# Mulesoft

**Introducing Any point Platform**

- Describe the benefits of Any point Platform and MuleSoft' s approach
- What is MuleSoft and ESB, and competitors to MuleSoft
- Advantages with MuleSoft
- Mule anypoint platform and its components
- MuleSoft key capabilities
- Describe the purpose of each file and folder in a Mule project
- Describe the role of each component in building application networks
- Logging message processors data
- Read and write message properties
- Build, run, and test a Mule application
- Application local testing
- Interview question

**Anypoint Studio**

- What is anypoint Studio
- How we use it to develop mule flows
- How to download and use it in our local machine
- Mule Palette, Canvas, Package Explorer, Console
- How to integrate with Maven and Github Repository

**Mule application structure & Basics**

- Mule attributes, payload and variables
- Flows, sub-flows, private flows, async, and flow references
- Processing strategies
- Read and write the mule event properties.
- Properties file and it's importance in runtime
- Global elements and Global configurations

**Debugging and Troubleshooting Mule Applications**

- Breakpoint and Inspecting the mule application
- Explanation on mule event processing
- Mule Message structure with all the events captured in message
- Step by step message processing explanation during mule event.

**Mule Elements**

  - Connectors
  - Components
  - Scopes
  - Transformers
  - Filters
  - Flow Control
  - Exception handling

- Security

**Mule Error Handling**

- Handle messaging errors at the application, flow, and processor level
- Handle different types of errors, including custom errors
- Use different error scopes to either handle an error and continue execution of the parent flow or propagate an error to the parent flow
- Set the success and error response settings for an HTTP Listener
- Set reconnection strategies for system errors

**Triggering the Flows**

- Read and write files
- Trigger flows when files are added, created, or updated with in the folder
- Trigger flows when new records are added to a database table
- Schedule flows to run at a certain time or frequency
- Persist and share data in flows using the Object Store
- Publish and consume JMS messages
- Scheduling the events on time basis

**Records Processing**

- Processing records using For Each
- Bulk data processing
- Streaming data processing
- Exporting data from excel, flat file, JSON and XML
- Dumping data from one database to other database

**API Led Connectivity**

- System Layer
- Process Layer
- Experience Layer
- Explaining what web services and API's
- Secure and unsecure API's
- Why we use multiple layers
- Difference between each layer

**Any point platform**

- What is Anypoint platform
- Design Center
- Exchange
- API Manager
- Runtime Manager
- Access Management
- Data Gateway
- Visualizer

**API Designing (RAML)**

- What is RAML
- Designing Restful API's
- History of RAML
- Mocking the RAML services
- Publishing the services
- Implementing the RAML form Anypoint Studio

**Rest API's and SOAP API's**

- Design and develop web API's
- SOAP(WSDL) vs Rest web services and importance
- Developing the SOAP services
- Developing the Rest services
- Consuming the SOAP services
- Consuming the Rest services

**Deploying and Managing API's**

- Deploying from Anypoint Studio to Runtime Manager
- Deploying the project as war file
- Deploying the application in cloud vs on premises.
- Managing the deployed applications
- Event or Error management

**API Manager**

- API Auto discovery
- What are the policies available
- How to apply policies
- How to secure your API's
- OAuth and OAuth 2 Security
- How to sync project using version

**Data Weave**

- What is Data Weave
- Why we use Data Weave
- Modules in Data Weave
- Best coding practices
- Data Weave transformation techniques
- Calling flows from Data Weave

**MUNIT (Testing)**

- Unit Testing
- Process Testing
- Create acceptance criteria
- Fail and pass tests

- Refactor test cases
- Refactor Mule applications
- Mocking test cases with examples

**Continuous Deployment using Jenkins**

- Git Repository
- Maven
- Jenkins
- Continuous integrating with Dev Ops tools to automate the deployment
- Managing the build versions
- Triggering the events on build failures

**Practice Examples**

- Designing API Error handling framework
- Mule Batch Processing
- How to host HTPPs
- Creating a proxy API using OAuth 2.0
- Advance Data Weave Tips
- How to secure sensitive data
- Parallel processing
- Many more practical examples discussed while explaining each topic…

**Interview Preparation**

- Discussing about interview question and answers
- Practical scenarios we face in real-time
- Guidance on preparing the Resume
- Guidance on real-time project description and explanation