Selling CI/CD

1. Fundamentals of CI/CD:
   - **Continuous Integration**: The practice of merging all developers' working copies to a shared mainline several times a day.
   - **Continuous Delivery**: An engineering practice in which teams produce and release value in short cycles.
   - **Continuous Deployment**: A software engineering approach in which the value is delivered frequently through automated deployments.
2. You might need CI/CD if you are:
   - Investing more time in a release cycle than delivering value.
   - Wasting time every time we finish a feature on merge issues.
   - Code gets lost because of botched merges.
   - Unit test suite hasn't been green in ages.
   - Deployments contribute to schedule slip.
   - Friction between ops and development departments.
   - Only one engineer can deploy system.
   - Deployments are not cause for celebration.

3. Benefits of CI/CD:

Technical Language	Value	Translation
Catch Compile Errors After Merge.	Reduce Cost	Less developer time on issues from new developer code.
Catch Unit Test Failures.	Avoid Cost	Less bugs in production and less time in testing.
Detect Security Vulnerabilities.	Avoid Cost	Prevent embarrassing or costly security holes.
Automate Infrastructure Creation.	Avoid Cost	Less human error, Faster deployments.

Automate Infrastructure Cleanup.	Reduce Cost	Less infrastructure costs from unused resources.
Faster and More Frequent Production Deployments.	Increase Revenue	New value-generating features released more quickly.
Deploy to Production Without Manual Checks.	Increase Revenue	Less time to market.
Automated Smoke Tests.	Protect Revenue	Reduced downtime from a deploy-related crash or major bug.
Automated Rollback Triggered by Job Failure.	Protect Revenue	Quick undo to return production to working state.