==========
Module-2 : Real-Time Tools (20+)
==========

=> Scrum Master will setup project in JIRA. Team members will create stories in jira and will assign stories to work (story means task).

=> Development team should send request to devops team to create github/bitbucket repository for code integration.

Note: DevOps team will create git repo and will share repository URL.

=> Developer should create a project skelton using maven/gradle build tools.

Note: Developer will push project skelton to git repository then other developers will clone git repository and will start coding.

=> As part coding developers should use debugging to understand code execution flow.

=> Based on project requirements we will use kafka + redis to implement our microservices.

=> Once coding is completed we need to test our backend apis functionality using POSTMAN + SWAGGER.

=> We need to perform code review using SONARQUBE.

=> Implement logging in the application using LOG4J. logging is used to trace execution details of the application and we can identify root cause of excetions.

=> Implement unit testing in the application using JUNIT and Mockito.

=> Implement Performance Testing on the Project using JMETER.

=> Developers should write dockerfile to containerize the application using DOCKER and push it to git repo.

=> Developers will send email to DEVOPS team to create JENINS CI CD Pipeline.

Note: As part of CI CD pipeline devops team will configure several tools to automate project build and deployment process.

=> DevOps team will deploy our project into kubernetes cluster using JENKINS CI CD pipeline

=> Developers can run jenkins pipeline when we want to deploy latet code.

=> Developers should monitor logs of the application using ELK/Splunk softwares.

==========
Tools That every java developer should know
==========

1) JIRA : Project Management + Bug Reporting

2) GitHub / Bitbucket : Source Code Repository servers

3) Maven / Gradle : Build Tools

4) Log4J + SLF4J : Logging tools (store log msgs)

5) ELK / Splunk : Log Monitoring tools (read log msgs)

6) JUnit + Mocking : Unit Testing

7) Kafka : Message Broker (Streaming Platform)

8) Redis : Distributed Cache

9) POSTMANT : To test backend apis

10) Swagger : To generate api documentation

11) Docker : Containerization

12) Kubernetes : Orchestration (manage app deployments)

13) Jenkins : for CI CD (to automate build + deployment)

14) SonarQube : Code Quality Checking

15) JMETER : for load/performance testing

16) Jacoco : Code Coverage

17) Nexus/Jfrog : Artifactory server (store jars/wars)