# SRE/DevOps Interview Q&A (Short Format)

## Panel vs Dashboard in Grafana

Panel: Single metric visualization.  
Dashboard: Group of panels for multiple metrics.

## Variable in Grafana?

Placeholder to filter data dynamically (e.g., $instance, $env).

## Use same dashboard for multiple instances?

Yes, use variables and design one dashboard.

## Variable Examples in Grafana

$instance, $env, $region, $job, $status.

## How to design such dashboards?

Add variables → Use in panel queries → Panels auto-update with dropdown.

## Single or multiple dashboards?

One dynamic dashboard is better using variables.

## Different visualization types?

Time series, Stat, Gauge, Table, Pie chart, Logs, Geomap, etc.

## If dashboard shows no data, how to troubleshoot?

Check data source, query, time range, variables, and Grafana logs.

## Splunk vs Grafana

Splunk: Log analysis.  
Grafana: Metrics visualization.

## Grafana vs Kibana

Kibana: Log search (via Elasticsearch).  
Grafana: Metric dashboards (multi-source).

## How Kibana interacts with Elasticsearch?

Kibana sends KQL/Lucene queries → Elasticsearch responds → Kibana visualizes.

## stats vs timechart in Splunk

stats: Summary (non-time).  
timechart: Time-based trends.

## Can we create the same view in both?

Yes, similar data shown as table (stats) or graph (timechart).

## Optimize slow SQL queries – steps?

Use EXPLAIN, indexes, avoid SELECT \*, filter data early.

## Limitations of joins?

Can be slow, complex, use more memory, and sensitive to NULLs.

## AWS Cognito – Uses in short?

User auth, federated login, JWT tokens, secure APIs.

## High severity incidents – your approach?

Acknowledge → Alert team → Mitigate → RCA → Document.

## Incident vs Change Management?

Incident: Unplanned fix.  
Change: Planned deployment.

## How to check the last status of an issue?

Check ticket, monitoring, logs, and confirm with the team.

## Increase app resilience – how?

Auto-scaling, health checks, retries, canary deploy, multi-AZ.

## Avoid downtime during patching (app)?

Rolling update, blue-green, load balancer, canary release.

## Avoid downtime during DB patching?

Multi-AZ RDS: AWS patches standby → automatic failover.

## How does Multi-AZ RDS patching work?

Patch standby → promote it → minimal downtime.

## Disagreement with client – your approach?

Listen, clarify, suggest win-win, escalate if needed, stay professional.

## App is critical & needs 24x7 – your thought?

On-call rotation, monitoring, automation, SOPs, DR setup.

## How to avoid DB patching downtime?

Use Multi-AZ, promote standby, schedule off-peak.