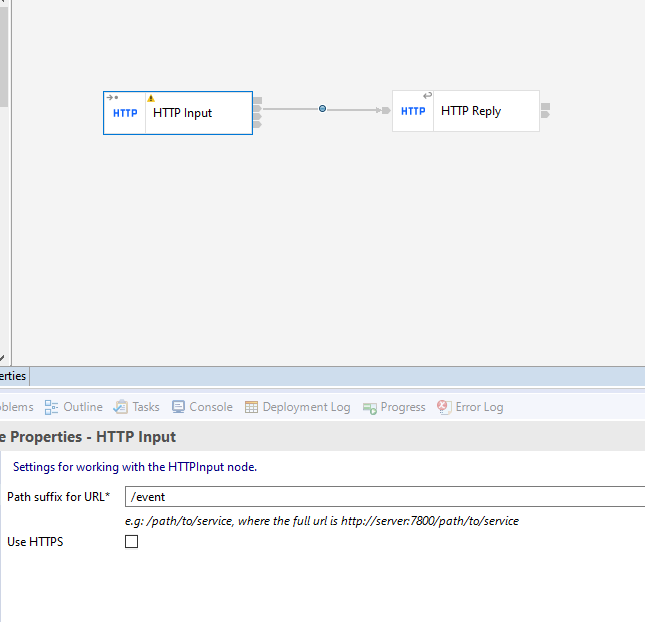
**Event Monitoring**

Introduction:

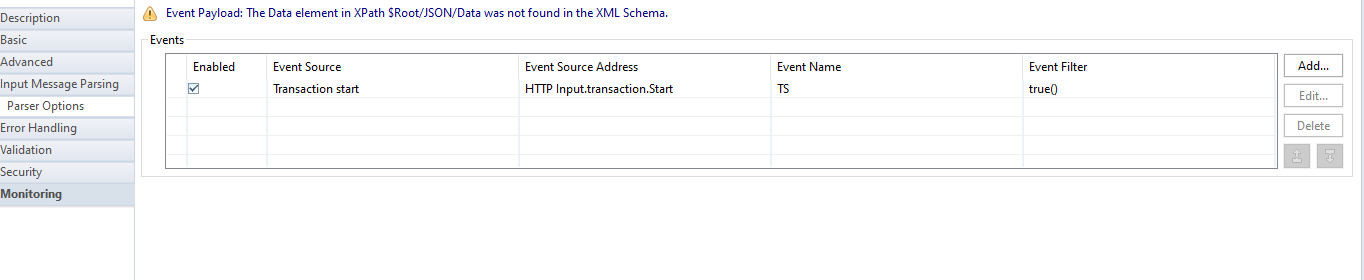
In IBM Integration Bus , event monitoring plays a crucial role in tracking the execution of integration flows.Message flows can be configured to emit events. The events can be read and used by other applications for transaction monitoring and auditing, and business process monitoring and we can store it in Database.

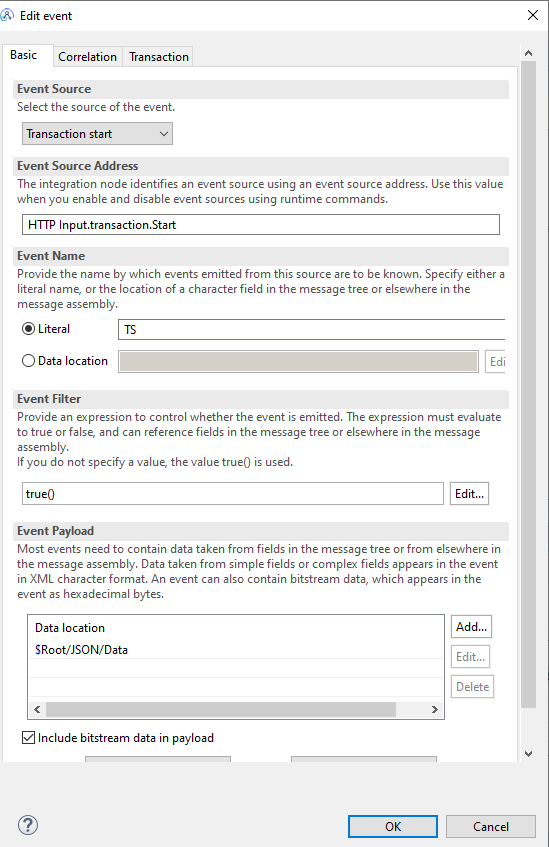
Process:

Create an application with HTTP Nodes.

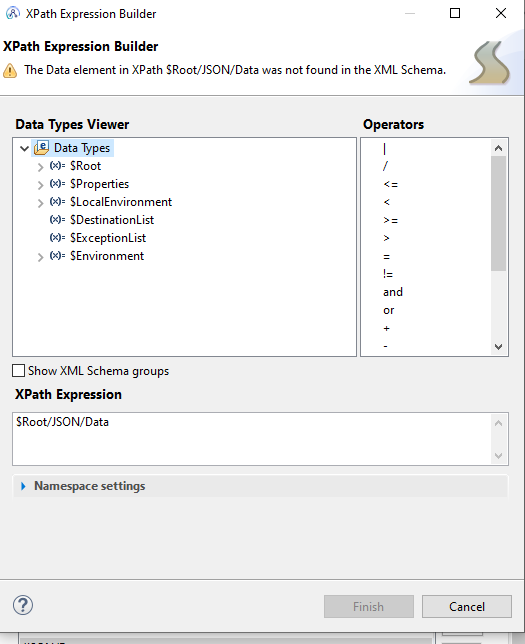


Open the HTTPInput Node properties and provide the Path suffix Url and Input message parsing as JSON.

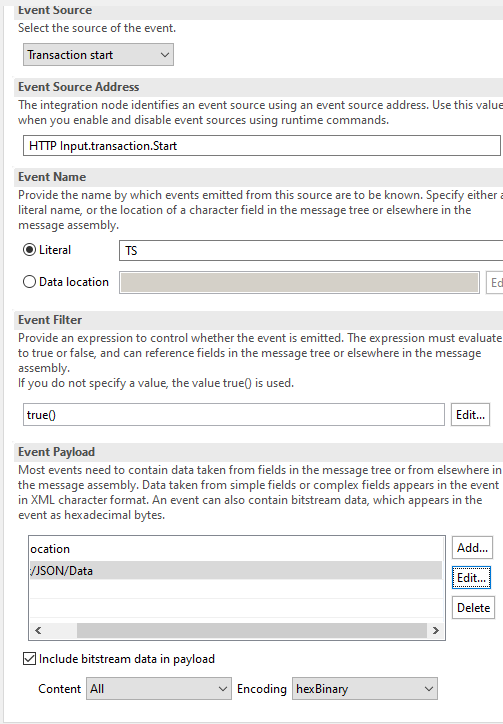
Go to monitoring tab and edit the event.



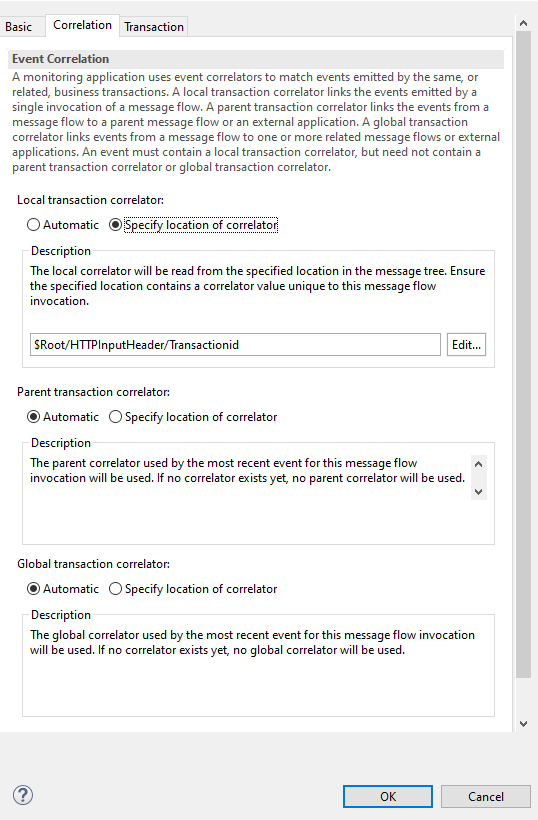
In the Event Payload tab add the x-path expression of your Input data.



Enable the include bitstream data in payload and select the Encoding option as hex Binary from the given drop down.

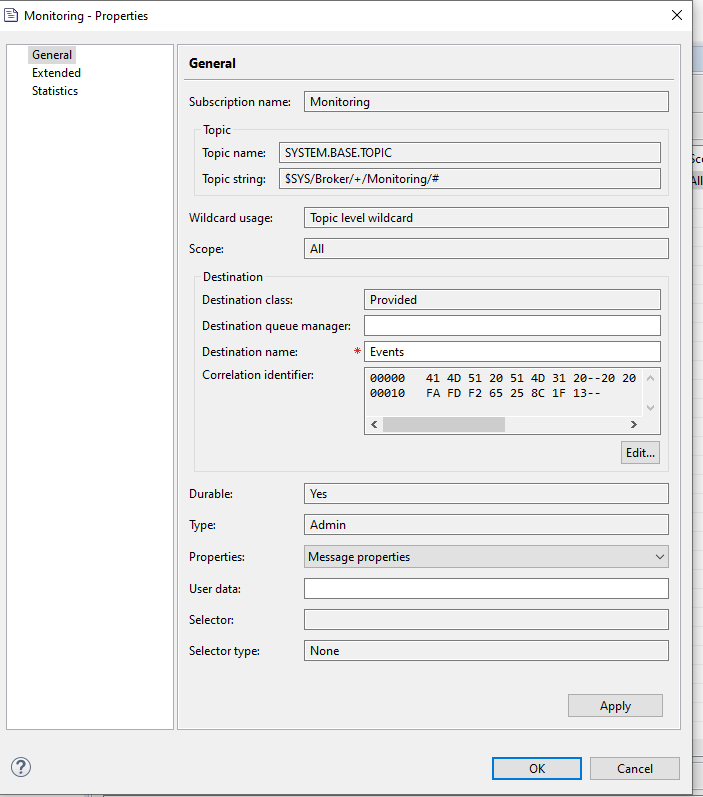


Go to correlation tab select the Specify Location for Correlator option.And provide the x-path expression and click on OK.

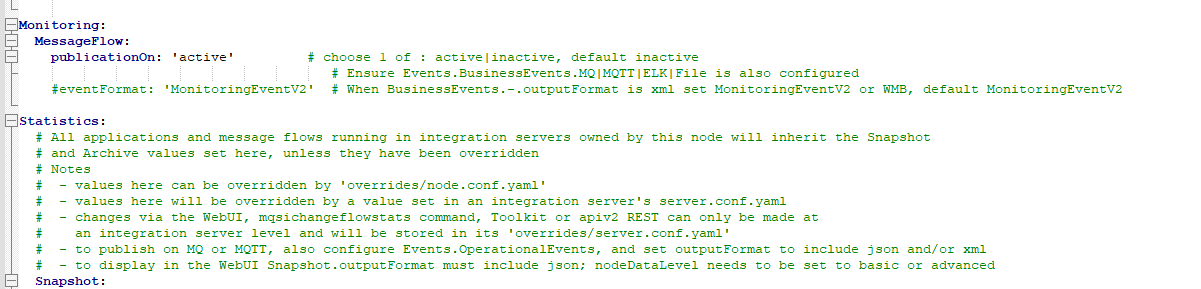


Configuration on MQ:

In MQ create a Subscription , in that give the queue name in which logs will be store.



Now go to the Node Yaml file and enable the Monitoring property.



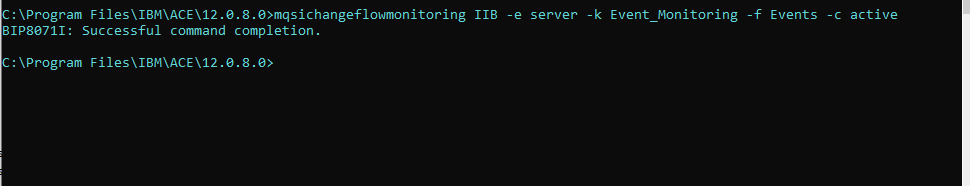
Now run the below mentioned commands in console

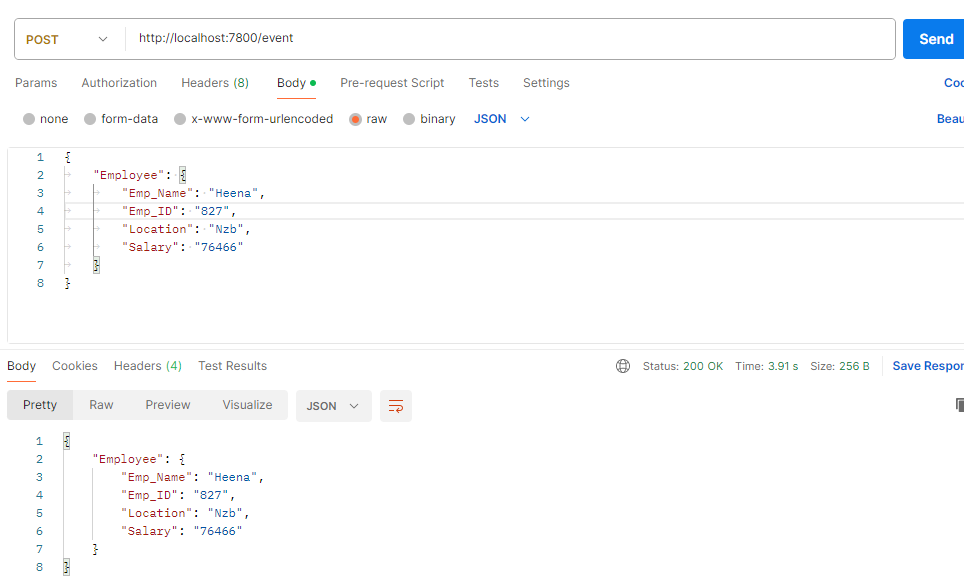
→ mqsichangeflowmonitoring BrokerName -e ServerName -k Applicationname -f messageflow -c active

→mqsireportflowmonitoring BrokerName -e ServerName -k Applicationname -f messageflow

* Stop and Start your Node to apply the changes
* Deploy the message flow

Note: Whenever redeploy the flow you need to run the above commands.





Message will be stored in the given ,from that queue you can store it in database.

Create a table in database.

CREATE TABLE "SYSTEM"."MONITORING"

( "TRANSACTIONID" VARCHAR2(20 BYTE),

"EVENT\_NAME" VARCHAR2(40 BYTE),

"BROKER\_NAME" VARCHAR2(50 BYTE),

"EXECUTION\_GROUP" VARCHAR2(30 BYTE),

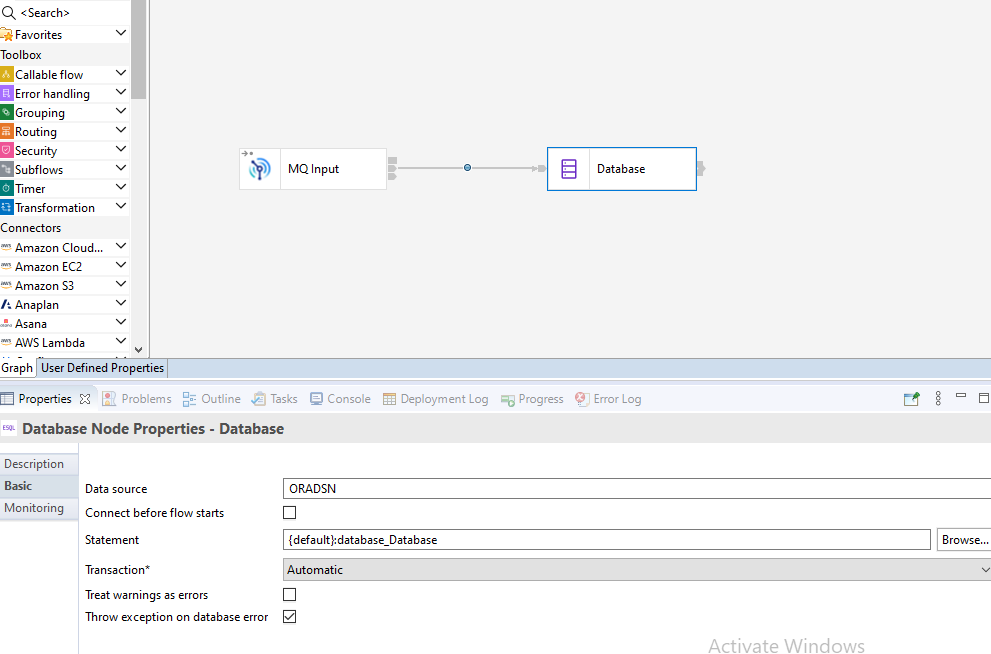
"MESSAGE\_FLOW" VARCHAR2(40 BYTE),

"PAYLOAD" BLOB,

"EVENT\_DATE" TIMESTAMP (6)

)

Create a messageflow to insert the data into the database.



For MQInput node give the same queue name as we used to store the event.

When you browse the queue the message will look like this.

<mon:event xmlns:mon="http://www.ibm.com/xmlns/monitoring/event/v2">

<mon:eventPointData>

<mon:eventData mon:productVersion="12080" mon:profileVersion="11" mon:eventSourceAddress="HTTP Input.transaction.Start">

<mon:eventIdentity mon:eventName="TS"/>

<mon:eventSequence mon:creationTime="2024-03-15T10:37:08.640341Z" mon:counter="1"/>

<mon:eventCorrelation mon:localTransactionId="tran\_76987" mon:parentTransactionId="" mon:globalTransactionId=""/>

</mon:eventData>

<mon:messageFlowData>

<mon:integrationNode mon:name="IIB" mon:hostName="DESKTOP-BL2IGCK"/>

<mon:integrationServer mon:name="server" mon:hostName="DESKTOP-BL2IGCK"/>

<mon:application mon:name="Event\_Monitoring"/>

<mon:messageFlow mon:uniqueFlowName="IIB.server.Event\_Monitoring.Events" mon:name="Events" mon:threadId="35676"/>

<mon:node mon:nodeLabel="HTTP Input" mon:nodeType="ComIbmWSInputNode"/>

</mon:messageFlowData>

</mon:eventPointData>

<mon:applicationData xmlns="">

<mon:complexContent mon:elementName="Data">

<Data>

<Employee>

<Emp\_Name>Heena</Emp\_Name>

<Emp\_id>827</Emp\_id>

<Location>Nzb</Location>

<Salary>76466</Salary>

</Employee>

</Data>

</mon:complexContent>

</mon:applicationData>

<mon:bitstreamData>

<mon:

mon:serializationEncoding="hexBinary" mon:dataCCSID="1208" mon:dataEncoding="546"></mon:bitstream>

</mon:bitstreamData>

</mon:event>

In database node write the code to insert the data into the table.

