

**WA1974 Modeling Business
Process Using IBM Process
Designer 8.0**

Student Labs

Web Age Solutions Inc.

EVALUATION ONLY

Table of Contents

Lab 1 - Introduction to Process Designer.....	3
Lab 2 - Basic Process Modeling.....	20
Lab 3 - Model UC001 Price Quote and Availability Process.....	23
Lab 4 - Working With Exclusive Gateway.....	27
Lab 5 - Gathering Requirements.....	38
Lab 6 - Design the Business Objects.....	42
Lab 7 - Create the Expense Claim Business Process.....	46
Lab 8 - Integrate With External Applications.....	52
Lab 9 - Integration Using Java Classes.....	62
Lab 10 - Kicking Off Phase II.....	69
Lab 11 - Implement User Task Using Coaches.....	73
Lab 12 - Expose a Process as Web Service.....	84
Lab 13 - Intermediate Message Event Handling.....	95
Lab 14 - Task Assignment.....	105
Lab 15 - Working with Timer Events.....	112
Lab 16 - Optimizing a Process Using Simulation.....	121

EVALUATION ONLY

Lab 1 - Introduction to Process Designer

IBM Process Designer (PD) is a desktop tool that allows users to model and define business processes. Modeling captures the structure of the process including the key activities and their sequence. Defining a process takes a model all the way to make it actually executable. The task of modeling can be done by business analysts and line of business managers. Definition of a process takes more technical knowledge. However, we expect that the business analysts, with some training, will be able to largely complete process definition. For more complex situations, they will need help from the developers.

In this lab, we will learn the basic steps in process modeling and definition. The goal is to get familiar with the tool and not to learn about modeling and definition in details.

The process will be very simple. It will print a simple greeting message in a file, such as "Good morning, Bill". The process will take as input:

1. The time of day message, such as, "Good morning" or "Good evening".
2. The name of the person being greeted.

The objective of the process is to format this input in a nice manner and store the greeting in a file.

After completing this lab, you will know how to:

1. Install PD.
2. Launch and log into PD.
3. Create a process application.
4. Create a new process definition.
5. Model a very simple process.
6. Complete the definition of the process.
7. Work with simple input data.
8. Execute a process and view result.

Part 1 - Start Process Center Server

PD is a very thin client. Most of the work actually happens within the Process Center server. PD acts as a client of this server. Every time you make a change to your project, changes are saved directly into the Process Center server. In other words, very little meaningful work actually goes on in the client machine.

Before we actually start this lab, we will need to start the Process Center server. In real life, this server will be maintained by a group of administrators. As a business analyst, you can assume that this server is running somewhere.

For the purpose of this lab, though, you will start the server.

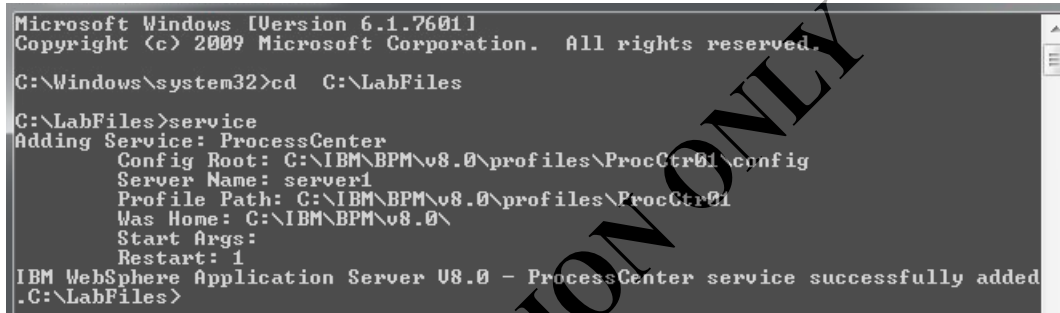
Note. If you are running on Windows 7 or Vista you need to run all windows as administrator and accept changes when appears. You can also change the settings in the user to Never notify for changes and turn off Windows Firewall.

- __1. Log into Windows using the user ID **wasadmin** and password **wasadmin**.
- __2. Open a command prompt window. In Windows 7 or Vista, run it as Administrator.
- __3. Change directory to: **C:\LabFiles**.

```
cd C:\LabFiles
```

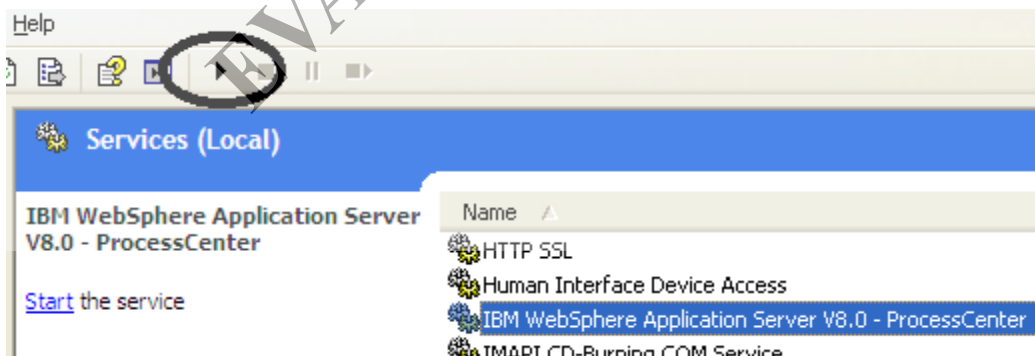
- __4. Register the server as a Windows service by running this command:

```
service
```

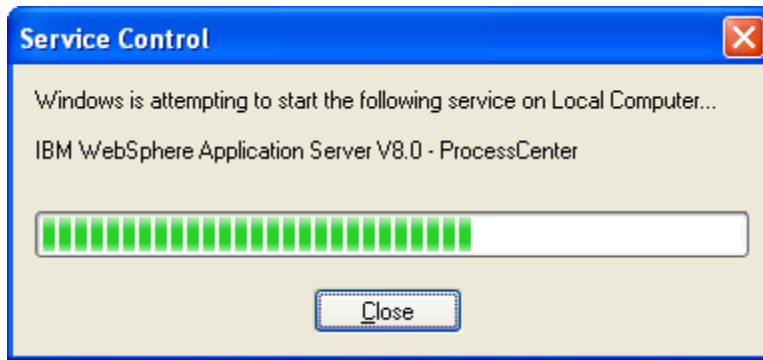


We will now start the service.

- __5. Open Windows control panel.
- __6. Double click **Administrative Tools**.
- __7. Double click **Services**.



- __8. Select **IBM WebSphere Application Server V8.0 – ProcessCenter**.
- __9. Start the service.



__10. Wait for the start dialog to go away.

WebSphere takes a few seconds to initialize. Even though Windows shows the service as started, WebSphere may be still starting. The best way to tell if it has finished initialization is to inspect the server's log file. A tool called PigTail has been given to you to monitor the log file. We will use it now.

__11. Run **C:\LabFiles\pigtail.exe**.

__12. Select **File > Open** from the menubar.

__13. Navigate to the **C:\IBM\BPM\v8.0\profiles\ProcCtr01\logs\server1** directory.

__14. Open **SystemOut.log**.

```
[6/6/12 22:13:50:671 EDT] 00000000 ActivationSpecImpl JZLH4U2Q The message endpoint for ActivationSpecImplEventImpl
(com.ibm.ws.sib.api.jmsra.impl.JmsJcaActivationSpecImpl) and MDB Application
IBM_BPM_Teamworks_onlineorderNode01_server1#twcore-eib.jar#EventManagerMessageListener is activated.
[6/6/12 22:13:50:671 EDT] 00000000 WsServerImpl A WSVR0001: Server server1 open for e-business
[6/6/12 22:13:50:671 EDT] 00000045 InstanceManag W The dependency javax.servlet.ServletContext is not supported!
[6/6/12 22:13:50:671 EDT] 00000044 InstanceManag W The dependency javax.servlet.ServletContext is not supported!
```

__15. Watch for the log entries to scroll by. If there are messages coming up means that the server is still starting. Finally, you will see the **Server open for e-business** message. This message is logged at the end of server initialization. If you have trouble finding the message, search for "e-business" as shown below.



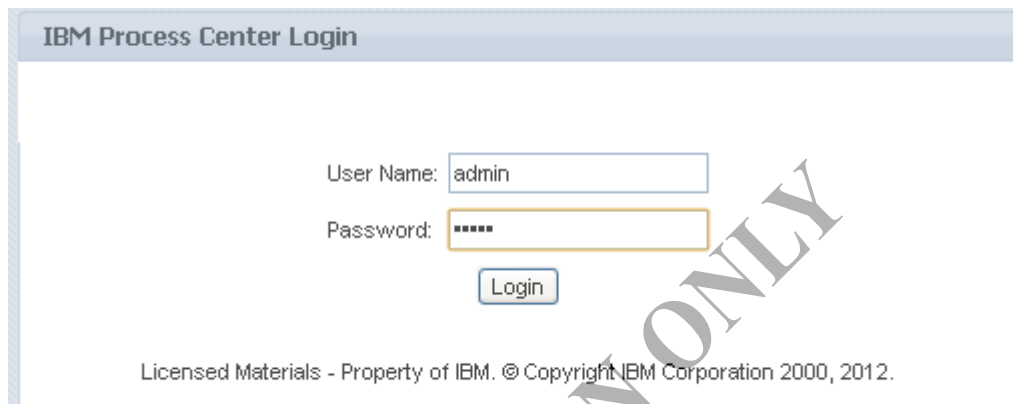
Part 2 - Installing PD

In your lab machine, to save time, PD is already installed. In real life, you can log into the Process Center console and then download and install PD. We will quickly go through this step.

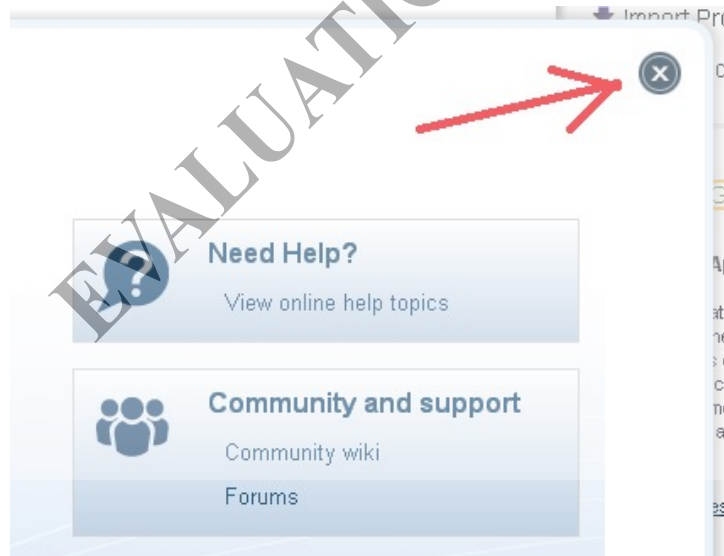
__1. Open a web browser and enter the URL:

`http://localhost:9080/ProcessCenter/`

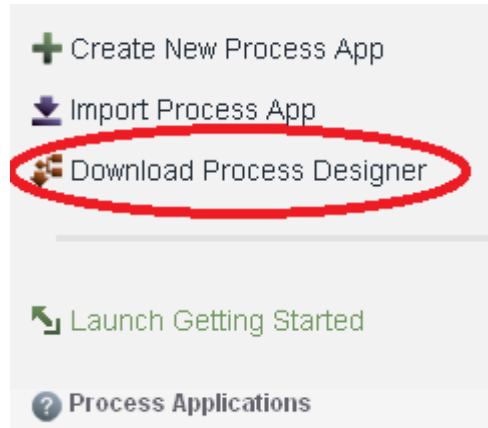
In real life, the host name will be different.



__2. Log in using user id **admin** and password **admin**.



__3. Close the getting started dialog.



__ 4. On the right hand side, notice the **Download Process Designer** link. This is the link you will need to use to download and install the tool. We are not going to use that link for this lab, since BPD is already installed.



__ 5. Click the **Logout** link on top right corner to sign off.

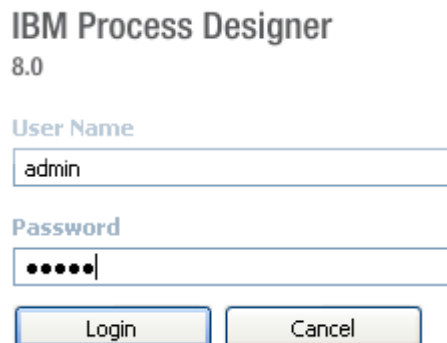
__ 6. Close the browser.

Part 3 - Launch PD

__ 1. Click Windows **Start** menu.



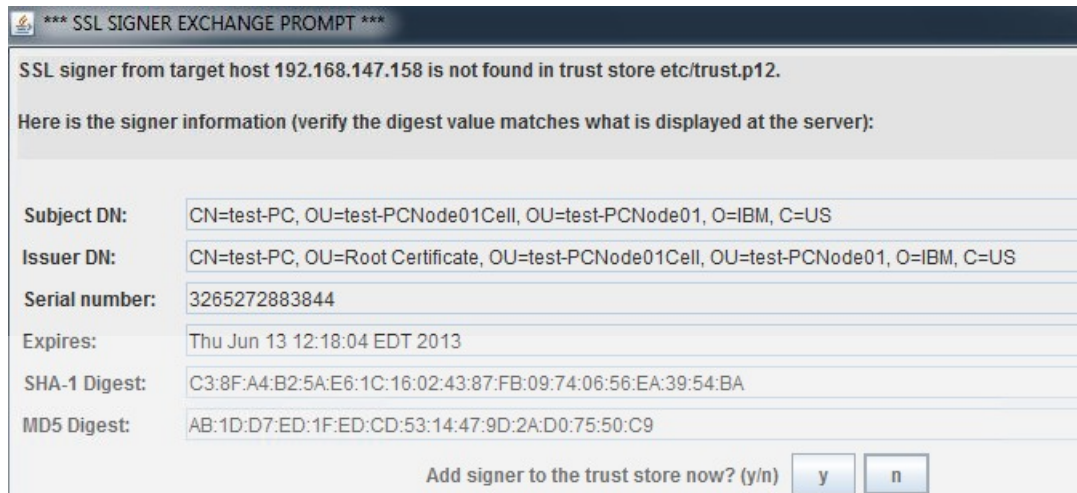
__ 2. Select **All Programs > IBM > IBM Process Designer > IBM Process Designer 8.0**.



__ 3. Enter **admin** as user ID and password.

__ 4. Click **Login**.

PD will connect to Process Center and authenticate you.



__5. You may see an SSL signer window, Click y.

Advanced

How does PD know in which machine Process Center is running? The URL is configured in C:\IBM\BPM\ProcessDesigner\v8.0\eclipse.ini file. The relevant line is as follows:

```
-Dcom.ibm.bpm.processcenter.url="http://<HOSTNAME>:<PORT>"
```

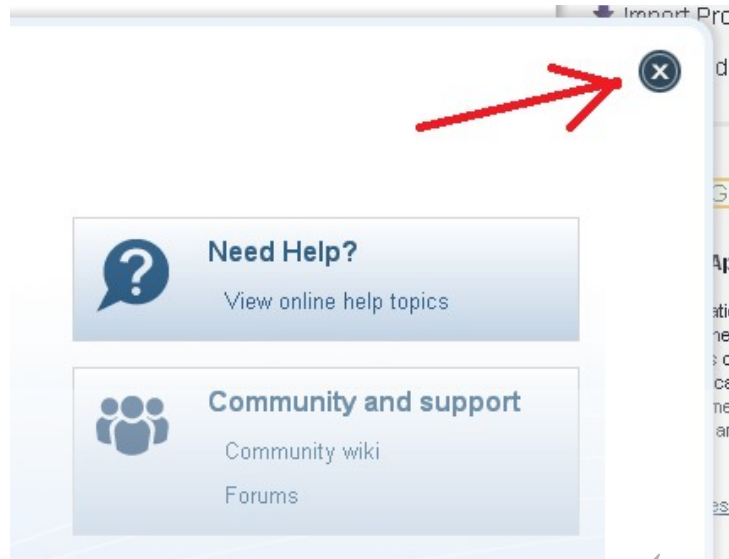
For example:

```
-Dcom.ibm.bpm.processcenter.url="http://onlineorder.webagesolutions.com:9080"
```

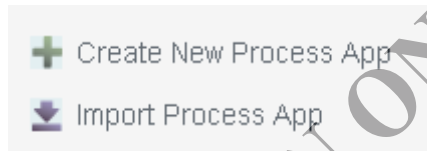
Part 4 - Create a New Business Process

A process is defined within a process application. A process application is eventually deployed inside Process Manager server.

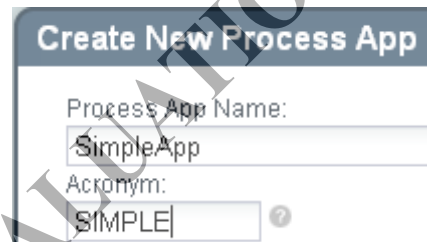
We will first create a process application.



__1. Close the getting started dialog.



__2. From the right hand side of PD, click **Create New Process App**.



__3. Enter **SimpleApp** as the application name.

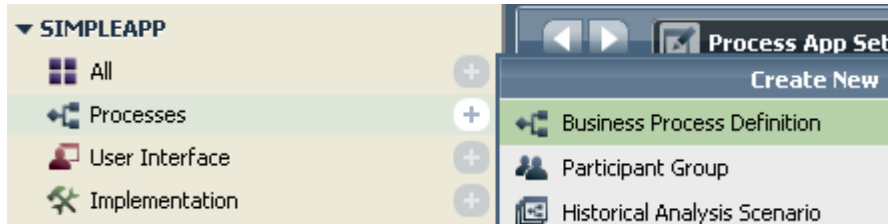
__4. Enter **SIMPLE** as the acronym. This name has to be unique among all other applications.

__5. Click **Create**.



__6. Click **Open in Designer** next to **SimpleApp** to start working on the application.

You are now looking at the PD tool. We will slowly get to know this tool. We will now create a new business process.



__7. Move your mouse above **Process** in the left pane and then click **+** icon next to **Process** and select **Business Process Definition**.

New Business Process Definition

A business process definition (BPD) is a reusable process definition that is common to all run-time instances of that process.

Name:

__8. Enter **GreetingProcess** as the name.

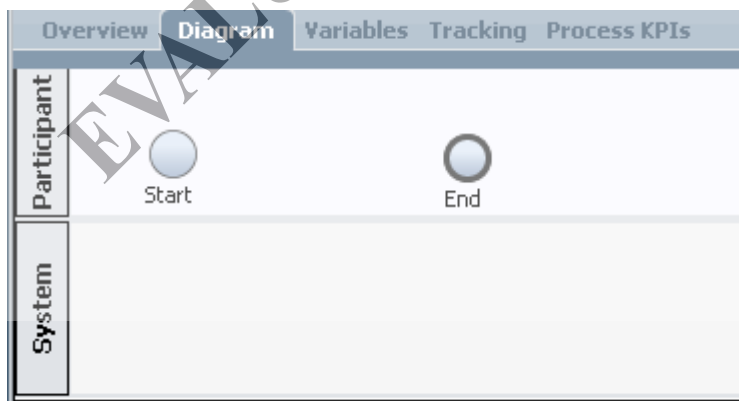
__9. Click **Finish**.

PD will create the process and open it in the editor.

Part 5 - Model the Process

Our process is very simple. It will have only one activity that will print out the greeting message in a file.

First, let's inspect what PD has already created for us.



__1. Note the two swim lanes – **Participant** and **System** – created by PD by default.

Each swim lane groups activities performed by a specific group of users or software systems. This visually explains who does what. You can always create new lanes. The two lanes created by PD have these implicit meanings:

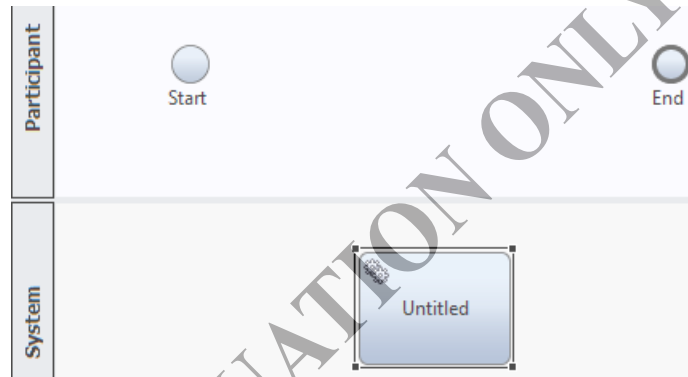
- Participant – Tasks performed by people.
- System – Tasks performed by external systems or organizations.

Again, this is just the default settings. You can always create more meaningful lanes. For now, we will keep using the default ones.

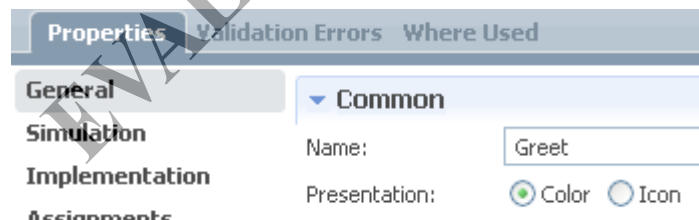
__2. Note that the **Start** and **End** nodes have been added by PD. Every process needs to have these.

We will now add an activity that will print the greeting message. Since this work will be done by software, we will add it to the System lane. We will do that just to convey who does the work. Otherwise, you can put the activity in any lane you like.

__3. From the right hand side palette, click the **activity**  icon.

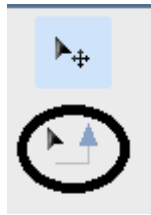


__4. Then, click on the System lane to add the activity.

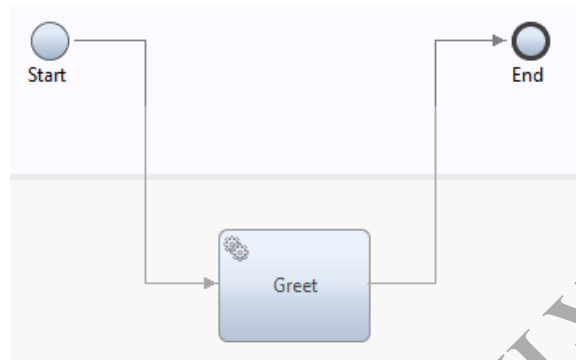


__5. Select the newly added activity from the properties pane at the bottom, enter **Greet** as name.

After you add activities, the next step in modeling is to define their sequence of execution. This is done using the connector arrow, known as Sequence Flow in BPMN.



__6. Click the **Sequence Flow** icon in the tool palette on the right.



__7. Establish connections as shown above.

__8. Save changes (Control+S).

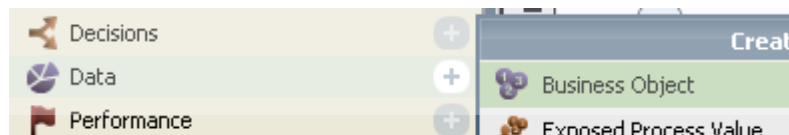
Our process model is now complete. Next we can move to defining the process so that it can be actually executed as software.

Part 6 - Define the Data Model

First, we need to capture the structure of information the process deals with. At the very beginning of the lab, we said that the process receives as input:

- The time of day message, such as, "Good morning" or "Good evening".
- The name of the person being greeted.

We will now create a data type or business object (BO) that models that structure. Later, we will configure the input type of the process to that business object.



__1. Click **+** icon next to **Data** and select **Business Object**.

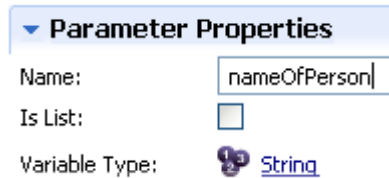
__2. Enter **GreetingInputBO** as the name of the BO.

__3. Click **Finish**.

We will now add two parameters to the BO as per our requirement.



__4. Next to the **Parameters** list, click the **Add** button.



__5. In the **Parameter Properties** pane, enter **nameOfPerson** as the name of the parameter.

__6. Note that the type of the parameter is String. This works for us and we will leave it unchanged.

__7. Similarly, add another parameter called **timeOfDayMessage**.



__8. Save changes.

Now, we will set the data type of the input for the process to *GreetingInput*.

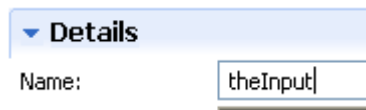


__9. Click **Processes** and then double click *GreetingProcess* to open it in the editor.



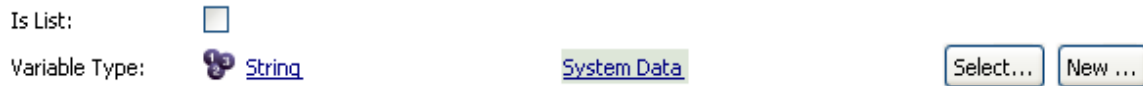
__10. Click the **Variables** tab.

__11. Click **Add Input**.



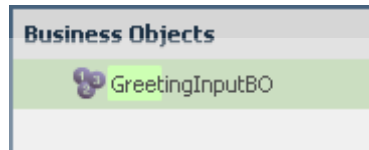
__12. Change the name of the variable to ***theInput***.

By default, the type of variable is set to String. We will need to change that to *GreetingInputBO*.



__13. Next to **Variable Type** click **Select**.

__14. Start typing the name of the BO – *GreetingInputBO* – to filter the list of all known types.



__15. Click **GreetingInputBO** to select it.

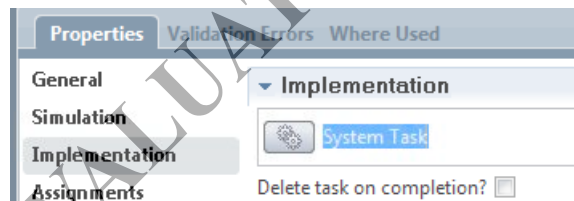
__16. Save changes.

Part 7 - Implement the Activity

Finally, we will implement the Greet activity. Recall, it needs to print out a message like "Good morning, Bill" in a file.

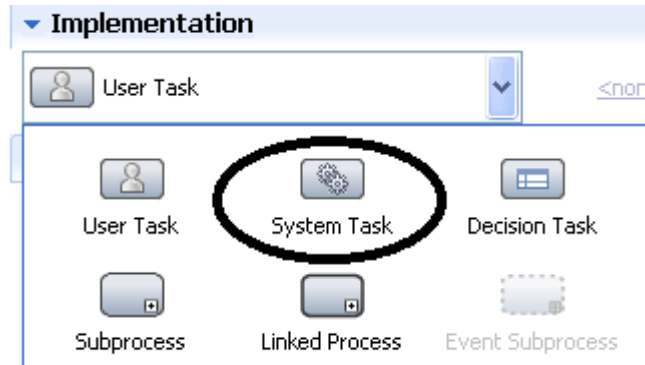
__1. Click the **Diagram** tab at the top of the process editor.

__2. Select the **Greet** activity.



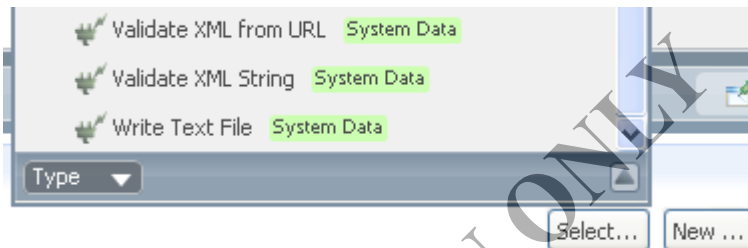
__3. At the bottom, from the **Properties** pane, select **Implementation**.

There are many different ways to implement an activity. Knowing how to implement an activity will be a large part of this class. *System Task* should be selected by default as shown above; the software remembers the last task implemented in your project so if you already changed the implementation to another task that task will be the default task. We will use in this lab a built-in service that is capable of writing data to a file. This type of task is called *System Task*.



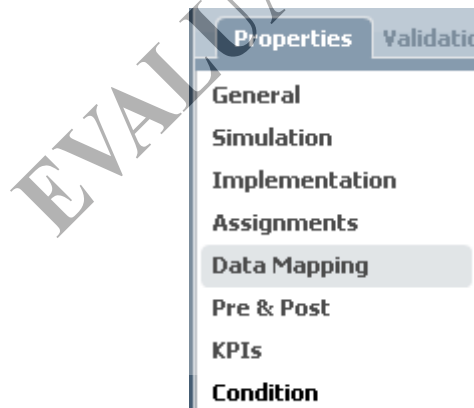
__4. If **System task** is not selected by default then click the drop down and select **System Task**.

__5. Click the **Select** button next to System Task dropdown.

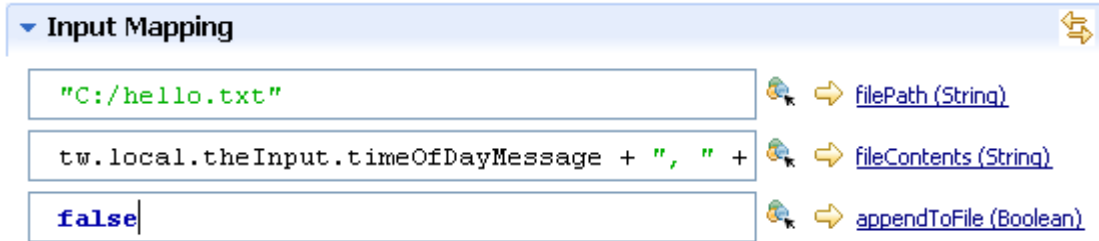


__6. Scroll down and select **Write Text File**.

Now, we will configure the "Write to File" task and specify the location of the file and the contents. Tasks take these configuration parameters as input. What we need to do now is supply these input parameters.



__7. Click the **Data Mapping** section.



8. Enter the following input parameter values.

Parameter	Value
filePath	"C:/hello.txt"
fileContents	tw.local.theInput.timeOfDayMessage + ", " + tw.local.theInput.nameOfPerson
appendToFile	FALSE

Recall, our input variable was called *theInput*. Expression like *tw.local.theInput.timeOfDayMessage* outputs specific elements within the input business object.

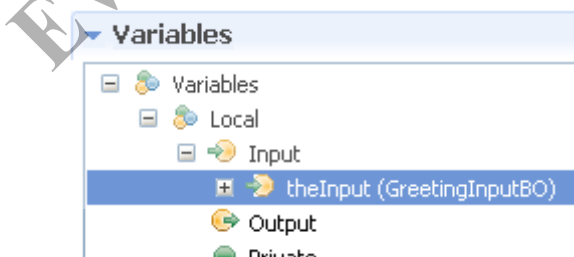
9. Save changes.

Our process is now fully defined. We can proceed to execute and test it out.

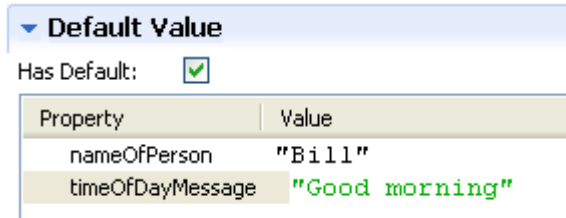
Part 8 - Testing

Before we run the process, we should configure the input data.

1. Click the **Variables** tab in the process editor.



2. Select **theInput** variable as shown above.



__3. Under **Default Value**, check **Has Default**.

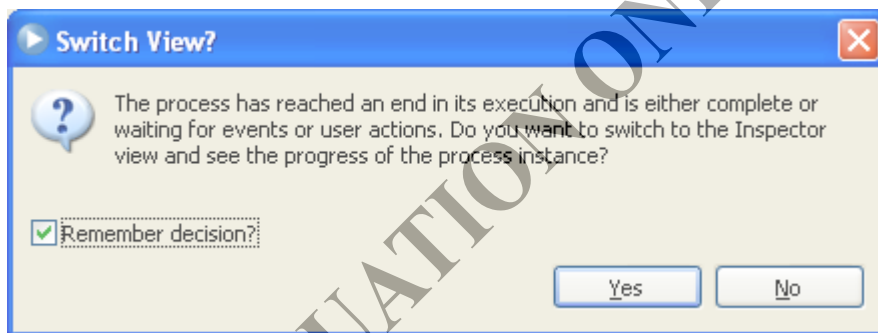
__4. Enter some values for the *nameOfPerson* and *timeOfDayMessage* parameters. Feel free to use your own name.

__5. Save changes.

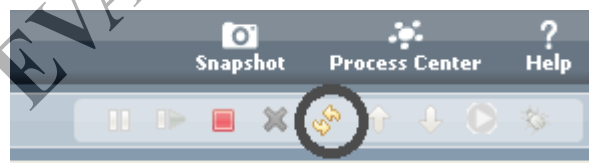
Now, we are ready to run the process.



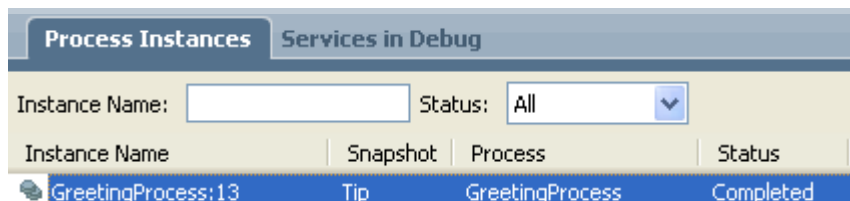
__6. From the top right hand corner, click the **Run Process** toolbar icon.



__7. PD will offer to switch the view so you can see the result. Check **Remember decision** and click **Yes**.

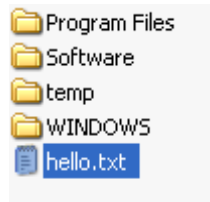


__8. Click the refresh button to view the latest status of the process instance.



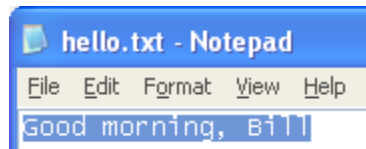
__9. Make sure that the status is shown as **Completed**.

__10. Open a file explorer window.



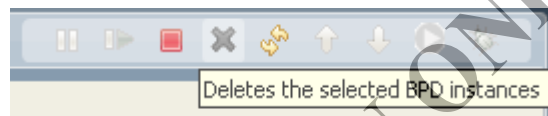
__11. Make sure that you can see **hello.txt** under C:\ drive.

__12. Double click the file to view it.

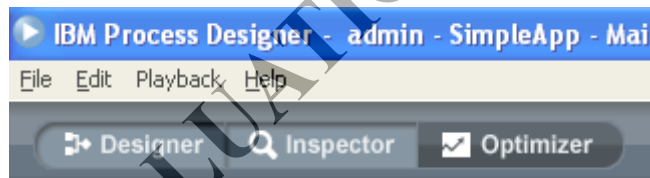


__13. Verify that the greeting message is correct.

__14. Close the file.



__15. Back in PD, click the delete button to delete the process instance and confirm you want to delete it.



__16. Click the **Designer** button at the top left to switch back to the designer view.

Part 9 - Review

This was a rather long lab for an introductory topic. But, we covered a lot of ground quickly. This will help us solve more complex problems in the upcoming labs.

Take a moment to go through these items and make sure that you learned how to do them:

- How to log into PD.
- How to create a new process application and process definition.
- How to add activities to a process and connect them using sequence flow arrows.
- How to accept input data from the process.
- How to implement activities and pass data to them.
- How to execute a process and test it out.

EVALUATION ONLY

Lab 2 - Basic Process Modeling

The main purpose of the next few labs will be to:

- Learn how to document and analyze business requirements and
- In the next lab, model a process using Business Process Modeling Notation (BPMN) following the requirements.

This is the basic skill that will come handy in future labs where we tackle more complex problems. We will not get into completing the process definition. For now, the focus is modeling.

Part 1 - The Case Study

A case study is a general document that outlines the needs and problems a business is currently facing. It does not have to follow any specific format. Here we will describe a case study that is quite common among businesses in real life. In fact, the case study comes from various interactions captured by a standards body called RosettaNet (www.rosettanet.org).

A business (from now on referred to as buyer organization or just buyer) procures goods from another business (from now on referred to as seller organization or just seller). A part of this transaction requires a price quote process. The process goes as follows:

When the time comes for the buyer to procure an item, it needs to query one or more sellers offering the item for a price quote and availability report. This query can be done automatically by a software system or by a human user using the buyer's portal web site. Presumably, the seller with lowest lead time and cost will win the business and receive the order.

Part 2 - Develop the Business Use Cases

Before starting to map a process, you should capture the flow of activities in a business use case. The training class probably hasn't covered business use case yet. So, we will give you the use cases for the two interactions we have identified above. Have a look at them and get familiar with the format.

UC001 Request Price Quote and Availability

Description

This use case provides an automated process for trading partners to request and provide product price and availability.

Actors

1. Buyer organization (referred here as buyer)
2. Seller organization (referred here as seller)

Main Flow

The buyer composes a price quote and availability request message. The message contains the following information:

- The buyer's business ID (DUNS number)
- For each requested item, the following information is provided:
 - The product Stock Keeping Unit (SKU) number.
 - Quantity of the product requested.

The buyer sends the request to the seller and waits for a response. The seller receives the request. The seller composes a reply message. In this message, for each requested item, the following information is provided:

- The product Stock Keeping Unit (SKU) number.
- Available quantity.
- Availability date.
- Quote ID.
- Price.

Part 3 - Identify the Processes and Set Scope

If you read the use case carefully, you will notice that it has two actors that are both organizations. This is a hint that the process is actually made up of two separate processes. Each actor organization will use its own process to deal with the interaction. In this case, the buyer side initiates the communication and the seller participates. We will call the buyer side process the **initiator** and the seller side process the **participant**.

As a business analyst you have to set the scope of the project. Do you model the initiator or the participant or both? The answer will depend on which organization you belong to and how much visibility and control you have over the other organization.

For the sake of completeness, we will model just the initiator (buyer side) process.

Part 4 - Review

We will actually model the processes in the next few labs. In this lab, we analyzed the problem and captured the requirements in business use cases. Then we identified the processes and set the project's scope.

EVALUATION ONLY