1. Invoke custom java class using dataweave.

Steps:

1. Create Java Class in Anypoint studio
2. Set the variable value
3. Drag and drop Transform message and write code to invoke java class in Dataweave.
4. In import the java class using import java!<packagename>::classname
5. In that called the static method with the method name.
6. The program resulted from the substring for the given string in the variable.
7. Tested the flow with the postman and log the result using the logger component.

2. Difference between map and mapObject.

map – Iterates over items in an array and outputs the results into a new array.

Map object – Iterates over an object using a mapper that acts on keys, values, or indices of that object.

3. Different ways of declaring a variable

1. Global variables are initialized in the header of the DataWeave script and can be referenced by name from anywhere in the body of a DataWeave script.

2. The header of a DataWeave script accepts a var directive that initializes a variable, for example, var string='Tarun'. You can declare multiple global variables on separate lines in the header.
3. Local variables are initialized in the body of the DW script and can be referenced by name only from within the scope of the expression where they are initialized.

4. Map operator in dataweave.

1. Using a map operator we can iterate through an array.
2. In this problem, I take an input of JSON array which contain the employee data
3. Using the transform message component I wrote the dataweave code which iterates through the array.
4. In that data, I collected the first name and last name of the employee and concatenated the strings.
5. Used the logger component to log the output.

5. Filter Operator in dataweave

1. Connected to the local database. Where I collected the employee information.
2. Using filter operator iterated through an array when satisfies the given condition it triggers the result.
3. Wrote the dataweave code in transform message
4. Used the logger component to log the output.
5. Used the postman to triggering the flow.

6. How to use the comment line in Dataweave?

```
// single line comment.

/*
* This is.
* multi-line.
* comments in.
* DataWeave.
*/
```

7, Custom Dataweave Function Using Custom Module

1. Created a project in anypoint studio and configure the listener.
2. Created the module under src/main/resources
3. In that module created the .dwl file
4. Dragged the transform component and configure the .dwl file.
5. Dragged the logger component to log the events.

8, Merging of arrays

Defines three arrays of numbers, creates another array containing those three arrays, and then uses the flatten function to convert the array of arrays into a single array with all values.

9. To convert string to the array you can use SplitBy function of dataweave 2.0

Using the split function we can convert the string into an array**.**

10. CONVERT ARRAY TO STRING IN DATAWEAVE 2.0

Using the join function we can convert the array into a string.

11. Working with File connectors and perform the following operation.

1. Created a mule project that dragged an "On New or Updated File".
2. And Added the Connecting Configuration to it which is the working directory.
3. And set the fixed frequency 2 and start delay 0 seconds for checking the working directory for every two seconds any file updated or not.
4. Set the auto-deletion is true and added the Move directory (Whenever the uploaded
   the file is traced it deleted and moved to the target folder)
5. Add a logger to display the output on the screen.

12. Transform JSON file to XML using file connector

1. Read the Particular file from the directory using the read
2. Performing the transform action using transform message
3. Write a new file using the file write component
4. Save the file in the target directory.

13. Created a YAML file and entered the properties of the local environment

1. Created a new mule project in mule4

2. And added the new YAML file called local.YAML and added the details HTTP properties and Database Properties
3. Configuring the YAML properties in HTTP and DB connectors.
4. Perform the Select operation from the database
5. Handling the arrays in data wave 2.0

14. Executing the python script in mule4

1. Drag and drop HTTP listener component from the mule palette and configure this to trigger the request.
2. Create the transform message component to pass the payload on which we will execute the script
3. Drag and drop script component from mule palette.
4. Write the python code under Code
5. Pass the parameters and add engine (Jython (python)).
6. Create the transforming message to print the payload
7. Logger component to logging the events and results.

15. Executing the ECMA(js) Script in mule4.

1.Drag and drop the HTTP listener component from the mule palette and configure this to trigger the request.
2.Create the transform message component to pass the payload on which we will execute the script
3.Drag and drop script component from mule palette.
4.Write the python code under Code
5.Pass the parameters and add the engine ECMA.
6.Create the transforming message to print the payload
7.Logger component to logging the events and results.

16. Access Secure properties in mule 4.

1. Created the sample.properties file
2. Added the properties in that file
3. Dragged the transform component we can add the dataweave code to access the properties in a file.
4. Dragged the logger component to log the events.

17. Performing null check in dataweave

1. Dragged the HTTP listener to configure the details.
2. Added the transforming message and write the dataweave code which accepts the payload and checks the message is null or not using the isNull function.
3. Added the logger component to log the events.

18. What is Munit Matchers

MUnit matches are a set of DataWeave functions to define assertion conditions for any value in an expression. When defining matches, include the prefix MunitTools:: in the expression.

19. Types of Munit Matchers:

1. Core Matchers
2. String Matchers
3. Comparable Matchers
4. Iterable and Map Matchers

Mulesoft Development:

1. **What is parallel processing in mule4?**

   In parallel processing, a list of actions is executed concurrently but independently.

2. **Achievement of parallel processing using scattergather**

- Drag the HTTP listener and configure the details.
- Connected the scatter-gather to HTTP listener.
- Started the three 3 independent flows in the project and named the flows
- Connected three flow references and configure the flow names to it.
- And again added the transforming message which transforms the results of different flows.
- Triggered the flow using the postman.

3. **What is the difference between map, mapObject and pluck?**

- ❖ map is used to iterate over array
- ❖ mapObject is used to iterate over object and return object
- ❖ pluck iterates over an object and returns an array of keys, values, or indices in that object.

4. **How many phases are there for batch job in Mule 4**

Each batch job contains 3 different Phase

Load and Dispatch.
Process
On Complete.

5. What is batch processing and work with mule4

1. Dragged the Batch flow
2. Dragged the onnewupload the component
3. Logg Component for debugging purpose.
4. In the Process take each line of the file.
5. Write the file using now().

6. Difference between batch and foreach

For each do the processing in single thread while Batch Process performs multi-threaded processing more information.

Error Handling in mule 4:

1. **On Error Continue :**

    1) Catches the error and don not repeat its as an error; hence flow processing continues.
    2) It can be used where flow execution should not be distributed even after error is reported.

2. **On Error Propagate :**

1. It process the error message and rethrows again to the parent.
2. Once error is rethrown no further processing is done.

3. **Try and catch scope :**

It allows to do error handling for some specific component in a flow

4. Error message structure :

- Casue
- Child Errors
- Description
- Detailed Description
- Error Message
- Error type