

1. ¿Cuál de estos es un comando adecuado para ejecutar una aplicación Java desde su jar?

Respuesta: ``java -jar target/refactorex-1.0-SNAPSHOT.jar``

2. Indica qué sentencia es correcta:

Respuesta: Todas son correctas.

3. ¿Qué código ejecuta la máquina virtual de Java?

Respuesta: Bytecode

4. ¿Qué nombre recibe el componente a partir de la cual se instancia un objeto?

Respuesta: Clase

5. Un ADT (abstract data type) se define en términos de:

Respuesta: Todas las propuestas.

6. Una clase es:

Respuesta: Todas las propuestas.

7. ¿Qué nombre recibe el acto de ocultar los detalles funcionales de una clase a los objetos que la usarán posteriormente?

Respuesta: Encapsulación

8. Dada la línea de código Java propuesta, ¿qué puedes deducir?

Respuesta: Que `checkCartucho()` es un método estático.

9. ¿Qué tipo de polimorfismo se utiliza en este código?

Respuesta: Paramétrico (por genéricos).

10. ¿Qué tipos de polimorfismo exhibe este código?

Respuesta: Overloading (por sobrecarga)

11. ¿Qué encontramos en un archivo jar de Java?

Respuesta: Todos los propuestos.

12. ¿Cuáles de las siguientes no forman parte del JRE de Java?

Respuesta: Todas las propuestas.

13. ¿Cuáles de las siguientes son motivos para refactorizar?

Respuesta: Todas las propuestas.

14. ¿Cuáles de las siguientes son refactorizaciones admisibles?

Respuesta: Todas las propuestas.

15. ¿Qué contiene el fichero POM.xml?

Respuesta: Todas las propuestas.

16. ¿Cuáles de las siguientes son características de la TDD?

Respuesta: Todas las propuestas.

17. La característica de un lenguaje de programación que permite manejar valores de diferentes tipos de datos a través de una interfaz unificada es:

Respuesta: Polimorfismo

18. ¿Cuándo finaliza un ciclo en la TDD?

Respuesta: Cuando se pasa el test.

19. ¿Cuáles de estas acciones pueden llevarse a cabo con Maven?

Respuesta: Todas las propuestas.

20. Maven es:

Respuesta: Un software de administración de proyectos.

21. ¿Qué crees que significa la idea convention over configuration o convención sobre la configuración, que está en el corazón de Maven?

Respuesta: Que el programador mantenga el código escrito bajo control.

22. ¿Qué anotación indica que un método ha de ejecutarse antes de cada caso test?

Respuesta: @Before

23. ¿Qué objetivos persigue la refactorización del código?

Respuesta: Todas las propuestas.

24. ¿Por qué conviene declarar como final los atributos de un objeto que hagan referencia a otros objetos?

Respuesta: Todas las propuestas.

25. Tienes que programar la actualización de la calidad de los ítems de un stock o inventario. Si tu código depende de sus tipos específicos en vez del tipo más general, ¿qué principio SOLID no se ha respetado?

Respuesta: DI

26. Con independencia del IDE o gestor de proyecto Java que utilices, ¿cuál de estos ficheros deberías dejar sin seguimiento en git?

Respuesta: Todos los propuestos.

27. Para iniciar un proyecto con Maven, ejecutas:

Respuesta: mvn archetype:generate

28. Si utilizas Maven, ¿cuál de estos directorios deberías dejar sin seguimiento en Git?

Respuesta: target

29. ¿Cuál de los principios SOLID asociarías con la sentencia "Un módulo debe ser responsable (de algo) ante un solo actor"?

Respuesta: SRP

30. ¿Cuál de las siguientes ramas no forma parte de la propuesta gitflow?

Respuesta: Ninguno de los propuestos.

31. Si quieres listar las ramas remotas y las locales, ejecutas:

Respuesta: git branch -a

32. ¿Qué comando se ejecuta para poner en seguimiento con git el directorio actual?

Respuesta: git init

33. El directorio .git:

Respuesta: Todas son correctas.

34. ¿Cuáles de las siguientes no forman parte de la TDD?

Respuesta: Ninguno de los propuestos.

35. ¿Qué significan las siglas TDD?

Respuesta: Test Driven Development.

36. ¿Cuáles de las siguientes son características de la TDD?

Respuesta: Todas las propuestas.

37. ¿Cuándo comienza el ciclo de la TDD?

Respuesta: Con la escritura de un test que falle.

38. En TDD, ¿qué tipo de código se escribe en la fase roja?

Respuesta: Código de test.

39. En TDD, en la fase verde:

Respuesta: Se escribe el código de producción hasta que el test pase.

40. El principio O de SOLID hace referencia:

Respuesta: A que los programas deben estar abiertos a la extensión y cerrados a la modificación.

41. Si has añadido en los ficheros CSS de tu proyecto las media query para conseguir que tu sitio sea responsive, ¿cuál sería el mensaje del commit para respetar la convención conventional commits?

Respuesta: feat(media.css): Incluir media query dispositivos móviles

42. Para crear la rama spam en git ejecutas el comando:

Respuesta: git branch spam

43. Para eliminar la rama spam en git ejecutas el comando:

Respuesta: git branch -d spam

44. Para listar las ramas locales que existen en un proyecto, ejecutas:

Respuesta: git branch

45. Si quieres obtener una copia de un repositorio remoto en un directorio local de tu máquina, ejecutas:

Respuesta: git clone <URL del proyecto>

46. Si quieres consultar la historia de commits de un repositorio git en tu máquina, ejecutas:

Respuesta: git log

47. ¿La depuración se incluye en qué fase del ciclo de vida del software?

Respuesta: Todas las propuestas.

48. ¿Cuáles de los siguientes mensajes de commit respetan la convención conventional commit?

Respuesta: Todas las propuestas.

49. Si quieres añadir al staging area todos los archivos de un repositorio git, ejecutas el comando:

Respuesta: git add .

50. Para realizar una instantánea del estado de tu proyecto y guardarla en el historial de versiones, en git utilizas el comando:

Respuesta: git commit -m "comentario"

51. Es deseable una rutina con:

Respuesta: Todas son correctas.

52. ¿Cuáles de los siguientes son principios clave en el diseño de un programa?

Respuesta: Todos los propuestos.

53. ¿Cuál o cuáles de las siguientes NO constituyen un nombre de rutina adecuado?

Respuesta: Todas las propuestas.

54. ¿Cuándo deberíamos definir una convención para los nombres de las variables?

Respuesta: Todos los propuestos.

55. ¿Cuál de los siguientes nombres usarías para nombrar a una rutina que calcula la media de puntos de los equipos de una liga en una determinada jornada?

Respuesta: Todos los propuestos.

56. ¿Cuál de los siguientes es un esquema de nombres adecuado para nombrar variables y rutinas, es decir, ¿cuál de las siguientes expresiones utilizarías para nombrar variables y rutinas?

Respuesta: Todos los propuestos.

57. ¿Cuál de las siguientes características debería tener el nombre de una variable?

Respuesta: Todas las propuestas.

58. ¿Qué es testing / testear?

Respuesta: Todas las propuestas.

59. ¿Qué es debugging / depurar?

Respuesta: Todas las propuestas.

60. ¿Con qué otros dos conceptos va de la mano la cohesión?

Respuesta: Todas las propuestas.

61. Si una rutina realiza una única operación, se dice que tiene:

Respuesta: Alta cohesión.

62. Si una rutina posee más de una responsabilidad, se dice que tiene:

Respuesta: Baja cohesión.

63. La sigla "S" en SOLID significa:

Respuesta: SRP

64. La instrucción pass de Java en un else:

Respuesta: No existe la palabra reservada pass en Java.

65. ¿Cuándo has de usar un bucle for?

Respuesta: Todas las propuestas.

66. Una sentencia break dentro de un bucle:

Respuesta: Todas las propuestas.

67. Una rutina que devuelve un valor recibe el nombre de:

Respuesta: Función

68. Para ignorar el directorio target independientemente de dónde se encuentre situado en el sistema de archivos, incluyes en el fichero gitignore la línea:

Respuesta: `**/target/`

69. Para saber desde consola qué ficheros se ignoran en el seguimiento git y por qué motivo, ejecutas el comando:

Respuesta: `git status --ignored`

70. En una clase con casos test:

Respuesta: Todas las propuestas son correctas.

71. Cuando ejecutas: `mvn clean`

Respuesta: Se elimina el directorio target

72. En las reuniones de pie o dailies (en Scrum), se persigue:

Respuesta: Todas las propuestas.

73. ¿De qué se habla en una reunión de pie o daily?

Respuesta: Todas las propuestas.

74. Las reuniones de pie o dailies se celebran:

Respuesta: Desde el punto de vista del comprador del software.

75. ¿Desde qué punto de vista se evalúa si una historia de usuario/a es valiosa?

Respuesta: Desde el punto de vista del comprador del software.

76. ¿Para añadir una nueva librería al proyecto mediante Maven, en qué sección del POM.xml has de añadir una nueva entrada?

Respuesta: <dependencies>

77. Indica qué deben cumplir los casos test:

Respuesta: Todos los propuestos.

78. ¿Cuáles de los siguientes son principios clave en el diseño de un programa mediante programación orientada a objetos?

Respuesta: Todos los propuestos.

79. El principio SRP es sinónimo de:

Respuesta: Alta cohesión.

80. ¿Por qué deberíamos usar convenciones para nombrar las variables de nuestro código?

Respuesta: Todas las propuestas.

81. El concepto de inyección de dependencias, ¿con qué principio SOLID se confunde tradicionalmente y de manera errónea?

Respuesta: DI

82. La deuda técnica:

Respuesta: Todas las propuestas.

83. Para crear una nueva rama en git y saltar a ella, en un solo paso, puedes utilizar el comando:

Respuesta: git checkout -b rama_nueva

84. Si una clase realiza varias operaciones poco relacionadas entre sí, se dice que tiene:

Respuesta: Baja cohesión.

85. Si quieres añadir y hacer un commit de los cambios del directorio git al mismo tiempo, ejecutas el comando:

Respuesta: git commit -a -m 'terminado footer [ticket 112]'

86. Cuando trabajas con JUnit para escribir casos test, estás realizando:

Respuesta: Pruebas de caja blanca.

87. Cuando averiguas la causa de un error y modificas el código, has hecho:

Respuesta: Debugging

88. Cuando en JUnit escribes los test que forman parte de la TDD para codificar una determinada clase que no involucra funcionalidades de otras clases, estás haciendo:

Respuesta: Unit testing.

89. La repetición de casos test previamente ejecutados con el propósito de encontrar defectos en el software que previamente había pasado los mismos casos test, se llama:

Respuesta: Test de regresión.

90. ¿Qué línea has de añadir al fichero .gitignore para evitar el seguimiento del directorio .vscode?

Respuesta: .vscode/

91. ¿Dónde has de situar el fichero .gitignore?

Respuesta: En el directorio raíz del proyecto que hayas puesto en seguimiento.

92. ¿Qué es una infección?

Respuesta: Un valor no válido de una de las variables del programa.