Assignment 1: Resilient Architecture with Sharding Strategies

As a Cloud Architect for an e-commerce company, your task is to design a resilient architecture that involves sharing of data or workload using various strategies. Specifically, you are asked to use the lookup strategy, range strategy, and hash strategy for load balancing or data sharing purposes.

Task:

1. Research and familiarize yourself with the lookup strategy, range strategy, and hash strategy for sharing data or workload.
2. Identify relevant use cases in which these strategies could be applied in an e-commerce context.
3. Develop a plan for implementing each strategy, including how to handle potential failures or issues that may arise.
4. Provide a detailed explanation of how each strategy works and how it contributes to the overall resilience of the architecture.
5. Outline the key performance indicators (KPIs) that will be used to measure the effectiveness of each strategy.

## Questions:

1. What is the lookup strategy, and how can it be used for load balancing or data sharing?
2. What is the range strategy, and how can it be used for load balancing or data sharing?
3. What is the hash strategy, and how can it be used for load balancing or data sharing?
4. In what specific use cases would each strategy be most appropriate?
5. What are the potential failure scenarios for each strategy, and how would you handle them?
6. How does each strategy contribute to the overall resilience of the architecture?
7. What KPIs will you use to measure the effectiveness of each strategy, and how will you track them?

## Checklist:

1. Research lookup strategy, range strategy, and hash strategy for data sharing.
2. Identify relevant use cases in e-commerce context.
3. Develop a plan for implementing each strategy.
4. Provide detailed explanation of how each strategy works and how it contributes to overall resilience.
5. Outline key performance indicators (KPIs) for measuring effectiveness of each strategy.
6. Plan for handling potential failures or issues for each strategy.
7. Create a roadmap for implementing the strategies.

## Deliverables:

1. A written report detailing the plan for implementing each strategy and how they contribute to the overall resilience of the architecture.
2. A diagram of the proposed architecture, including how the lookup, range, and hash strategies will be implemented.
3. A summary of the KPIs that will be used to measure the effectiveness of each strategy.
4. A detailed plan for handling potential failures or issues for each strategy.
5. A roadmap for implementing the strategies.

## Example:

The e-commerce company has a large inventory of products that are stored in multiple warehouses across different regions. To ensure high availability and performance, the company decides to implement a shared architecture for its inventory management system.

The system should be able to handle a large number of requests from customers and also be resilient to failures in individual warehouses. The company also wants to ensure that the system is scalable and can easily accommodate new warehouses as the business grows.

To achieve this, the company decides to use a hash strategy for sharing the inventory data across the different warehouses. Each warehouse is assigned a unique hash value, and the inventory data for that warehouse is stored in a shard with the same hash value.

The company also implements a lookup strategy for routing customer requests to the appropriate warehouse. When a customer makes a request for a particular product, the system looks up the hash value for that product and then routes the request to the warehouse with the corresponding hash value.

In addition to this, the company also implements a range strategy for load balancing. If one warehouse experiences a sudden surge in traffic, the system can automatically redirect some of the traffic to other warehouses in the same range to balance the load and prevent overloading of any single warehouse.

By using this shared architecture with different strategies, the e-commerce company can ensure high availability, scalability, and resiliency for its inventory management system.

## Steps to Design Sharding strategy

Task:

Design a resilient architecture with sharing strategies for the e-commerce company using the following techniques:

1. Lookup strategy: This strategy involves using a lookup table or a directory to locate the specific resource or service required by the application. The lookup table contains information about the location, availability, and performance of each resource or service.
2. Range strategy: This strategy involves dividing the resources or services into different ranges based on their characteristics such as type, size, capacity, or location. The application can then select the appropriate range based on its requirements.
3. Hash strategy: This strategy involves hashing the resource or service identifier to map it to a specific location or node in the system. The hashing algorithm ensures that the load is distributed evenly across the nodes and that each node can handle a similar load.

Requirements:

1. Define the business requirements and use cases for the e-commerce company, including the types of applications, workloads, users, and data involved.
2. Identify the resources or services that need to be shared and the reasons for sharing them, such as performance, availability, scalability, or cost.
3. Define the lookup table or directory for the lookup strategy, including the schema, data sources, access control, caching, and synchronization mechanisms.
4. Define the range categories and criteria for the range strategy, including the allocation, rebalancing, and scaling mechanisms.
5. Define the hashing algorithm and distribution strategy for the hash strategy, including the key space, consistency, replication, and fault tolerance mechanisms.
6. Define the monitoring, logging, and alerting mechanisms for the sharing strategies, including the health checks, metrics, thresholds, and notifications.
7. Define the testing, deployment, and rollback procedures for the sharing strategies, including the integration, automation, and validation mechanisms.
8. Evaluate the performance, availability, scalability, and cost of the sharing strategies, using the appropriate benchmarks and metrics.
9. Identify the potential risks, constraints, and trade-offs of the sharing strategies, such as data consistency, performance overhead, complexity, or vendor lock-in.
10. Define the mitigation strategies and contingency plans for the risks, such as redundancy, backups, snapshots, or failover mechanisms.

Checklist:

1. Have you defined the business requirements and use cases for the e-commerce company?
2. Have you identified the resources or services that need to be shared and the reasons for sharing them?
3. Have you defined the lookup table or directory for the lookup strategy?
4. Have you defined the range categories and criteria for the range strategy?
5. Have you defined the hashing algorithm and distribution strategy for the hash strategy?
6. Have you defined the monitoring, logging, and alerting mechanisms for the sharing strategies?
7. Have you defined the testing, deployment, and rollback procedures for the sharing strategies?
8. Have you evaluated the performance, availability, scalability, and cost of the sharing strategies?
9. Have you identified the potential risks, constraints, and trade-offs of the sharing strategies?
10. Have you defined the mitigation strategies and contingency plans for the risks?