ME-Q1) Given the following definition of the classes Animal, Lion, and Jumpable, select the correct combinations of assignments of a variable that don't result in compilation errors or runtime exceptions (select 2 options).

```
interface Jumpable {} class Animal {}
class Lion extends Animal implements Jumpable {}
```

a Jumpable var1 = new Jumpable();
b Animal var2 = new Animal();
c Lion var3 = new Animal();
d Jumpable var4 = new Animal();
e Jumpable var5 = new Lion();
f Jumpable var6 = (Jumpable)(new Animal());

ME-Q2) Given the following code, which option, if used to replace /* INSERT CODE HERE */, will make the code print 1? (Select 1 option.)

```
try {
String[][] names = {{"Andre", "Mike"}, null, {"Pedro"}}; System.out.println (names[2][1].substring(0, 2));
} catch (/*INSERT CODE HERE*/) { System.out.println(1);
}
```

a IndexPositionException e
b NullPointerException e
c ArrayIndexOutOfBoundsException e
d ArrayOutOfBoundsException e

ME-Q3) What is the output of the following code? (Select 1 option.)

```
public static void main(String[] args) { int a = 10; String name = null;
try {
a = name.length(); //line1
a++; //line2
} catch (NullPointerException e){
++a;
return;
} catch (RuntimeException e){ a--;
return;
} finally {
System.out.println(a);
}
}
```

a 5
b 6
c 10
d 11
e 12
f Compilation error
g No output

h Runtime exception

ME-Q4) Given the following class definition,

class Student { int marks = 10; }

what is the output of the following code? (Select 1 option.)

```
class Result {
public static void main(String... args) { Student s = new Student();
switch (s.marks) {
default: System.out.println("100"); case 10: System.out.println("10");
case 98: System.out.println("98");
}
}
}
```

a 100
10
98
b 10
98
c 100
d 10

ME-Q5) Given the following code, which code can be used to create and initialize an object of the class ColorPencil? (Select 2 options.)

```
class Pencil {}
class ColorPencil extends Pencil { String color;
ColorPencil(String color) {this.color = color;}
}
```

a ColorPencil var1 = new ColorPencil();
b ColorPencil var2 = new ColorPencil(RED); **c** ColorPencil var3 = new ColorPencil("RED"); **d** Pencil var4 = new ColorPencil("BLUE");

ME-Q6) What is the output of the following code? (Select 1 option.)

```
class Doctor { protected int age;
protected void setAge(int val) { age = val; } protected int getAge() { return age; }
}
class Surgeon extends Doctor { Surgeon(String val) {
specialization = val;
}
String specialization;
String getSpecialization() { return specialization; }
}
class Hospital {
public static void main(String args[]) { Surgeon s1 = new Surgeon("Liver"); Surgeon s2 = new Surgeon("Heart"); s1.age = 45;
System.out.println(s1.age + s2.getSpecialization()); System.out.println(s2.age + s1.getSpecialization());
}
}
```

a 45Heart 0Liver
b 45Liver 0Heart
c 45Liver 45Heart
d 45Heart 45Heart
e Class fails to compile.

ME-Q7) What is the output of the following code? (Select 1 option.)

```
class RocketScience {
public static void main(String args[]) { int a = 0;
while (a == a++) { a++;
System.out.println(a);
}
}
}
```

a The while loop won't execute; nothing will be printed.
b The while loop will execute indefinitely, printing all numbers, starting from 1.
c The while loop will execute indefinitely, printing all even numbers, starting from 0.
d The while loop will execute indefinitely, printing all even numbers, starting from 2.
e The while loop will execute indefinitely, printing all odd numbers, starting from 1.
f The while loop will execute indefinitely, printing all odd numbers, starting from 3.

ME-Q8) Given the following statements,

- com.ejava is a package
- class Person is defined in package com.ejava
- class Course is defined in package com.ejava

which of the following options correctly import the classes Person and Course in the class MyEJava? (Select 3 options.)
a import com.ejava.*; class MyEJava {}
b import com.ejava; class MyEJava {}
c import com.ejava.Person; import com.ejava.Course; class MyEJava {}
d import com.ejava.Person; import com.ejava.*; class MyEJava {}

ME-Q9) Given that the following classes Animal and Forest are defined in the same package, examine the code and select the correct statements (select 2 options).

```
line1> class Animal {
line2> public void printKing() { line3> System.out.println("Lion"); line4> }
line5> }
line6> class Forest {
line7> public static void main(String... args) { line8> Animal anAnimal = new Animal();
line9> anAnimal.printKing();
line10> }
line11> }
```

a The class Forest prints Lion.
b If the code on line 2 is changed as follows, the class Forest will print Lion:
private void printKing() {
c If the code on line 2 is changed as follows, the class Forest will print Lion:
void printKing() {
d If the code on line 2 is changed as follows, the class Forest will print Lion:
default void printKing() {

ME-Q10) Given the following code,

```
class MainMethod {
public static void main(String... args) { System.out.println(args[0]+":"+ args[2]);
}
}
```

what is its output if it's executed using the following command? (Select 1 option.)
java MainMethod 1+2 2*3 4-3 5+1

a java:1+2
b java:3
c MainMethod:2*3 **d** MainMethod:6 **e** 1+2:2*3
f 3:3
g 6
h 1+2:4-3
i 31
j 4

ME-Q11) What is the output of the following code? (Select 1 option.)
interface Moveable {
int move(int distance);
}
class Person {
static int MIN_DISTANCE = 5; int age;
float height; boolean result; String name;
}
public class EJava {
public static void main(String arguments[]) { Person person = new Person();
Moveable moveable = (x) -> Person.MIN_DISTANCE + x; System.out.println(person.name +
person.height + person.result
+ person.age + moveable.move(20));
}
}
a null0.0false025
b null0false025
c null0.0ffalse025
d 0.0false025
e 0false025
f 0.0ffalse025
g null0.0true025
h 0true025
i 0.0ftrue025
j Compilation error
k Runtime exception

ME-Q12) Given the following code, which option, if used to replace /* INSERT CODE HERE */, will make the code print the value of the variable pagesPerMin? (Select 1 option.)
class Printer { int inkLevel;
}
class LaserPrinter extends Printer { int pagesPerMin;
public static void main(String args[]) { Printer myPrinter = new LaserPrinter(); System.out.println(/*
INSERT CODE HERE */);
}
}
a (LaserPrinter)myPrinter.pagesPerMin
b myPrinter.pagesPerMin
c LaserPrinter.myPrinter.pagesPerMin
d ((LaserPrinter)myPrinter).pagesPerMin
ME-Q13) What is the output of the following code? (Select 1 option.)
interface Keys {

```
String keypad(String region, int keys);
}
public class Handset {
public static void main(String... args) { double price;
String model;
Keys varKeys = (region, keys) ->
{if (keys >= 32)
return region; else return "default";}; System.out.println(model + price + varKeys.keypad("AB", 32));
}
}
```
a null0AB
b null0.0AB
c null0default
d null0.0default
e 0
f 0.0
g Compilation error

ME-Q14) What is the output of the following code? (Select 1 option.)
```
public class Sales {
public static void main(String args[]) { int salesPhone = 1;
System.out.println(salesPhone++ + ++salesPhone +
++salesPhone);
}
}
```
a 5
b 6
c 8
d 9

ME-Q15) Which of the following options defines the correct structure of a Java class that compiles successfully? (Select 1 option.)

a
```
package com.ejava.guru; package com.ejava.oracle; class MyClass {
int age = /* 25 */ 74;
}
```
b
```
import com.ejava.guru.*; import com.ejava.oracle.*; package com.ejava;
class MyClass {
String name = "e" + "Ja /*va*/ v";
}
```
c
```
class MyClass {
import com.ejava.guru.*;
}
```
d
```
class MyClass {
int abc;
String course = //this is a comment "eJava";
}
```
e None of the above

ME-Q16) What is the output of the following code? (Select 1 option.)
```
class OpPre {
public static void main(String... args) { int x = 10;
int y = 20;
```

```
int z = 30;
if (x+y%z > (x+(-y)*(-z))) { System.out.println(x + y + z);
}
}
}
```
a 60
b 59
c 61
d No output.
e The code fails to compile.

ME-Q17) Select the most appropriate definition of the variable name and the line num- ber on which it should be declared so that the following code compiles successfully (choose 1 option).

```
class EJava {
// LINE 1
public EJava() { System.out.println(name);
}
void calc() {
// LINE 2
if (8 > 2) {
System.out.println(name);
}
}
public static void main(String... args) {
// LINE 3
System.out.println(name);
}
}
```

a Define static String name; on line 1.
b Define String name; on line 1.
c Define String name; on line 2.
d Define String name; on line 3.

ME-Q18) Examine the following code and select the correct statement (choose 1 option).

```
line1> class Emp {
line2> Emp mgr = new Emp(); line3> }
line4> class Office {
line5> public static void main(String args[]) { line6> Emp e = null;
line7> e = new Emp();
line8> e = null; line9> }
line10> }
```

a The object referred to by object e is eligible for garbage collection on line 8.
b The object referred to by object e is eligible for garbage collection on line 9.
c The object referred to by object e isn't eligible for garbage collection because its member variable mgr isn't set to null.
d The code throws a runtime exception and the code execution never reaches line 8 or line 9.

ME-Q19) Given the following,

```
long result;
```

which options are correct declarations of methods that accept two String arguments and an int argument and whose return value can be assigned to the variable result? (Select 3 options.)
a Short myMethod1(String str1, int str2, String str3)
b Int myMethod2(String val1, int val2, String val3)
c Byte myMethod3(String str1, str2, int a)
d Float myMethod4(String val1, val2, int val3)
e Long myMethod5(int str2, String str3, String str1)
f Long myMethod6(String... val1, int val2)
g Short myMethod7(int val1, String... val2)

ME-Q20) Which of the following will compile successfully? (Select 3 options.)
a int eArr1[] = {10, 23, 10, 2};
b int[] eArr2 = new int[10];
c int[] eArr3 = new int[] {};
d int[] eArr4 = new int[10] {};
e int eArr5[] = new int[2] {10, 20};
ME-Q21) Assume that Oracle has asked you to create a method that returns the con- catenated value of two String objects. Which of the following methods can accom- plish this job? (Select 2 options.)
a public String add(String 1, String 2) { return str1 + str2;
}
b private String add(String s1, String s2) { return s1.concat(s2);
}
c protected String add(String value1, String value2) { return value2.append(value2);
}
d String subtract(String first, String second) { return first.concat(second.substring(0));
}