Q1

Given:

1. **public abstract class** Shape {

2. **private int** x;

3. **private int** y;

4. **public abstract void** draw();

5. **public void** setAnchor(**int** x, **int** y) {

6. **this**.x = x;

7. **this**.y = y;

8. }

9. }

Which two classes use the Shape class correctly? (Choose two.)

A. **public class** Circle **implements** Shape {

**private int** radius;

}

B. **public abstract class** Circle **extends** Shape {

**private int** radius;

}

C. **public class** Circle **extends** Shape {

**private int** radius;

**public void** draw();

}

D. **public abstract class** Circle **implements** Shape {

**private int** radius;

**public void** draw();

}

E. **public class** Circle **extends** Shape {

**private int** radius;

**public void** draw() {/\* code here \*/}

}

F. **public abstract class** Circle **implements** Shape {

**private int** radius;

**public void** draw() {/\* code here \*/}

}

Q2.

1. **public class** Barn {

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

3. **new** Barn().go("hi", 1);

4. **new** Barn().go("hi", "world", 2);

5. }

6. **public void** go(String... y, **int** x) {

7. System.out.print(y[y.length - 1] + " ");

8. }

9. }

What is the result?

A. hi hi

B. hi world

C. world world

D. Compilation fails.

E. An exception is thrown at runtime.

Q3.

Given:

1. **class** Nav{

2. **public enum** Direction { NORTH, SOUTH, EAST, WEST }

3. }

1. **public class** Sprite{

2. // insert code here

3. }

Which code, inserted at line 14, allows the Sprite class to compile?

A. Direction d = NORTH;

B. Nav.Direction d = NORTH;

C. Direction d = Direction.NORTH;

D. Nav.Direction d = Nav.Direction.NORTH;

Q4.

Which statement is true about the classes and interfaces in the exhibit?

1. **public interface** A {

2. **public void** doSomething(String thing);

3. }

1. **public class** AImpl **implements** A {

2. **public void** doSomething(String msg) {}

3. }

1. **public class** B {

2. **public** A doit(){

3. //more code here

4. }

5. **public** String execute(){

6. //more code here

7. }

8. }

1. **public class** C **extends** B {

2. **public** AImpl doit(){

3. //more code here

4. }

5.

6. **public** Object execute() {

7. //more code here

8. }

9. }

A. Compilation will succeed for all classes and interfaces.

B. Compilation of class C will fail because of an error in line 2.

C. Compilation of class C will fail because of an error in line 6.

D. Compilation of class AImpl will fail because of an error in line 2.

Q5.

Click the Exhibit button.

What is the result?

**11. public class** Person {

12. String name = "No name";

13. **public** Person(String nm) { name = nm; }

14. }

15.

**16. public class** Employee **extends** Person {

17. String empID = "0000";

18. **public** Employee(String id) { empID = id; }

19. }

20.

**21. public class** EmployeeTest {

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

23. Employee e = **new** Employee("4321");

24. System.out.println(e.empID);

25. }

26. }

A. 4321

B. 0000

C. An exception is thrown at runtime.

D. Compilation fails because of an error in line 18.

Q6.

Given:

1. **public class** Rainbow {

2. **public enum** MyColor {

3. RED(0xff0000), GREEN(0x00ff00), BLUE(0x0000ff);

4. **private final int** rgb;

5. MyColor(**int** rgb) { **this**.rgb = rgb; }

6. **public int** getRGB() { **return** rgb; }

7. };

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

9. //insert code here

10. }

11.}

Which code fragment, inserted at line 19, allows the Rainbow class to compile?

A. MyColor skyColor = BLUE;

B. MyColor treeColor = MyColor.GREEN;

C. if(RED.getRGB() < BLUE.getRGB()) { }

D. Compilation fails due to other error(s) in the code.

E. MyColor purple = new MyColor(0xff00ff);

F. MyColor purple = MyColor.BLUE + MyColor.RED;

Q7.

Given:

22. StringBuilder sb1 = new StringBuilder("123");

23. String s1 = "123";

24. // insert code here

25. System.out.println(sb1 + " " + s1);

Which code fragment, inserted at line 24, outputs "123abc 123abc"?

A. sb1.append("abc"); s1.append("abc");

B. sb1.append("abc"); s1.concat("abc");

C. sb1.concat("abc"); s1.append("abc");

D. sb1.concat("abc"); s1.concat("abc");

E. sb1.append("abc"); s1 = s1.concat("abc");

F. sb1.concat("abc"); s1 = s1.concat("abc");

G. sb1.append("abc"); s1 = s1 + s1.concat("abc");

H. sb1.concat("abc"); s1 = s1 + s1.concat("abc");

Q8.

Given:

**interface** TestA { String toString(); }

**public class** Test {

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

System.out.println(**new** TestA() {

**public** String toString() { **return** "test"; }

});

}

}

What is the result?

A. test

B. null

C. An exception is thrown at runtime.

D. Compilation fails because of an error in line 1.

E. Compilation fails because of an error in line 4.

F. Compilation fails because of an error in line 5.

Q9.

Given:

1. **public static void** parse(String str) {

2. **try** {

3. **float** f = Float.parseFloat(str);

4. } **catch** (NumberFormatException nfe) {

5. f = 0;

6. } **finally** {

7. System.out.println(f);

8. }

9. }

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

11. parse("invalid");

12. }

What is the result?

A. 0.0

B. Compilation fails.

C. A ParseException is thrown by the parse method at runtime.

D. A NumberFormatException is thrown by the parse method at runtime.

Q10

Given that adder() returns an int, which are valid Predicate lambdas? (Choose all that apply.)



Q11



Which are true? (Choose all that apply.)

A. The output is: 2018-08-16 2018-08-17 2018-08-18

B. The output is: 2018-08-16 2018-08-18 2018-08-17

C. The output is: 2018-08-16 2018-08-17 2018-08-17

D. At line X, zero LocalDate objects are eligible for garbage collection

E. At line X, one LocalDate object is eligible for garbage collection

F. At line X, two LocalDate objects are eligible for garbage collection

G. Compilation fails

Q12.

Given:

**1. class** Super {

2. **private int** a;

3. **protected** Super(**int** a) { **this**.a = a; }

4. }

**11. class** Sub **extends** Super {

12. **public** Sub(**int** a) { **super**(a); }

13. **public** Sub() { **this**.a = 5; }

14. }

Which two, independently, will allow Sub to compile? (Choose two.)

A. Change line 2 to:

public int a;

B. Change line 2 to:

protected int a;

C. Change line 13 to:

public Sub() { this(5); }

D. Change line 13 to:

public Sub() { super(5); }

E. Change line 13 to:

public Sub() { super(a); }

Q13.

Given a valid DateFormat object named df, and

16. LocalDate d=null;

17. String ds = "December 15, 2004";

18. //insert code here

What updates d's value with the date represented by ds?(choose two)

A. 18. d = LocalDate.parse(ds);

B. 18. d = LocalDate.parse(ds, DateTimeFormatter.ofPattern(“MMM d, yyyy”));

C. 18 d = LocalDate.parse(ds, DateTimeFormatter.ofPattern(“MMMM d, yyyy”));

D. 18. D = LocalDate.parse(ds, DateTimeFormatter.ofPattern(“MMMM dd, yyyy”));

Q14.

Given:

1. **package** test;

2.

3. **class** Target {

4. **public** String name = "hello";

5. }

What can directly access and change the value of the variable name?

A. any class

B. only the Target class

C. any class in the test package

D. any class that extends Target

Q15.

Which three statements describe the object-oriented features of the Java language? (Choose three.)

A. Objects cannot be reused.

B. A subclass can inherit from a superclass.

C. Objects can share behaviors with other objects.

D. A package must contain more than one class.

E. Object is the root class of all other objects.

F. A main method must be declared in every class.

Q16.

What is the result of the following code?  
4: int total = 0;  
5: StringBuilder letters = new StringBuilder("abcdefg");  
6: total += letters.substring(1, 2).length();  
7: total += letters.substring(6, 6).length();  
8: total += letters.substring(6, 5).length();  
9: System.out.println(total);  
• A. 1  
• B. 2  
• C. 3  
• D. 7  
• E. An exception is thrown.  
• F. The code does not compile.

Q17.  
What is the result of the following code?  
4: int total = 0;  
5: StringBuilder letters = new StringBuilder("abcdefg");  
6: total += letters.substring(1, 2).length();  
7: total += letters.substring(6, 6).length();  
8: total += letters.substring(6, 5).length();  
9: System.out.println(total);  
• A. 1  
• B. 2  
• C. 3  
• D. 7  
• E. An exception is thrown.  
• F. The code does not compile.

Q18.

What is the output of the following code?  
LocalDate date = LocaleDate.of(2018, Month.APRIL, 40);  
System.out.println(date.getYear() + " " + date.getMonth() + " " + date.getDayOfMonth());  
• A. 2018 APRIL 4  
• B. 2018 APRIL 30  
• C. 2018 MAY 10  
• D. Another date.  
• E. The code does not compile.  
• F. A runtime exception is thrown.

Q19.

What is the output of the following code?  
LocalDateTime d = LocaleDateTime.of(2015, 5, 10, 11, 22, 33);  
Period p = Period.ofDays(1).ofYears(2);  
d = d.minus(p);  
DateTimeFormatter f = DateTimeFormatter.ofLocalizedTime(FormatStyle.SHORT);  
System.out.println(d.format(f));  
• A. 5/9/13 11:22 AM  
• B. 5/10/13 11:22 AM  
• C. 5/9/14  
• D. 5/10/14  
• E. 11:22 AM  
• F. The code does not compile.  
• G. A runtime exception is thrown.

Q20.

Given:  
try { int x = Integer.parseInt(“two”); }  
Which could be used to create an appropriate catch block? (Choose all that apply.)  
A. ClassCastException  
B. IllegalStateException  
C. NumberFormatException  
D. IllegalArgumentException  
E. ExceptionInInitializerError  
F. ArrayIndexOutOfBoundsException