1. Which of the following statements is true about arrays in Java?

R: An array has a fixed size

2. ¿Cuál de los siguientes componentes es parte de una solicitud HTTP?

R: URL, headers, cuerpo de la solicitud

```
3. public class LogicQuestion {
 public static void main(String[] args) {
 int count = 0;
 for (int i = 0; i < 10; i++) {
 if (i % 2 == 0) {
 count++;
 }
 }
 System.out.println("Count: " + count);
 }
}
```

R:Count 5

4. Which method is used to sort elements of a List in natural order in Java? (*)

R:Collections.sort()

5. ¿Qué significa que un cambio en el software sea retro-compatible?

R: El cambio garantiza que el software será funcional sin necesidad de modificaciones en el código que depende de él

```
6. class Base {
 public Base() {
 System.out.println("Base constructor");
 }
 public Base(String message) {
 System.out.println("Base constructor with message: " + message);
 }
}
class Derived extends Base {
 public Derived() {
 super("Hello");
 System.out.println("Derived constructor");
 }
}
public class Test {
 public static void main(String[] args) {
 Derived derived = new Derived();
 }
}
```

R: Base constructor with message: Hello
Derived constructor

7. ¿Cuál es la principal función de Jfrog Artifactory en un entorno de desarrollo de software?
R: Gestionar y almacenar artefactos de software, como dependencias y bibliotecas, de manera centralizada

8. ¿Cuál de los siguientes métodos de Mockito se utiliza para verificar que un método de un mock ha sido llamado un número específico de veces? (*)
R: verify() / verify(mock, times(n))

9.
```
abstract class Animal {
 public abstract void makeSound();
}
class Dog extends Animal {
 @Override
 public void makeSound() {
 System.out.println("Bark!");
 }
}
public class Test {
 public static void main(String[] args) {
 Animal myDog = new Dog();
 myDog.makeSound();
 }
}
```
R: Bark!

10: ¿Cuál es el propósito principal de la anotación @Test en JUnit?
R: Marcar un método como un método de prueba

11. ¿Cuál de las siguientes características es fundamental en una base de datos relacional?
R: Organización de datos en tablas con filas y columnas

12. Which line of code will compile successfully without any additional import statements?
```
public class Program{
 public static void main(String[] args) {
 // Line A
 String str = "Hello World!";
 // Line B
 ArrayList<String> list = new ArrayList<>();
 // Line C
 File file = new File("example.txt");
 // Line D
 URL url = new URL("http://example.com");
 }
}
```
R: Line A

```
13. abstract class Shape {
 public abstract void draw();
 public void printShape() {
 System.out.println("This is a shape");
 }
}
class Circle extends Shape {
 @Override
 public void draw() {
 System.out.println("Drawing a circle");
 }
}
public class Test {
 public static void main(String[] args) {
 Circle circle = new Circle();
 circle.draw();
 circle.printShape();
 }
}
```

R:Drawing a circle
This is a shape

14: Cuál es el propósito principal del archivo pom.xml en un proyecto Maven?
R: Definir las dependencias, plugins y configuraciones del proyecto

15: Which section in the pom.xml file specifies the external libraries and dependencies required by the project?
R: <dependencies>

16. What will be the output of the following code snippet?

```
public class ScopeTest {
 private int value = 10;

 public void printValue() {
 int value = 20;
 System.out.println(this.value);
 }

 public static void main(String[] args) {
 ScopeTest test = new ScopeTest();
 test.printValue();
 }
}
```

R: 10

17: What is the primary purpose of a Data Transfer Object (DTO) in software design?
R: To transfer data between different layers or tiers of an application

18: Which declaration correctly initializes a boolean variable in Java?

a. boolean f = "true";

b. boolean f = () => f;

c. boolean f = (1 + 0);

d. boolean d = (a < b);

e. boolean b = 0 < 1;

f. boolean a = (10 > 5 && 2 < 3);

R: e y f

19: Which of the following statements about the 'throw' keyword is true?

R: It is used to manually throw an exception

20. Which of the following code snippets will throw a ClassCastException

```
a. class A {}
class B extends A {}
public class Test {
 public static void main(String[] args) {
 B obj = new B();
 A a = (A) obj;
 }
}
```

```
b. class A {}
class B extends A {}
public class Test {
 public static void main(String[] args) {
 A obj = new B();
 A a = (A) obj;
 }
}
```

```
c. class A {}
class B extends A {}
public class Test {
 public static void main(String[] args) {
 A obj = new B();
 B b = (B) obj;
 }
}
```

```
d. class A {}
class B extends A {}
public class Test {
 public static void main(String[] args) {
 A obj = new A();
 B b = (B) obj;
 }
}
```

R: d.

21. ¿Cuál de las siguientes afirmaciones es correcta sobre la aserción assert() en jUnit?
R: Se utiliza para verificar que una condición es verdadera

22. ¿Cual es la función principal del JDK (Java Development Kit)?
R: Ofrecer herramientas necesarias para compilar, depurar y ejecutar aplicaciones Java.

23. Which of the following statements accurately describe the differences between Comparator and Comparable interfaces in Java?
R: All of the above.

24. What is the purpose of the "throws" keyword in a method declaration in Java?
R: To indicate the exceptions that the method can throw to the caller.

25. ¿Cuáles de los siguientes comandos de Git se utilizan para gestionar ramas en un repositorio? (Seleccione todas las que correspondan).
R: git checkout, git branch, git merge

26. ¿Cuál de los siguientes patrones de diseño es adecuado para crear una estructura de objetos en forma de árbol para representar jerarquías parte-todo, permitiendo a los clientes tratar objetos individuales y compuestos de manera uniforme?
R: Patrón Compuesto (Composite Pattern).

27. Which file is used to configure user specific settings in Maven?
R: settings.xml

28. What will be the output of the following code snippets?

```
import java.util.ArrayList;
import java.util.List;
public class GenericTest {
 public static <T> void addIfAbsent(List<T> list, T element) {
 if (!list.contains(element)) {
 list.add(element);
 }
 }
 public static void main(String[] args) {
 List<String> items = new ArrayList<>();
 items.add("apple");
 items.add("banana");
 addIfAbsent(items, "cherry");
 addIfAbsent(items, "apple");
 System.out.println(items);
 }
}
```

R: [apple, banana, cherry]

29. What will be the output of the following code snippet?

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

```
 String str1 = "Hello";
 String str2 = "World";
 String str3 = str1 + " " + str2;
 String str4 = str1.concat(" ").concat(str2);
 String str5 = new StringBuilder().append(str1).append(" ").append(str2).toString();
 System.out.println(str1.equals(str2) + " ");
 System.out.println(str3.equals(str4) + " ");
 System.out.println(str3 == str5 + " ");
 System.out.println(str4 == str5);
 }
}
```

R: false true false false

30. Which of the following code snippets will result in a compilation error when implementing the vehicle interface?

```
interface Vehicle{
void start();
void stop(); //public abstract
}
```

```
R: public class Bike implements Vehicle {
 public void start() { ... }
 void stop() { ... }
}
```

31. What will be the output of the following code snippet?

```
public class StaticNonStaticBlockTest {
 static {
 System.out.println("Static block");
 }
 {
 System.out.println("Instance block");
 }
 public StaticNonStaticBlockTest() {
 System.out.println("Constructor");
 }
 public static void staticMethod() {
 System.out.println("Static method");
 }
 public static void main(String[] args) {
 StaticNonStaticBlockTest test = new StaticNonStaticBlockTest();
 new StaticNonStaticBlockTest();
 }
}
```

R:

```
Static block
Instance block
```

Constructor
Instance block
Constructor

32. En el contexto de bases de datos relacionales, si una transacción cumple con la propiedad de Aislamiento (Isolation) del principio ACID, esto significa que:
R: Las transacciones se ejecutan como si fueran la única operación en el sistema, sin interferencia de otras transacciones concurrentes.

33. Which of the following statements accurately describe the relationships that can exist between classes in Java?

a. Both inheritance and composition can be used together to model complex relationships.
b. Inheritance represents an "is-a" relationship where one class derives from another class.
c. Composition represents a "has-a" relationship where one class contains an instance of another class.
d. Composition should be preferred over inheritance to promote code reuse and flexibility.
e. Usage (or association) represents a "uses-a" relationship where one class uses methods or instances of another class.
f. Inheritance should be preferred over composition to promote code reuse and flexibility
R: a, b, c ,d, e

34. Todo el acceso a los datos debe estar encapsulado en una biblioteca para facilitar la reutilización y control de acceso.
R: Verdadero

35. En el contexto de Maven, ¿Cuál es la función principal del archivo settings.xml?
R: Configurar la información del repositorio local y remoto, así como las credenciales y perfiles de usuario.

36. Where do you configure the plugins used for various build tasks in Maven's pom.xml?
R: <build>

37. What will be the result of the following code execution?

```
import java.util.ArrayList;
import java.util.List;

public class ArrayListTest {
public static void main(String[] args) {
List<Integer> list = new ArrayList<>();
list.add(1);
list.add(2);
list.add(3);
list.remove(1);
System.out.println(list);
}
}
```

R: [1, 3]

38. Which of the following methods can be used to remove all elements from an ArrayList?
R: clear()

39. ¿Cuál es la principal desventaja del antipatrón "contenedor mágico" en el desarrollo de software?
R: Oculta demasiada lógica de negocio en un contenedor genérico, lo que hace que el código sea difícil de entender y depurar.

40. ¿Cuál de las siguientes afirmaciones describe mejor un Step en el contexto de Spring Batch?
R: Un objeto de dominio que encapsula una fase independiente y secuencial de un trabajo por lotes.

41. Which method override is valid given the following classes?

```
class Parent {
void display() {
System.out.println("Parent");
}
}
class Child extends Parent {
// Override here
}
```

R: public void display() { System.out.println("Child");
}

42. ¿Cuál es la rama principal de Github en la que se integran las nuevas funcionalidades antes de lanzarlas a producción?
R: develop

43. ¿Cuál es el comando en Bash para cambiar el directorio actual a uno especificado?
R: cd

44. Which of the following is NOT part of the Agile Software development lifecycle?
R: Documenting

45. What is the result of: "" instanceof Object ?
true — A String is an Object.

46. What is the result of: null instanceof Object?
false — null is not an Object

47. What is the result of: "" instanceof java.util.HashMap?
Does not compile. The compiler recognizes that a String cannot be a HashMap. (This check is enforced for classes but not interfaces.)

48. What does equals() take as a parameter?
Object obj

49. If obj1.equals(obj2) returns true, what is true of hashCode() ?
obj1.hashCode() and obj2.hashCode() must return the same result.

50. True or false? An enum may be compared to an int in a switch statement
False. An enum is only allowed to use the enum value in a case.

51. What is polymorphism?
The property in Java for an object to take on many different forms

52. Why is loose coupling preferred to tight coupling when designing classes?
Loosely coupled classes have fewer dependencies on one another, are more resistant to change, and can be used in more places

53. What is the difference between an object and a reference in Java?
An object is a representation of the entity that exists in memory, while a reference provides a way of pointing to the object.

54. True or false? A functional interface may include any number of static or default methods?
True

55. True or false? An interface requires the @FunctionalInterface annotation to be interpreted by the compiler as a functional interface.
False. The compiler treats any interface with exactly one abstract method as a functional interface.

56. What is the name of the technique for delaying the creation of objects until they are needed?
Lazy instantiation

57. What is an invariant property of an object?
A property that remains true regardless of how the object is modified

58. Name the three method prefixes used to access variables in a JavaBean.
get, set, is

59. How do you modify an immutable object?
You can't, although you can create a copy of the object with similar information

60. Write code to declare an ArrayList of Integer objects using the diamond operator.
ArrayList<Integer> list = new ArrayList<>(); or List<Integer> list = new ArrayList<>(); or Collection<Integer> list = new ArrayList<>();

61. Fill in the blank to make this class Comparable using generics:
public class MyComparable _______________ {
/ imagine implementation here /
}
implements Comparable<MyComparable>

62. Fill in the blank to make this code compile:
public static __________ returnObject (T o) { return o;}
<T> T

63. Does this compile?
List<Object> l = newArrayList<String>();
No. ArrayList implements List, but the generic types are not compatible.

64. Does this compile?
List<List> listOfLists = new ArrayList<? super List>();
No. A lower/upper bound may be specified on the left side of a declaration but not on the right side.

65. Fill in the blank to make this code compile:
List <_______ Number> listOfNumbers = new ArrayList<Object>();
? super (This is a lower bound.)

66. Fill in the blank to make this code compile:
List<_______ Number> listOfNumbers = new ArrayList<Integer>();
? extends (This is an upper bound.)

67. Does this code compile?
List<? extends Number> listOfNumbers = new ArrayList<Number>();
Yes. For upper and lower bounds, the extends and super keywords include the class itself.

68. Does this code compile: List<? implements Collection> byInterface = new ArrayList<Collection>();
No. Upper bounds use extends for both classes and interfaces as the upper bound.

69. Do lower or upper bounds allow you to add elements to a list?
Lower bounds do because they specify the class that all types must be above using super.

70. What method takes a Consumer parameter and is available on both Stream and ArrayList?
forEach

71. What functional interface takes one parameter and returns a (possibly) different Object type?
Function

72. What functional interface takes two parameters and returns a boolean?
BiPredicate

73. What functional interface takes zero parameters and returns void?
Runnable

74. What functional interface takes zero parameters and has a get() method.
Supplier

75. What functional interface takes two parameters and has an accept() method?
BiConsumer

76. What functional interface takes one parameter and returns void?
Consumer

77. What functional interface takes one parameter and returns the same Object type?
UnaryOperator

78. How do you create an Optional without a value?
OptionalEmpty()

79. How do you create an Optional with the value "hi"?
Optional.of("hi")

80. How do you check if an Optional variable named o contains a value and return a boolean result?
o.isPresent()

81. How do you retrieve the value of an Optional variable named o if you know for sure there is a value inside?
o.get()

82. Which of these are required in a stream pipeline: source, intermediate operation, terminal operation?
Source and terminal operation

83. When creating a Locale with one parameter, what is it?
The language code

84. When creating a Locale with two parameters, what are they and in what order?
The language code followed by the country code

85. In a Locale, is the language code in uppercase or lowercase?
Lowercase

86. In a Locale, is the country code in uppercase or lowercase?
Uppercase

87. What is the most common format for name/value pairs in a property file?
Name=value

88. What is the most common format for a comment in a property file?
# comment

89. If the matching resource bundle is RB_en_US.properties, what other filenames will Java look for when searching for matching keys?
RB_en.properties and RB.properties

90. In what package are the new Java 8 date/time classes?
java.time

91. Is a class that extends RuntimeException a checked or unchecked exception?
Unchecked

92. True or false? A program should handle or declare java.lang.Error
False. A program must handle or declare checked exceptions. It should not handle or declare java.lang.Error.

93. When do you use the throw and throws keywords?
throw is used when throwing an exception. throws is used in a method declaration.

94. A regular try statement (not a try-with-resources) is allowed to have how many finally blocks if no catch blocks are present?
One

95. A regular try statement (not a try-with-resources) statement is allowed to have how many catch blocks if no finally block is present?
One or more

96. If a try statement has both a catch block and a finally block, in what order do they run if an exception is thrown?
try followed by catch followed by finally

97. If a try statement has both a catch block and a finally block, in what order do they run if an exception is not thrown?
try followed by finally

98. If the try block, catch block, and finally block all throw an exception, which gets thrown to the caller?
The one from the finally block

99. What are the two differences between Runnable and Callable?
Unlike Runnable, Callable returns a generic type and can throw checked exceptions.

100. What is the difference between calling shutdown() and shutdownNow() on a thread executor?
The shutdown() method prevents new tasks from being added and waits for all previously submitted tasks to finish. The shutdownNow() method does the same except that it attempts to stop all running tasks and cancels any tasks that have not been started yet.

101. How many threads are allocated in a pool created by calling Executors.newCachedThreadPool()?
Unspecified. The pool will add more threads as they are needed.

102. True or false? A livelock is an undesirable result that occurs when two tasks are executed at the same time.
False. This is the description for a race condition.

103. Name two concurrent collection classes that are sorted.
ConcurrentSkipListMap and ConcurrentSkipListSet

104. What is the name of the ConcurrencyAPI list class that creates a new array structure anytime the list is modified?
CopyOnWriteArrayList

105. What is the difference between FileOutputStream and FileWriter?
FileOutputStream writes data in bytes, while FileWriter uses characters

106. Name two techniques for reading non password text input from a java.io.Console instance.
readLine() and reader().read()

107. True or false? InputStream is used to write data to a stream.
False. It is used to read data from a stream

108. What is the standard way to determine if there is no more data available from an ObjectInputStream?
By catching EOFException

109. Name two java.io OutputStream/Writer stream classes that do not have corresponding InputStream/Reader classes.
PrintWriter and PrintStream

110. What may happen if you fail to properly close a stream resource that uses a file?
Data could be lost if the program is abruptly terminated. A resource leak could also occur, in which the file is unable to be read/written until the program is terminated.

111. True or false? You do not need to call flush() right before closing an OutputStream.
True. The close() method will automatically call flush().

112. Which NIO.2 method reads all files in the directory tree?
Files.walk()

113. True or false? The java.io.File class is used only for files and the java.nio.file.Path interface is used only for directories.
False. Both can be used on files and directories.

114. What does NIO.2 stand for?
Non-blocking Input/Output, Version 2. Commonly referred to as the New I/O.

115. What does it mean for a move to be atomic?
The move is performed as single unit of execution that appears to the rest of the system as occurring instantaneously.

116. True or false? The first index used in Path.getName(int) method is 0.
True. Path.getName(int) is zero-indexed.

117. True or false? Files.walk() and Files.find() use a breadth-first algorithm
False. They use a depth-first algorithm

118. Which NIO.2 method constructs a relative path from one Path object to another Path object?
path1.relativize(path2)

119. What are the four key JDBC interfaces you need to know for the exam?
Driver, Connection, Statement, and ResultSet

120. Where is the implementation of ResultSet defined?
In a databasespecific JAR file

121. What are the main parts of a JDBC URL?
jdbc, vendor/product name, and database specific string

122. What character is used to separate the three main parts of a JDBC URL?
: (Colon)

123. True or false? In a JDBC URL, the port is required.
False

124. What are the name and location of the file inside a JDBC 4.0+ JAR that tells Java the name of the Driver implementation?
java.sql.Driver in META-INF/service

125. When is Class.forName() called?
In old (JDBC 3.0 and lower) code that doesn't specify a java.sql.Driver file

126. What exception does DriverManager.getConnection() throw in JDBC 4.0+ if the JAR is not in the classpath?
SQLException

127. When creating a Statement without using the defaults, what are the two parameters in order?
ResultSet type and ResultSet concurrency mode

128. On what class do you call a method to create a Statement?
Connection

129. In an enum, when is the semicolon required after the values list?
If there is anything in the enum besides the values, such as instance variables, constructors or methods.

130.What access modifiers are allowed for enum constructors?
Only private or default access is allowed.

131. True or false? An enum can have methods defined on the individual values
True. An abstract method can be defined or a concrete method can be overridden in the value specific body.

132. Suppose B is an outer class and A is a member inner class within it. You have access to a B instance called b. How do you instantiate A?
b.new A();

133. Suppose B is an outer class and A is a member inner class within it. Both have instance variables called c. How does a method in an instance of A refer to the c declared in B?
B.this.c

133. When can a local inner class refer to a local variable from the method in which it was created?
When the local variable is final or effectively final

134. In what way does Java support multiple inheritance?
Classes can implement multiple interfaces.

135. Under what situation can a class that implements two interfaces, each with a default method that has the same signature, still compile?
The class must provide its own overridden implementation of the default method to be able to compile.

136. Name at least three requirements to implement the immutable object pattern.
Any of these five are correct: instance variables must be private and final, the class must block external access to mutable objects it contains, class methods may not be overridden, the class contains no setters, and the constructor sets all instance variable values.

137. What is the name of the design pattern that is focused on creating read only objects in memory?
The immutable object pattern
138. True or false? A lambda expression requires parentheses on the left side of the expression only when there are two or more arguments.

False. It also requires parentheses when there are zero arguments on the left side of the expression

139. True or false? Data types are optional in the left side of a lambda expression.
True. Although once a data type has been set for one parameter, it must be provided for all parameters.

140. True or false? A return statement is not required on the right side of a lambda expression when braces {} are used if the return type is void.
True

141. A common solution to a problem that can lead to difficult to manage code is referred to as ____________.
An anti-pattern

142. True or false? We can add an element to a List<?>.
False. List<Object> allows adding to the List. List<?> does not because it could be a list of any type.

143. Does this code compile?
List legacy = new ArrayList<String>();
Yes. It generates a compiler warning, but it doesn't give a compiler error.

144. What is the natural sort ordering of "123", "abc", and "ABC" ?
"123", "ABC", "abc"

145. What method do you need to implement in a Comparator, and how many parameters does it take?
compare(), and it takes two parameters.

146. What method do you need to implement in a Comparable, and how many parameters does it take?
compareTo(), and it takes one parameter.

147. Which would you implement using a lambda: Comparator or Comparable?
Comparator

148. What are the four main interfaces in the Java Collections Framework?
List, Map, Queue, and Set

149. What Java Collections Framework interface requires unique elements?
Set

150. What Java Collections Framework interface uses indexes?
List

151. What Java Collections Framework interface has key/value pairs?
Map

152. What does FIFO stand for, and what Java Collections Framework interface is it associated with?
Firstin/firstout and Queue

153. Rewrite this code using a method reference: () > Math.random()
Math::random

154. Which of these can exist multiple times in a stream pipeline: source, intermediate operation, terminal operation?
Intermediate operation

155. Which of these have a return type of Stream: source, intermediate operation, terminal operation?
Source and intermediate operation

156. Which of these are evaluated immediately after the method rather than participating in deferred execution: source, intermediate operation, terminal operation?
Terminal operation

157. Name a terminal operation that always terminates on an infinite stream.
findAny() or findFirst()

158. Name a terminal operation that returns an Optional.
findAny(), findFirst(), min(), or max()

159. Which of these are reductions? anyMatch(), collect(), count(), findAny()
collect() and count()

160. Which of these are mutable reductions? anyMatch(), collect(), count(), findAny()
collect()

161. Which intermediate operation makes the stream smaller based on a Predicate?
filter()

162. Which intermediate operation returns one element for every element passed into it using a Function to transform?
map()

163. Which intermediate operation is used for debugging a finite stream?
peek()

164. What are the names of all of the primitive streams?
DoubleStream, IntStream, LongStream

165. Which two static methods create an infinite stream of int primitives?
IntStream.generate() and IntStream.iterate()

166. What method do you call on an IntStream instance to convert it to a Stream<Integer>?
mapToObj()

167. What are the four key date and time classes, and what do they represent?
LocalDate is just the date without time or time zone. LocalTime is just the time without date or time zone. LocalDateTime is the date and time without time zone. ZonedDateTime is the date and time with time zone.

168. How do you create an object with the current date and time?
LocalDateTime.now();

169. How do you add an hour to a LocalTime object?
localTime.plusHours(1);

170. How do you create a Period of a month and a day?
Period.of(0, 1, 1);

171. How do you create a Duration of an hour and a half?
Duration.ofMinutes(90); // other ways are correct as well.

172. To what classes can you add a Duration?
LocalTime, LocalDateTime, and ZonedDateTime

173. To what classes can you add a Period?
LocalDate, LocalDateTime, and ZonedDateTime

174. What is the class used for formatting a date?
DateTimeFormatter

175. True or false? If a try statement has catch blocks for FileNotFoundException and IOException, the catch blocks can be in either order.
False. The more general catch blocks must appear below the more specific ones. Since FileNotFoundException is a subclass of IOException, it must come first in the catch block order.

176. Which of the following exceptions are checked exceptions: IllegalFormatException, IOException, MissingResourceException, SQLException?
IOException, SQLException

177. What symbol is used to separate exception types in a multi-catch statement?
|(the pipe character)

178. True or false? The variable in a multi-catch block is allowed to be reassigned.
False. This is allowed in a catch clause only when there is one exception type present

179. True or false? The order is important for exception types listed inside a multi-catch block.

False

180. A try-with-resources statement is allowed to have how many finally blocks if no catch blocks are present?
Zero or one

181. A try-with-resources statement is allowed to have how many catch blocks if no finally block is present?
Zero or more

182. True or false? The accumulator argument passed to a stream reduce() method should be associative and stateless.
True

183. Which Concurrency API framework allows you to perform tasks recursively?
Fork/join framework

184. True or false? A serial stream completes tasks in order, one record at a time
True

185. What's the difference between HashMap and ConcurrentHashMap?
Unlike HashMap, ConcurrentHashMap will not throw a ConcurrentModificationException if multiple threads modify it at the same time or inside an iterator.

186. What's the difference between a deadlock and a livelock?
A deadlock occurs when two threads are blocked forever, whereas a livelock occurs when two threads appear to be active and working but are conceptually deadlocked forever.

187. What is the name of the ConcurrencyAPI class used to allow multiple threads to synchronize at a common point?
CyclicBarrier

188. Which class in the JavaAPI is most like HashMap but also supports access by multiple threads?
ConcurrentHashMap

189. Which java.io class should be used to write and format text data to a low level character stream?
PrintWriter

190. Name two java.io low-level stream classes used to read file data directly.
FileInputStream and FileWriter

191. Why does the Console method readPassword() return a character array instead of a String value?

For security reasons. While character arrays can be explicitly overwritten in memory, String values are added to the shared memory pool, making the password available in memory long after it is used.

192. What methods is a class that implements the Serializable interface required to override?
There are no required methods. The serialVersionUID attribute is recommended but not required.

193. Let's say you wanted to read binary data from a file in an efficient manner; which two java.io stream classes might you use?
FileInputStream and BufferedInputStream

194. Which java.io class should be used to deserialize a stored data element back to an Object?
ObjectInputStream

195. Name two advantages of using the NIO.2 API to read files versus the java.io.File class.
The NIO.2 API supports symbolic links and reading filesystem specific attributes, whereas java.io.File does not.

196. True or false? Files.lines() does not load the entire file into memory.
True. The file is read lazily.

197. What is a symbolic link?
A file within an operating system that serves a reference to another file or directory

198. Which NIO.2 method tests if two Path objects represent the same file within the file system?
Files.isSameFile(Path,Path)

199. True or false? The java.nio.file.Path interface includes a method that returns true if a file is a directory
False. The Files factory class is used to determine if a file is a directory.

200. What does the NIO.2 method subpath() do?
It returns the relative subpath between the start and end indexes of an existing path.

201. True or false? For absolute paths, Path.getName(0) is the root directory.
False. Path.getName() does not include the root directory.

202. What are the constants for the ResultSet types?
TYPE_FORWARD_ONLY, TYPE_SCROLL_INSENSITIVE, and TYPE_SCROLL_SENSITIVE

203. What are the constants for the ResultSet concurrency modes?
CONCUR_READ_ONLY, CONCUR_UPDATABLE

204. What exception is thrown if you request an unsupported ResultSet type or concurrency mode?
None. The JDBC driver downgrades the request to one that is supported.

205. What methods on Statement can run a SELECT SQL statement?
executeQuery() or execute()

206. What methods on Statement can run a DELETE SQL statement?
executeUpdate() or execute()

207. What does executeQuery() return?
ResultSet

208. What does executeUpdate() return?
int

209. What does execute() return?
boolean

210. On a forward only ResultSet, what method moves the cursor to the first row?
rs.next()

211. What method gets the first column in a ResultSet if it is an int?
rs.getInt(1)

212. What does "effectively final" mean in the context of an inner class?
If the local variable were to have final added to the declaration, the code would still compile.

213. What is the rule about what an anonymous inner class can extend/implement?
An anonymous inner class must either extend a class by name or implement exactly one interface.

214. What are the four access levels in Java?
Public (all classes can access), protected (subclasses and code in the same package can access), private (only code in the same class can access), and default (only code in the same package can access)

215. What is the difference between overloading and overriding?
Overriding requires the same method signature, and it is used when a subclass has a version of the method. Overloading requires the same method name but a different argument list.

215. With what character does an annotation begin?
@
216. Which of the following are standard annotations in core Java: @Extend, @Overload, @Override?
@Override

217. At most how many instances of a singleton class should exist in memory?
One

218. A class that does not expose its private variables demonstrates which design principle?
Encapsulation

219. A mammal has-a backbone, a big cat isa mammal, and an ocelot isa big cat. What can we say about ocelots and backbones?
An ocelot has-a backbone.

220. Which class design strategy primarily makes use of the has a principle: class inheritance or object composition?
Object composition

221. True or false? Lazy instantiation is the process of creating an instance of an object or resource the first time it is requested.
True

222. True or false? An interface can extend an interface and a class.
False. An interface can only extend another interface.

223. Name the three types of methods allowed in an interface definition
default, static, abstract

224. True or false? Design patterns try to solve problems that you encounter, which other developers before you have already solved.
True

225. Rewrite this code using a lambda: Math::random
() > Math.random()

226. Rewrite this code using a method reference: w -> l.contains(w)
l::contains

227. Rewrite this code using a lambda: l::contains
w -> l.contains(w)

228. Rewrite this code using a method reference given that a List is passed in: x -> x.isEmpty()
List::isEmpty

229. Rewrite this code using a lambda: ArrayList::isEmpty
x -> x.isEmpty()

230. Rewrite this code using a method reference: () -> new ArrayList<String>()
ArrayList<String>::new

231. Rewrite this code using a lambda: ArrayList<String>::new
() -> new ArrayList<String>()

232. What two methods taking a lambda as a parameter can you call directly on an ArrayList (without using a stream)?
forEach and removeIf

233. What functional interface does forEach take?
Consumer

234. What functional interface does removeIf take?
Predicate

235. What method do you call on an OptionalInt to get the value if you know a value is contained?
getAsInt()

236. What are the primitive optional types?
OptionalDouble, OptionalInt, OptionalLong

237. What is the only primitive functional interface that doesn't involve double, int, or long?
BooleanSupplier

238. Which Collector creates a single String from a Stream?
joining()

239. Which Collector returns both the average and sum in a single call to an IntStream?
summarizingInt()

240. Which Collector turns a Stream into an ArrayList?
toCollection(ArrayList::new) or toCollection(l -> new ArrayList())

241. Which Collector creates a Map with the keys true and false?
partitioningBy()

242. Which Collector creates a Map with many keys?
groupingBy()

243. To which method do you pass a Collector?
collect()

244. Which class defines common Collector implementations?
Collectors

245. On an infinite stream containing nothing but the number 4, which methods will terminate when passed a Predicate that checks whether the element is 1? allMatch(), anyMatch(), or noneMatch()
allMatch()

246. On an infinite stream containing nothing but the number 4, which methods will terminate when passed a Predicate that checks whether the element is 4? allMatch(), anyMatch(), or noneMatch()
anyMatch() and noneMatch()

247. What class is used for formatting a percentage?
NumberFormat

248. Which of these times is earlier and by how much? 2016-03-14T05:00-04:00 and 2016-03-14T09:00+01:00
The first is 9:00 GMT and the second is 8:00 GMT, making the second one hour earlier.

249. Which of these times is earlier and by how much? 2016-03-14T07:00-05:00 and 2016-03-14T08:00+06:00
The first is 12:00 GMT and the second is 2:00 GMT (using a 24hour clock), making the first two hours earlier.

250. What happens to the clocks in the March clock change of daylight savings time?
At 1:59 a.m., the time jumps to 3:00 a.m.

251. What happens to the clocks in the November clock change of daylight savings time?
At 1:59 a.m., the time jumps back to 1:00 a.m.

252. What does this display? System.out.println(Period.of(1, 0, 3));
P1Y3D

253. What does this display? System.out.println(Period.ofWeeks(1));
P7D

254. What does this display? System.out.println(Duration.ofMinutes(5));
PT5M

255. What does this display? System.out.println(Duration.ofMills(123));
PT0.123S

256. What interface must a class implement to be listed in the declaration of a try-with-resources statement?
AutoCloseable

257. What does idempotent mean?
The method is safe to run multiple times

258. In a try-with-resources statement, what is the second exception called if both the try clause and a close() method throw an exception?
Suppressed exception

259. Which of the classes is closed first in
try (C c = new C(); D d = new D()) {}?

D (The last to be created is the first to be closed.)

260. True or false? The following statement guarantees an AssertionError is thrown: assert false;
False. An AssertionError will not be thrown if assertions are not enabled at runtime.

261. The two command-line flags for enabling assertions are ___________ and _____.
-enableassertions and -ea

262. What parameter is optional in an assertion?
The String message

263. What type does the mandatory parameter in an assertion return?
Boolean

264. True or false? Synchronizing on a class object will synchronize on all instances of the class.
False. The class object and the instances of the class are not related in terms of synchronization

265. What does it mean for a task to be atomic?
An atomic task is completed as a single unit of execution without any interference by another thread.

266. What method do you call to create a parallel stream from an existing stream?
parallel()

267. True or false? All Java programs running in a JVM are multithreaded.
True. The JVM includes daemon worker threads, such as the garbage collector.

268. Which class that implements the ForkJoinTask interface is best to use if you do not need the return value for each task?
RecursiveAction

269. True or false? The synchronized keyword can only be applied to non-static methods and on blocks of code.
False. It may also be applied to static methods.

270. Name a concurrent deque class that supports blocking.
LinkedBlockingDeque

271. True or false? The findAny() method always returns the first result in the stream.
False. On serial streams it returns the first element, but on parallel streams it can return any element of the stream.

272. On which type of stream, serial or parallel, is it appropriate to use the groupingByConcurrent() collector?

Parallel stream

273. True or false? The java.io.Fileclass can be used to delete nonempty directories
False. The java.io.File class can delete only files and empty directories.

274. Name the two java.io stream classes used to improve performance when writing data to sequential disk-based systems.
BufferedInputStream and BufferedWriter

275. What syntax structure can you use to avoid needing to explicitly close a java.io stream?
try-with-resource block

276. True or false? Calling getParent() on a root directory returns null.
True. The root directory has no parent.

277. True or false? Calling flush() closes an OutputStream.
False. The flush() method may be called repeatedly while the stream is open.

278. The __________ path is the full path from the root to the file or directory
absolute

279. Name two methods for writing to a java.io.Console instance.
printf() and format()

280. True or false? Files.list() traverses a directory tree.
False. Files.list() reads only a single directory, not the tree.

281. What does Path.toAbsolutePath() return on a relative path?
The existing path of the current directory as a prefix

282. How do you delete a directory in NIO.2?
First, delete all of the contents of the directory; then call Files.delete() on it

283. What happens to a Path supplied to a compatible NIO.2 method that includes the NOFOLLOW_LINKS option?
If the Path is a symbolic link, the operation will be performed on the link itself, rather than the target of the symbolic link.

284. What's the difference between Files.createDirectory() and Files.createDirectories()?
Files.createDirectory() creates only a single path, whereas Files.createDirectories() creates all subdirectories along the input path that do not exist.

285. Which NIO.2 method reads the entire contents of file to memory when it is first called?
readAllLines()

286. What method gets a date and time from a ResultSet for column 5?
rs.getTimeStamp(5)

287. True or false? Calling rs.getString(1) before calling rs.next() throws a SQLException on a forward only ResultSet.
True

288. On a scrollable ResultSet, how do you move the cursor to the third row?
rs.absolute(3)

289. On a scrollable ResultSet, how do you move the cursor to the last row?
rs.last()

290. On a scrollable ResultSet, how do you move the cursor to the row immediately before the current one?
rs.previous() or rs.relative(-1)

291. On a scrollable ResultSet, how do you move the cursor to the row immediately after the last one?
rs.afterLast()

292. In what order should the three JDBC resources be closed?
ResultSet, Statement, Connection

293. When closing a ResultSet, what other resources get closed?
None

294. When closing a Statement, what other resources get closed?
ResultSet

295. When closing a Connection, what other resources get closed?
ResultSet and Statement

296. What structure is Queue?
FIFO, It has methods offer(e)/add(e) and poll()/remove() for this purpose. Note that offer and add are similar while poll and remove are similar.

297. What structure is Stack?
LIFO, It provides methods push(e) and pop() for this purpose, where push adds to the front and pop removes from the front.

298. Properties of Deque.
It is a subtype of the java.util.Queue interface. It represents a queue where you can insert and remove elements from both ends of the queue.

299. What bound can be used to define a generic class?
Only an upper bound (the one one defined using extends) can be used. A lower bound (super) can never be used.

300. What exception is throw when the classes are not compatible with each other in a class that implements Comparable interface.

java.lang.ClassCastException

301. Can you use primitives in generics?
No, you can't use primitives in Generics

302. File class has two methods for returning the contents of a directory. What are these?
String[] list() - Returns an array of strings naming the files and directories in the directory denoted by this abstract pathname.
File[] listFiles() - Returns an array of abstract pathnames denoting the files in the directory denoted by this abstract pathname.

303. What's return allMatch(predicate), noneMatch(predicate), anyMatch(predicate)?
A primitive boolean.

304. What's return Stream<T>.sorted() and Stream<T>.sorted(comparator) ?
Stream<T>

305. What's return findFirst(), findAny()?
Optional<T>

306. What's return IntStream.reduce(intBinaryOperator)?
OptionalInt

307. What's return IntStream.reduce(int, intBinaryOperator)?
A primitive int

308. The LocalTime.until(localTime, temporalUnit) method return the difference between the two time periods in given units. Here, the difference is 45 minutes but the unit is HOURS, what's the result?
It will return 0

309. True o false, The Path.relativize(path) method requires that both paths be absolute or both relative.
True, it will throw an IllegalArgumentException if a relative path value is mixed with an absolute path value.

310. Is valid Path.resolve(pathIsAbsolute)?
Yes, it returns only pathIsAbsolute

311. What's return x.equals(null) ?
false

311. On x.equals(Object y) methdo is correct define in x==y
It's correct.

312. What's return Stream.count();
long primitive. Returns the count of elements in this stream.

313. What's return Stream.sort();
Stream.sort() doesn't exist.

314. Which class use to generate a random number generator isolated to the current thread.
java.util.concurrent.ThreadLocalRandom

315. If the receiver has loaded a class for the object that has a different serialVersionUID than that of the corresponding sender's class. In what the deserialization result?
It result in an InvalidClassException.

316. What's a nested class?
It's any class whose declaration occurs within the body of another class or interface.

317. What's an inner class?
It's a nested class that is not explicitly or implicitly declared static.

318. List<Integer> names = Arrays.asList(1, 2, 3);
names.forEach(x->x=x+1);
names.forEach(System.out::print);
123

319. What's function List.replaceAll(operator) requires?
UnaryOperator

320. Which methods may pause/stop the current thread?
Thread.yield() and someObject.wait()

321. Can a int be boxed to a Double?
It can't.

322. What's return Map.values() ?
Collection<V>

323. What's return Map.keySet() ?
Set<K>

324. What's happen when while deserializing, the JVM will not find any no-arg constructor that can be invoked to initialize?
It will throw an InvalidClassException.

325. Having ambiguous fields does cause any problems?
Not, but referring to such fields in an ambiguous way will cause an error at compile time.
326. Anonymous class can be static?. if it was created in a static method.
No, anonymous class can never be static. Even if created in a static method.

327. Is this one correct? The object issuing the call to wait() will halt until another object sends a notify() or notifyAll() method
No, note that it is not the object that calls the wait/notify but the thread.

328. What function expects peek(obj) as an argument and what returns?
Stream<T> peek(Consumer);

329. What's mean connection.rollback(savepoint) ?
Ensures that any statements executed after that save point was created will not be commited. Any statement executed before the save point will be committed as and when commit is called.

330. Does BufferedReader provide markSupported() ?
Yes and It returns a boolean.

331. What's necessary for a binarySearch works?
binarySearch method works ONLY if the collection or array being searched is already sorted. Further, the sort must have been done using the same Comparator that is being used to search otherwise search results will not be reliable. If no Comparator is used explicitly (either in sort or binarySearch method calls), elements are sorted by their natural order as determined by their Comparable interface implementation.

332. Is private a valid access modifier for a top level class?
No, private is not a valid access modifier for a top level class.

333. What interface implements Hashtable<K,V>?
Map<K,V>

334. What's the function of Hashtable<K,V>?
Hashtable methods are synchronized.

335. What's return xxxxxx.collect(Collectors.counting());
Collector.counting returns a Collector that returns a long

336. What's return Exception e.getSuppressed()?
Throwable[] resultado

337. What's happen when System.console() run in the background?
System.console() will return null.

338. What's return Optional.of(null) ?
Optional.of method throws NullPointerException if you try to create an Optional with a null value. If you expect the argument to be null, you should use Optional.ofNullable method, which returns an empty Optional if the argument is null.

339. During deserialization, what components are not executed?
The constructor of the class (or any static or instance blocks) is not executed.

340. As a part of a application, you have serialized and stored some objects of a class in the database. At another place in the same application, you deserialize those objects. After a

few months you determine that you need to add one new String field in the class. Which of the following statements are correct regarding the above described situation?
It is possible to deserialize the older objects into the update class objects even if the original class did not explicitly define the serialVersionUID field.

If a class that implements Serializable does not explicitly define serialVersionUID field, the compiler automatically adds this field. It assignes this field a value that is computed based on the attributes of the class such as the fields and the implemented interfaces.

341. Consider the following situation: Thread T2 has called obj.wait() and is waiting. Thread T1 holds the lock for an object obj. What will allow the thread T2 to become runnable?
Thread T1 calls the notify() method on obj and releases the lock on obj.

342. What's return mapToInt ?
It returns an IntStream

343. What's return Stream.reduce(BinaryOperator<T> accumulator) ?
Optional<T>

344. What's return Stream.reduce(T identity, BinaryOperator<T> accumulator) ?
T

345. What's return Stream.reduce (
U identity,
BiFunction<U,? super T,U> accumulator, BinaryOperator<U> combiner) ?
<U> U
Since the identity values type is not restricted to the T, see previous question.

346. Is Stream.collect(Collector<? super T,A,R> collector) a terminal operation?
True

347. What's return in IntSummaryStatistics:
getAverage()
getCount()
getMax()
getMin()
getSum()
double
long
int
int
long

348. Which statement will print "/test.txt" when executed on a *nix system?
System.out.println(Paths.get("/", "test.txt"))

public static Path Paths.get(String first, String... more)

349. Does the Map interface defines the method remove(Object) ?
True

350What's return BufferedReader.read() ?
int, Reads a single character.

351. What's return BufferedReader.readLine() ?
String, Reads a line of text.

```
352. List<> list = new ArrayList<String>();
```

Compile error, The diamond operator is used at the place of creation of the object and not at the place of variable type declaration. So it should be: List<String> list = new ArrayList<>();

353. In jdbc, when does execute returns true?
If the first object that the query returns is a ResultSet object. Use this method if the query could return one or more ResultSet objects. Retrieve the ResultSet objects returned from the query by repeatedly calling Statement.getResultSet.

354. ¿De que tipo es el objeto System.out?
Del tipo java.io.PrintStream

```
355. IntStream.range(1, 4)
.forEach(System.out::print);
```

123

356. What's return:

```
Arrays.stream(int[] array),
Arrays.stream(double[] array),
Arrays. stream(long[] array)
```

IntStream
DoubleStream
LongStream

```
357. Stream.of(1.0, 2.0, 3.0)
.mapToInt(Double::intValue)
.mapToObj(i -> "a" + i)
.forEach(System.out::println);
```

a1, a2, a3.

```
358. IntStream.range(1, 4)
.mapToObj(i -> new Foo("Foo" + i))
.peek(f -> IntStream.range(1, 4)
.mapToObj(i -> new Bar("Bar" + i + " <- " f.name))
.forEach(f.bars::add))
.flatMap(f -> f.bars.stream())
.forEach(b -> System.out.println(b.name));
```

Bar1 <- Foo1
Bar2 <- Foo1

Bar3 <- Foo1
Bar1 <- Foo2
Bar2 <- Foo2
Bar3 <- Foo2
Bar1 <- Foo3
Bar2 <- Foo3
Bar3 <- Foo3

359. What's return Stream.flatMap(Function)
<R> Stream<R>

360. Stream<String> s = Stream.of("c","b","a");
Collection<String> names = s
.collect(Collectors.toCollection(TreeSet::new));
System.out.println(names);
[a,b,c]

361. Can a constructor have a final modifier?
The constructor cannot have a final modifier.

362. This exception may be thrown by methods that have detected concurrent modification of an object when such modification is not permissible.
ConcurrentModificationException

363. Define method hashCode
int hashCode()

364. To sort a List, what's need, a Comparator or Comparable?
A list needs a Comparable.

365. Static variables value can be stored while serializing if the same is provided while initialisation.
True

366. If variable is defined as Static and Transient both, which govern?
Static modifier will govern the behavior of variable and not Transient.

367. Does calling notify/notifyAll release the lock?
Calling notify/notifyAll does not release the lock. A lock is not released until the thread exits a synchronized block or method.

368. When the program is trying to manipulate the scheduling of thread using priorities which result?
It's a bad idea. Simply because operating systems behave differently about priorities