Q1. Given the definition of the Vehicle class:

```
Class Vehhicle {

int distance;//line n1

Vehicle (int x) {

this distance = x;

}

public void increSpeed(int time) {//line n2

int timeTravel = time;//line n3

class Car {

int value = 0;

public void speed () {

value = distance /timeTravel;

System.out.println ("Velocity with new speed"+value+"kmph");

}

}

new Car().speed();

}

}
```

and this code fragment:

```
Vehicle v = new Vehicle (100);
V. increSpeed(60);
```

What is the result?
A. Velocity with new speed 1 kmph
B. Velocity with new speed
C. A compilation error occurs atline n1.
D. A compilation error occurs atline n2.

E. A compilation error occurs atline n3.

Answer: A

Explanation:
Q2. Given:

```
IntStream stream = IntStream.of (1,2,3);

IntFunction<Integer> inFu= x -> y -> x*y;//line n1

IntStream newStream = stream.map(inFu.apply(10));//line n2
```

newStream.forEach(System.out::print);

Which modification enables the code fragment to compile?

A. Replaceline n1with:
IntFunction<UnaryOperator> inFu = x -> y -> x*y;
B. Replaceline n1with:
IntFunction<IntUnaryOperator> inFu = x -> y -> x*y;
C. Replaceline n1with:
BiFunction<IntUnaryOperator> inFu = x -> y -> x*y;
D. Replaceline n2with:
IntStream newStream = stream.map(inFu.applyAsInt (10));

Answer:B

Explanation:
Q3. Given the code fragment:

List<Integer> values = Arrays.asList (1, 2, 3);

values.stream ()

.map(n -> n*2)//line n1

.peek(System.out::print)//line n2

.count();

What is the result?

A. 246
B. The code produces no output.
C. A compilation error occurs atline n1.
D. A compilation error occurs atline n2.

Answer: A

Explanation:
Q4. Given the code fragment:

public class Foo {

public static void main (String [ ] args) {

Map<Integer, String> unsortMap = new HashMap< > ( );

unsortMap.put (10, "z");

unsortMap.put (5, "b");

unsortMap.put (1, "d");

unsortMap.put (7, "e");

unsortMap.put (50, "j");

Map<Integer, String> treeMap = new TreeMap <Integer, String> (new

Comparator<Integer> ( ) {

```
@Override public int compare (Integer o1, Integer o2) {return o2.compareTo

(o1); } } );

treeMap.putAll (unsortMap);

for (Map.Entry<Integer, String> entry : treeMap.entrySet () ) {

System.out.print (entry.getValue () + " ");

}

}

}
```

What is the result?

A. A compilation error occurs.
B. d b e z j
C. j z e b d
D. z b d e j

Answer: C

Explanation:
Q5. Which two reasons should you use interfaces instead of abstract classes?

A. You expect that classes that implement your interfaces have many common methods or fields, or require access modifiers other than public.
B. You expect that unrelated classes would implement your interfaces.
C. You want to share code among several closely related classes.
D. You want to declare non-static on non-final fields.
E. You want to take advantage of multiple inheritance of type.

Answer: A,E

Reference:http://www.programmerinterview.com/index.php/java-questions/interface-vs-abstract- class/
Q6. Given:

```
public class Counter {

public static void main (String[ ] args) {

int a = 10;

int b = -1;

assert (b >=1) : "Invalid Denominator";

int c = a / b;

System.out.println (c);

}

}
```

What is the result of running the code with the 璉 a option?

A. -10

B. 0
C. AnAssertionErroris thrown.
D. A compilation error occurs.

Answer: A

Explanation:
Q7. Given:

```
class Bird {

public void fly () { System.out.print("Can fly"); }

}

class Penguin extends Bird {

public void fly () { System.out.print("Cannot fly"); }

}
```

and the code fragment:

```
class Birdie {

public static void main (String [ ] args) {

fly( ( ) -> new Bird ( ));

fly (Penguin : : new);

}

/* line n1 */

}
```

Which code fragment, when inserted at line n1, enables the Birdie class to compile?

A. static void fly (Consumer<Bird> bird) {
bird :: fly ();
}
B. static void fly (Consumer<? extends Bird> bird) {
bird.accept( ) fly ();
}
C. static void fly (Supplier<Bird> bird) {
bird.get( ) fly ();
}
D. static void fly (Supplier<? extends Bird> bird) {
LOST

Answer: C

Explanation: NOTE: Very confusing question. There is no logic in the options.
Q8. Given:

```
1. abstract class Shape {

2. Shape ( ) { System.out.println ("Shape"); }

3. protected void area ( ) { System.out.println ("Shape"); }
```

```
4. }
5.
6. class Square extends Shape {
7. int side;
8. Square (int side ){
9./* insert code here */
10. this.side = side;
11. }
12. public void area ( ) { System.out.println ("Square"); }
13. }
14. class Rectangle extends Square {
15. int len, br;
16. Rectangle (int x, int y) {
17. /* insert code here */
18. len = x, br = y;
19. }
20. void area ( ) { System.out.println ("Rectangle"); }
21. }
```

Which two modifications enable the code to compile?

A. At line 1, removeabstract
B. At line 9, insertsuper ( );
C. At line 12, removepublic
D. At line 17, insertsuper (x);
E. At line 17, insertsuper (); super.side = x;
F. At line 20, usepublic void area ( ) {

Answer: D,F

Explanation:
Q9. Given:

```
class Sum extends RecursiveAction { //line n1
static final int THRESHOLD_SIZE = 3;
int stIndex, lstIndex;
int [ ] data;
public Sum (int [ ]data, int start, int end) {
```

```
this.data = data;

this stIndex = start;

this. lstIndex = end;

}

protected void compute ( ) {

int sum = 0;

if (lstIndex<=stIndex + THRESHOLD_SIZE) {

for (int i = stIndex; i < lstIndex; i++) {

sum += data [i];

}

System.out.println(sum);

} else {

new Sum (data, stIndex + THRESHOLD_SIZE, lstIndex).fork( );

new Sum (data, stIndex,

Math.min (lstIndex, stIndex + THRESHOLD_SIZE)

).compute ();

}

}

}
```

and the code fragment:

```
ForkJoinPool fjPool = new ForkJoinPool ( );

int data [ ] = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10}

fjPool.invoke (new Sum (data, 0, data.length));
```

and given that the sum of all integers from 1 to 10 is 55.

Which statement is true?

A. The program prints several values that total 55.
B. The program prints 55.
C. A compilation error occurs atline n1.
D. The program prints several values whose sum exceeds 55.

Answer: A

Explanation:
Q10. Given:

```
public class Foo<K, V> {
```

```
private K key;

private V value;

public Foo (K key, V value) (this.key = key; this value = value;)

public static <T> Foo<T, T> twice (T value) (return new Foo<T, T> (value, value); )

public K getKey () (return key;)

public V getValue () (return value;)

}
```

Which option fails?

A. Foo<String, Integer> mark = new Foo<String, Integer> ("Steve", 100);
B. Foo<String, String> pair = Foo.<String>twice ("Hello World!");
C. Foo<?, ?> percentage = new Foo <> (97, 32);
D. Foo<String, String> grade = new Foo <> ("John", "A");

Answer: C

Explanation:
Q11. Given the code fragment:

```
Stream<List<String>> iStr= Stream.of (

Arrays.asList ("1", "John"),

Arrays.asList ("2", null));

Stream<<String> nInSt = iStr.flatMapToInt ((x) -> x.stream ());

nInSt.forEach (System.out :: print);
```

What is the result?

A. 1John2null
B. 12
C. ANullPointerExceptionis thrown at run time.
D. A compilation error occurs.

Answer: D

Explanation:
Q12. Given the code fragment:

```
Path file = Paths.get ("courses.txt");

// line n1
```

Assume the courses.txt is accessible.

Which code fragment can be inserted at line n1 to enable the code to print the content of the courses.txt file?

A. List<String> fc = Files.list(file);

fc.stream().forEach(s - > System.out.println(s));
B. Stream<String> fc = Files.readAllLines (file);
fc.forEach (s - > System.out.println(s));
C. List<String> fc = readAllLines(file);
fc.stream().forEach (s - > System.out.println(s));
D. Stream<String> fc = Files.lines (file);
fc.forEach (s - > System.out.println(s));

Answer: D

Explanation:
Q13. Given the code fragment:

```
public void recDelete (String dirName) throws IOException {

File [ ] listOfFiles = new File (dirName) .listFiles();

if (listOfFiles ! = null && listOfFiles.length >0) {

for (File aFile : listOfFiles) {

if (aFile.isDirectory ()) {

recDelete (aFile.getAbsolutePath ());

} else {

if (aFile.getName ().endsWith (".class"))

aFile.delete ();

}

}

}

}
```

Assume that Projects contains subdirectories that contain .class files and is passed as an argument to the recDelete () method when it is invoked.

What is the result?

A. The method deletes all the.classfiles in theProjectsdirectory and its subdirectories.
B. The method deletes the.classfiles of theProjectsdirectory only.
C. The method executes and does not make any changes to theProjectsdirectory.
D. The method throws an IOException.

Answer: A

Explanation:
Q14. Given the code fragments:

```
4. void doStuff() throws ArithmeticException, NumberFormatException, Exception {

5. if (Math.random() >-1 throw new Exception ("Try again");

6. }
```

and

24. try {

25. doStuff ( ):

26. } catch (ArithmeticException | NumberFormatException | Exception e) {

27. System.out.println (e.getMessage()); }

28. catch (Exception e) {

29. System.out.println (e.getMessage()); }

30. }

Which modification enables the code to print Try again?

A. Comment the lines 28, 29 and 30.
B. Replace line 26 with:
} catch (Exception | ArithmeticException | NumberFormatException e) {
C. Replace line 26 with:
} catch (ArithmeticException | NumberFormatException e) {
D. Replace line 27 with:
throw e;

Answer: C

Explanation:
Q15. Given the definition of the Country class:

public class Country {

public enum Continent {ASIA, EUROPE}

String name;

Continent region;

public Country (String na, Continent reg) {

name = na, region = reg;

}

public String getName () {return name;}

public Continent getRegion () {return region;}

}

and the code fragment:

List<Country> couList = Arrays.asList (

new Country ("Japan", Country.Continent.ASIA),

new Country ("Italy", Country.Continent.EUROPE),

new Country ("Germany", Country.Continent.EUROPE));

```
Map<Country.Continent, List<String>> regionNames = couList.stream ()

.collect(Collectors.groupingBy (Country ::getRegion,

Collectors.mapping(Country::getName, Collectors.toList())));

System.out.println(regionNames);
```

What is the output?

A. {EUROPE = [Italy, Germany],ASIA = [Japan]}
B. {ASIA = [Japan], EUROPE = [Italy, Germany]}
C. {EUROPE = [Germany, Italy], ASIA = [Japan]}
D. {EUROPE = [Germany], EUROPE = [Italy], ASIA = [Japan]}

Answer: A

Explanation:
Q16. Given the code fragment:

```
Map<Integer, String> books = new TreeMap<>();

books.put (1007, "A");

books.put (1002, "C");

books.put (1001, "B");

books.put (1003, "B");

System.out.println (books);
```

What is the result?

A. {1007 = A, 1002 = C, 1001 = B, 1003 = B}
B. {1001 = B, 1002 = C, 1003 = B, 1007 = A}
C. {1002 = C, 1003 = B, 1007 = A}
D. {1007 = A, 1001 = B, 1003 = B, 1002 = C}

Answer: B

Explanation:
Q17. Given:

```
class Book {

int id;

String name;

public Book (int id, String name) {

this.id = id;

this.name = name;

}

public boolean equals (Object obj) { //line n1
```

```
boolean output = false;

Book b = (Book) obj;

if (this.name.equals(b name)){

output = true;

}

return output;

}

}
```

and the code fragment:

```
Book b1 = new Book (101, "Java Programing");

Book b2 = new Book (102, "Java Programing");

System.out.println (b1.equals(b2)); //line n2
```

Which statement is true?

A. The program printstrue.
B. The program printsfalse.
C. A compilation error occurs. To ensure successful compilation, replaceline n1with:
boolean equals (Book obj) {
D. A compilation error occurs. To ensure successful compilation, replaceline n2with:
System.out.println (b1.equals((Object) b2));

Answer: A

Explanation:

Q18. Given the content of /resourses/Message.properties:

```
welcome1="Good day!"
```

and given the code fragment:

```
Properties prop = new Properties ();

FileInputStream fis = new FileInputStream ("/resources/Message.properties");

prop.load(fis);

System.out.println(prop.getProperty("welcome1"));

System.out.println(prop.getProperty("welcome2", "Test"));//line n1

System.out.println(prop.getProperty("welcome3"));
```

What is the result?

A. Good day!
Test
followed by anExceptionstack trace

B. Good day!
followed by anExceptionstack trace
C. Good day!
Test
null
D. A compilation error occurs atline n1.

Answer: C

Explanation:
Q19. Which action can be used to load a database driver by using JDBC3.0?

A. Add the driver class to the META-INF/services folder of the JAR file.

B. Include the JDBC driver classin ajdbc.propertiesfile.
C. Use thejava.lang.Class.forNamemethod to load the driver class.
D. Use theDriverManager.getDrivermethod to load the driver class.

Answer: D

Explanation:
Q20. Given the code fragment:

```
Path p1 = Paths.get("/Pics/MyPic.jpeg");

System.out.println (p1.getNameCount() +

":" + p1.getName(1) +

":" + p1.getFileName());
```

Assume that the Pics directory does NOT exist.

What is the result?

A. An exception is thrown at run time.
B. 2:MyPic.jpeg: MyPic.jpeg
C. 1:Pics:/Pics/ MyPic.jpeg
D. 2:Pics: MyPic.jpeg

Answer: B

Explanation:
Q21. Given the code fragments:

```
class MyThread implements Runnable {

private static AtomicInteger count = new AtomicInteger (0);

public void run () {

int x = count.incrementAndGet();

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

}

}
```

and

```
Thread thread1 = new Thread(new MyThread());

Thread thread2 = new Thread(new MyThread());

Thread thread3 = new Thread(new MyThread());

Thread [] ta = {thread1, thread2, thread3};

for (int x= 0; x < 3; x++) {

ta[x].start();

}
```

Which statement is true?

A. The program prints 12 3 and the order is unpredictable.
B. The program prints 1 2 3.
C. The program prints 1 1 1.
D. A compilation error occurs.

Answer: A

Explanation:
Q22. Given the code fragment:

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

BufferedReader br = new BufferedReader (new InputStremReader (System.in));

System.out.print ("Enter GDP: ");

//line 1

}
```

Which code fragment, when inserted at line 1, enables the code to read the GDP from the user?

A. int GDP = Integer.parseInt (br.readline());
B. int GDP = br.read();
C. int GDP = br.nextInt();
D. int GDP = Integer.parseInt (br.next());

Answer: A

Explanation:
Q23. Given the code fragment:

```
Path source = Paths.get ("/data/december/log.txt");

Path destination = Paths.get("/data");

Files.copy (source, destination);
```

and assuming that the file /data/december/log.txt is accessible and contains:

10-Dec-2014 ?Executed successfully

What is the result?

A. A file with the namelog.txtis created in the/datadirectory and the content of the/data/december/log.txtfile is copied to it.
B. The program executes successfully and does NOT change the file system.
C. AFileNotFoundExceptionis thrown at run time.
D. AFileAlreadyExistsExceptionis thrown at run time.

Answer: D

Explanation:
Q24. Given:

```
class Student {

String course, name, city;

public Student (String name, String course, String city) {

this.course = course; this.name = name; this.city = city;

}

public String toString() {

return course + ":" + name + ":" + city;

}
```

and the code fragment:

```
List<Student> stds = Arrays.asList(

new Student ("Jessy", "Java ME", "Chicago"),

new Student ("Helen", "Java EE", "Houston"),

new Student ("Mark", "Java ME", "Chicago"));

stds.stream()

.collect(Collectors.groupingBy(Student::getCourse))

.forEach(src, res) -> System.out.println(scr));
```

What is the result?

A. [Java EE: Helen:Houston]
[Java ME: Jessy:Chicago, Java ME: Mark:Chicago]
B. Java EE
Java ME
C. [Java ME: Jessy:Chicago, Java ME: Mark:Chicago]
[Java EE: Helen:Houston]
D. A compilation error occurs.

Answer: D

Explanation:

Q25. Given the code fragments:

```
interface CourseFilter extends Predicate<String> {
```

```
public default boolean test (String str) {

return str.equals ("Java");

}

}
```

and

```
List<String> strs = Arrays.asList("Java", "Java EE", "Java ME");

Predicate<String> cf1 = s - > s.length() > 3;

Predicate cf2 = new CourseFilter() { //line n1

public boolean test (String s) {

return s.contains ("Java");

}

};

long c = strs.stream()

.filter(cf1)

.filter(cf2//line n2

.count();

System.out.println(c);
```

What is the result?

A. 2
B. 3
C. A compilation error occurs atline n1.
D. A compilation error occurs atline n2.

Answer: B

Explanation:
Q26. Given:

```
public class Emp {

String fName;

String lName;

public Emp (String fn, String ln) {

fName = fn;

lName = ln;

}
```

```
public String getfName() { return fName; }

public String getlName() { return lName; }

}
```

and the code fragment:

```
List<Emp> emp = Arrays.asList (

new Emp ("John", "Smith"),

new Emp ("Peter", "Sam"),

new Emp ("Thomas", "Wale"));

emp.stream()

//line n1

.collect(Collectors.toList());
```

Which code fragment, when inserted at line n1, sorts the employees list in descending order of fName and then ascending order of lName?

A. .sorted (Comparator.comparing(Emp::getfName).reserved().thenComparing(Emp::getlName))
B. .sorted (Comparator.comparing(Emp::getfName).thenComparing(Emp::getlName))
C. .map(Emp::getfName).sorted(Comparator.reserveOrder())
D. .map(Emp::getfName).sorted(Comparator.reserveOrder().map(Emp::getlName).reserved

Answer: A

Explanation:
Q27. Given:

```
public enum USCurrency {

PENNY (1),

NICKLE(5),

DIME (10),

QUARTER(25);

private int value;

public USCurrency(int value) {

this.value = value;

}

public int getValue() {return value;}

}

public class Coin {
```

```
public static void main (String[] args) {

USCurrency usCoin =new USCurrency.DIME;

System.out.println(usCoin.getValue()):

}

}
```

Which two modifications enable the given code to compile?

A. Nest theUSCurrencyenumeration declaration within theCoinclass.
B. Make theUSCurrencyenumeration constructorprivate.
C. Remove thenewkeyword from the instantion ofusCoin.
D. Make the getter method ofvalueas astaticmethod.
E. Add thefinalkeyword in the declaration ofvalue.

Answer: A, B, C

Explanation:
Q28. Given:

```
class ImageScanner implements AutoCloseable {

public void close () throws Exception {

System.out.print ("Scanner closed.");

}

public void scanImage () throws Exception {

System.out.print ("Scan.");

throw new Exception("Unable to scan.");

}

}

class ImagePrinter implements AutoCloseable {

public void close () throws Exception {

System.out.print ("Printer closed.");

}

public void printImage () {System.out.print("Print."); }

}
```

and this code fragment:

```
try (ImageScanner ir = new ImageScanner();

ImagePrinter iw = new ImagePrinter()) {

ir.scanImage();
```

```
iw.printImage();

} catch (Exception e) {

System.out.print(e.getMessage());

}
```

What is the result?

A. Scan.Printer closed. Scanner closed. Unable to scan.
B. Scan.Scanner closed. Unable to scan.
C. Scan. Unable to scan.
D. Scan.Unable to scan.Printer closed.

Answer: A

Explanation:
Q29. Given the structure of the STUDENT table:

Student (id INTEGER, name VARCHAR)

Given:

```
public class Test {

static Connection newConnection =null;

public static Connection get DBConnection () throws SQLException {

try (Connection con = DriveManager.getConnection(URL, username, password)) {

newConnection = con;

}

return newConnection;

}

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

get DBConnection ();

Statement st = newConnection.createStatement();

st.executeUpdate("INSERT INTO student VALUES (102, `Kelvin')");

}

}
```

Assume that:

The required database driver is configured in the classpath.

The appropriate database is accessible with the URL, userName, and passWord exists.

The SQL query is valid.

What is the result?

A. The program executes successfully and the STUDENT table is updated with one record.
B. The program executes successfully and the STUDENT table is NOT updated with any record.
C. ASQLExceptionis thrown as runtime.
D. ANullPointerExceptionis thrown as runtime.

Answer: C

Explanation:
Q30. Given the code fragments:

```
class Employee {

Optional<Address> address;

Employee (Optional<Address> address) {

this.address = address;

}

public Optional<Address> getAddress() { return address; }

}

class Address {

String city = "New York";

public String getCity (){ return city: }

public String toString() {

return city;

}

}
```

and

```
Address address = null;

Optional<Address> addrs1 = Optional.ofNullable (address);

Employee e1 = new Employee (addrs1);

String eAddress = (addrs1.isPresent()) ? addrs1.get().getCity() : "City Not

 available";

System.out.println (eAddress);
```

What is the result?
A. New York
B. City Not available
C. null
D. ANoSuchElementExceptionis thrown at run time.

Answer: B

Explanation:

Q31. Given the code fragment:

```
Stream<Path> files = Files.walk(Paths.get(System.getProperty("user.home")));

files.forEach (fName -> {//line n1

try {

Path aPath = fName.toAbsolutePath();//line n2

System.out.println(fName + ":"

+ Files.readAttributes(aPath, Basic.File.Attributes.class).creationTime

());

} catch (IOException ex) {

ex.printStackTrace();

});
```

What is the result?

A. All files and directories under thehomedirectory are listed along with their attributes.
B. A compilation error occurs atline n1.
C. The files in the home directory are listed along with their attributes.
D. A compilation error occurs atline n2.

Answer: A

Explanation:
Q32. Given:

```
class Vehicle {

int vno;

String name;

public Vehicle (int vno, String name) {

this.vno = vno,;

this.name = name;

}

public String toString () {

return vno + ":" + name;

}

}
```

and this code fragment:

```
Set<Vehicle> vehicles = new TreeSet <> ();

vehicles.add(new Vehicle (10123, "Ford"));
```

vehicles.add(new Vehicle (10124, "BMW"));

System.out.println(vehicles);

What is the result?

A. 10123 Ford
10124 BMW
B. 10124 BMW
10123 Ford
C. A compilation error occurs.
D. AClassCastExceptionis thrown at run time.

Answer: D

Explanation:
Q33. Given that course.txt is accessible and contains:

Course : : Java

and given the code fragment:

public static void main (String[ ] args) {

int i;

char c;

try (FileInputStream fis = new FileInputStream ("course.txt");

InputStreamReader isr = new InputStreamReader(fis);) {

while (isr.ready()) { //line n1

isr.skip(2);

i = isr.read ();

c = (char) i;

System.out.print(c);

}

} catch (Exception e) {

e. printStackTrace();
}
}
What is the result?
A. ur ::va
B. ueJa
C . The program prints nothing.
D. A compilation error occurs atline n1.

Answer: C

Explanation:
Q34. Given:

```
public class Test<T> {

private T t;

public T get () {

return t;

}

public void set (T t) {

this.t = t;

}

public static void main (String args [ ] ) {

Test<String> type = new Test<>();

Test type 1 = new Test ();//line n1

type.set("Java");

type1.set(100);//line n2

System.out.print(type.get() + " " + type1.get());

}

}
```

What is the result?

A. Java 100
B. java.lang.string@<hashcode>java.lang.Integer@<hashcode>
C. A compilation error occurs. To rectify it, replaceline n1with:
Test<Integer> type1 = new Test<>();
D. A compilation error occurs. To rectify it, replaceline n2with:
type1.set (Integer(100));

Answer: A

Explanation:
Q35. Given the definition of the Vehicle class:

```
class Vehicle {

String name;

void setName (String name) {

this.name = name;

}

String getName() {

return name;

}
```

}

Which action encapsulates the Vehicle class?

A. Make theVehicleclasspublic.
B. Make thenamevariablepublic.
C. Make thesetNamemethodpublic.
D. Make thenamevariableprivate.
E. Make thesetNamemethodprivate.
F. Make thegetNamemethodprivate.

Answer: D

Explanation:
Q36. Given:

```
public class Product {

int id; int price;

public Product (int id, int price) {

this.id = id;

this.price = price;

}

public String toString() { return id + ":" + price; }

}
```

and the code fragment:

```
List<Product> products = Arrays.asList(new Product(1, 10),

new Product (2, 30),

new Product (2, 30));

Product p = products.stream().reduce(new Product (4, 0), (p1, p2) -> {

p1.price+=p2.price;

return new Product (p1.id, p1.price);});

products.add(p);

products.stream().parallel()

.reduce((p1, p2) - > p1.price > p2.price ? p1 : p2)

.ifPresent(System.out: :println);
```

What is the result?

A. 2 : 30
B. 4: 0
C. 4 : 60
D. 4 : 60

2 : 30
3 : 20
1 : 10
E. The program prints nothing.
F.An Vnsupported Operation Exception is thrown at run time.

Answer: F

Explanation:
Q37. Given the code fragments:

```
public class Book implements Comparator<Book> {

String name;

double price;

public Book () {}

public Book(String name, double price) {

this.name = name;

this.price = price;

}

public int compare(Book b1, Book b2) {

return b1.name.compareTo(b2.name);

}

public String toString() {

return name + ":" + price;

}

}
```

and

```
List<Book>books = Arrays.asList (new Book ("Beginning with Java", 2), new book ("A

Guide to Java Tour", 3));

Collections.sort(books, new Book());

System.out.print(books);
```

What is the result?

A. [A Guide to Java Tour:3, Beginning with Java:2]
B. [Beginning with Java:2.0,A Guide to Java Tour:3.0]
C. A compilation error occurs because theBookclass does not override the abstract methodcompareTo().
D. AnExceptionis thrown at run time.

Answer: B

Explanation:
Q38. Given the code fragment:

```
List<String> listVal = Arrays.asList("Joe", "Paul", "Alice", "Tom");

System.out.println (

// line n1

);
```

Which code fragment, when inserted at line n1, enables the code to print the count of string elements whose length is greater than three?

A. listVal.stream().filter(x -> x.length()>3).count()
B. listVal.stream().map(x -> x.length()>3).count()
C. listVal.stream().peek(x -> x.length()>3).count().get()
D. listVal.stream().filter(x -> x.length()>3).mapToInt(x -> x).count()

Answer: B

Explanation:
Q39. Given the code fragments:

```
class Caller implements Callable<String> {

String str;

public Caller (String s) {this.str=s;}

public String call()throws Exception { return str.concat ("Caller");}

}

class Runner implements Runnable {

String str;

public Runner (String s) {this.str=s;}

public void run () { System.out.println (str.concat ("Runner"));}

}
```

and

```
public static void main (String[] args) InterruptedException, ExecutionException {

ExecutorService es = Executors.newFixedThreadPool(2);

Future f1 = es.submit (new Caller ("Call"));

Future f2 = es.submit (new Runner ("Run"));

String str1 = (String) f1.get();

String str2 = (String) f2.get();//line n1

System.out.println(str1+ ":" + str2);

}
```

What is the result?

A. The program prints:
Run Runner
Call Caller : null
And the program does not terminate.
B. The program terminates after printing:
Run Runner
Call Caller : Run
C. A compilation error occurs atline n1.
D. AnExecutionis thrown at run time.

Answer: A

Explanation:
Q40. Given:

```
public class Canvas implements Drawable {

public void draw () { }

}

public abstract class Board extends Canvas { }

public class Paper extends Canvas {

protected void draw (int color) { }

}

public class Frame extends Canvas implements Drawable {

public void resize () { }

}

public interface Drawable {

public abstract void draw ();

}
```

Which statement is true?

A. Boarddoes not compile.
B. Paperdoes not compile.
C. Framedoes not compile.
D. Drawabledoes not compile.
E. All classes compile successfully.

Answer: E

Explanation:
Q41. Given the code fragment:

```
List<String> str = Arrays.asList ("my", "pen", "is", "your', "pen");

Predicate<String> test = s -> {

int i = 0;
```

```
boolean result = s.contains ("pen");

System.out.print(i++ + ":");

return result;

};

str.stream()

.filter(test)

.findFirst()

.ifPresent(System.out ::print);
```

What is the result?

A. 0 : 0 : pen
B. 0 : 1 : pen
C. 0 : 0 : 0 : 0 : 0 : pen
D. 0 : 1 : 2 : 3 : 4 :
E. A compilation error occurs.

Answer: A

Explanation:
Q42. Given the code fragment:

```
List<String> empDetails = Arrays.asList("100, Robin, HR",

"200, Mary, AdminServices",

"101, Peter, HR");

empDetails.stream()

.filter(s-> s.contains("1"))

.sorted()

.forEach(System.out::println); //line n1
```

What is the result?

A. 100, Robin, HR
101, Peter, HR
B. E. A compilation error occurs atline n1.
C. 100, Robin, HR
101, Peter, HR
200, Mary, AdminServices
D. 100, Robin, HR

200, Mary, AdminServices
101, Peter, HR

Answer: A

Explanation:
Q43. Given:

```
interface Rideable {Car getCar (String name); }

class Car {

private String name;

public Car (String name) {

this.name = name;

}

}
```

Which code fragment creates an instance of Car?

A. Car auto = Car ("MyCar"): : new;
B. Car auto = Car : : new;
Car vehicle = auto : : getCar("MyCar");
C. Rideable rider = Car : : new;
Car vehicle = rider.getCar("MyCar");
D. Car vehicle = Rideable : : new : : getCar("MyCar");

Answer: C

Explanation:
Q44. Which statement is true about the single abstract method of the java.util.function.Function interface?

A. It accepts one argument and returnsvoid.
B. It accepts one argument and returnsboolean.
C. It accepts one argument and always produces a result of the same type as the argument.
D. It accepts an argument and produces a result of any data type.

Answer: C

Reference:http://winterbe.com/posts/2014/03/16/java-8-tutorial/(functions)
Q45. Which statement is true about the DriverManager class?

A. It returns an instance ofConnection.
B. it executes SQL statements against the database.
C. It only queries metadata of the database.
D. it is written by different vendors for their specific database.

Answer: A

Explanation: The DriverManager returns an instance of Doctrine\DBAL\Connection which is a wrapper around the underlying driver connection (which is often a PDO instance).
Reference:http://doctrine-dbal.readthedocs.org/en/latest/reference/configuration.html
Q46. Given the code fragment:

```
List<Integer> nums = Arrays.asList (10, 20, 8):

System.out.println (

//line n1

);
```

Which code fragment must be inserted at line n1 to enable the code to print the maximum number in the nums list?

A. nums.stream().max(Comparator.comparing(a -> a)).get()
B. nums.stream().max(Integer : : max).get()

C. nums.stream().max()
D. nums.stream().map(a -> a).max()

Answer: A

Explanation:
Q47. Given:

```
public final class IceCream {

public void prepare() {}

}

public class Cake {

public final void bake(int min, int temp) {}

public void mix() {}

}

public class Shop {

private Cake c = new Cake ();

private final double discount = 0.25;

public void makeReady () { c.bake(10, 120); }

}

public class Bread extends Cake {

public void bake(int minutes, int temperature) {}

public void addToppings() {}

}
```

Which statement is true?

A. A compilation error occurs inIceCream.
B. A compilation error occurs inCake.
C. A compilation error occurs inShop.

D. A compilation error occurs inBread
E. All classes compile successfully.

Answer: D

Explanation:
Q48. Which two statements are true about localizing an application?

A. Support for new regional languages does not require recompilation of the code.
B. Textual elements (messages and GUI labels) are hard-coded in the code.
C. Language and region-specific programs are created using localized data.
D. Resource bundle files include data and currency information.

E. Language codes use lowercase letters andregion codes use uppercase letters.

Answer: A,E

Reference:http://docs.oracle.com/javase/7/docs/technotes/guides/intl/
Q49. Which statement is true about java.util.stream.Stream?

A. A stream cannot be consumed more than once.
B. The execution mode of streams can be changed during processing.
C. Streams are intended to modify the source data.
D. A parallel stream is always faster than an equivalent sequential stream.

Answer: B

Explanation:
Q50. The data.doc, data.txt and data.xml files are accessible and contain text.

Given the code fragment:

```
Stream<Path> paths = Stream.of (Paths. get("data.doc"),

Paths. get("data.txt"),

Paths. get("data.xml"));

paths.filter(s-> s.toString().endWith("txt")).forEach(

s -> {

try {

Files.readAllLines(s)

.stream()

.forEach(System.out::println); //line n1

} catch (IOException e) {

System.out.println("Exception");

}

}

);
```

What is the result?

A. The program prints the content ofdata.txtfile.
B. The program prints:
Exception
<<The content of the data.txt file>>
Exception
C. A compilation error occurs atline n1.
D. The program prints the content of the three files.

Answer: C

Explanation:
Q51. Given:

```
final class Folder {//line n1

//line n2

public void open () {

System.out.print("Open");

}

}

public class Test {

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

try (Folder f = new Folder()) {

f . open();
}
}
}
```

Which two modifications enable the code to print Open Close?

A. Replaceline n1with:
class Folder implementsAutoCloseable {

B. Replaceline n1with:
class Folder extends Closeable {

C. Replaceline n1with:
class Folder extends Exception {

D. At line n2, insert:
final void close () {
System.out.print("Close");
}

E. At line n2, insert:
public void close () throws IOException {
System.out.print("Close");
}

Answer: A,E

Explanation:

Q52. You want to create a singleton class by using the Singleton design pattern.

Which two statements enforce the singleton nature of the design?

A. Make the classstatic.
B. Make the constructorprivate.
C. Overrideequals() andhashCode() methods of the java.lang.Object class.
D. Use astaticreference to point to the single instance.
E. Implement theSerializableinterface.

Answer: A,B

Explanation:

Q53. Given the code fragment:

9. Connection conn = DriveManager.getConnection(dbURL, userName, passWord);

10. String query = "SELECT id FROM Employee";

```
11. try (Statement stmt = conn.createStatement()) {

12. ResultSet rs = stmt.executeQuery(query);

13.stmt.executeQuery("SELECT id FROM Customer");

14. while (rs.next()) {

15. //process the results

16.System.out.println("Employee ID: "+ rs.getInt("id"));

17.}

18. } catch (Exception e) {

19. System.out.println ("Error");

20. }
```

Assume that:

The required database driver is configured in the classpath.

The appropriate database is accessible with the dbURL, userName, and passWord exists.

The Employee and Customer tables are available and each table has id column with a few records and the SQL queries are valid.

What is the result of compiling and executing this code fragment?

A. The program prints employee IDs.
B. The program prints customer IDs.
C. The program printsError.
D. compilation fails on line 13.

Answer: A

Explanation:
Q54. Given the code fragment:

```
List<Integer> codes = Arrays.asList (10, 20);

UnaryOperator<Double> uo = s -> s +10.0;

codes.replaceAll(uo);

codes.forEach(c -> System.out.println(c));
```

What is the result?

A. 20.0
30.0
B. 10
20
C. A compilation error occurs.
D. ANumberFormatExceptionis thrown at run time.

Answer: C

Explanation:
Q55.
Given:

```
public class Customer {

private String fName;

private String lName;

private static int count;

public customer (String first, String last) {fName = first, lName = last;

++count;}

static { count = 0; }

public static int getCount() {return count; }

}

public class App {

public static void main (String [] args) {

Customer c1 = new Customer("Larry", "Smith");

Customer c2 = new Customer("Pedro", "Gonzales");

Customer c3 = new Customer("Penny", "Jones");

Customer c4 = new Customer("Lars", "Svenson");

c4 = null;

c3 = c2;

System.out.println (Customer.getCount());

}

}
```

What is the result?

A. 0
B. 2
C. 3
D. 4
E. 5

Answer: D

Explanation:
Q56. Given:

Item table

?ID, INTEGER: PK

?DESCRIP, VARCHAR(100)

?PRICE, REAL

?QUANTIT<; INTEGER

And given the code fragment:

9. try {

10.Connection conn = DriveManager.getConnection(dbURL, username, password);

11. String query = "Select * FROM Item WHERE ID = 110";

12. Statement stmt = conn.createStatement();

13. ResultSet rs = stmt.executeQuery(query);

14.while(rs.next()) {

15.System.out.println("ID:" + rs.getInt("Id"));

16.System.out.println("Description:" + rs.getString("Descrip"));

17.System.out.println("Price:" + rs.getDouble("Price"));

18. System.out.println(Quantity:" + rs.getInt("Quantity"));

19.}

20. } catch (SQLException se) {

21. System.out.println("Error");

22. }

Assume that:

The required database driver is configured in the classpath.

The appropriate database is accessible with the dbURL, userName, and passWord exists.

The SQL query is valid.

What is the result?

A. An exception is thrown at runtime.
B. Compilation fails.
C. The code printsError.
D. The code prints information about Item 110.

Answer: C

Explanation:
Q57. Given:

class Worker extends Thread {

CyclicBarrier cb;

```
public Worker(CyclicBarrier cb) { this.cb = cb; }

public void run () {

try {

cb.await();

System.out.println("Worker...");

} catch (Exception ex) { }

}

}

class Master implements Runnable { //line n1

public void run () {

System.out.println("Master...");

}

}
```

and the code fragment:

```
Master master = new Master();

//line n2

Worker worker = new Worker(cb);

worker.start();
```

You have been asked to ensure that the run methods of both the Worker and Master classes are executed.

Which modification meets the requirement?

A. Atline n2, insertCyclicBarrier cb = new CyclicBarrier(2, master);
B. Replaceline n1withclass Master extends Thread {
C. Atline n2, insertCyclicBarrier cb = new CyclicBarrier(1, master);
D. Atline n2, insertCyclicBarrier cb = new CyclicBarrier(master);

Answer: C

Explanation:
Q58. Given the code fragment:

```
String str = "Java is a programming language";

ToIntFunction<String> indexVal = str: : indexOf; //line n1

int x = indexVal.applyAsInt("Java");//line n2

System.out.println(x);
```

What is the result?

A. 0

B. 1

C. A compilation error occurs atline n1.
D. A compilation error occurs atline n2.

Answer: A

Explanation:
Q59. Given the code fragment:

```
List<String> codes = Arrays.asList ("DOC", "MPEG", "JPEG");

codes.forEach (c -> System.out.print(c + " "));

String fmt = codes.stream()

.filter (s-> s.contains ("PEG"))

.reduce((s, t) -> s + t).get();

System.out.println("\n" + fmt);
```

What is the result?

A. DOC MPEG JPEG
MPEGJPEG
B. DOC MPEG MPEGJPEG
MPEGMPEGJPEG
C. MPEGJPEG
MPEGJPEG
D. The order of the output is unpredictable.

Answer: A

Explanation:
Q60. Given the code fragment:

```
List<String> nL = Arrays.asList("Jim", "John", "Jeff");
Function<String, String> funVal = s -> "Hello : ".contact(s);

nL.Stream()

.map(funVal)

.peek(System.out::print);
```

What is the result?

A. Hello : Jim Hello : John Hello : Jeff
B. Jim John Jeff
C. The program prints nothing.
D. A compilation error occurs.

Answer: C

Explanation:
Q61. Given:

```
public interface Moveable<Integer> {

public default void walk (Integer distance) {System.out.println("Walking");)
```

public void run(Integer distance);

}

Which statement is true?

A. Moveablecan be used as below:
Moveable<Integer> animal = n - > System.out.println("Running" + n); animal.run(100);
animal.walk(20);
B. Moveablecan be used as below:
Moveable<Integer> animal = n - >n + 10;
animal.run(100);
animal.walk(20);
C. Moveablecan be used as below:
Moveable animal = (Integer n) - > System.out.println(n); animal.run(100);

Moveable.walk(20);
D. Movablecannot be used in a lambda expression.

Answer: A

Explanation:
Q62. Which two code blocks correctly initialize a Locale variable?

A. Locale loc1 = "UK";
B. Locale loc2 = Locale.getInstance("ru");
C. Locale loc3 = Locale.getLocaleFactory("RU");
D. Locale loc4 = Locale.UK;
E. Locale loc5 = new Locale ("ru", "RU");

Answer: D,E

Explanation:
Q63. Given:

class FuelNotAvailException extends Exception { }

class Vehicle {

void ride() throws FuelNotAvailException {//line n1

System.out.println("Happy Journey!");

}

}

class SolarVehicle extends Vehicle {

public void ride () throws Exception {//line n2

super ride ();

}

}

and the code fragment:

public static void main (String[] args) throws FuelNotAvailException, Exception {

Vehicle v = new SolarVehicle ();
V. ride();
}
Which modification enables the code fragment to print Happy Journey!?
A. Replaceline n1withpublic void ride() throws FuelNotAvailException {
B. Replaceline n1withprotected void ride() throws Exception {
C. Replaceline n2withvoid ride() throws Exception {
D. Replaceline n2withprivate void ride() throws FuelNotAvailException {

Answer: B

Explanation:
Q64. Given the definition of the Emp class:

public class Emp

private String eName;

private Integer eAge;

Emp(String eN, Integer eA) {

this.eName = eN;

this.eAge = eA;

}

public Integer getEAge () {return eAge;}

public String getEName () {return eName;}

}

and code fragment:

List<Emp>li = Arrays.asList(new Emp("Sam", 20), New Emp("John", 60), New Emp("Jim", 51));

Predicate<Emp> agVal = s -> s.getEAge() > 50;//line n1

li = li.stream().filter(agVal).collect(Collectors.toList());

Stream<String> names = li.stream()map.(Emp::getEName);//line n2

names.forEach(n -> System.out.print(n + " "));

What is the result?

A. Sam John Jim
B. John Jim
C. A compilation error occurs atline n1.
D. A compilation error occurs atline n2.

Answer: B

Explanation:
Q65. For which three objects must a vendor provide implementations in its JDBC driver?

A. Time

B. Date
C. Statement
D. ResultSet
E. Connection
F. SQLException
G. DriverManager

Answer: C,D,E

Explanation: Database vendors support JDBC through the JDBC driver interface or through the ODBC connection. Each driver must provide implementations of java.sql.Connection, java.sql.Statement, java.sql.PreparedStatement, java.sql.CallableStatement, and java.sql.Re sultSet. They must also implement the java.sql.Driver interface for use by the generic java.sql.DriverManager interface.

Q66. Given the code fragment:

```
LocalDate valentinesDay =LocalDate.of(2015, Month.FEBRUARY, 14);

LocalDate nextYear = valentinesDay.plusYears(1);

nextYear.plusDays(15); //line n1

System.out.println(nextYear);
```

What is the result?

A. 2016-02-14
B. ADateTimeExceptionis thrown.
C. 2016-02-29
D. A compilation error occurs atline n1.

Answer: A

Explanation:
Q67. Given the code fragment:

```
BiFunction<Integer, Double, Integer> val = (t1, t2) -> t1 + t2;//line n1

System.out.println(val.apply(10, 10.5));
```

What is the result?

A. 20
B. 20.5
C. A compilation error occurs atline n1.
D. A compilation error occurs atline n2.

Answer: C

Explanation:

Q68. Which statement is true about java.time.Duration?

A. It tracks time zones.
B. It preserves daylight savingtime.
C. It defines time-based values.
D. It defines date-based values.

Answer: C

Reference:http://tutorials.jenkov.com/java-date-time/duration.html#accessing-the-time-of-a- duration

Q69. Given the code fragment:

UnaryOperator<Integer> uo1 = s -> s*2;line n1

List<Double> loanValues = Arrays.asList(1000.0, 2000.0);

loanValues.stream()

.filter(lv -> lv >= 1500)

.map(lv -> uo1.apply(lv))

.forEach(s -> System.out.print(s + " ")); line n2

What is the result?

A. 4000.0
B. 4000
C. A compilation error occurs atline n1.
D. A compilation error occurs atline n2.

Answer: D

Explanation:

Q70. You have been asked to create a ResourceBundle which uses a properties file to localize an application.

Which code example specifies valid keys of menu1 and menu2 with values of File Menu and View Menu?

A. <key name = `menu1">File Menu</key>
<key name = `menu2">View Menu</key>
B. <key>menu1</key><value>File Menu</value>
<key>menu2</key><value>View Menu</value>
C. menu1, File Menu, menu2, View Menu
D. menu1 = File Menu
menu2 = View Menu

Answer: B

Explanation:
Q71. Given the records from the Employee table:

eid	ename
111	Tom
112	Jerry
113	Donald

and given the code fragment:

try {

Connection conn = DriverManager.getConnection (URL, userName, passWord);

Statement st = conn.createStatement(ResultSet.TYPE_SCROLL_INSENSITIVE,

ResultSet.CONCUR_UPDATABLE);

st.execute("SELECT*FROM Employee");

ResultSet rs = st.getResultSet();

```
while (rs.next()) {

if (rs.getInt(1) ==112) {

rs.updateString(2, "Jack");

}

}

rs.absolute(2);

System.out.println(rs.getInt(1) + " " + rs.getString(2));

} catch (SQLException ex) {

System.out.println("Exception is raised");

}
```

Assume that:

The required database driver is configured in the classpath.

The appropriate database accessible with the URL, userName, and passWord exists.

What is the result?

A. The Employee table is updated with the row:
112 Jack
and the program prints:
112 Jerry
B. The Employee table is updated with the row:
112 Jack
and the program prints:
112 Jack
C. The Employee table is not updated and the program prints:
112 Jerry
D. The program printsException is raised.

Answer: D

Explanation:
Q72. Given:

```
class RateOfInterest {

public static void main (String[] args) {

int rateOfInterest = 0;

String accountType = "LOAN";

switch (accountType) {

case "RD";

rateOfInterest = 5;

break;
```

```
case "FD";

rateOfInterest = 10;

break;

default:

assert false: "No interest for this account"; //line n1

}

System.out.println ("Rate of interest:" + rateOfInterest);

}

}
```

and the command:

瑾

What is the result?

A. Rate of interest: 0
B. An AssertionError is thrown.
C. No interest for this account
D. A compilation error occurs atline n1.

Answer: A

Explanation:

Q73. Given the code fragment:

```
class CallerThread implements Callable<String> {

String str;

public CallerThread(String s) {this.str=s;}

public String call() throws Exception {

return str.concat("Call");

}

}
```

and

```
public static void main (String[] args) throws InterruptedException, ExecutionException

{

ExecutorService es = Executors.newFixedThreadPool(4); //line n1

Future f1 = es.submit (newCallerThread("Call"));

String str = f1.get().toString();
```

```
System.out.println(str);

}
```

Which statement is true?

A. The program printsCall Calland terminates.
B. The programprintsCall Calland does not terminate.
C. A compilation error occurs atline n1.
D. AnExecutionExceptionis thrown at run time.

Answer: B

Explanation:

Q74. Given the code fragment:

```
public class FileThread implements Runnable {

String fName;

public FileThread(String fName) { this.fName = fName; }

public void run (){System.out.println(fName);}

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

ExecutorService executor = Executors.newCachedThreadPool();

Stream<Path> listOfFiles = Files.walk(Paths.get("Java Projects"));

listOfFiles.forEach(line -> {

executor.execute(new FileThread(line.getFileName().toString())); //

line n1

});

executor.shutdown();

executor.awaitTermination(5, TimeUnit.DAYS);//

line n2

}

}
```

The Java Projects directory exists and contains a list of files.

What is the result?

A. The program throws a runtime exception atline n2.
B. The program prints files names concurrently.
C. The program prints files names sequentially.
D. A compilation error occurs atline n1.

Answer: C

Explanation:

Q75. Given:

```
class CheckClass {

public static int checkValue (String s1, String s2) {

return s1 length()-s2.length();

}

}
```

and the code fragment:

```
String[] strArray = new String [] {"Tiger", "Rat", "Cat", "Lion"}

//line n1

for (String s : strArray) {

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

}
```

Which code fragment should be inserted at line n1 to enable the code to print Rat Cat Lion Tiger?

A. Arrays.sort(strArray, CheckClass : : checkValue);
B. Arrays.sort(strArray, (CheckClass : : new) : : checkValue);
C. Arrays.sort(strArray, (CheckClass : : new).checkValue);
D. Arrays.sort(strArray, CheckClass : : new : : checkValue);

Answer: A

Explanation:
Q76. Given the code fragments:

```
class TechName {

String techName;

TechName (String techName) {

this.techName=techName;

}

}
```

and

```
List<TechName> tech = Arrays.asList (

new TechName("Java-"),

new TechName("Oracle DB-"),

new TechName("J2EE-")

);
```

Stream<TechName> stre = tech.stream();

//line n1

Which should be inserted at line n1 to print Java-Oracle DB-J2EE-?

A. stre.forEach(System.out::print);
B. stre.map(a-> a.techName).forEach(System.out::print);
C. stre.map(a-> a).forEachOrdered(System.out::print);
D. stre.forEachOrdered(System.out::print);

Answer: B

Explanation:
Q77. Given that /green.txt and /colors/yellow.txt are accessible, and the code fragment:

Path source = Paths.get("/green.txt);

Path target = Paths.get("/colors/yellow.txt);

Files.move(source, target, StandardCopyOption.ATOMIC_MOVE);

Files.delete(source);

Which statement is true?

A. Thegreen.txtfile content is replaced by theyellow.txtfile content and theyellow.txtfile is deleted.
B. Theyellow.txtfile content is replaced by thegreen.txtfile content and an exception is thrown.
C. The filegreen.txtis moved to the/colorsdirectory.
D. AFileAlreadyExistsExceptionis thrown at runtime.

Answer: B

Explanation:
Q78. Given:

interface Doable {

public void doSomething (String s);

}

Which two class definitions compile?

A. public abstract class Task implements Doable {
public void doSomethingElse(String s) { }
}
B. public abstract class Work implements Doable {
public abstract void doSomething(String s) { }
public void doYourThing(Boolean b) { }
}
C. public class Job implements Doable {
public void doSomething(Integer i) { }
}
D. public class Action implements Doable {
public void doSomething(Integer i) { }
public String doThis(Integer j) { }
}
E. public class Do implements Doable {
public void doSomething(Integer i) { }

```
public void doSomething(String s) { }
public void doThat (String s) { }
}
```

Answer: E

Explanation:

Q79. Given the code fragment:

```
List<Integer> list1 = Arrays.asList(10, 20);

List<Integer> list2 = Arrays.asList(15, 30);

//line n1
```

Which code fragment, when inserted at line n1, prints 10 20 15 30?

```
A. Stream.of(list1, list2)
.flatMap(list -> list.stream())
.forEach(s -> System.out.print(s + " "));
B. Stream.of(list1, list2)
.flatMap(list -> list.intStream())
.forEach(s -> System.out.print(s + " "));
C. list1.stream()
.flatMap(list2.stream().flatMap(e1 -> e1.stream())
.forEach(s -> System.out.println(s + " "));
D. Stream.of(list1, list2)
.flatMapToInt(list -> list.stream())
.forEach(s -> System.out.print(s + " "));
```

Answer: A

Explanation:

Q80. Given the code fragment:

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

BufferedReader brCopy = null;

try (BufferedReader br = new BufferedReader (new FileReader("employee.txt"))) { //

line n1

br.lines().forEach(c -> System.out.println(c));

brCopy = br;//line n2

}

brCopy.ready(); //line n3;

}
```

Assume that the ready method of the BufferedReader, when called on a closed BufferedReader, throws an exception, and employee.txt is accessible and contains valid text.

What is the result?

A. A compilation error occurs atline n3.

B. A compilation error occurs atline n1.
C. A compilation error occurs atline n2.
D. The code prints the content of theemployee.txtfile and throws an exception atline n3.

Answer: D

Explanation:
Q81. Given:

Book.java:

```
public class Book {

private String read(String bname) { return "Read" + bname }

}
```

EBook.java:

```
public class EBook extends Book {

public String read (String url) { return "View" + url }

}
```

Test.java:

```
public class Test {

public static void main (String[] args) {

Book b1 = new Book();

b1.read("Java Programing");

Book b2 = new EBook();

b2.read("http://ebook.com/ebook");

}

}
```

What is the result?

A. Read Java Programming
View http:/ ebook.com/ebook
B. Read Java Programming
Read http:/ ebook.com/ebook
C. TheEBook.javafile fails to compile.
D. TheTest.javafile fails to compile.

Answer: D

Explanation:
Q82. Given the code fragment:

```
ZonedDateTime depart = ZonedDateTime.of(2015, 1, 15, 3, 0, 0, 0, ZoneId.of("UTC-7"));

ZonedDateTime arrive = ZonedDateTime.of(2015, 1, 15, 9, 0, 0, 0, ZoneId.of("UTC-5"));
```

long hrs = ChronoUnit.HOURS.between(depart, arrive); //line n1

System.out.println("Travel time is" + hrs + "hours");

What is the result?

A. Travel time is 4 hours
B. Travel time is 6 hours
C. Travel time is 8 hours
D. An exception is thrown atline n1.

Answer: A

Explanation:
Q83. Given the code fragment:

Path path1 = Paths.get("/app/./sys/");

Path res1 = path1.resolve("log");

Path path2 = Paths.get("/server/exe/");

Path res2 = path1.resolve("/readme/");

System.out.println(res1);

System.out.println(res2);

What is the result?

A. /app/sys/log
/readme/server/exe
B. /app/log/sys
/server/exe/readme
C. /app/./sys/log
/readme
D. /app/./sys/log
/server/exe/readme

Answer: C

Explanation:
Q84. Given the code fragment:

List<String> colors = Arrays.asList("red", "green", "yellow");

Predicate<String> test = n - > {

System.out.println("Searching...");

return n.contains("red");

};

colors.stream()

.filter(c -> c.length() > 3)

.allMatch(test);

What is the result?

A. Searching...
B. Searching...
Searching...
C. Searching...
Searching...
Searching...
D. A compilation error occurs.

Answer: A

Explanation:
Q85. Given:

```
class UserException extends Exception { }

class AgeOutOfLimitException extends UserException { }
```

and the code fragment:

```
class App {

public void doRegister(String name, int age)

throws UserException, AgeOutOfLimitException {

if (name.length () < 6) {

throw new UserException ();

} else if (age >= 60) {

throw new AgeOutOfLimitException ();

} else {

System.out.println("User is registered.");

}

}

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

App t = new App ();

t. doRegister("Mathew", 60);
}
}
```

What is the result?
A. User is registered.
B. AnAgeOutOfLimitExceptionis thrown.
C. AUserExceptionis thrown.
D. A compilation error occurs in themainmethod.

Answer: B

Explanation: