    }
    public static void main(String... args) {
        List<String> li = new ArrayList<>();
        Collection<String> co = li;
        Bar b = new Bar();
        b.foo(li);
        b.foo(co);
    }
}
```

What is the output?

- Bonjour le monde!
  Hello world!
- Hello world!
  Bonjour le monde!
- Bonjour le monde!
  Bonjour le monde!
- Hello world!
  Hello world!

Answer the question(s) on this page.
answer before submitting the test. Click Finish Test when you are done.

Time Remaining 01:05:51

43. Given:

```
Path p1 = Paths.get("/scratch/exam/topsecret/answers");
Path p2 = Paths.get("/scratch/exam/answers/temp.txt");
Path p3 = Paths.get("/scratch/answers/topsecret");
```

Which two statements print ..\..\..\answers\topsecret?

- [ ] `System.out.print(p1.relativize(p3));`
- [ ] `System.out.print(p3.relativize(p1));`
- [ ] `System.out.print(p2.relativize(p3));`
- [ ] `System.out.print(p1.relativize(p2));`
- [ ] `System.out.print(p2.relativize(p1));`
- [ ] `System.out.print(p3.relativize(p2));`

19. Given the code fragment:

```
module citizen {
    exports com.name to greeting;
}
```

and

```
module greeting {
}
```

Doubt

Which statement is true?

- `public` members in the `com.name` package are accessible only to the `greeting` module.
- All members of `com.name` are accessible only to the `citizen` and `greeting` modules.
- All members in the `com.name` package are accessible only to the `greeting` module.
- Inserting "`requires citizen;`" at `greeting`'s module-info.java, enables `com.name` members accessible to the `greeting` module.
- `public` members in the `com.name` package are accessible to all modules.

Answer the question(s) on this page, and click Next to go to the next test page. Click Sum
answer before submitting the test. Click Finish Test if you are ready to submit your test.

Time Remaining 00:56:06

32. Given the code fragment:

```
StringBuilder txt1 = new StringBuilder("PPQRRRSTT");
int i = 0;
a:
while (i < txt1.length()) {
   char x = txt1.charAt(i);
   int j = 0;
   i++;
   b:
   while (j < txt1.length()) {
      char y = txt1.charAt(j);
      if (i != j && y == x) {
         txt1.deleteCharAt(j);
         // line 1
      }
      j++;
   }
}
System.out.println(txt1);
```

Which two statements inserted independently at line 1 enable this code to print PRRT?

- [ ] `continue a;`
- [ ] `i--;`
- [ ] `j--;`
- [ ] `continue b;`
- [ ] `break b;`
- [ ] `break a;`

27. Given:

Mark for Review

```
<ListInteger> numbers = List.of(2, 3, 0, 8, 1, 9, 5, 7, 6, 4);
int sum = numbers.stream().reduce(0, (n, m) -> n + m);  // line 1
```

You want to make the reduction operation parallelized.

Which two modifications will accomplish this?

- [ ] Replace line 1 with `int sum = numbers.parallel().stream().reduce(0, (n, m) -> n + m);`.
- [ ] Replace line 1 with `int sum = numbers.parallelStream().reduce(0, (n, m) -> n + m);`.
- [ ] Replace line 1 with `int sum = numbers.stream().iterate(0, a -> a+1).reduce(0, (n, m) -> n + m);`.
- [ ] Replace line 1 with `int sum = numbers.stream().parallel().reduce(0, (n, m) -> n + m);`.
- [ ] Replace line 1 with `int sum = numbers.stream().flatMap(a -> a).reduce(0, (n, m) -> n + m);`.

answer before submitting the test. Click Finish Test if you are ready to submit your test.

Time Remaining 00:56:50

23. Given:

```
public class App {
    // line 1
    public static void main(String[] args) {
        new App().new Greeting().greet("Joe");
    }
}
```

Which code fragment added to line 1 enables the code to compile and print `Hello Joe`?

```
class Greeting {
    private void greet(String name) {
        System.out.println("Hello " + name);
    }
}
```

```
class Greeting {
    public static void greet(String s) {
        System.out.println("Hello "+ s);
    }
}
```

in java 11, we can't declare static method inside non static classes

```
static class Greeting {
    public void greet(String name) {
        System.out.println("Hello " + name);
    }
}
```

```
interface Greeting {
    public default void greet(String name) {
        System.out.println(greet+name);
    }
}
```

Previous | Page 23 of 50 | Next | Summary | Finish

28. Given:

Mark for Review

```
public class Employee {
      private String name;
      private String neighborhood;
      private int salary;
      // Constructors and setter and getter methods go here
}
```

and the code fragment:

```
List roster = new ArrayList<>();
Predicate p = e -> e.getSalary() > 30;
Function> f =
    e -> Optional.ofNullable(e.getNeighborhood());
```

Which two `Map` objects group all employees with a salary greater than 30 by neighborhood?

- [ ]
```
Map<String, List<Employee>> r2 = roster.stream().filter(p)
    .collect(Collectors.groupingBy(f, Employee::getNeighborhood));
```

- [ ]
```
Map<Optional<String>, List<Employee>> r5 = roster.stream()
    .collect(Collectors.groupingBy(Employee::getNeighborhood,
        Collectors.filtering(p, Collectors.toList())));));
```

- [ ]
```
Map<Optional<String>, List<Employee>> r4 = roster.stream()
    .collect(Collectors.groupingBy(f, Collectors.filtering(p,
Collectors.toList())));
```

- [ ]
```
Map<String, List<Employee>> r1 = roster.stream()
    .collect(Collectors.groupingBy(Employee::getNeighborhood,
        Collectors.filtering(p, Collectors.toList())));
```

- [ ]
```
Map<Optional<String>, List<Employee>> r3 = roster.stream().filter(p)
    .collect(Collectors.groupingBy(p));
```

answer before submitting the test. Click Finish Test if you are ready to submit your test.

Time Remaining 00:46:35

Mark for Review

31. Given `TripleThis.java`:

```
6.  import java.util.function.*;
7.  public class TripleThis {
8.      public static void main(String[] args) {
9.          Function tripler = x -> { return (Integer) x * 3; };
10.         TripleThis.printValue(tripler, 4);
11.     }
12.     public static void printValue(Function f, T num) {
13.         System.out.println(f.apply(num));
14.     }
15. }
```

Compiling `TripleThis.java` gives this compiler warning:

```
Note: TripleThis.java uses unchecked or unsafe operations.
```

Which two replacements remove this compiler warning and prints `12`?

- [ ] Replace line 9 with `Function tripler = x -> { return x * 3; }`
- [ ] Replace line 12 with `public static void printValue(Function f, int num) {`
- [ ] Replace line 12 with `public static void printValue(Function f, T num) {`
- [ ] Replace line 12 with `public static void printValue(Function f, Integer num) {`
- [ ] Replace line 9 with `Function tripler = x -> { return x * 3; }`
- [ ] Replace line 9 with `Function tripler = x -> { return (Integer) x * 3; }`

answer before submitting the test. Click Finish Test if you are ready to

Time Remaining 00:50:17

29. Given:

```
public class Tester {
    private int x;
    private static int y;
    public static void main(String[] args) {
        Tester t1 = new Tester();
        t1.x = 2;
        Tester.y = 3;
        Tester t2 = new Tester();
        t2.x = 4;
        t2.y = 5;
        System.out.println(t1.x+","+t1.y);
        System.out.println(t2.x+","+Tester.y);
        System.out.println(t2.x+","+t1.y);
    }
}
```

What is the result?

- 2,3
  4,5
  4,5
- 2,3
  4,5
  4,5
- 2,5
  4,5
  4,5
- 2,3
  4,3
  4,5

answer before submitting the test. Click Finish Test if you are ready to submit your test.

Time Remaining 00:46:16

34. Given:

```
public class ResourceTest {
   public static void main(String[] args){
     final MyResource res1 = new MyResource();
     MyResource res2 = new MyResource();
     try(res1 ; res2) {
       // do something
     } catch(Exception e) {}
   }
   static class MyResource implements AutoCloseable {
     public void close() throws Exception { }
   }
}
```

Which statement is true?

- The code fails to compile as `res2` should be declared as `final`.
- The code fails to compile as `MyResource` must implement `Closeable`.
- The code compiles successfully.
- The code fails to compile as `try-with-resource` needs a variable declaration such as `MyResource r1 = res1; MyResource r2 = res2;`.

Test: 819 - Java SE 11 Developer

Answer the question(s) on this page, and click Next to go to the nex
answer before submitting the test. Click Finish Test if you are ready

Time Remaining 00:46:07

35. Given:

```
public class Tester {
  public static void main(String[] args) {
    String s = "hat at store";
    int x = s.indexOf("at");
    s.substring(x + 3);
    x = s.indexOf("at");
    System.out.println(s + " " + x);
  }
}
```

What is the result?

- `hat at store 1`
- `at once 1`
- An `IndexOutOfBoundsException` is thrown at runtime.
- `at once 0`
- `hat at store 4`

Test: 819 - Java SE 11 Developer

Answer the question(s) on this page, and click Next to go to the nex
answer before submitting the test. Click Finish Test if you are ready

Time Remaining 00:46:24

33. Given the code fragment:

```
public class Test {
  class L extends Exception { }
  class M extends L { }
  class N extends RuntimeException { }
  public void p() throws L { throw new M(); }
  public void q() throws N { throw new N(); }
  public static void main(String[] args) {
    try {
      Test t = new Test();
      t.p();
      t.q();
    } /* line 1 */ {
      System.out.println("Exception caught");
    }
  }
}
```

What change on line 1 will make this code compile?

- Add `catch(M | L e)`
- Add `catch(N | L | M e)`
- Add `catch(L | N e)`
- Add `catch(L | M | N e)`
- Add `catch(L e)`

Previous Page 33 of 50 Next

Mark for Review

28. Given:

```
public class Employee {
      private String name;
      private String neighborhood;
      private int salary;
      // Constructors and setter and getter methods go here
}
```

and the code fragment:

```
List roster = new ArrayList<>();
Predicate p = e -> e.getSalary() > 30;
Function> f =
    e -> Optional.ofNullable(e.getNeighborhood());
```

Which two Map objects group all employees with a salary greater than 30 by neighborhood?

- [ ]
```
Map<String, List<Employee>> r2 = roster.stream().filter(p)
  .collect(Collectors.groupingBy(f, Employee::getNeighborhood));
```

- [ ]
```
Map<Optional<String>, List<Employee>> r5 = roster.stream()
  .collect(Collectors.groupingBy(Employee::getNeighborhood,
      Collectors.filtering(p, Collectors.toList()))););
```

- [ ]
```
Map<Optional<String>, List<Employee>> r4 = roster.stream()
  .collect(Collectors.groupingBy(f, Collectors.filtering(p,
Collectors.toList())));
```

- [ ]
```
Map<String, List<Employee>> r1 = roster.stream()
  .collect(Collectors.groupingBy(Employee::getNeighborhood,
      Collectors.filtering(p, Collectors.toList())));
```

- [ ]
```
Map<Optional<String>, List<Employee>> r3 = roster.stream().filter(p)
  .collect(Collectors.groupingBy(p));
```

Mark for Review

28. Given:

```
public class Employee {
      private String name;
      private String neighborhood;
      private int salary;
      // Constructors and setter and getter methods go here
}
```

and the code fragment:

```
List roster = new ArrayList<>();
Predicate p = e -> e.getSalary() > 30;
Function> f =
    e -> Optional.ofNullable(e.getNeighborhood());
```

Which two Map objects group all employees with a salary greater than 30 by neighborhood?

- [ ]
```
Map<String, List<Employee>> r2 = roster.stream().filter(p)
  .collect(Collectors.groupingBy(f, Employee::getNeighborhood));
```

- [ ]
```
Map<Optional<String>, List<Employee>> r5 = roster.stream()
  .collect(Collectors.groupingBy(Employee::getNeighborhood,
      Collectors.filtering(p, Collectors.toList()))););
```

- [ ]
```
Map<Optional<String>, List<Employee>> r4 = roster.stream()
  .collect(Collectors.groupingBy(f, Collectors.filtering(p,
Collectors.toList())));
```

- [ ]
```
Map<String, List<Employee>> r1 = roster.stream()
  .collect(Collectors.groupingBy(Employee::getNeighborhood,
      Collectors.filtering(p, Collectors.toList())));
```

- [ ]
```
Map<Optional<String>, List<Employee>> r3 = roster.stream().filter(p)
  .collect(Collectors.groupingBy(p));
```

answer before submitting the test. Click Finish Test if you are r

Time Remaining 00:41:42

36. Given:

```
public class Tester {
  public static void main(String[] args) {
    StringBuilder sb = new StringBuilder(5);
    sb.append("HOWDY");
    sb.insert(0, ' ');
    sb.replace(3, 5, "LL");
    sb.insert(6, "COW");
    sb.delete(2, 7);
    System.out.println(sb.length());
  }
}
```

What is the result?

- An exception is thrown at runtime.
- 3
- 5
- 4

Previous Page 36 of 50 N

Time Remaining 00:41:07

37. Given:

```
class ConSuper {
    protected ConSuper(){
        this(2);
        System.out.print("3");
    }
    protected ConSuper(int a){
        System.out.print(a);
    }
}
```

and

```
public class ConSub extends ConSuper{
    ConSub(){
        this(4);
        System.out.print("1");
    }
    ConSub(int a) {
        System.out.print(a);
    }
    public static void main (String[] args){
        new ConSub(4);
    }
}
```

What is the result?

- 2134
- 234
- 2341
- 214

Mark for Review

28. Given:

```
public class Employee {
      private String name;
      private String neighborhood;
      private int salary;
      // Constructors and setter and getter methods go here
}
```

and the code fragment:

```
List roster = new ArrayList<>();
Predicate p = e -> e.getSalary() > 30;
Function> f =
    e -> Optional.ofNullable(e.getNeighborhood());
```

Which two Map objects group all employees with a salary greater than 30 by neighborhood?

- [ ]
```
Map<String, List<Employee>> r2 = roster.stream().filter(p)
  .collect(Collectors.groupingBy(f, Employee::getNeighborhood));
```

- [ ]
```
Map<Optional<String>, List<Employee>> r5 = roster.stream()
  .collect(Collectors.groupingBy(Employee::getNeighborhood,
      Collectors.filtering(p, Collectors.toList()))););
```

- [ ]
```
Map<Optional<String>, List<Employee>> r4 = roster.stream()
  .collect(Collectors.groupingBy(f, Collectors.filtering(p,
Collectors.toList())));
```

- [ ]
```
Map<String, List<Employee>> r1 = roster.stream()
  .collect(Collectors.groupingBy(Employee::getNeighborhood,
      Collectors.filtering(p, Collectors.toList())));
```

- [ ]
```
Map<Optional<String>, List<Employee>> r3 = roster.stream().filter(p)
  .collect(Collectors.groupingBy(p));
```

10. Given:

```
public class Test {
   public static void main(String... args) {
      int number = 20;
      Predicate<Integer> p = a -> a % 2 != 0;
      // line 1
         System.out.println(number + " is odd.");
      } else {
         System.out.println(number + " is even.");
      }
   }
}
```

Which statement on line 1 enables the `Test` class to compile?

- `if(p.test(number)) {`
- `if(p.accept(number)) {`
- `if(p.get(number)) {`
- `if(p.apply(number)) {`

Test: 819 - Java SE 11 Developer

Answer the question(s) on this page, and click Next to go to the next test
answer before submitting the test. Click Finish Test if you are ready to sub

Time Remaining 00:40:50

38. Given:

```
public class Main {
   public static void main(String[] args) {
      Thread t1 = new Thread(new MyThread());
      Thread t2 = new Thread(new MyThread());
      Thread t3 = new Thread(new MyThread());

      t1.start();
      t2.run();
      t3.start();

      t1.start();
   }
}
class MyThread implements Runnable {
  public void run() {
     System.out.println("Running.");
  }
}
```

Which one is correct ?

- Four threads are created.
- The compilation fails.
- An `IllegalThreadStateException` is thrown at runtime.
- Three threads are created.

Mark fo

31. Given TripleThis.java:

```
6.  import java.util.function.*;
7.  public class TripleThis {
8.     public static void main(String[] args) {
9.        Function tripler = x -> { return (Integer) x * 3; };
10.       TripleThis.printValue(tripler, 4);
11.    }
12.    public static void printValue(Function f, T num) {
13.       System.out.println(f.apply(num));
14.    }
15. }
```

Compiling TripleThis.java gives this compiler warning:

Note: TripleThis.java uses unchecked or unsafe operations.

Which two replacements remove this compiler warning and prints 12?

- [ ] Replace line 9 with `Function tripler = x -> { return x * 3; }`
- [ ] Replace line 12 with `public static void printValue(Function f, int num) {`
- [ ] Replace line 12 with `public static void printValue(Function f, T num) {`
- [ ] Replace line 12 with `public static void printValue(Function f, Integer num) {`
- [ ] Replace line 9 with `Function tripler = x -> { return x * 3; }`
- [ ] Replace line 9 with `Function tripler = x -> { return (Integer) x * 3; }`

34. Given:

```
public class ResourceTest {
  public static void main(String[] args){
    final MyResource res1 = new MyResource();
    MyResource res2 = new MyResource();
    try(res1 ; res2) {
      // do something
    } catch(Exception e) {}
  }
  static class MyResource implements AutoCloseable {
    public void close() throws Exception { }
  }
}
```

Which statement is true?

- The code fails to compile as `res2` should be declared as `final`.
- The code fails to compile as `MyResource` must implement `Closeable`.
- The code compiles successfully.
- The code fails to compile as `try-with-resource` needs a variable declaration such as `MyResource r1 = res1; MyResource r2 = res2;`.

Time Remaining 00:34:42

17. Given:

```
public interface Converter {
    public static final double POUNDS_PER_KILOGRAM = 2.20462; //
line 1
    public double tare();
    public double net();
    public default double gross() {                              //
line 2
        return tare() + net();
    }
    public default double tare(String units) {
        return toUnit(tare(), units);
    }
    public default double net(String units) {
        return toUnit(net(), units);
    }
    public default double gross(String units) {
        return toUnit(gross(), units);
    }
    private static double toUnit(double kilograms, String unit)
{  // line 3
        switch (unit) {
            case "KILO": return kilograms;
            case "POUND": return kilograms * POUNDS_PER_KILOGRAM;
            default: throw new IllegalArgumentException();
        }
    }
}
```

Which is true?

- Line 3 is the first line to cause a compilation error.
- Line 2 is the first line to cause a compilation error.
- Line 1 is the first line to cause a compilation error.
- It compiles without errors.

40. Given the code fragment:

```
public static void main(String[] args) {
 var lst = List.of(1, 2.0f, "4.0");
 for (var c : lst) {
  System.out.print("> " + c);
 }
 System.out.println();
 lst.add(2, 3);                                    // line n1
 for (int c = 0; c < lst.size(); c++) {
  display(lst.get(c));
 }
}
public static void display(var c) {               // line n2
 System.out.print("> " + c);
}
```

What is the result?

- > 1> 2.0> 4.0
  > 1> 2.0> 4.0
- A compile time error occurs at `line n2`.
- > 1> 2.0> 4.0
  > 1> 2.0> 3> 4.0
- An exception is thrown at `line n1`.

Answer the question(s) on this page, and click Next to go to the next te
answer before submitting the test. Click Finish Test if you are ready to s

Time Remaining 00:30:23

41. Which declaration of an annotation type is legal?

```
@interface Author {
    String name();
    String date;
}
```

```
@interface Author {
    String name();
    String date default "";
}
```

```
@interface Author extends Serializable {
    String name() default "";
    String date();
}
```

```
@interface Author {
    String name() default "";
    String date();
}
```

```
@interface Author {
    String name() default null;
    String date();
}
```

Answer the question(s) on this page, and click Next to go to the next test page. Click Summary to see which questions you answer before submitting the test. Click Finish Test if you are ready to submit your test.

Time Remaining 00:30:39

39. Given the code fragment:

```
Locale locale = Locale.US;
// line 1
double currency = 1_00.00;
System.out.println(formatter.format(currency));
```

You want to display the value of `currency` as `$100.00`.

Which code inserted on line 1 will accomplish this?

- `NumberFormat formatter = NumberFormat.getInstance(locale);`
- `NumberFormat formatter = NumberFormat.getCurrencyInstance(locale);`
- `NumberFormat formatter = NumberFormat.getCurrency(locale);`
- `NumberFormat formatter = NumberFormat.getInstance(locale).getCurrency();`

40. Given the code fragment:

```
public static void main(String[] args) {
 var lst = List.of(1, 2.0f, "4.0");
 for (var c : lst) {
  System.out.print("> " + c);
 }
 System.out.println();
 lst.add(2, 3);                                    // line n1
 for (int c = 0; c < lst.size(); c++) {
  display(lst.get(c));
 }
}
public static void display(var c) {               // line n2
 System.out.print("> " + c);
}
```

What is the result?

- > 1> 2.0> 4.0
  > 1> 2.0> 4.0
- A compile time error occurs at `line n2`.
- > 1> 2.0> 4.0
  > 1> 2.0> 3> 4.0
- An exception is thrown at `line n1`.

42. Given the code fragment:

```
public class Color {
  String hue;
  int value;
  public Color(String hue, int value) {
    this.hue = hue;
    this.value = value;
  }
  public String toString() {
    return this.hue + ":" + this.value;
  }
  public static void main(String[] args) {
    List clrs = List.of(new Color("Red", 100),
                        new Color("Yellow", 50),
                        new Color("Red", 75),
                        new Color("Yellow", 75));
    Comparator hueSrtr = (h1, h2) -> h1.hue.compareTo(h2.hue);
    Comparator valueSrtr = (h1, h2) -> { if (h1.value >= h2.value) {
                                           return 1;
                                         } else {
                                           return -1;
                                         }
                                       };
    clrs.sort(hueSrtr.thenComparing(valueSrtr));
    System.out.println(clrs);
  }
}
```

What is the result?

- [Red:75, Red:100, Yellow:50, Yellow:75]
- [Yellow:75, Yellow:50,Red:100, Red:75]
- [Yellow:50, Yellow:75,Red:75, Red:100]
- An Exception is thrown at runtime.
- [Red:100, Red:75, Yellow:75, Yellow:50]

```
    }
}

public class ExSub extends ExSuper {
  public ExSub(int eCode, String msg, Throwable cause)
    { super(eCode, msg, cause); }
}
```

and the code fragment:

```
try {
   String param1 = "Oracle";
   if (param1.equalsIgnoreCase("oracle")) {
     throw new ExSub(9001, "APPLICATION ERROR-9001", new
FileNotFoundException("MyFile.txt"));
   }
     throw new ExSuper(9001, new FileNotFoundException("MyFile.txt")); // Line 1
} catch (ExSuper ex) {
   System.out.println(ex.getMessage());
}
```

What is the result?

- 9001: APPLICATION ERROR-9001-MyFile.txt
  9001: java.io.FileNotFoundException: MyFile.txt-MyFile.txt
- 9001: java.io.FileNotFoundException: MyFile.txt-MyFile.txt
- 9001: APPLICATION ERROR-9001-MyFile.txt
- Compilations fails at Line 1.

27. Given:

```
<ListInteger> numbers = List.of(2, 3, 0, 8, 1, 9, 5, 7, 6, 4);
int sum = numbers.stream().reduce(0, (n, m) -> n + m);  // line 1
```

You want to make the reduction operation parallelized.

Which two modifications will accomplish this?

- [ ] Replace line 1 with `int sum = numbers.parallel().stream().reduce(0, (n, m) -> n + m);`.
- [ ] Replace line 1 with `int sum = numbers.parallelStream().reduce(0, (n, m) -> n + m);`.
- [ ] Replace line 1 with `int sum = numbers.stream().iterate(0, a -> a+1).reduce(0, (n, m) -> n + m);`.
- [ ] Replace line 1 with `int sum = numbers.stream().parallel().reduce(0, (n, m) -> n + m);`.
- [ ] Replace line 1 with `int sum = numbers.stream().flatMap(a -> a).reduce(0, (n, m) -> n + m);`.

Exam

Mark for Review

21. Given:

```
public class ExSuper extends Exception {
   private final int eCode;
   public ExSuper(int eCode, Throwable cause) {
     super(cause);
     this.eCode = eCode;
   }

   public ExSuper(int eCode, String msg, Throwable cause) {
     super(msg, cause);
     this.eCode = eCode;
   }
   public String getMessage() {
     return this.eCode+": "+super.getMessage()+"-"+this.getCause().getMessage();
    }
}

public class ExSub extends ExSuper {
   public ExSub(int eCode, String msg, Throwable cause)
     { super(eCode, msg, cause); }
}
```

and the code fragment:

```
try {
   String param1 = "Oracle";
   if (param1.equalsIgnoreCase("oracle")) {
     throw new ExSub(9001, "APPLICATION ERROR-9001", new
FileNotFoundException("MyFile.txt"));
   }
     throw new ExSuper(9001, new FileNotFoundException("MyFile.txt")); // Line 1
} catch (ExSuper ex) {
   System.out.println(ex.getMessage());
}
```

What is the result?

- 9001: APPLICATION ERROR-9001-MyFile.txt

40. Given the code fragment:

```
public static void main(String[] args) {
 var lst = List.of(1, 2.0f, "4.0");
 for (var c : lst) {
  System.out.print("> " + c);
 }
 System.out.println();
 lst.add(2, 3);                                    // line n1
 for (int c = 0; c < lst.size(); c++) {
  display(lst.get(c));
 }
}
public static void display(var c) {               // line n2
 System.out.print("> " + c);
}
```

What is the result?

- > 1> 2.0> 4.0
  > 1> 2.0> 4.0
- A compile time error occurs at `line n2`.
- > 1> 2.0> 4.0
  > 1> 2.0> 3> 4.0
- An exception is thrown at `line n1`.

42. Given the code fragment:

Mark for Review

```
public class Color {
   String hue;
   int value;
   public Color(String hue, int value) {
     this.hue = hue;
     this.value = value;
   }
   public String toString() {
     return this.hue + ":" + this.value;
   }
   public static void main(String[] args) {
     List clrs = List.of(new Color("Red", 100),
                         new Color("Yellow", 50),
                         new Color("Red", 75),
                         new Color("Yellow", 75));
     Comparator hueSrtr = (h1, h2) -> h1.hue.compareTo(h2.hue);
     Comparator valueSrtr = (h1, h2) -> { if (h1.value >= h2.value) {
                                            return 1;
                                          } else {
                                            return -1;
                                          }
                                        };
     clrs.sort(hueSrtr.thenComparing(valueSrtr));
     System.out.println(clrs);
   }
}
```

What is the result?

- [Red:75, Red:100, Yellow:50, Yellow:75]
- [Yellow:75, Yellow:50,Red:100, Red:75]
- [Yellow:50, Yellow:75,Red:75, Red:100]

34. Given:

```
public class ResourceTest {
  public static void main(String[] args){
    final MyResource res1 = new MyResource();
    MyResource res2 = new MyResource();
    try(res1 ; res2) {
      // do something
    } catch(Exception e) {}
  }
  static class MyResource implements AutoCloseable {
    public void close() throws Exception { }
  }
}
```

Which statement is true?

- The code fails to compile as `res2` should be declared as `final`.
- The code fails to compile as `MyResource` must implement `Closeable`.
- The code compiles successfully.
- The code fails to compile as `try-with-resource` needs a variable declaration such as `MyResource r1 = res1; MyResource r2 = res2;`.

```
    this.value = value;
  }
  public String toString() {
    return this.hue + ":" + this.value;
  }
  public static void main(String[] args) {
    List clrs = List.of(new Color("Red", 100),
                        new Color("Yellow", 50),
                        new Color("Red", 75),
                        new Color("Yellow", 75));
    Comparator hueSrtr = (h1, h2) -> h1.hue.compareTo(h2.hue);
    Comparator valueSrtr = (h1, h2) -> { if (h1.value >= h2.value) {
                                           return 1;
                                         } else {
                                           return -1;
                                         }
                                       };
    clrs.sort(hueSrtr.thenComparing(valueSrtr));
    System.out.println(clrs);
  }
}
```

What is the result?

- [Red:75, Red:100, Yellow:50, Yellow:75]
- [Yellow:75, Yellow:50,Red:100, Red:75]
- [Yellow:50, Yellow:75,Red:75, Red:100]
- An Exception is thrown at runtime.
- [Red:100, Red:75, Yellow:75, Yellow:50]