**Lab Book**

**TIBCO BusinessWorks 6**

Contents

[1 Software Perquisites 3](#_Toc462226919)

[1.1 TIBCO BW Installation 3](#_Toc462226920)

[1.2 TIBCO EMS Installation 3](#_Toc462226921)

[1.3 GEMS Installation 3](#_Toc462226922)

[1.4 MySQL Installation 3](#_Toc462226923)

[1.5 SoapUI Installation 3](#_Toc462226924)

[1.6 Notepad++ 3](#_Toc462226925)

[2 Lab Exercises 3](#_Toc462226926)

[2.1 Lab Exercise-1 3](#_Toc462226927)

[2.1.1 Problem Definition 3](#_Toc462226928)

[2.1.2 Solution Overview 4](#_Toc462226929)

[2.1.3 Required Artifacts 4](#_Toc462226930)

[2.1.4 Development Steps 4](#_Toc462226931)

[2.1.5 Debug/Testing 17](#_Toc462226932)

[3 Homework 22](#_Toc462226933)

[3.1 Homework-1 22](#_Toc462226934)

# Software Perquisites

## TIBCO BW Installation

Download and install TIBCO ActiveMatrix BusinessWorks 6.X from <https://tap.tibco.com/>

Refer Installation Guide available at <https://docs.tibco.com/products/a_z_products> for any help on installation.

## TIBCO EMS Installation

Download and install TIBCO Enterprise Message Service.

Refer Installation Guide available at <https://docs.tibco.com/products/a_z_products> for any help on installation.

## GEMS Installation

Download and install GEMS from <https://community.tibco.com/modules/graphical-administration-tool-tibcor-ems>. Configure GEMS to connect to default instance of EMS (7222).

## MySQL Installation

Download and install MySQL server from <http://dev.mysql.com/downloads/>.

Download and install MySQL sample database *employee* from <https://dev.mysql.com/doc/employee/en/>.

Configure MySQL Workbench to connect to *employee* database.

## SoapUI Installation

Download and install Open Source version of SoapUI from <https://www.soapui.org/>.

## Notepad++

Download and install Notepad++ from <https://notepad-plus-plus.org>.

# Lab Exercises

## Lab Exercise-1

### Problem Definition

Capgemini has multiple offices across the globe. Capgemini uses a centralized application – Employee Data Management System (EDMS) –for managing employee data. Whenever a new employee joins at any office, a message on a particular queue gets published.

The solution should read the message from the queue create record in the database.

### Solution Overview

Topics Covered: Project Creation, JMS, JDBC, and Logger.

### Required Artifacts

Queue Name: Capgemini.TIBCOTraining.Exercise1.Queue

XML Schema: EmployeeData.xsd

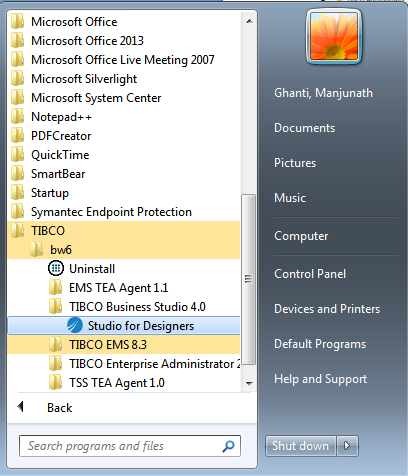


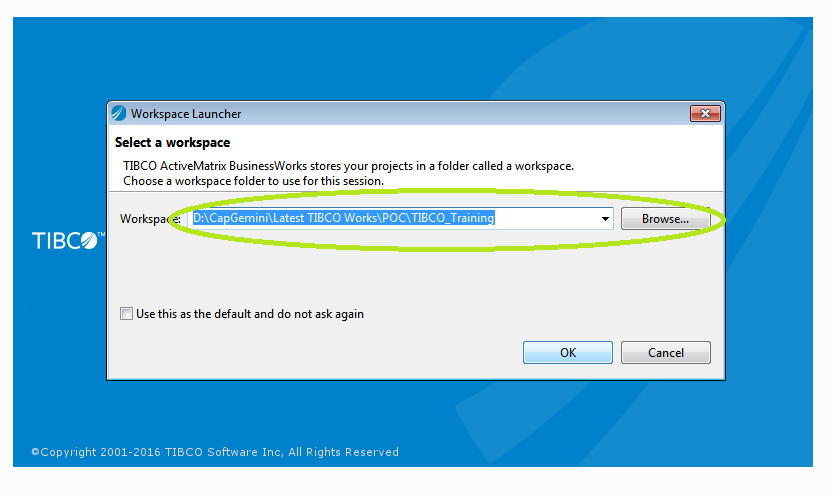
Database Table:

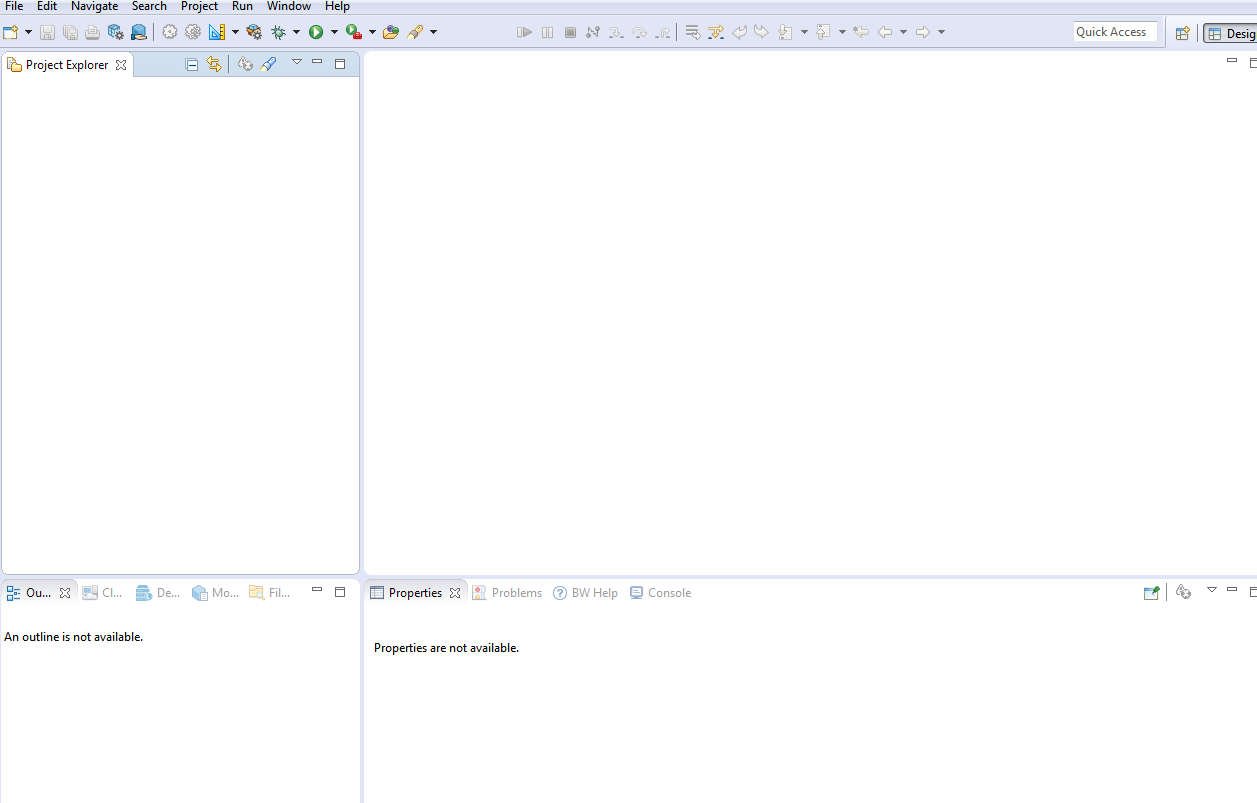


### Development Steps

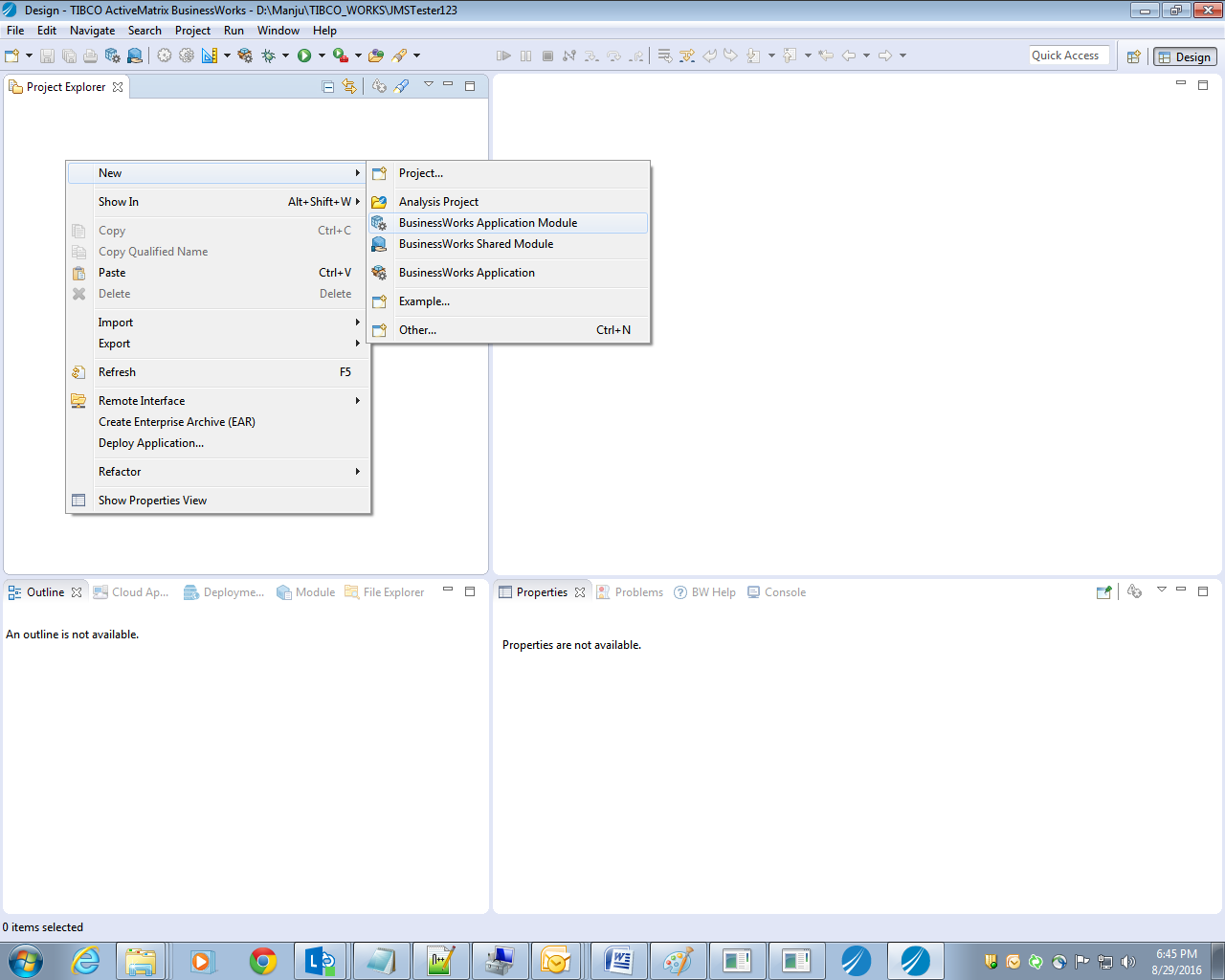
* Open Business Studio by clicking on Start 🡪 All Programs 🡪 TIBCO 🡪bw6 🡪TIBCO Business Studio 4.0 🡪 Studio for Designers

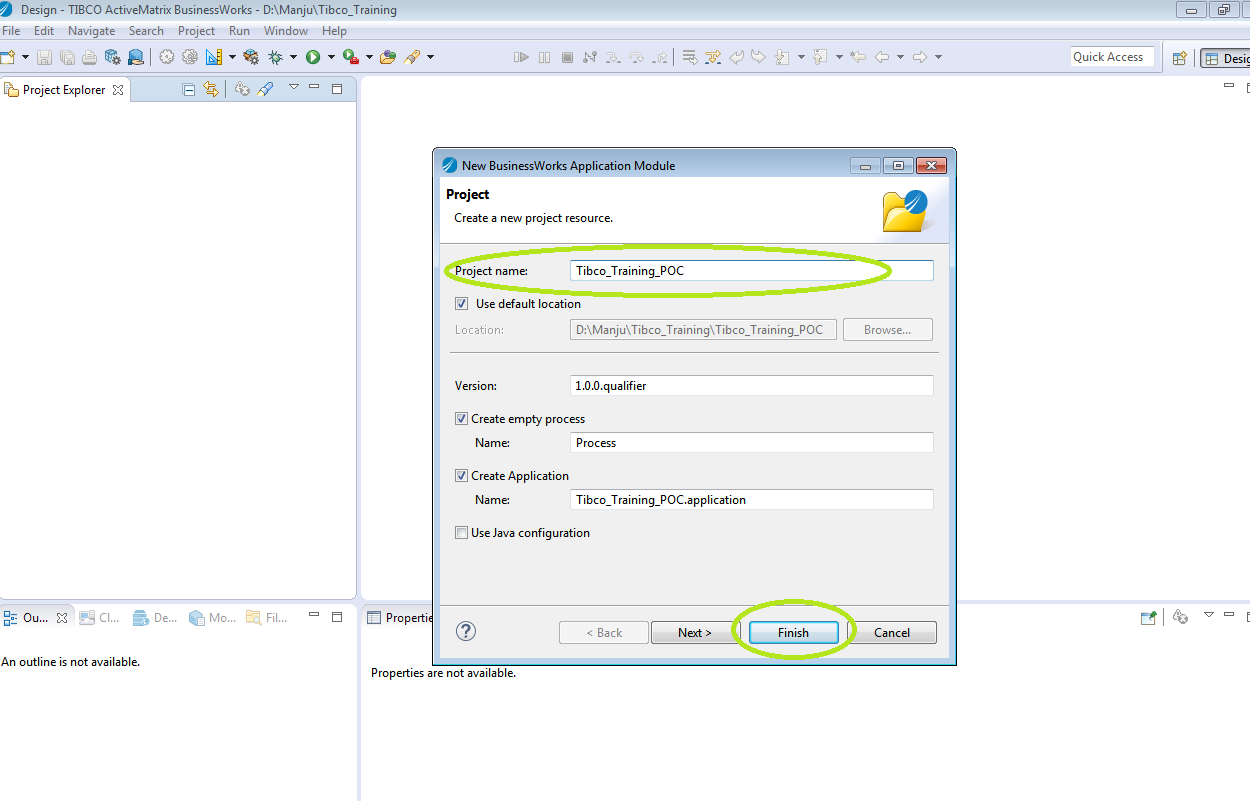


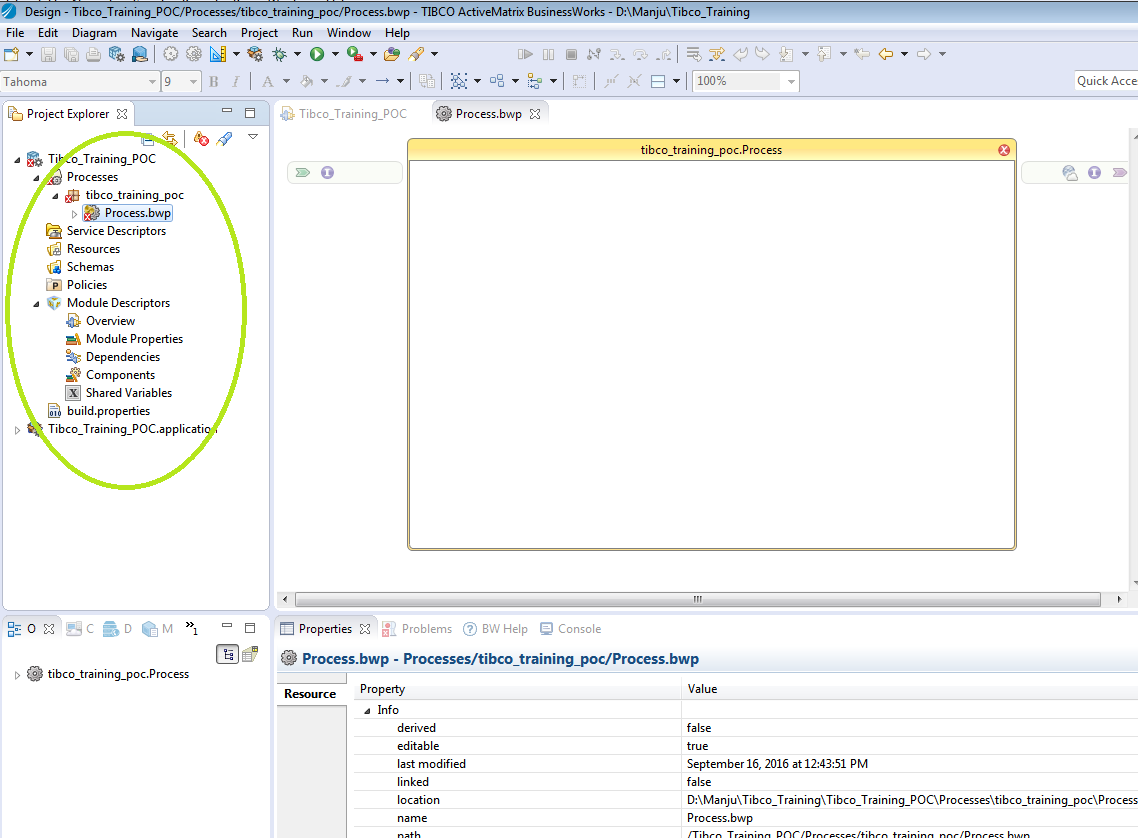
* After launching the designer refer the below screenshot which will ask us for the project workspace name and the directory where we want to create the workspace. Click on ok and proceed further. Refer the below screenshot.
* Workspace is displayed as below:



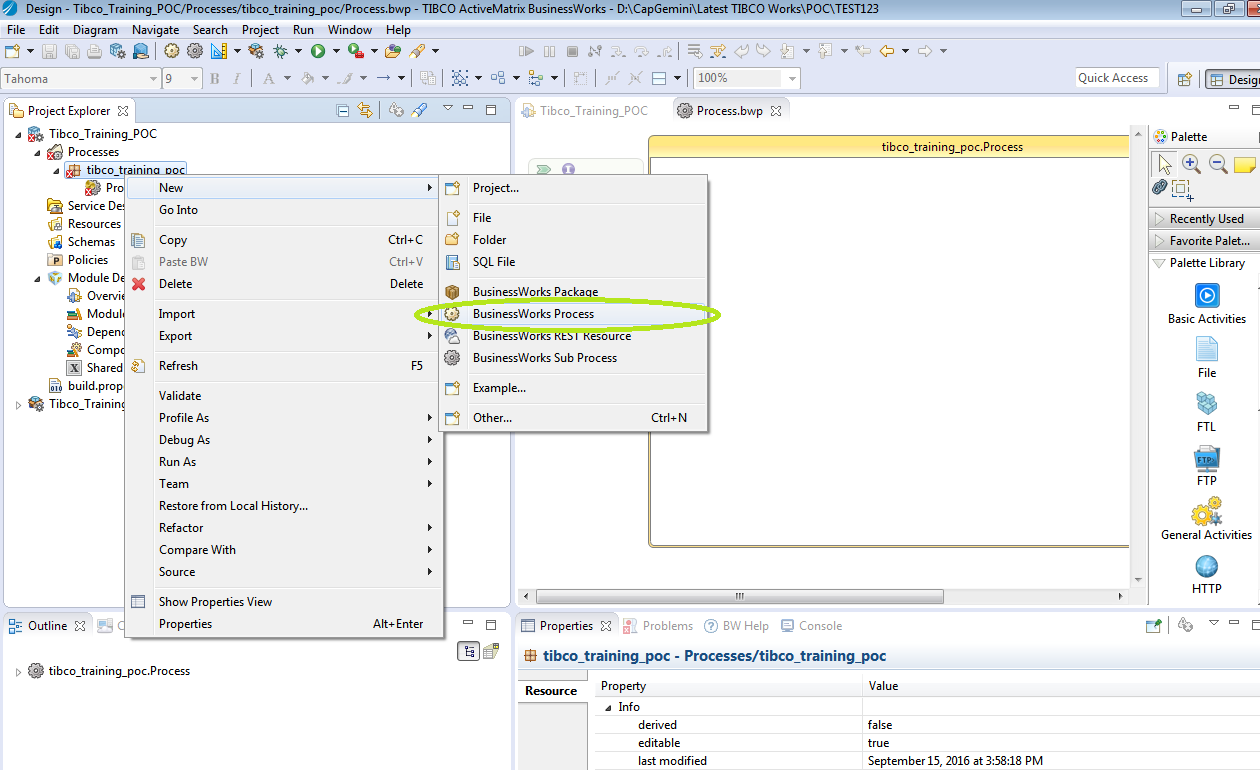
* In TIBCO Business Works Studio, click on File 🡪 New 🡪 Business Works Application Module to create a new business works module

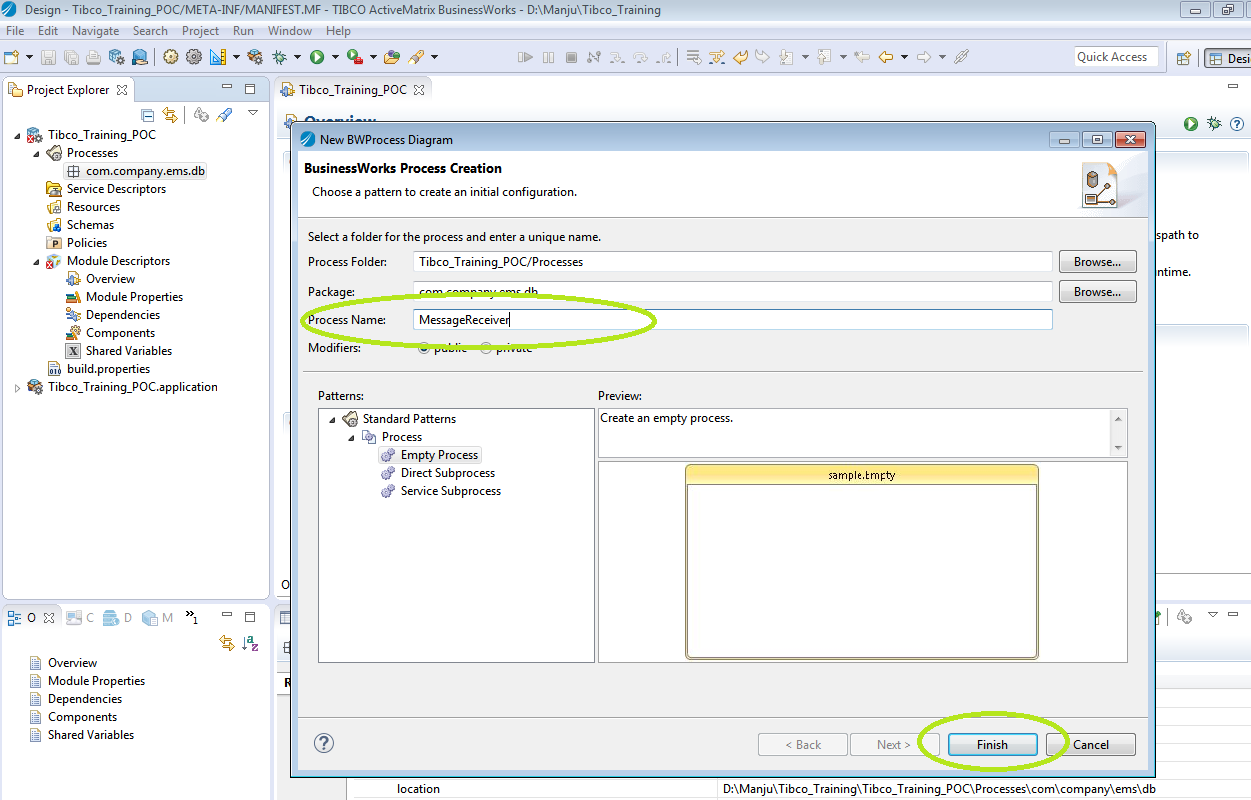


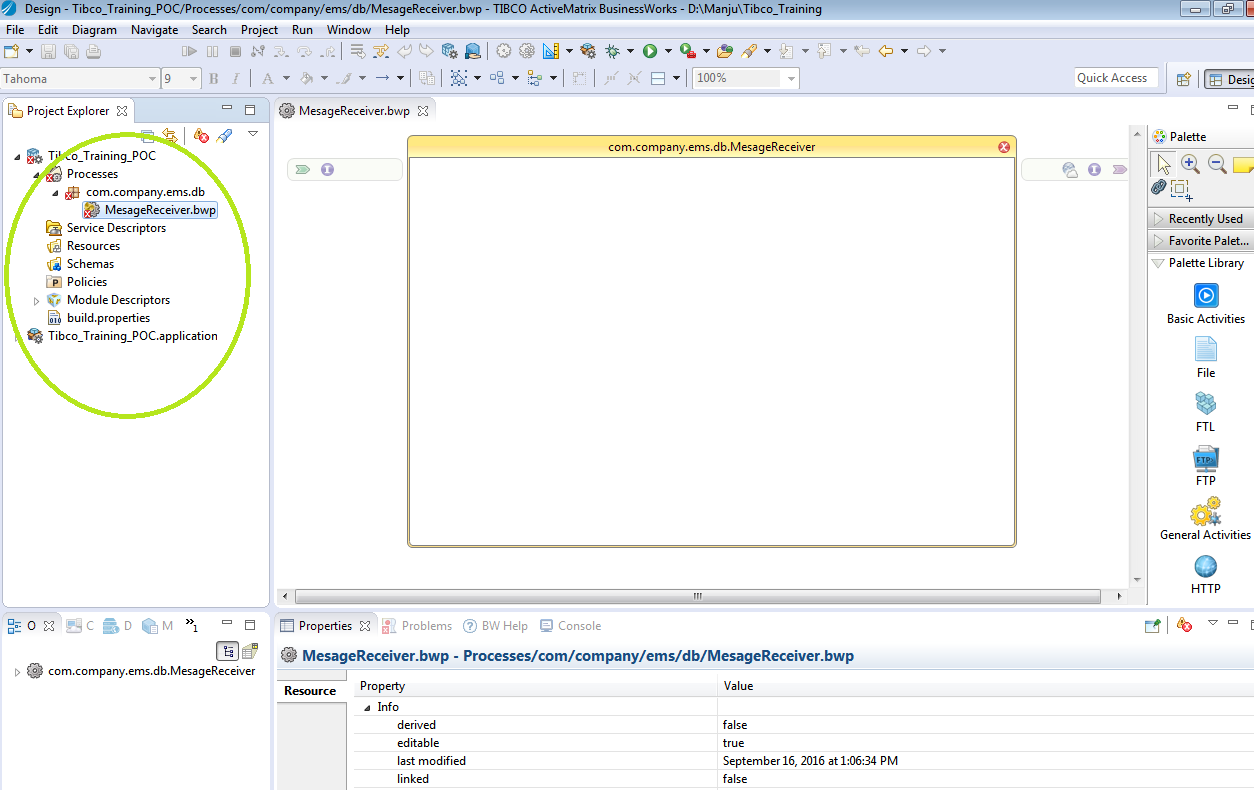
* Give the appropriate name for BusinessWorks module and click on finish as shown:
* Module and application will be as below with the entire structure created:

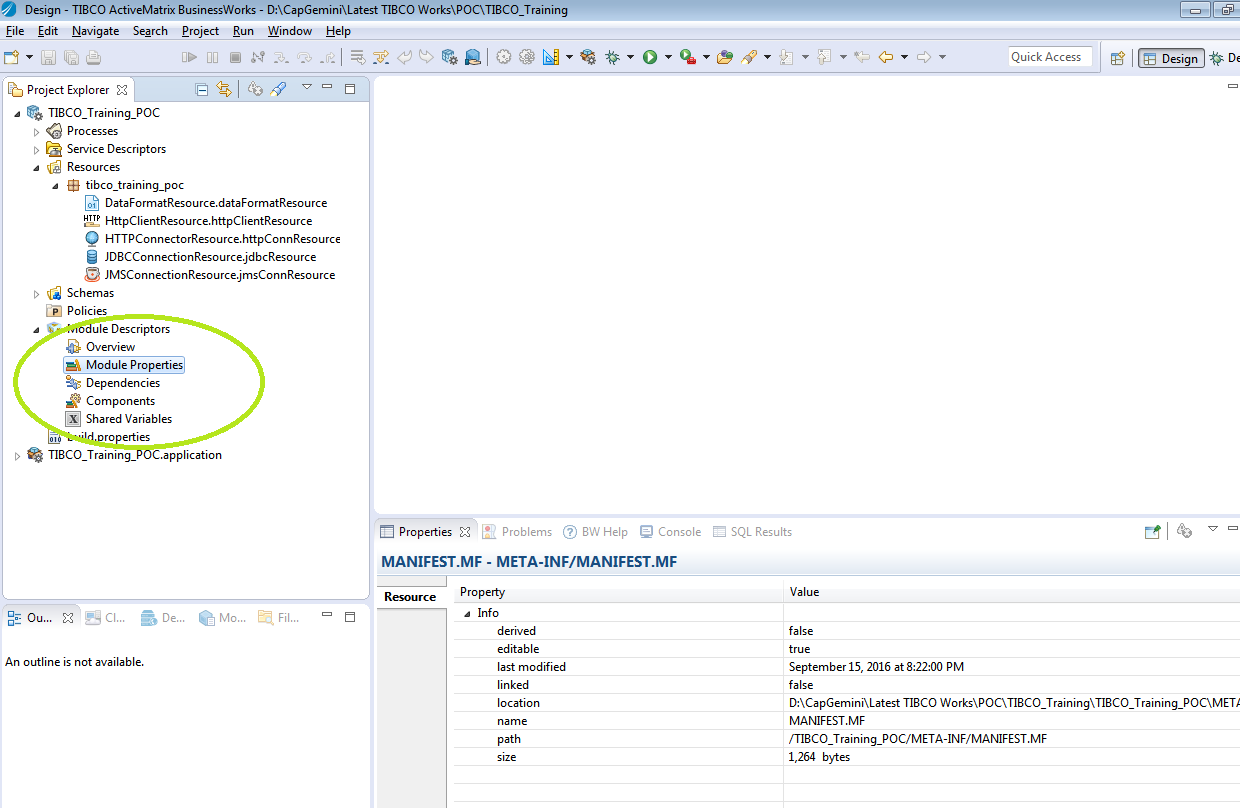
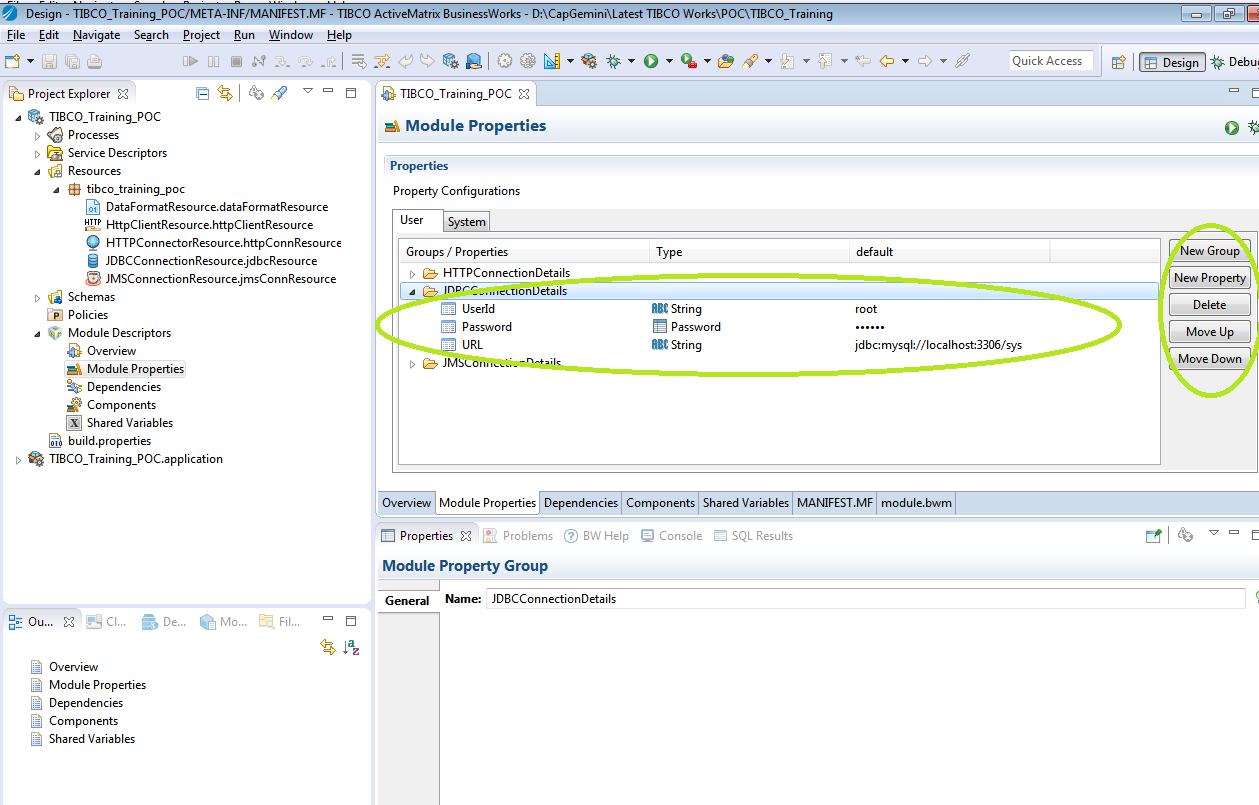
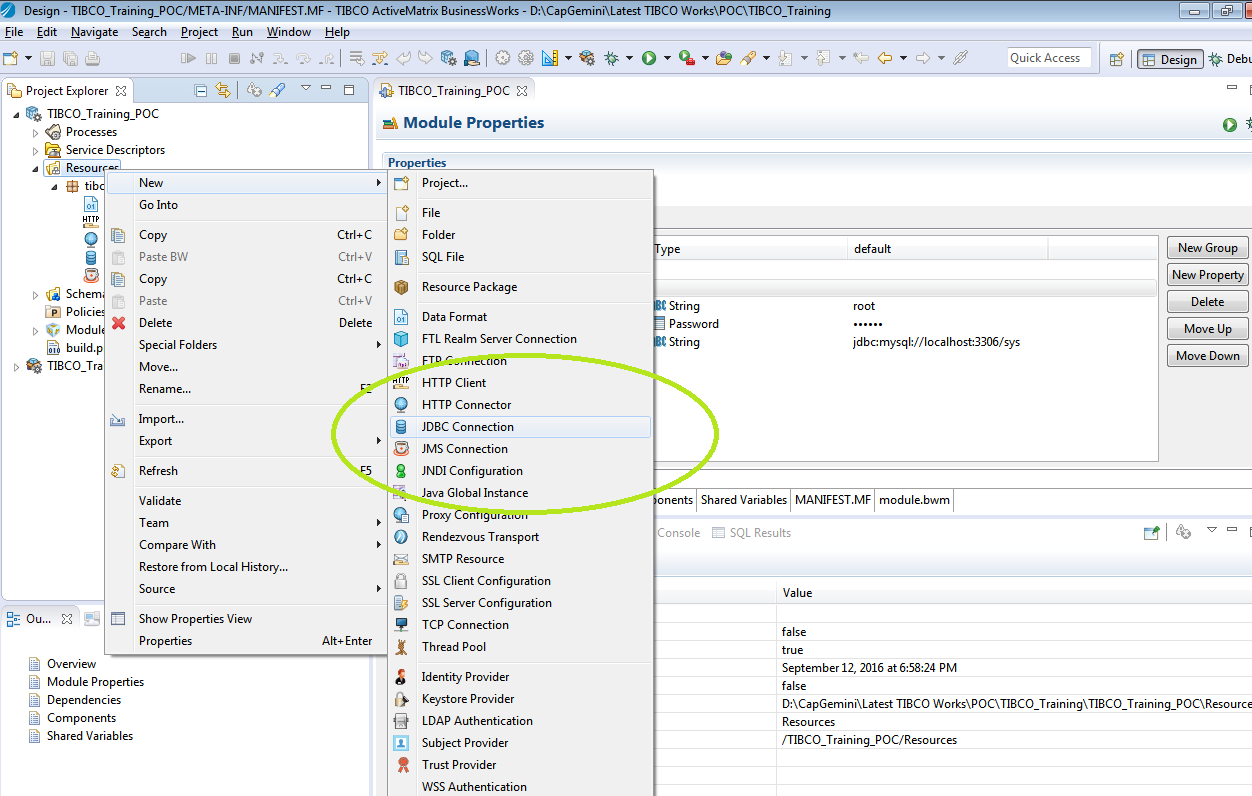


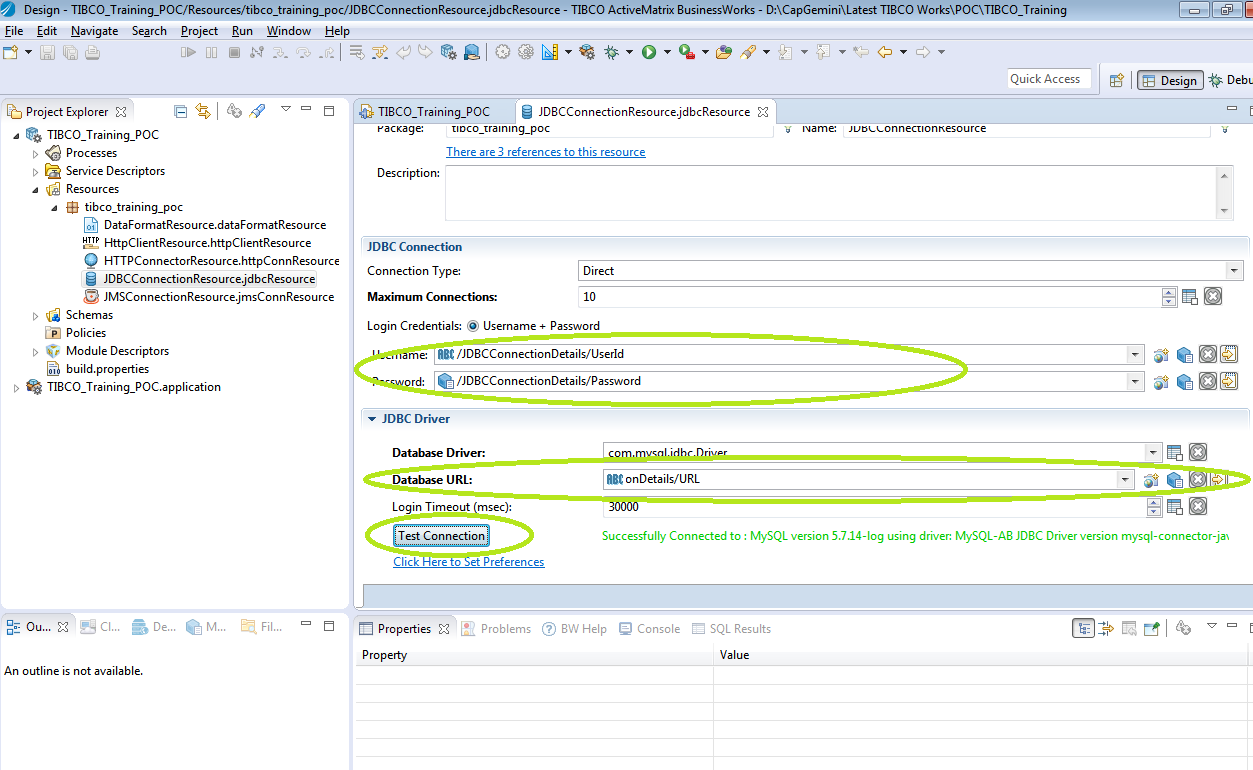
* Additional process if required can be created as below: Right Click on the process package🡪New🡪BusinessWorks Process:



* Provide and appropriate process name and click on Finish
* The project explorer will be as below with new process created:



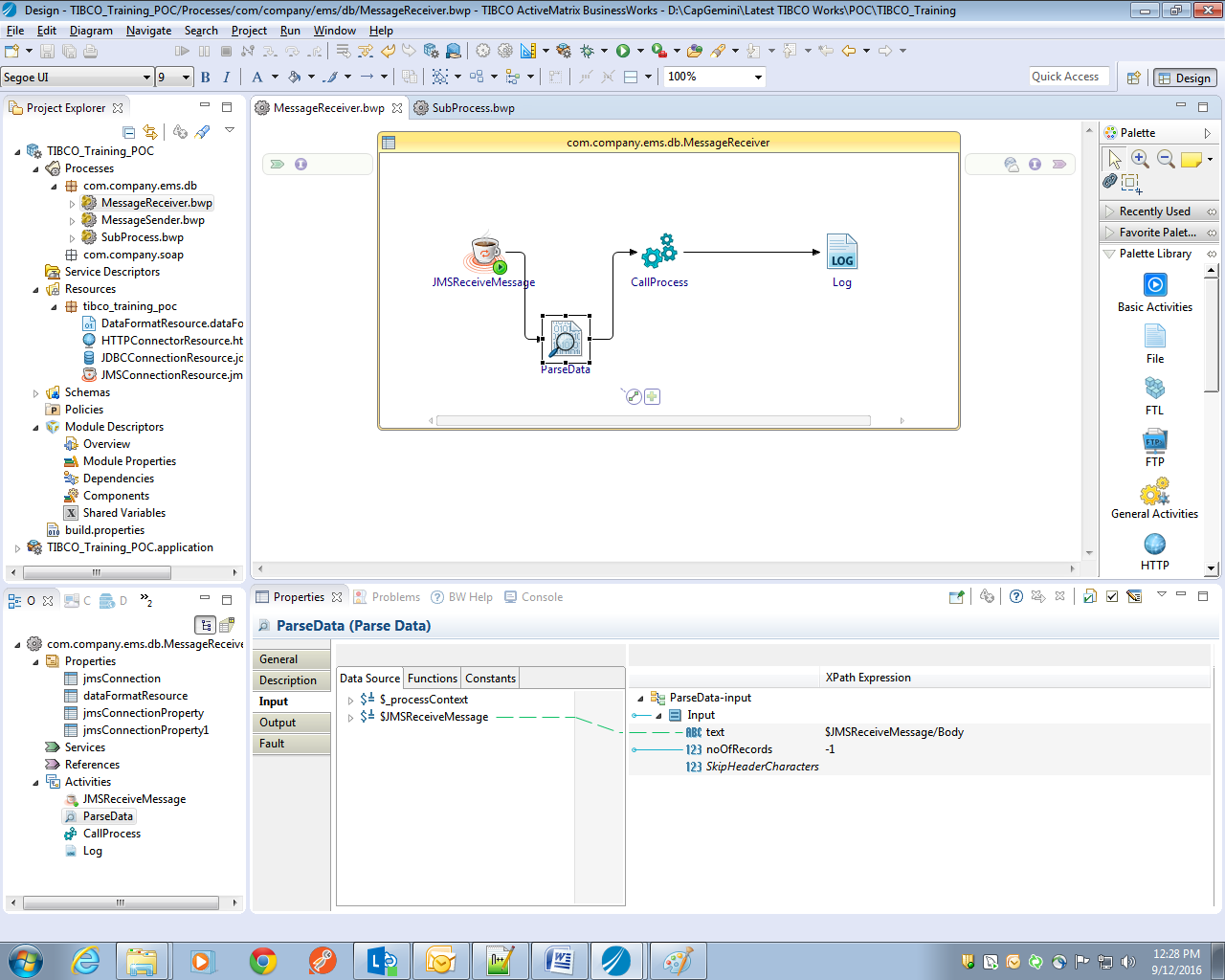
* We will add module properties to use in shared connections: JMS and JDBC.
* Module Descriptors🡪 Module Properties as shown below:
* After that provide all the necessary details to connect to the servers. Please refer for JDBC connection details:
* Please refer below to create a new shared resource
* Configure connection parameters from module properties defined and test the connection.
* Please refer the below screenshots;

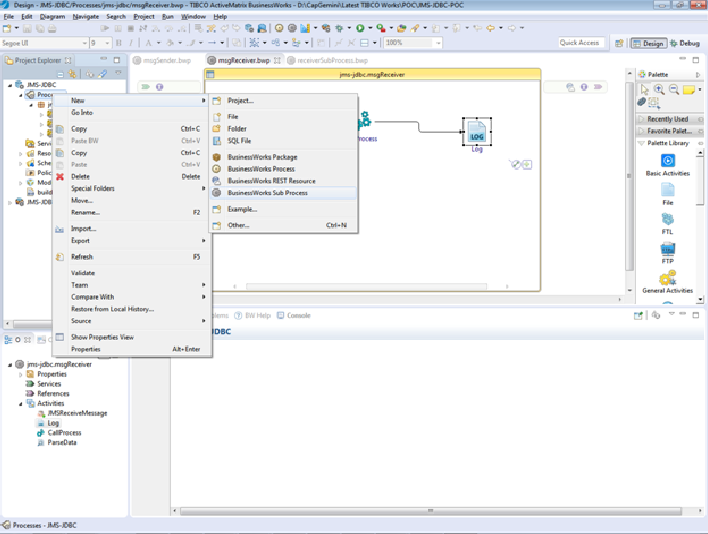


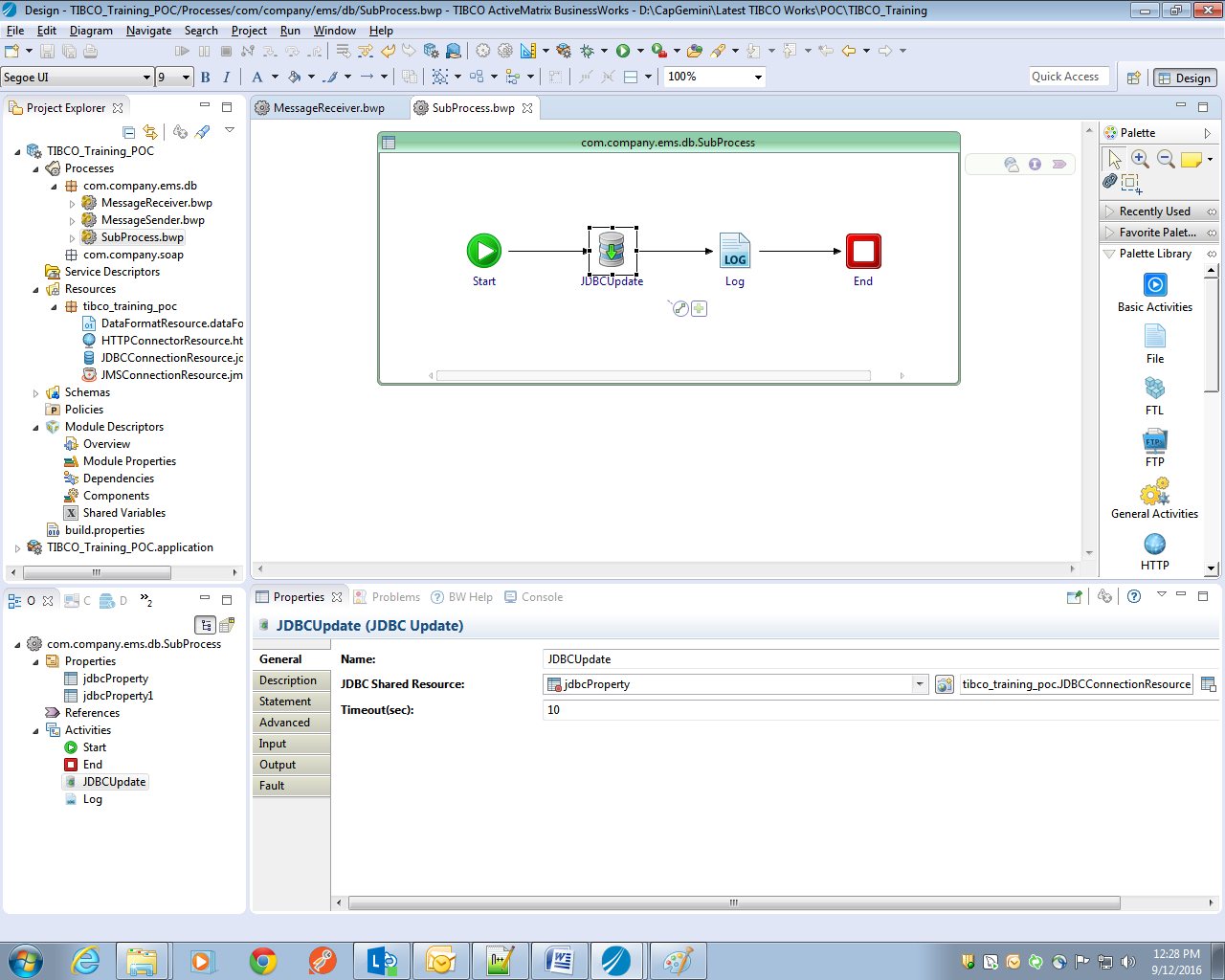
* Now let’s proceed further to create a sender process which will read a file and then send it to the queue which we created earlier (queue: Capgemini.TIBCOTraining.Exercise1.Queue).
* MessageReceiver process will get the message from the above mentioned queue, parse it and send it to a sub process which will update the record to database.

We will add the required palettes.

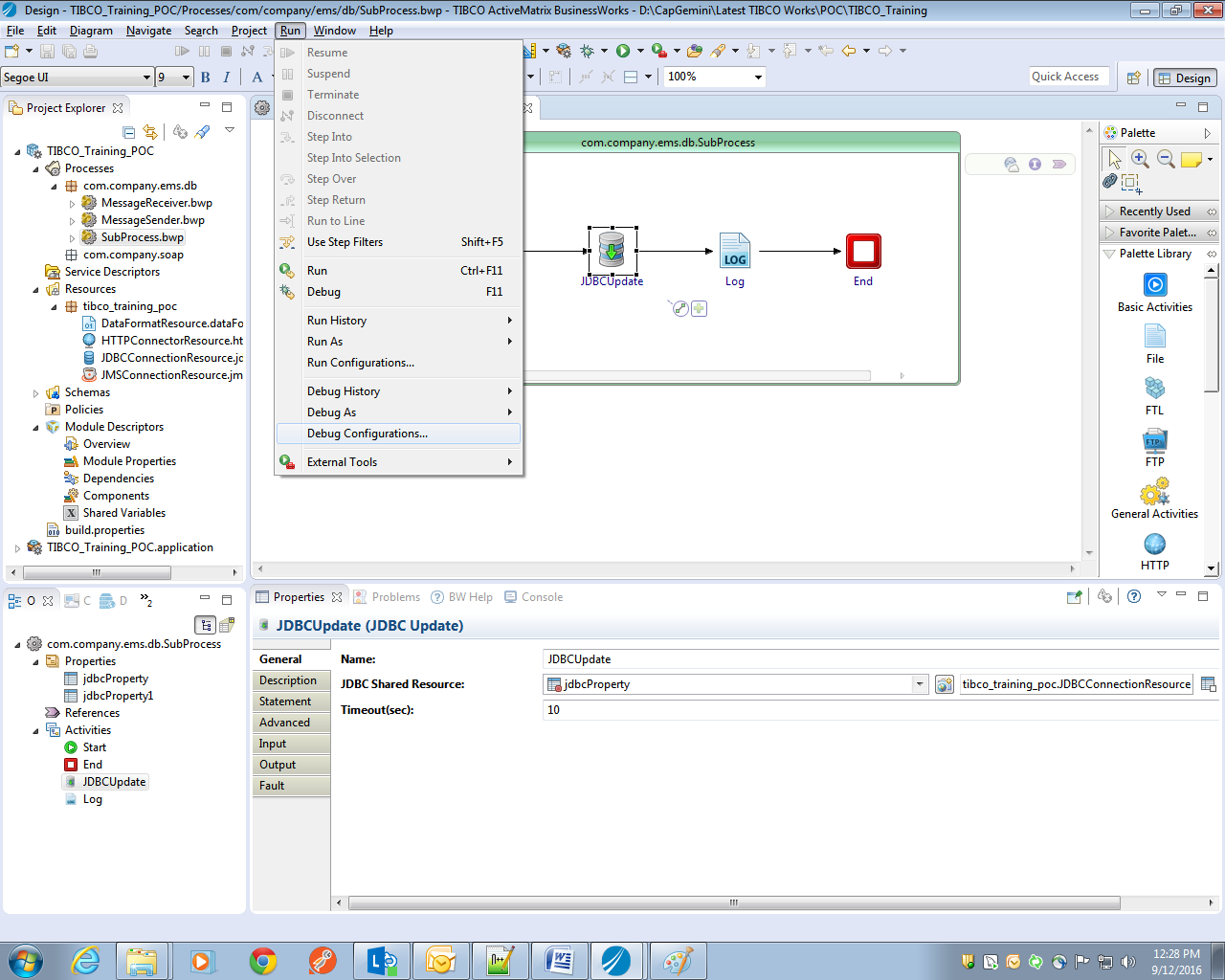
* Sub process will insert data into database.
* The below screenshot depicts the above scenario;

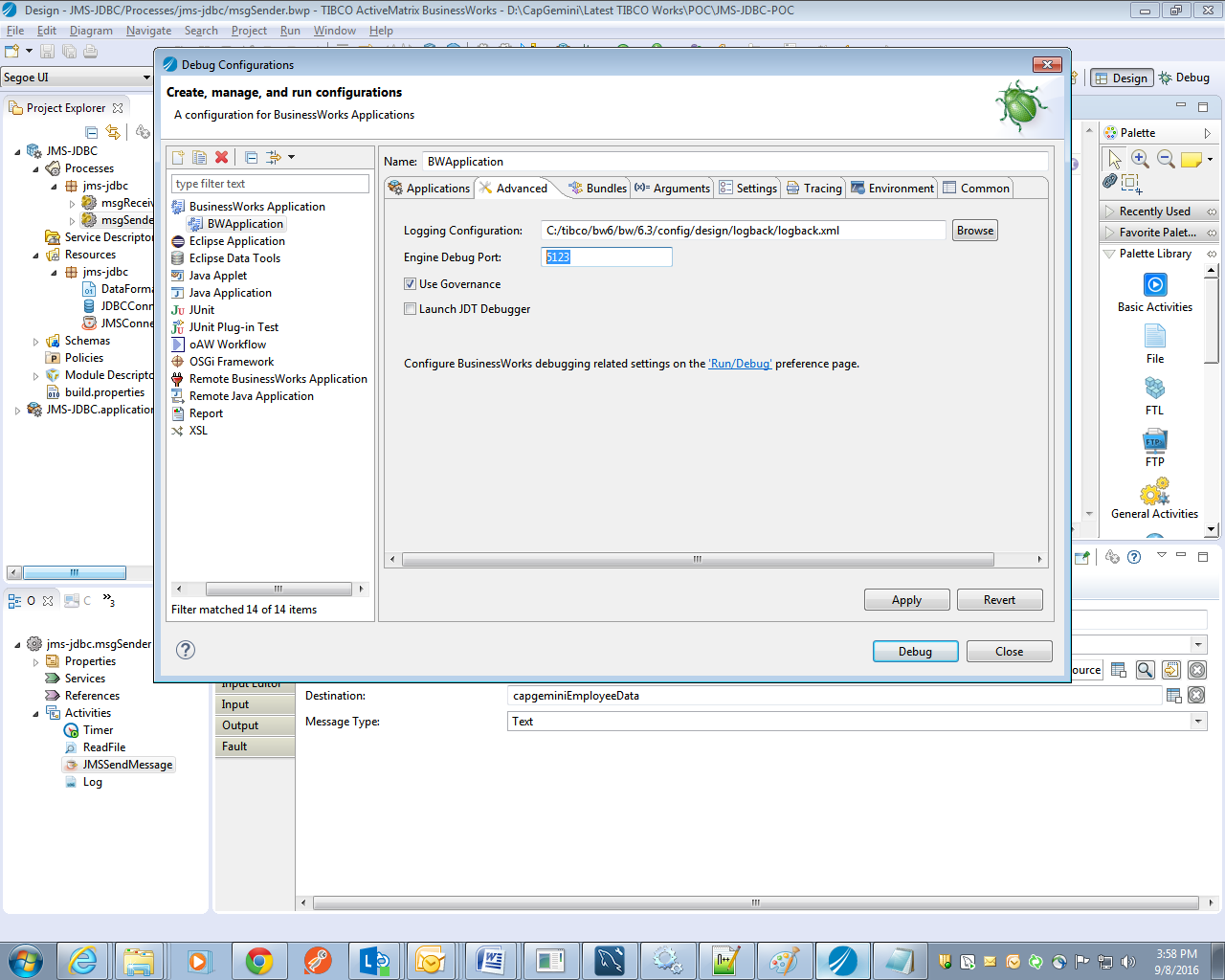


* For creating the sub-process right click on the Process and select for Business works sub-process as shown below;
* Map all the required connection details and fields.

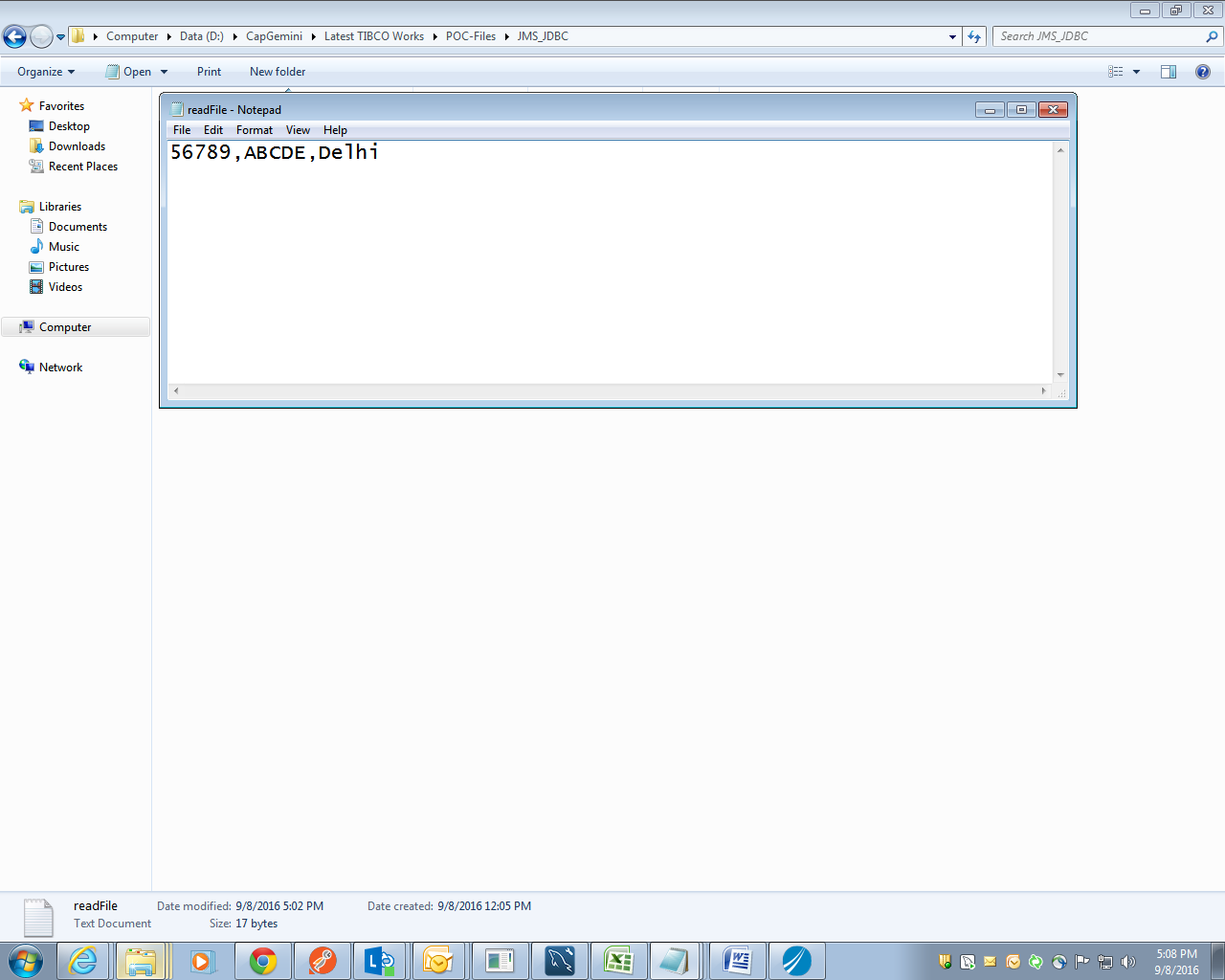


### Debug/Testing

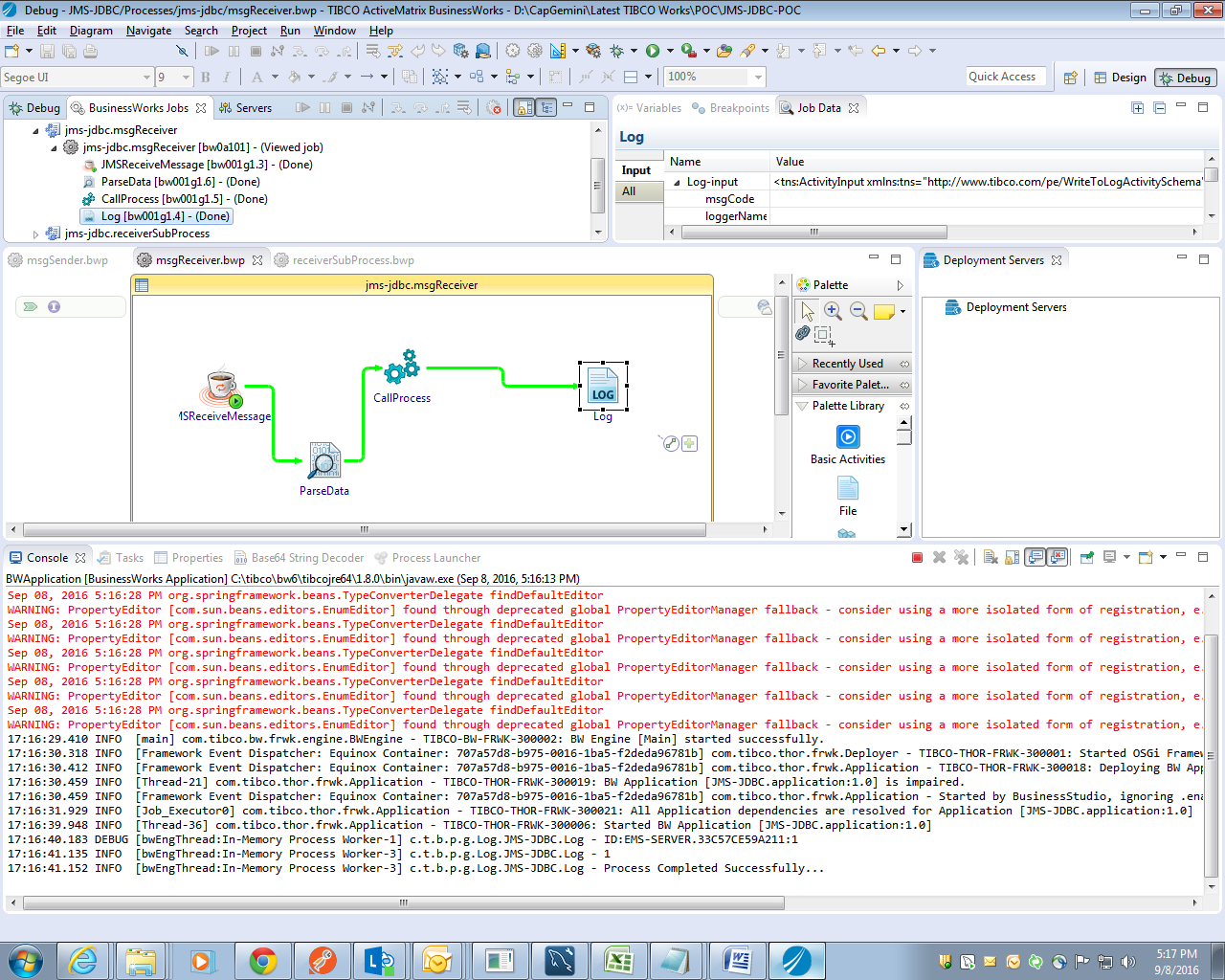
* For debugging the application, Click on Run which is at the top panel of the window and select Debug Configurations:
* Select Application to be debugged.

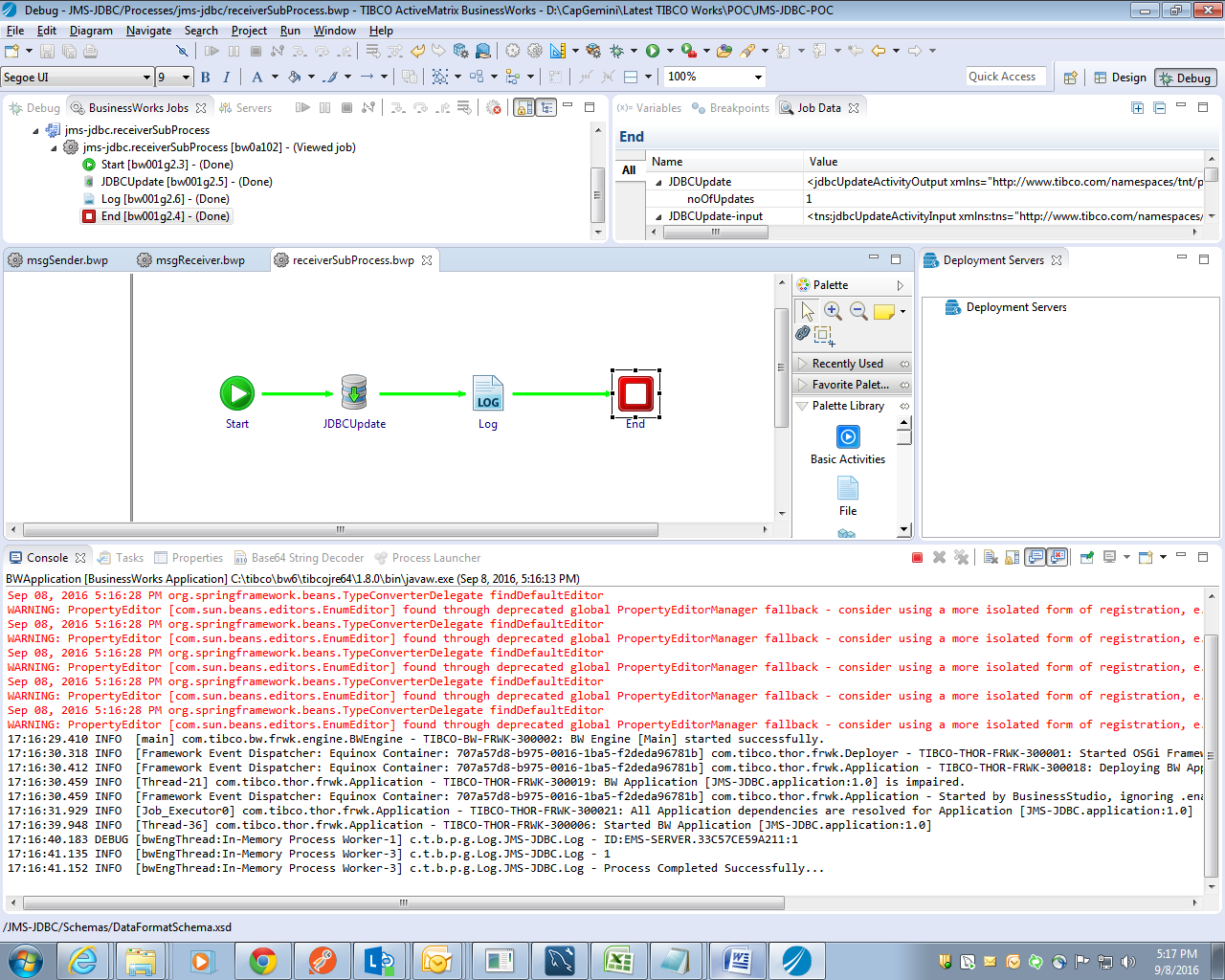


* Once it is done apply the changes and click on debug button.
* The application will get triggered and the logs will be written on the console.
* Here is the sample file which our process will read from and push to queue.

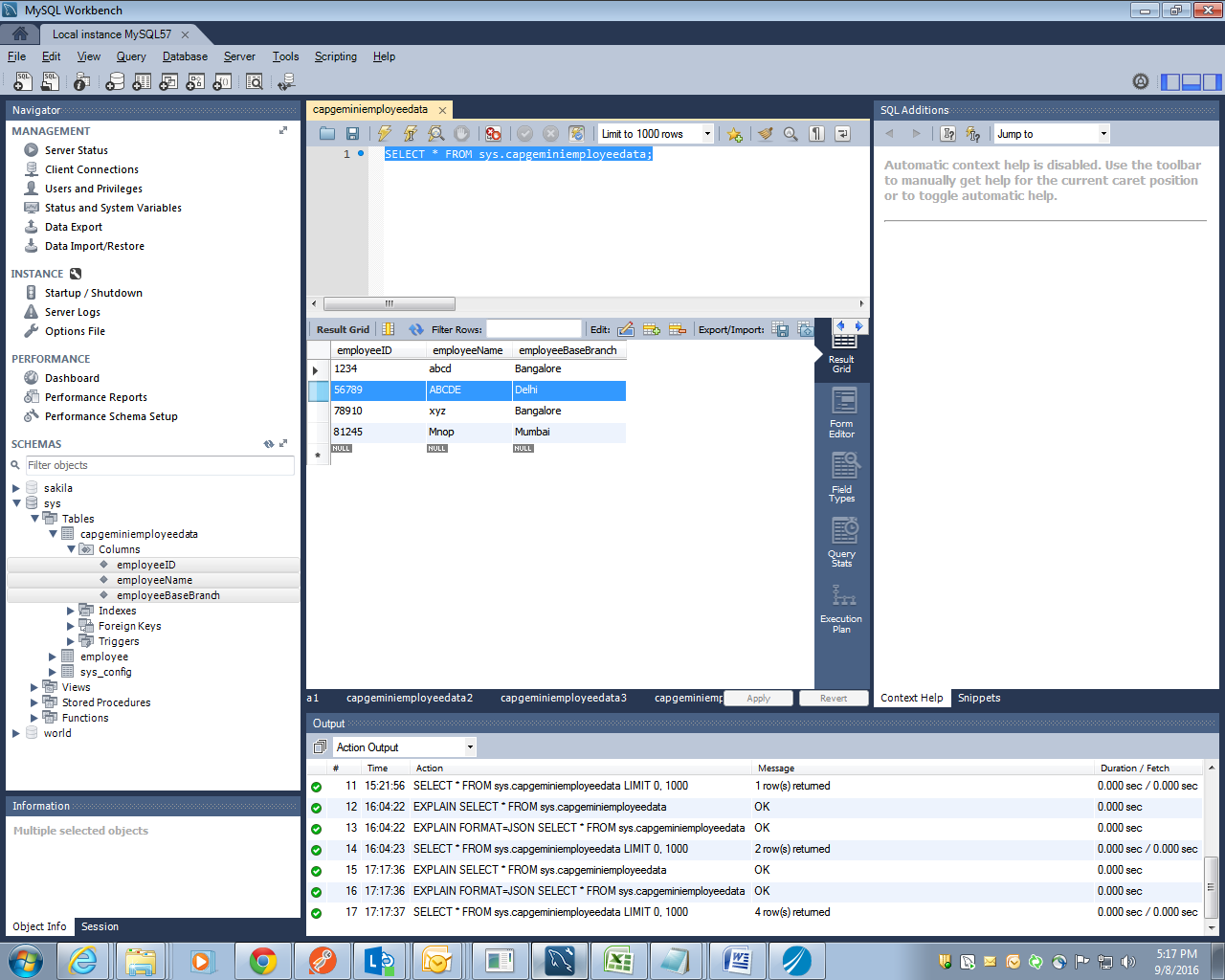


* Once the application is started, the sender process will send the message to the queue.
* The console output of the process along with the database table where record has been inserted successfully, please see below:





* Database rows updated with new values :



# Homework

## Homework-1

Extend Exercise-1 and create solution which receives a message with employee id from a queue Capgemini.TIBCOTraining.HomeWork1.Queue and delete corresponding records from database.

Extend Exercise-1 and create solution which receives a message with employee id from a queue Capgemini.TIBCOTraining.HomeWork2.Queue, search the employee based on the employee id and publish on another queue Capgemini.TIBCOTraining.HomeWork3.Queue.