

PRUEBA TEÓRICA

NOMBRE: Luis Guillermo Diaz Villalobos

IDENTIFICACIÓN: 1047402718

FECHA: 05/10/2023

1. ¿Qué es Java?
 - ☐ Lenguaje de programación para Linux
 - ☐ Lenguaje de programación para Windows
 - ☒ Lenguaje de programación multiplataforma
- 2.Cuál de las siguientes no es una característica de JAVA
 - ☐ Orientado a Objetos
 - ☐ Encapsulación
 - ☐ Polimorfismo
 - ☒ Herencia Múltiple
 - ☐ Abstracción.
3. ¿Como se crea una instancia de una variable estática?
4. ¿Cuándo se inicializan las variables estáticas en la ejecución de un programa?
5. List, Deque, Queue, Set son subinterfaces de?
 - ☐ ArrayList
 - ☐ AbstractQueue
 - ☐ AbstractSet
 - ☒ Collection
 - ☐ Iterable
 - ☐ Vector
 - ☐ AbstractCollection
6. Iterator() es un método de?
 - ☒ Iterator
 - ☐ Collection
 - ☐ List
 - ☐ Vector
 - ☐ Iterable
7. Map es un collection?
 - ☐ Verdadero
 - ☒ Falso
8. hasNext(),next() son métodos de que interfaz?
9. List es una clase abstracta?
 - ☐ True
 - ☒ False
10. ArrayList, AbstractQueue Y ConcurrentLinkDeque son subclases de?
 - ☐ Collection
 - ☐ Iterable
 - ☐ Iterator
 - ☒ AbstractCollection

11. Map.Entry<K,V> que es?

- ☒ Interface
- ☐ Class
- ☐ Nested Interface
- ☐ Nested Class

12. ¿Los métodos utilitarios para el manejo de Colecciones se encuentran en que clase utilitaria de java?

13. ¿Con que tags se identifican un proyecto en Maven?

- ☐ Version
- ☐ Dependencias
- ☒ Groupid
- ☐ ArtifactId
- ☐ Packaging

14. De un ejemplo de cómo agregar una dependencia en el pom

15. ¿Si agregamos una propiedad como se muestra a continuación, como se usa al agregar una nueva dependencia?

```
<properties>
  <spring.version>4.3.5.RELEASE</spring.version>
</properties>
```

16. ¿Si se requiere crear un proyecto Maven por la línea de comandos y que sea un proyecto web, que comando se debe correr?

- ☐ maven -DgroupId=com.sample.webproject -DartifactId=SampleWebApp
- ☒ mvn archetype:generate -DgroupId=com.sample.webproject -DartifactId=SampleWebApp -DarchetypeArtifactId=maven-archetype-webapp -DinteractiveMode=false
- ☐ maven -DartifactId=SampleWebApp -DarchetypeArtifactId=maven-archetype-webapp
- ☐ mvn generate compile archetype:generate -DgroupId=com.sample.webproject -DartifactId=SampleWebApp -DarchetypeArtifactId=maven-archetype-webapp -DinteractiveMode=false
- ☐ mvn install -DarchetypeArtifactId=maven-archetype-webapp -DinteractiveMode=false

17. ¿Las dependencias con scope Provided, de donde obtienen el JAR?

- ☐ JRE
- ☐ JDK x
- ☒ Contenedor
- ☐ Repositorio Maven

18. ¿Dónde queda el resultado al ejecutar el comando mvn package?

19. ¿Usualmente dónde queda el repositorio del proyecto, hablando de Maven?

20. ¿Qué comando se debe ejecutar para verificar que falta por versionar en el repositorio GIT?

21. ¿Qué comando ejecutar para saber cuál es el repositorio remoto GIT?

22. Para que sirve el siguiente comando **git checkout -b branch?**

23. ¿Que debe ir en las xxxx del siguiente comando de **git push origin xxxxxx?**

24. De los siguientes comandos cual es el orden adecuado para subir un cambio al repositorio remoto, justifique su respuesta.
- ☐ git push origin Branch
 - ☐ git pull origin Branch
 - ☐ git add .
 - ☐ git commit -m"Se sube los cambios"
25. Si se tenemos un proyecto con springframework y tenemos una clase anotada con **@RestController @RequestMapping("api/parametros/")** y luego un método con la siguiente anotación **@RequestMapping(value = "tipoDocumento", method = RequestMethod.GET, produces = MediaType.APPLICATION_JSON_VALUE)** que podemos inferir de lo anterior?
26. Si tenemos un servicio web y en la url está construida de la siguiente forma **/api/servicio/recursos?key=1&valor=2**, como se obtiene el valor de "key" y "valor"?
- ☐ RequestHeader
 - ☐ PathVariable
 - ☒ RequestParam
 - ☐ RequestBody
27. ¿Como se construye la url si el método del controlador solicita que los valores vengan por el PathVariable de spring?
- ☐ /api/servicio/recurso?valor=1
 - ☒ /api/servicio/recurso/1
 - ☐ /api/servicio/recurso con body {"valor": "1"}
28. ¿Si tenemos la siguiente firma de método y sabemos que el atributo value del RequestMapping está mal, como debería quedar?

```
@RequestMapping(value = "validar", method = RequestMethod.GET, produces = MediaType.APPLICATION_JSON_VALUE)
public ResponseEntity<ResponseStandardVo> validarSolicitud(
    @RequestHeader("Authorization"),
    @PathVariable("uno") String uno,
    @PathVariable("dos") String dos,
    @PathVariable("tres") String tres
) {}
```

- ☐ Validar?uno&dos&tres
 - ☐ Validar?uno=1&dos=2&tres=3
 - ☒ Validar/uno/dos/tres
 - ☐ Validar/1/2/3
29. Si tenemos una arquitectura de varias capas, donde hay una capa controller que tendrás las clases que exponen los servicios web, otra capa fachada que tendrá las clases con la lógica de negocio y finalmente una capa dao que tendrá las clases que obtendrán la información de la base de datos, donde debería ir la anotación @Service?
- ☐ Controller
 - ☒ Fachada
 - ☐ Dao
30. ¿Como se debe inyectar un bean de spring?
- ☐ @ejb
 - ☐ @inject
 - ☒ @autowired

- @transactional

Respuestas:

3. Por definición, las variables estaticas son valores definidos en una clase de manera inmutable y pueden ser accedidos de la siguiente forma: nombre_de_la_clase.nombre de variable estatica, no se crean instancias de variables estaticas para acceder a ellas.

4. se inicializan antes que las variables de instancia de una clase cuando se ejecuta un programa.

8. Interfaz Iterator

12. java.util

14. Para agregar una dependencia existen varias formas, una es ir directamente a la pagina de repositorios de maven, buscar la dependencia requerida y copiarla, normalmente estará de la siguiente forma:

```
<dependency>
  <groupId>org.mongodb</groupId>
  <artifactId>mongo-java-driver</artifactId>
  <version>3.12.14</version>
</dependency>
```

luego esta se debe pegar dentro del archivo pom.xml dentro de las etiquetas <dependencies></dependencies>

La otra forma es, dependiendo del IDE, se puede escribir directamente en el archivo el nombre de la dependencia y automaticamente apareceran sugerencias de que dependencia usar.
por ultimo se debe recargar el archivo para que tome los cambios.

15.

```
<dependency>
  <groupId>org.mongodb</groupId>
  <artifactId>mongo-java-driver</artifactId>
  <version>3.12.14</version>
</dependency>
```

18. al ejecutar mvn package se empaqueta el codigo compilado en un .jar que queda almacenado en la carpeta target

19. queda en la carpeta lib

20. git status

21. git remote -v

22. git checkout -b branch permite cambiarnos a una rama nueva.

23. git push origin xxxx, en las xxxx debe ir el nombre de la rama a la cual se quiere subir los cambios.

24. git pull origin branch (se descarga lo que actualmente tiene la rama para iniciar a hacer cambios en ella)
git add . (se ejecuta este comando para pasar todos los archivos modificados de la rama a staging)
git commit -m "comentario" (se añade un comentario respecto a todos los cambios a subir)
git push origin branch (se suben todos los cambios realizados que previamente se ubicaron en staging por el comando git add .)

25. La url para consumir el servicio de tipo get es: <in><puerto>/api/parametros/tipoDocumento