Task 3

1. Steps to follow and deploy on azure api management to connect to an on-premises get and post api’s.

First, we need to create api management instance in azure portal, then we need to create vnet and configuration for the connectivity with the on-premises network.

Then we need to deploy the self hosted gateway in the gateways tab.

Later on we need to add api and define get and post endpoints and methods, then setup the url as per the self hosted gateway and test the connection.

We need to provided authentication and need to configure policies for the request and response to manage the api traffic.

1. For implementing ETL using azure data factory to look up the data from another tables, initially we need to setup the azure data factory then we need to use the connector with Microsoft dataverse for the data, where as the data is stored in the sql.

Then create the datasets, then extract the data from the sql, then we need to transform the data by looking up the details, then data flow will occurs and load the data with the help of copy activity.

To implement ETL to lookup for an guid from another dynamics 365, we need to extract the data from the sql using copy activity, then lookup for the guid, then transform the data, and the load the data back to the sql.

3.

To create the azure data pipeline for creating and updating the existing records from the core system to dynamics 365, we need to follow ETL process only, first we need to extract the records from the sql using copy activity, then lookup to check if the record already exists or not, then transform the data and load the data back to the dynamics 365.

And the component we have used is relationship concept, conditions component we used to decide for inserting or updating the data and lookup we used to look for the specific record or the data from the particular customer and the copy activity we used to extract the data from the base system.

4.

Creating azure data pipeline to migrate leads and tasks from on-premises dynamics 365, we need to setup the azure data factory environment first, then create the linked service and datasets which are leads dataset and tasks dataset, for leads it would be leadid, fname, lname, email, contact, createdon, createdby, modifiedby.

Then create the pipeline by extracting the data from the base system using copy activity, and lookup activity to retrieve related contacts, then transform the data if necessary and load the data and share the leads, we can use the share api to share the leads by adding http session to share the record to the teams which maintains the relationships with contacts and teams.

Also we can use the triggers to schedule the pipeline