**URI Parameters:**

\* URI Parameter is a variable element enclosed in curly braces {} inside relative

URI of resources.

\* A URI is a resource identifier that uniquely identifies a specific instance of a

resource. URI is Unique Resource Identifier as the name suggests, it should get a

unique resource.

\* The parameter which is part of the URL that is passed to get the unique resource is the URI parameter.

Let's consider an example where you want identify the employee on the basis of employeeID, and in that case, you will be using the URI params.

GET /employee/{employeeID}

**Query Parameters:**

Query Parameter is basically used to filter or sort the resources. It is passed in the URL as query string in key-value form.

you want to filter the employee on the basis of designation, and in that case, you will be using Query Parameter.

GET /employee?Designation=SSE

Header Parameters:

**Header parameters** are included in the request **header**. Usually, the **header** just includes authorization **parameters** that are common across all endpoints; as a result, the **header parameters** aren't usually documented with each endpoint.

**Flow be like:**

Https listener -> set variable for uri -> Logger -> set variable for query -> Logger ->

Set variable for headers-> Loggers

**HTTP listener:**

The HTTP listener is an event source that enables you to set up an HTTP server and trigger flows when HTTP requests are received.

You can choose what methods the source accepts, such as GET, POST or a list of methods, and on which path to accept requests, thereby allowing the routing of requests through different flows.

Once a request is accepted by the listener, the corresponding flow is triggered with the HTTP body as payload and the HTTP data as attributes (headers, query parameters and so on).

When the flow finishes its execution, the HTTP listener enables you to customize the HTTP response based on whether the execution was successful or not, so that different status codes can be returned.

**Set Variable:**

The Set Variable (Set variable) component is for creating or updating a variable to store values for use within the flow of a Mule app. You can store simple literal values such as strings or messages, message payloads, or attribute objects. For example, you might store the original payload of a message (before it is processed) so you can use it later in the flow or within an error handler.

The Set variable component is not recommended for complex expressions or transformations. You should instead use it for simple ones, such as selections, and use the [Transform Component](https://docs.mulesoft.com/mule-runtime/4.2/transform-component-about) for complex scenarios. In which, you give syntax that related to what is the uri params you are going to use at the time of postman as a key for posting a value. As we took 3 set variables, in remaining two set variables we have to take query params key and headers key for posting value at the time of postman

**Logger:**

This Core component helps you monitor and debug your Mule application by logging important information such as error messages, status notifications, payloads, and so on. You can add a Logger anywhere in a flow, and you can configure it to log a string that you specify, the output of a Data Weave expression you write, or any combination of strings and expressions.