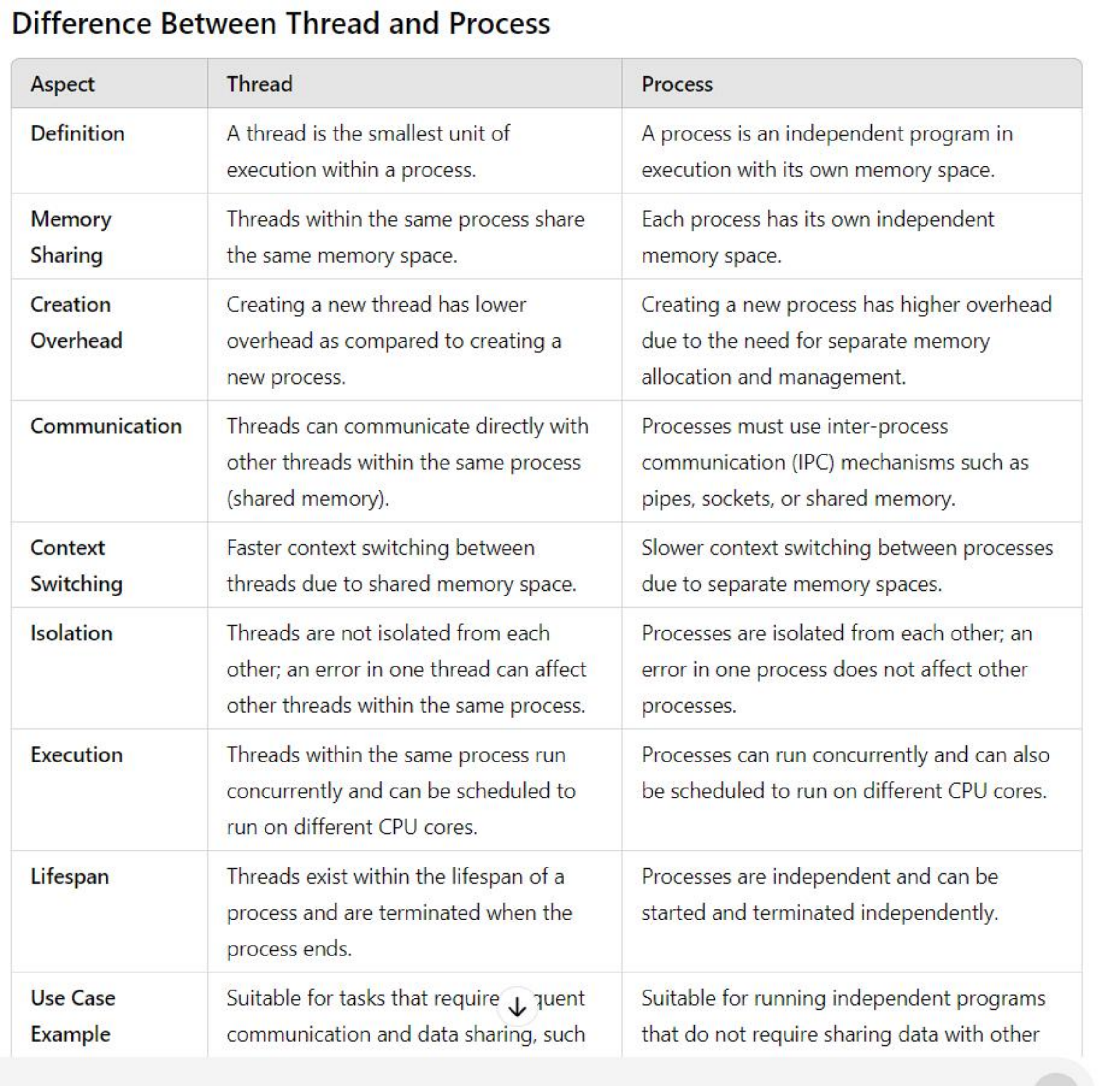
Chewy

1. Describe static and final keywords

The static keyword in Java is used to indicate that a member (variable, method, or nested class) belongs to the class itself rather than to any specific instance of the class. The final keyword in Java is used to restrict the user. It can be applied to variables, methods, and classes, each having different implications:

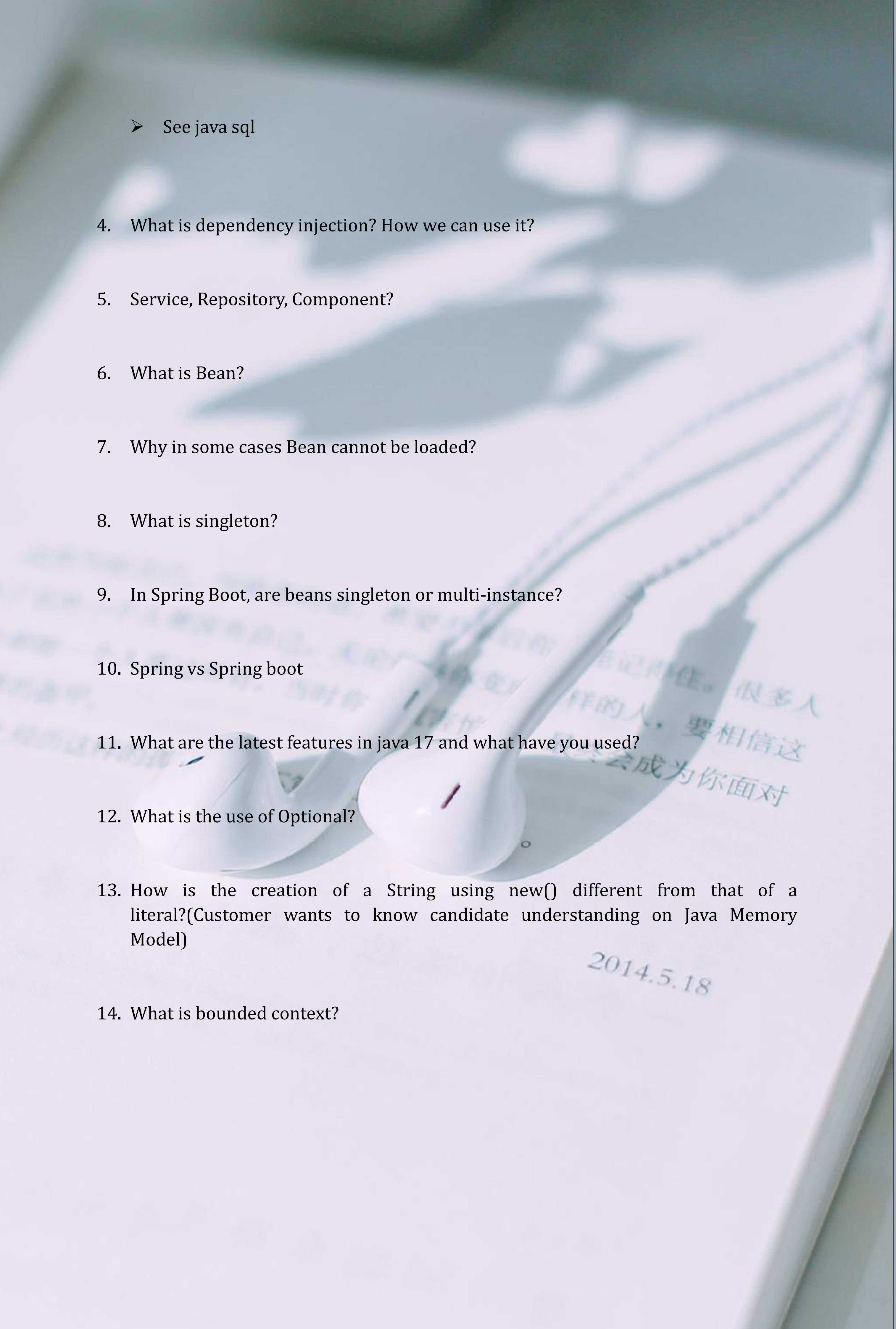
2. What is the difference between Thread vs Process?



Difference Between Thread and Process

Aspect	Thread	Process
Definition	A thread is the smallest unit of execution within a process.	A process is an independent program in execution with its own memory space.
Memory Sharing	Threads within the same process share the same memory space.	Each process has its own independent memory space.
Creation Overhead	Creating a new thread has lower overhead as compared to creating a new process.	Creating a new process has higher overhead due to the need for separate memory allocation and management.
Communication	Threads can communicate directly with other threads within the same process (shared memory).	Processes must use inter-process communication (IPC) mechanisms such as pipes, sockets, or shared memory.
Context Switching	Faster context switching between threads due to shared memory space.	Slower context switching between processes due to separate memory spaces.
Isolation	Threads are not isolated from each other; an error in one thread can affect other threads within the same process.	Processes are isolated from each other; an error in one process does not affect other processes.
Execution	Threads within the same process run concurrently and can be scheduled to run on different CPU cores.	Processes can run concurrently and can also be scheduled to run on different CPU cores.
Lifespan	Threads exist within the lifespan of a process and are terminated when the process ends.	Processes are independent and can be started and terminated independently.
Use Case Example	Suitable for tasks that require quent communication and data sharing, such	Suitable for running independent programs that do not require sharing data with other

3. What is database index?



- See java sql

4. What is dependency injection? How we can use it?

5. Service, Repository, Component?

6. What is Bean?

7. Why in some cases Bean cannot be loaded?

8. What is singleton?

9. In Spring Boot, are beans singleton or multi-instance?

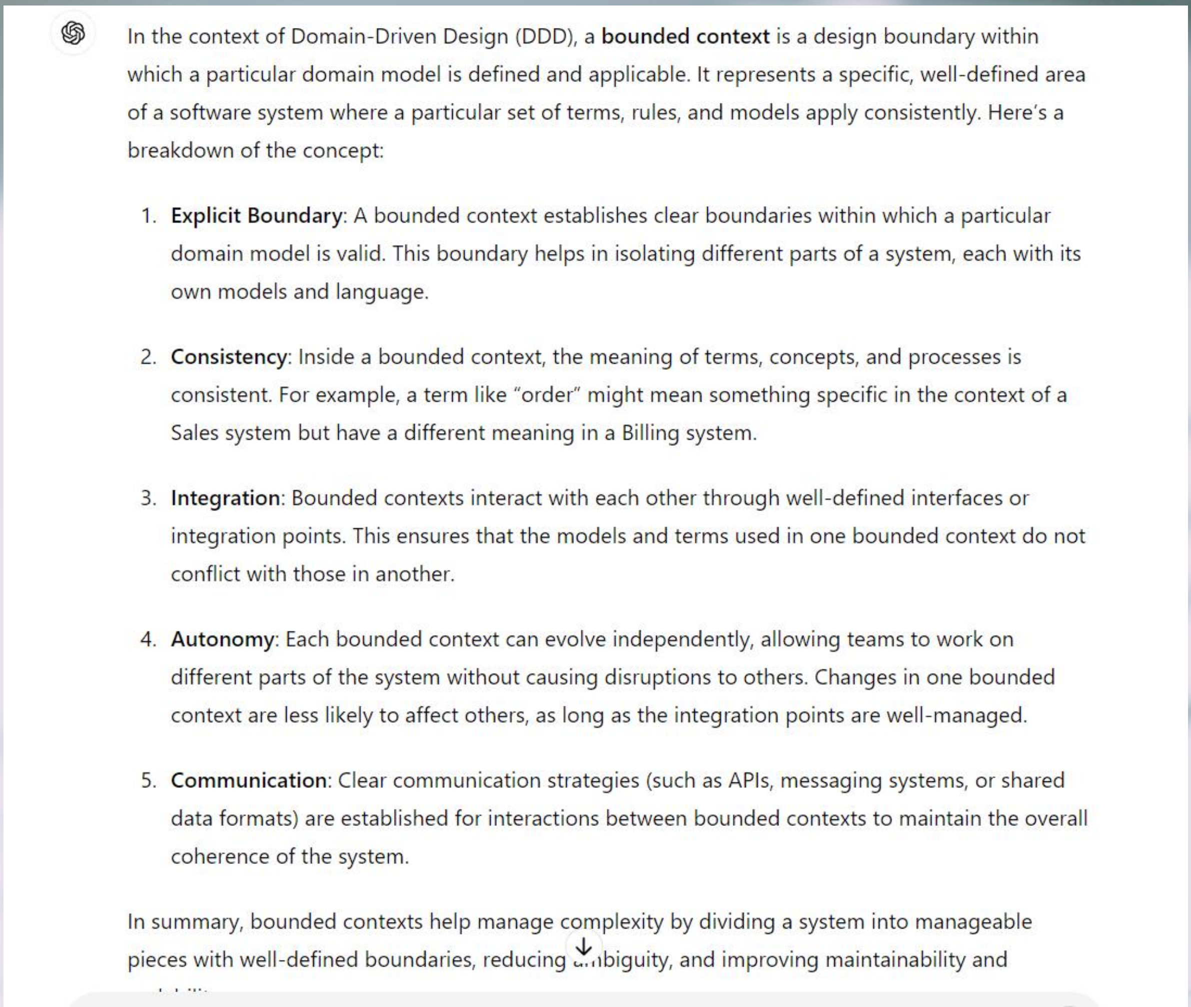
10. Spring vs Spring boot

11. What are the latest features in java 17 and what have you used?

12. What is the use of Optional?

13. How is the creation of a String using new() different from that of a literal?(Customer wants to know candidate understanding on Java Memory Model)

14. What is bounded context?



In the context of Domain-Driven Design (DDD), a **bounded context** is a design boundary within which a particular domain model is defined and applicable. It represents a specific, well-defined area of a software system where a particular set of terms, rules, and models apply consistently. Here's a breakdown of the concept:

1. **Explicit Boundary**: A bounded context establishes clear boundaries within which a particular domain model is valid. This boundary helps in isolating different parts of a system, each with its own models and language.

2. **Consistency**: Inside a bounded context, the meaning of terms, concepts, and processes is consistent. For example, a term like "order" might mean something specific in the context of a Sales system but have a different meaning in a Billing system.

3. **Integration**: Bounded contexts interact with each other through well-defined interfaces or integration points. This ensures that the models and terms used in one bounded context do not conflict with those in another.

4. **Autonomy**: Each bounded context can evolve independently, allowing teams to work on different parts of the system without causing disruptions to others. Changes in one bounded context are less likely to affect others, as long as the integration points are well-managed.

5. **Communication**: Clear communication strategies (such as APIs, messaging systems, or shared data formats) are established for interactions between bounded contexts to maintain the overall coherence of the system.

In summary, bounded contexts help manage complexity by dividing a system into manageable pieces with well-defined boundaries, reducing ambiguity, and improving maintainability and

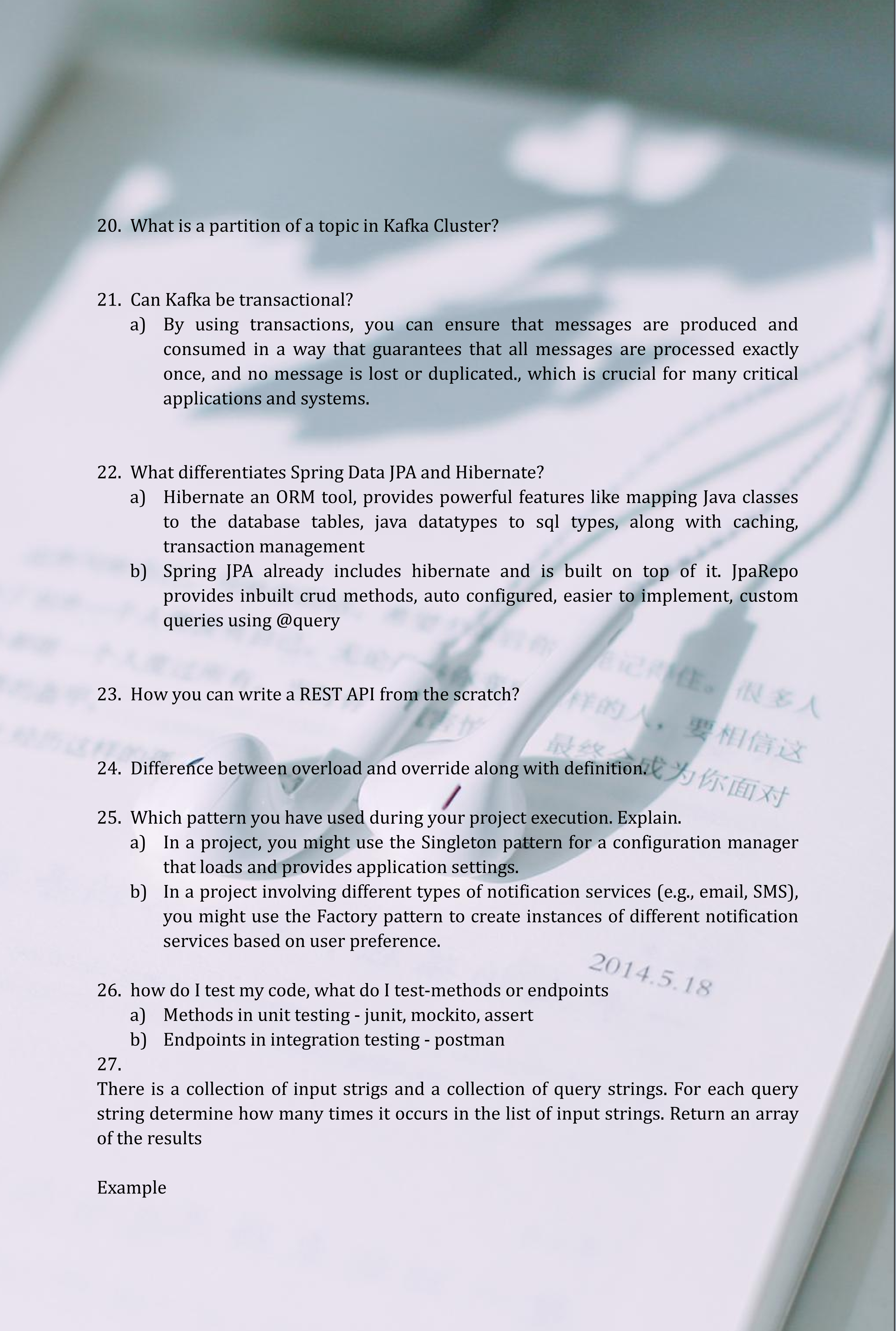
15. How to implement multiple inheritance in Java ?

16. What is Service, Repository, Component?

17. How are the @RestController and @Controller Annotation different?

18. Can the static methods be overloaded?

19. Can you describe sharding in MongoDB?
    a) Sharding in MongoDB is a method used to distribute data across multiple servers to ensure horizontal scalability, improve performance, and handle large volumes of data. Each shard is a separate MongoDB instance or a replica set that holds a subset of the data. Shards work together to provide the complete dataset.



20. What is a partition of a topic in Kafka Cluster?

21. Can Kafka be transactional?
    a) By using transactions, you can ensure that messages are produced and consumed in a way that guarantees that all messages are processed exactly once, and no message is lost or duplicated., which is crucial for many critical applications and systems.

22. What differentiates Spring Data JPA and Hibernate?
    a) Hibernate an ORM tool, provides powerful features like mapping Java classes to the database tables, java datatypes to sql types, along with caching, transaction management
    b) Spring JPA already includes hibernate and is built on top of it. JpaRepo provides inbuilt crud methods, auto configured, easier to implement, custom queries using @query

23. How you can write a REST API from the scratch?

24. Difference between overload and override along with definition.

25. Which pattern you have used during your project execution. Explain.
    a) In a project, you might use the Singleton pattern for a configuration manager that loads and provides application settings.
    b) In a project involving different types of notification services (e.g., email, SMS), you might use the Factory pattern to create instances of different notification services based on user preference.

26. how do I test my code, what do I test-methods or endpoints
    a) Methods in unit testing - junit, mockito, assert
    b) Endpoints in integration testing - postman

27.

There is a collection of input strigs and a collection of query strings. For each query string determine how many times it occurs in the list of input strings. Return an array of the results

Example

stringList = ['ab','ab','abc']
queries = ['ab,'abc','bc']

There are 2 instances of 'ab', 1 of 'abc' and 0 of 'bc'. For each query add an element to the return array results = [2,1,0]