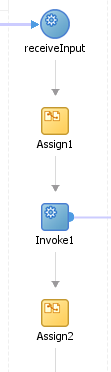
**egar condicional al flujo del BPEL**

* Este tutorial está dedicado a el uso de la sentencia “**if**” dentro del BPEL.
* Vamos a comenzar a trabajar a partir del flujo creado en el tutorial anterior (2)

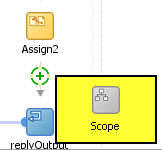
Partiendo de:



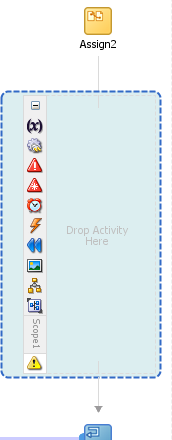
Lo que queremos hacer:

* La idea es preguntar la prioridad del “**TODO**” traído del “**TodoServiceBus**”.
  + En caso que sea mayor que 20, cambiar el valor de la cabecera (**descripcion**) a: La tarea tiene prioridad alta.
  + En caso que esté entre 20 (sin incluir) y 10: La prioridad de la tarea es media.
  + En caso de que sea menor que 10 (sin incluir): La tarea no tiene importancia.
    - Además, si entramos aquí, vamos a crear una variable boolean para otro “**if”** y aumentar el valor de la tarea de 10 en 10 hasta llegar a 100.
    - Y concatenamos con el mensaje de “La tarea no tiene importancia” con “por lo que fue elevada”
* Y después devolver el resultado final.

1. Antes de comenzar, vamos a crear un “**scope**” entre el segundo asignar y el “**output**”, para organizar el trabajo y crear variables propias del “**scope**”.
   1. En el menú de componentes, buscamos el nodo llamado “**scope**” [].
   2. Lo presionamos y lo arrastramos entre el segundo asignar y el “**output**”; esto queda de la siguiente forma:

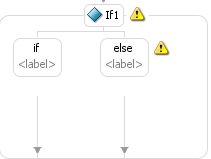


* 1. Resultado final:

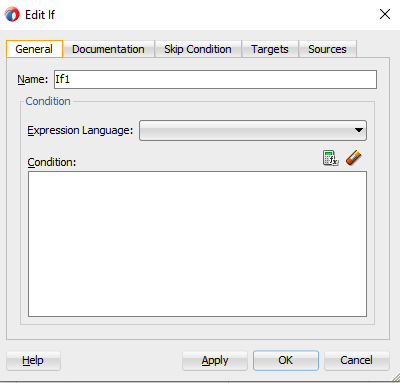


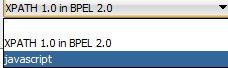
* 1. Dentro del recuadro marcado por bordes azules, podemos añadir los nodos correspondientes al flujo de la condicional.

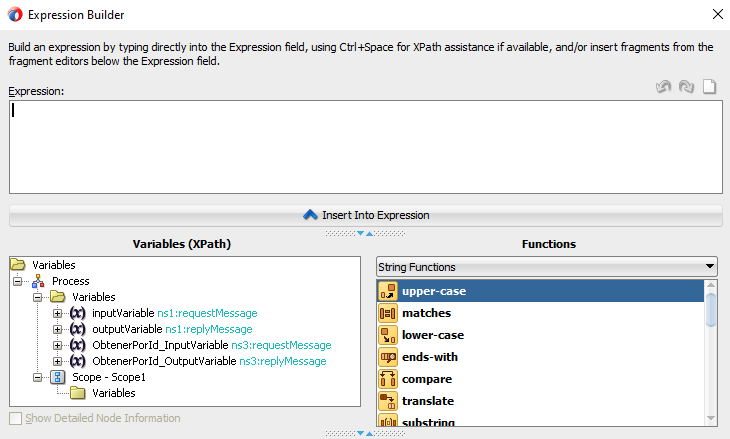
1. Ahora buscamos, entre el menú de componentes, el nodo correspondiente a una condicional [].
   1. Lo presionamos y lo arrastramos dentro de nuestro nuevo “**Scope**”
   2. Una vez puesto dentro, queda de la siguiente forma:



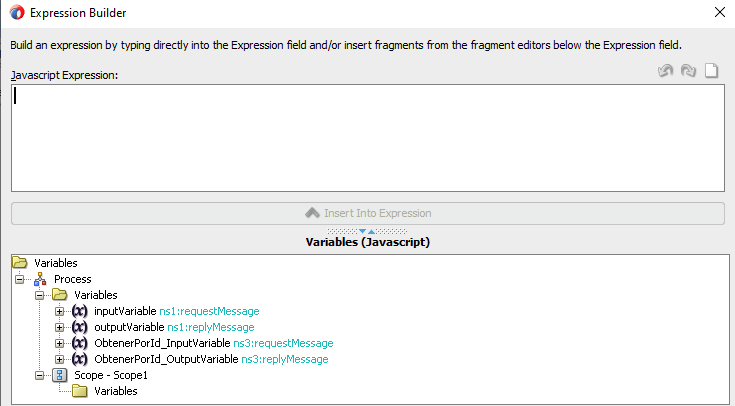
* 1. Las advertencias () son debido a que no tenemos declarada ninguna condición dentro del “**if**”.
  2. Damos doble clic sobre el componente “**if**” del nodo creado (Por defecto, la condicional principal, se realiza en el “**if**” principal (), los “**elseif**” tienen su propia pestaña).



* + 1. **Name**: Nombre que le podremos a nuestra condicional (La condicional global). Es solo nombre, no es importante para el tratamiento de la condición. Cada rama que se separe de la condicional tiene un “**label**” que se puede cambiar dando clic arriba.
    2. **Expression Languege**: Aquí podremos definir el tipo de lenguaje que queramos utilizar para definir la expresión.
    3. 
       1. La primera opción es mediante el uso de los componentes predefinidos para este tipo de operaciones:



* + - 1. El segundo (**javascript**) permite la creación de una condicional utilizando el lenguaje “**js**” puro:



* + 1. **Condition**: Aquí se podrá poner la condición a utilizar de forma manual o mediante el uso del creador de funciones (). Las dos imágnes superiores son la representación de esta opción por cada tipo de lenguaje a utilizar.
  1. Llenamos la condicional según el lenguaje que más nos sea factible:
     1. XPATH:

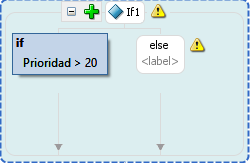


* + 1. Javascript:

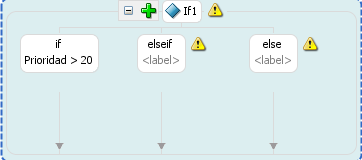


* + 1. El cambio entre ambos lenguajes radica en la forma de acceso a los datos.
  1. Presionamos [**OK**] y ya se aplicaron todos los cambios para esa condicional.

1. Al principio de este tutorial se dijo que tendríamos 3 condicionales; pero el nodo “**if**” por defecto crea la estructura “**IF** - **ELSE**”. En este caso tenemos que agregar un “**ELSE IF**” delante del “**ELSE**”.
   1. Para ello seleccionamos el recuadro completo del if:



* 1. Al hacerlo, nos aparecerá una opción [] para agregar “**ramas/condiciones**” a nuestro flujo condicional. Lo presionamos y el resultado obtenido será el siguiente:

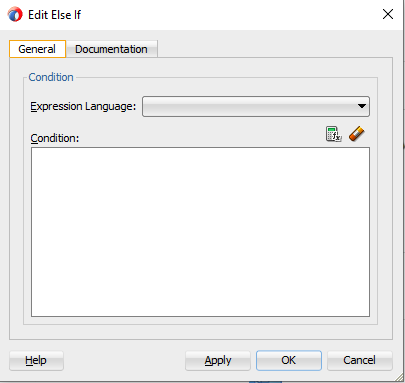


* 1. Ahora solo tocaría modificar la condicional del “**ELSE** **IF**”.

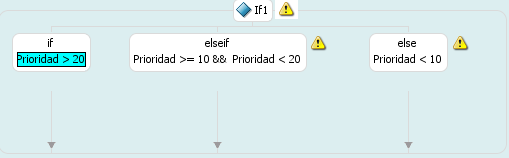
1. Para modificar la condicional del “**ELSE** **IF**” damos doble clic sobre su sección de flujo:



* 1. Esto nos levantará una ventana parecida a la de la condicional principal; la única diferencia radica en que, en esta última, solo se puede modificar la expresión a utilizar.



* 1. Con todas las modificaciones realizadas a las expresiones de las condicionales y los “**labels**” el resultado final es:



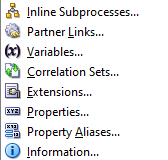
1. Ahora toca asignar los valores según la condicional que se entrase.
   1. **IF**: Aquí vamos a asignar el texto “La tarea tiene prioridad alta”
      1. Añadimos un nodo de asignación dentro del primer **if**:



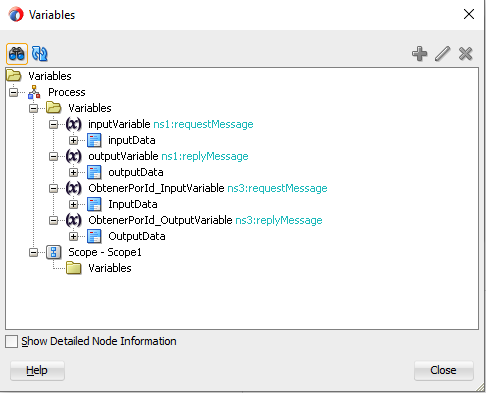
* 1. **ELSE IF:** Aquí pondremos el mensaje “La prioridad de la tarea es media” concatenado con el mensaje que ya tenía antes.
     1. La expresión a utilizar en el asignar es la siguiente:



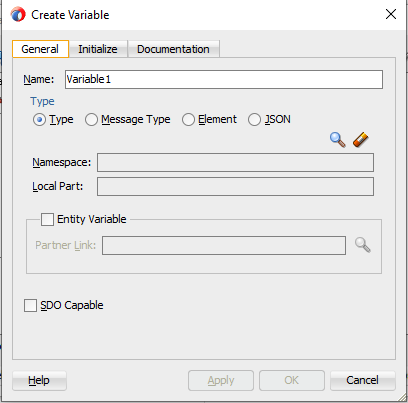
* 1. **ELSE:** Aquí se dijo primero que se iba a crear una variable para controlar en el “**scope**” si se pasó por este flujo de la condicional. Para ello vamos a el ícono [] en la barra superior del BPEL.
     1. Por ahora, con lo que sabemos, la mejor forma de decirle a una variable que comportamiento debe seguir es mediante el uso de los “Schemas”; por lo que se creará uno para almacenar todas las variables auxiliares que se van a utilizar.
     2. Esto es un menú desplegable que nos brinda las siguientes opciones:



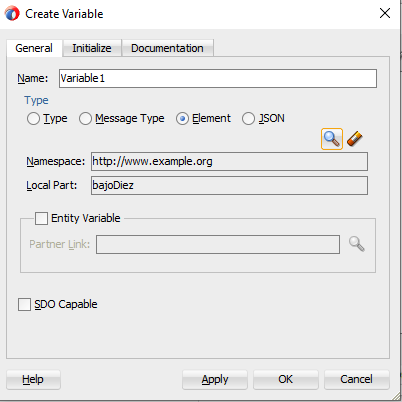
* + 1. La que nos interesa del conjunto es ()
    2. Damos clic en ella para que nos aparezca la siguiente ventana:



* + 1. Aquí se van a mostrar todas las variables, globales () o de “**scope**” (). En este caso, vamos a modificar las variables del “**Scope1**”.
    2. Con la carpeta () del “**scope**” () seleccionada, damos clic en ().
    3. Ahora nos aparece una ventana similar a la que utilizamos para crear las variables del “**invoke**”.



* + 1. Aquí seleccionamos el “**Element**” dentro de las opciones de “**Type**”.
       1. Le ponemos el nombre que queramos.
       2. Y presionamos en la [] para buscar el esquema donde se declararon las variables auxiliares.
    2. Después de llenadas todas estas opciones, el resultado de la creación de la variable es la siguiente:



* + 1. Damos [**OK**] y la estructura de las variables queda de la siguiente forma:



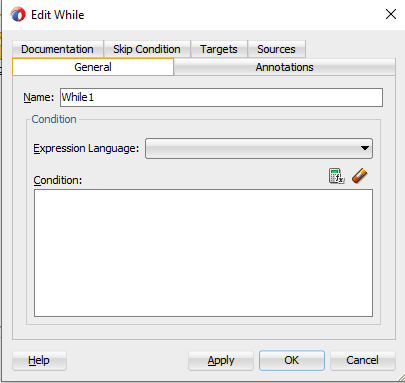
* 1. **“ELSE”** teniendo la variable creada, pasamos a asignarle el valor de true.
     1. La expresión para lograr esto es:



* 1. Ahora debemos añadir un ciclo “**while**” [] para incrementar el valor de la prioridad en el objeto “**Todo**” llamado.



* + 1. Lo primero que se debe hacer es crear la condición de parada del while. Doble clic sobre el símbolo [] para acceder a la ventana de configuración. Muy parecida a la del “**if**”.



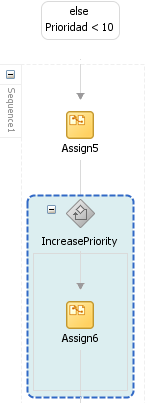
* + 1. La expresión resultante es:

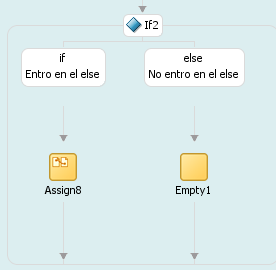


* + 1. La asignación queda de la siguiente forma:



* + 1. El flujo por esa rama del “**if**” queda:



1. Saliendo del “**if**” nos toca crear otra condicional para saber si entró al “**else**” o no mediante la variable de control creada. Sería repetir los mismos pasos seguidos arriba, por lo que en esta parte solo voy a mostrar el resultado final.
2. ****
   1. En el caso del “**else**” se utilizó un nodo **“empty”** [] debido a que no se puede dejar un flujo sin sus respectivos elementos internos.