

Práctico de Pruebas sobre Servicios

Los servicios web son cada vez más numerosos en las aplicaciones y por eso es necesario poder contar con alternativas para verificarlos de forma rápida y eficiente.

La herramienta SOAPUI permite realizar de forma automática alguna de las verificaciones sobre servicios web. La idea es que usted pueda automatizar estas verificaciones para obtener así sus beneficios asociados.

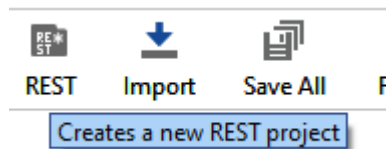
Para este taller se utilizará dos posibilidades de la herramienta:

- 1 – Para el escenario en cual requerimos exponer un servicio web y se desea verificar su correcto despliegue.
- 2 – Para lograr aislar una unidad de test que consume un servicio web de terceros.

Para el primer se utilizará la posibilidad de crear suites de pruebas que brinda SOAPUI

El caso de prueba se generará en base a tecnología REST. Para eso es necesario dirigirse a SOAPUI y seguir los siguientes pasos:

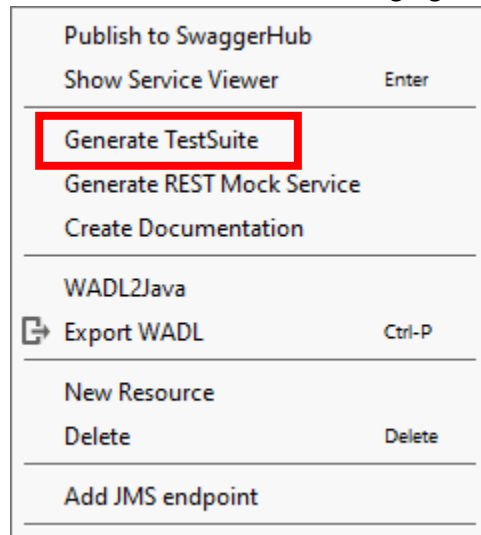
A. Crear un nuevo proyecto REST



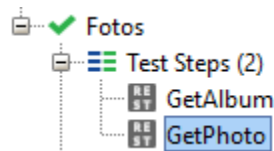
- a.
- b. Agregar el siguiente servicio web: jsonplaceholder.typicode.com
- c. Agregar los recursos Posts, Users, Photos

Posts			
Resource Path /posts			
Resource Parameters			
	Name	Value	Style
	userid		QUERY
Users			
Resource Path /users			
Resource Parameters			
	Name	Value	Style
	id		QUERY
Photos			
Resource Path /photos			
Resource Parameters			
	Name	Value	Style
	albumId		QUERY
	id		QUERY

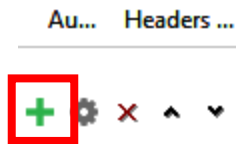
- B. Crear una suite de pruebas
- Click derecho sobre el servicio agregado y seleccionamos "Generate Test Suite"



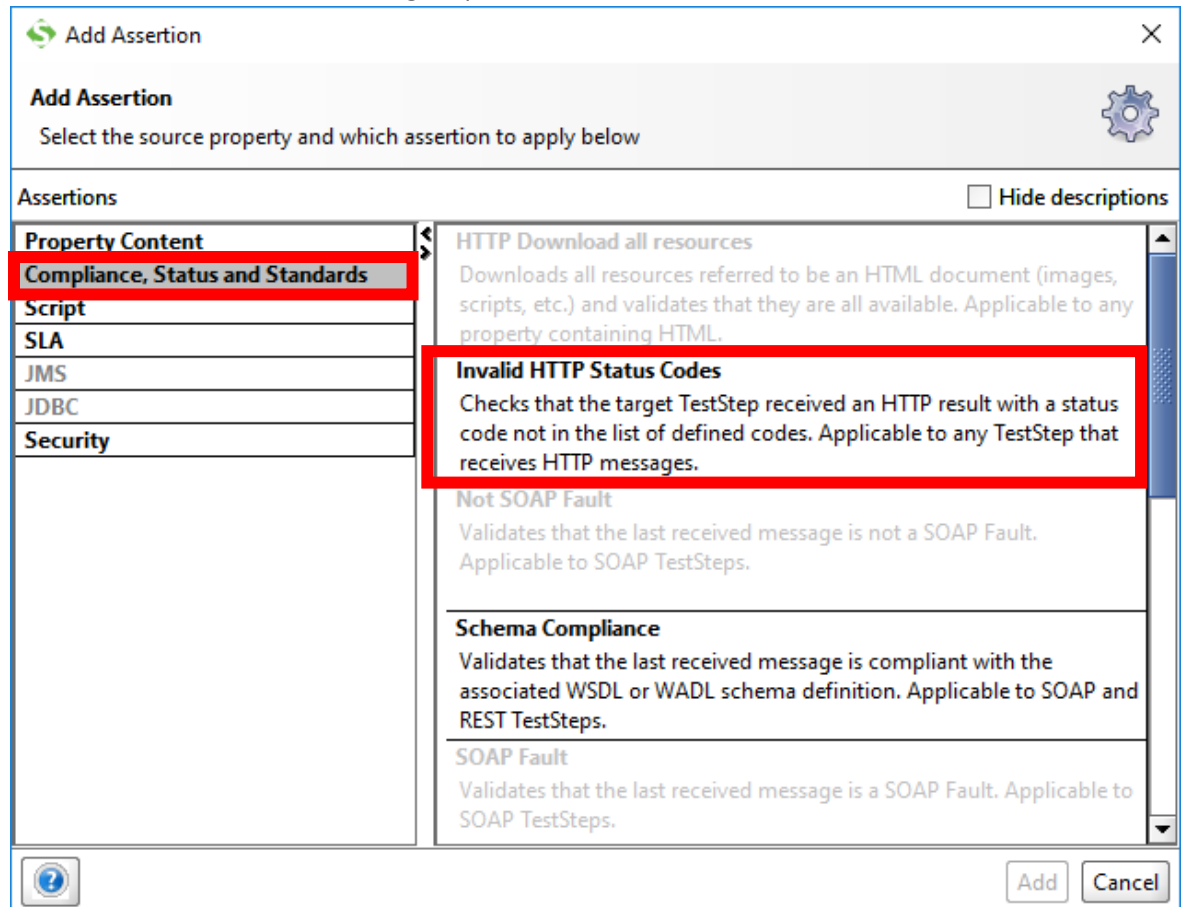
- Esto creará una suite de prueba con un conjunto de test de prueba para cada request que posea el servicio
- Seleccionar el test case "Fotos" y el paso "GetPhoto"



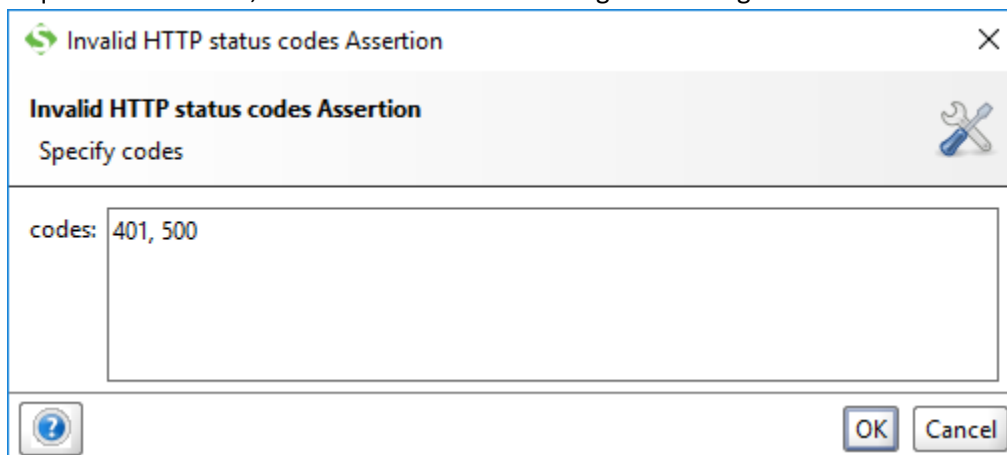
- Abrir el paso y hacer click sobre el botón "Assertions"
- Presionar el botón "Add Assertion"



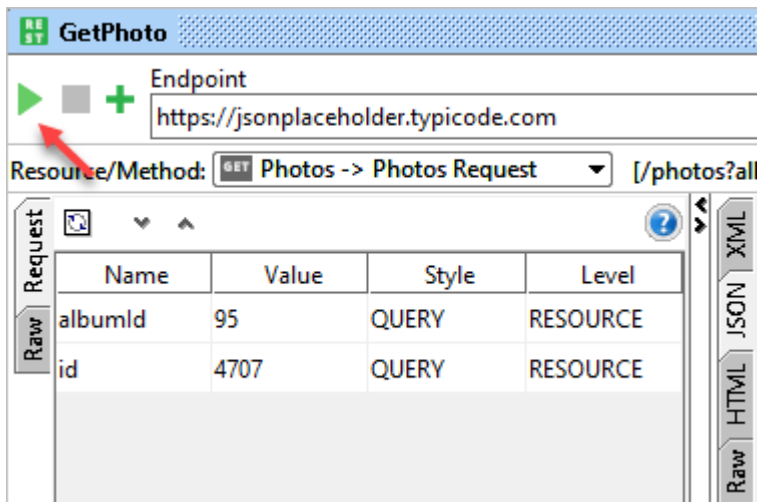
- f. Seleccionar el assertion de la imagen que se muestra a continuación



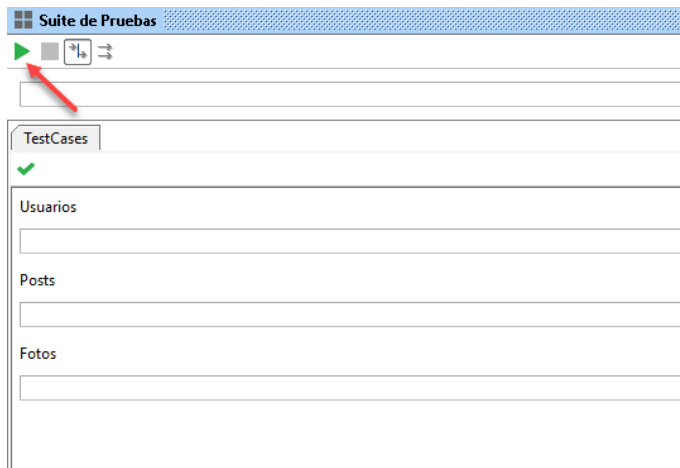
- C. Presionar el botón "Add"
- D. Agregar los valores que se desea checkear que no se reciban como resultado de la ejecución del request "Get Photo", tal como se muestra en la siguiente imagen:



E. Ejecutar Prueba

**Ejercicio:**

1. Generar un assertion para el método "GetAllUsers" del test case "Usuarios"
2. Luego de generado ejecutar toda la suite de pruebas



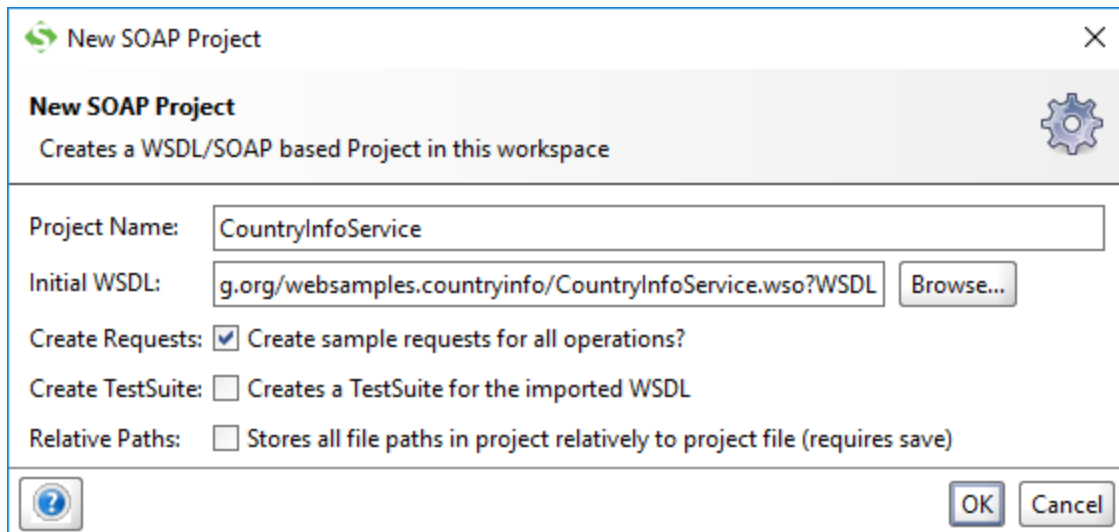
Para el segundo caso, el objetivo es lograr aislar una funcionalidad dependiente de un servicio web de un tercero para así lograr por ejemplo poder realizar un test unitario sobre la funcionalidad.

Con el fin de lograr este objetivo es necesario crear una respuesta automática del servicio web a través de un mock del servicio. Para ello es necesario realizar los siguientes pasos:

A. Crear un nuevo proyecto

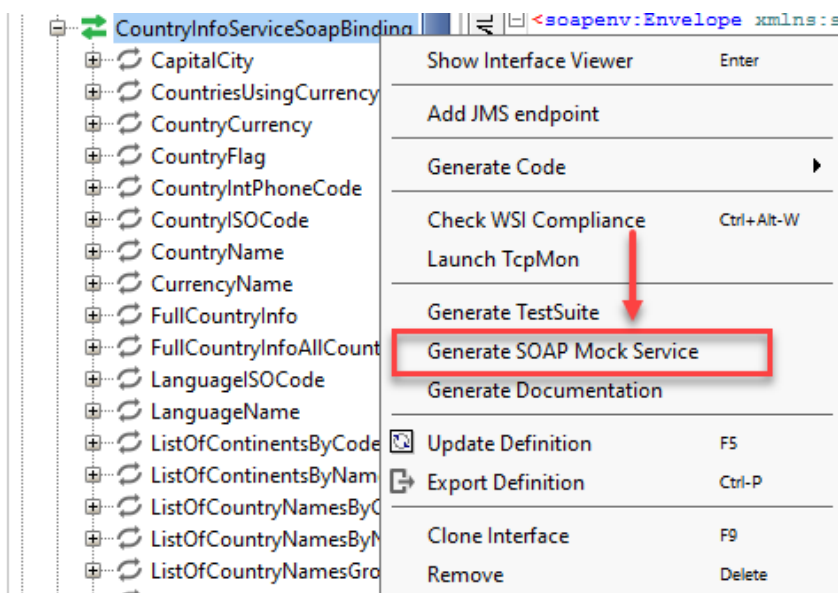
- a. Crear un nuevo proyecto SOAP usando como base el siguiente WSDL

<http://webservices.oorsprong.org/websamples.countryinfo/CountryInfoService.wso?WSDL>



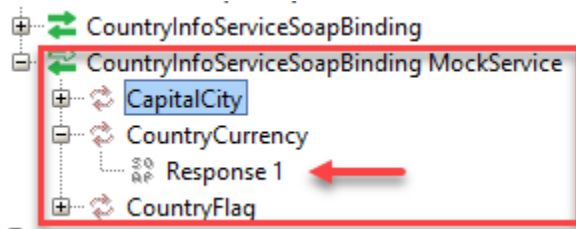
- B. Crear un MockService (Virtualizar el servicio)

- a. Click derecho en el servicio al que se le desea generar un Mock en la interfaz de SOAP y seleccionar "Generate MockService".



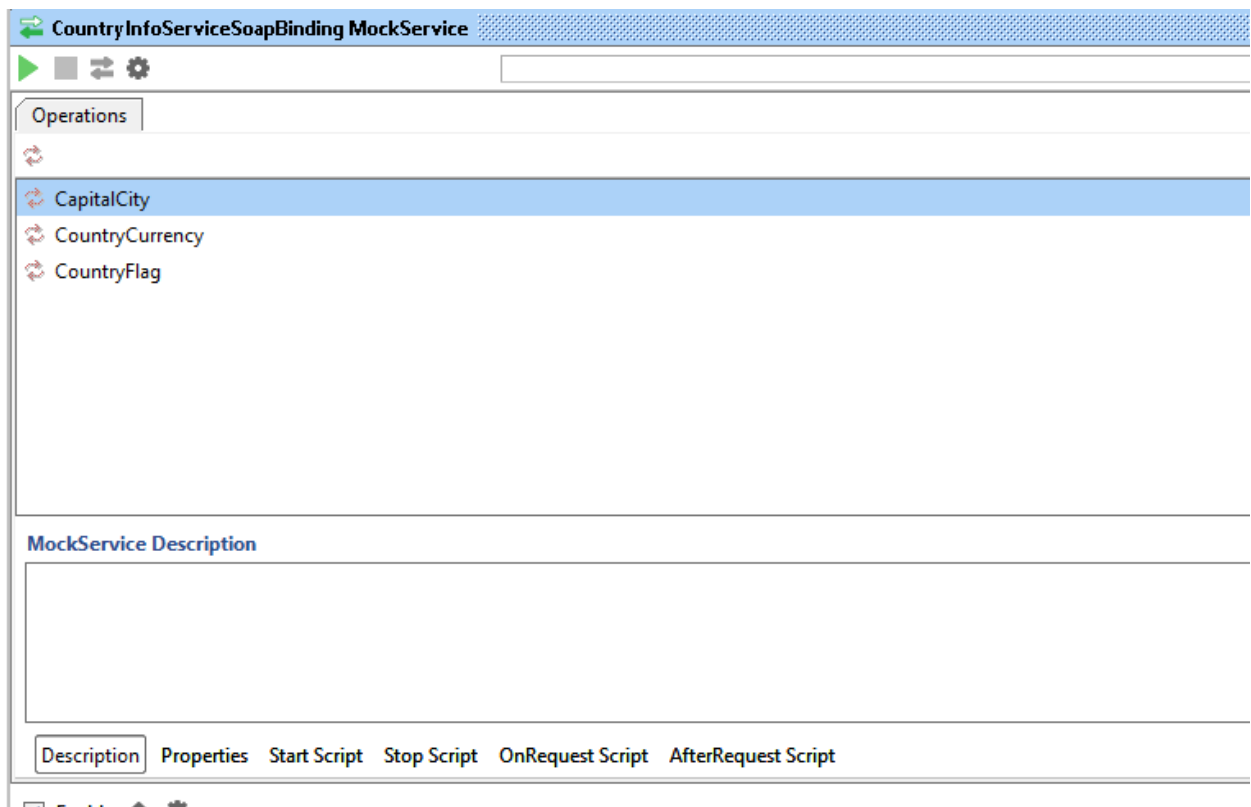
- b. En el diálogo *Generate Mock Service* es posible especificar el Puerto/ruta del servicio web. Hacer click en OK.

- c. Agregar un nombre al MockService en la casilla Name dialog y realizar click en OK.
- d. Luego de creado el MockService, el resultado será un MockService con una operación y una respuesta por cada operación generada.

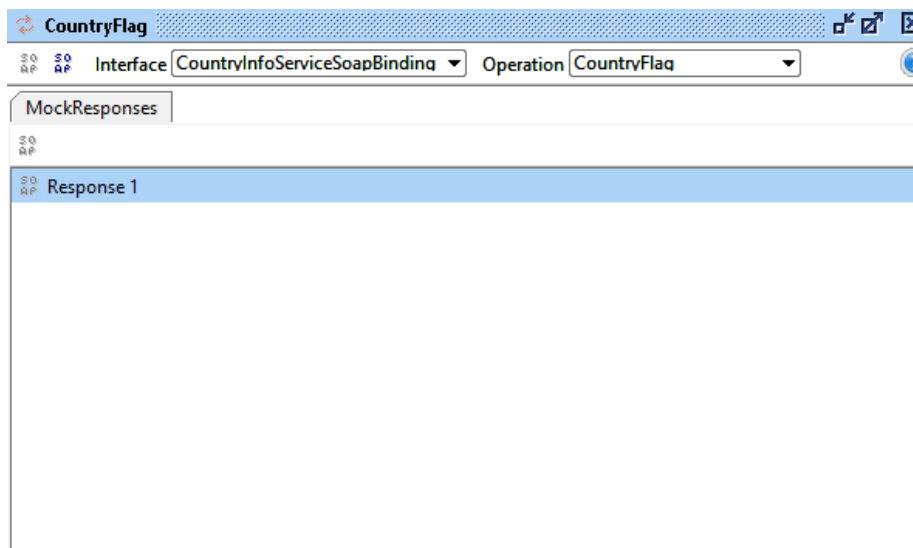


c. Editar un MockService

- a. Realizar doble click sobre el MockService para visualizar el editor. En el editor se puede observar una lista de operaciones para el servicio, así como una operación y un log de respuesta con todas las respuestas que han sido manejadas por el MockService.



- b. Doble click en CountryCity para visualizar la respuesta que posee el MockService:



- c. Como se puede apreciar, solo se posee una. Proceda a editarla. Para ello realizar doble click para abrir el editor de la respuesta.

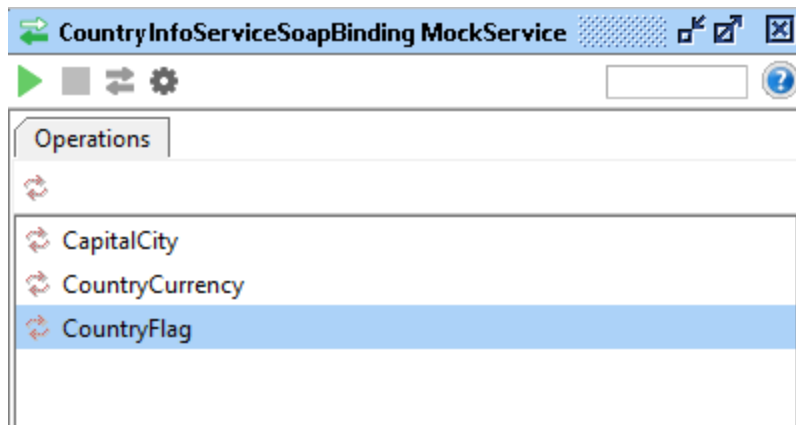


- d. Edite la respuesta a:

<http://www.oorsprong.org/WebSamples.CountryInfo/Images/Uruguay.jpg>.

D. Invocar el MockService

- a. Primero se debe iniciar el MockService. Click  en el editor del MockService. Esto iniciara el servicio en el puerto y ruta configurados.



E. Consumir desde GeneXus el Mock creado

- Desde GeneXus abrir la kb (nombre de KB: **KBUpToDateGeneXus15**) -> Ctrl + Shift + O
- Para poder utilizar el servicio web consumido primero necesitamos realizar algunas modificaciones en la transacción "Country"

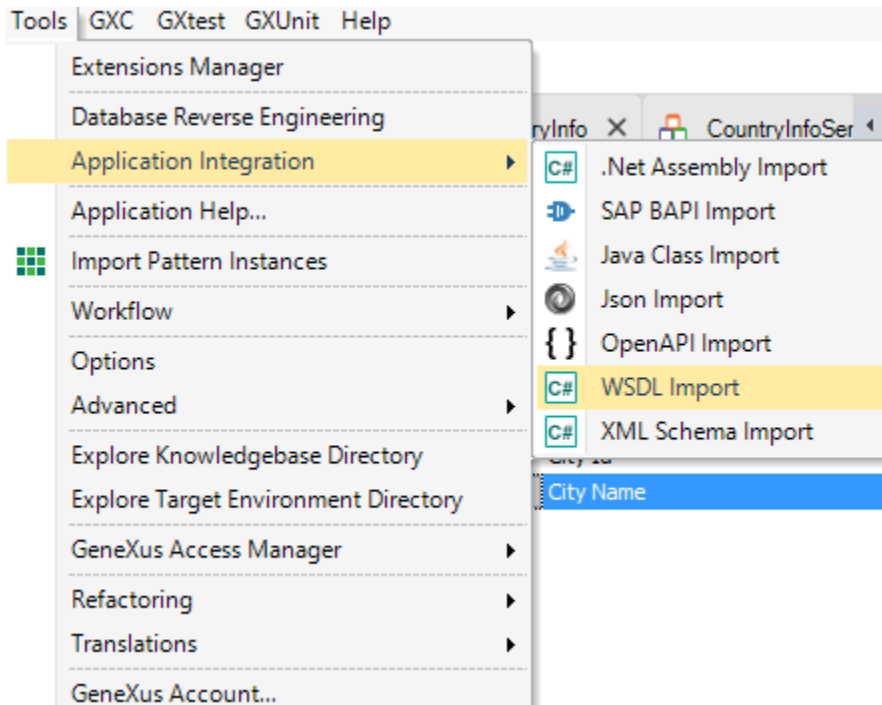
Agregaremos el atributo "CountryISO" tipo Character(20)

Y un sub-nivel "City" al que agregaremos los atributos "CityId" tipo Id y "CityName" tipo Name

Name	Type	Description	Formula	N
Country	Country	Country		
CountryId	Id	Country Id		No
CountryName	Name	Country Name		No
CountryISO	Character(20)	Country ISO		No
City	City	City		
CityId	Id	City Id		No
CityName	Name	City Name		No

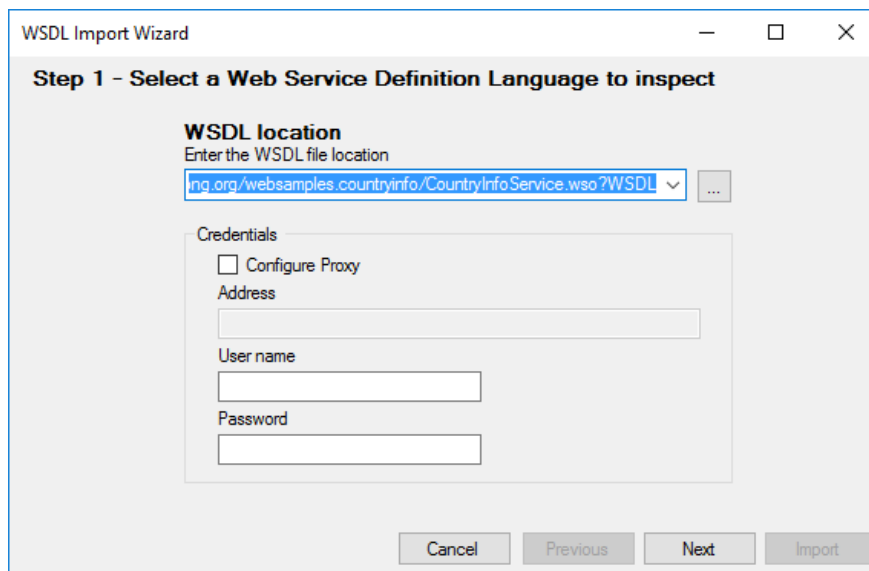
Luego se deberá ejecutar la aplicación y agregar a los países existentes el correspondiente ISO

- Se deberá importar el servicio web desde:



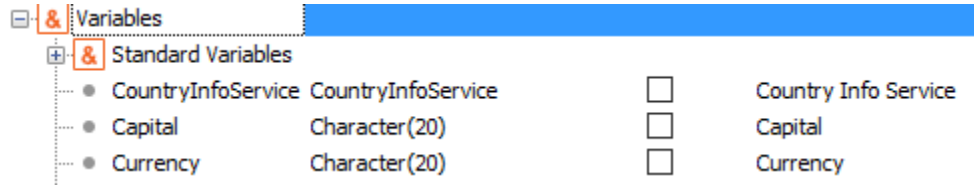
- d. Importar a través del WSDL Import Wizard el servicio web:
<http://webservices.oorsprong.org/websamples.countryinfo/CountryInfoService.wso>

Esto creará un "Objeto externo" de tipo WSDL (Web Service) para ser invocado como un tipo de datos más



- e. Posteriormente crear un nuevo procedimiento GeneXus denominado "CountryInfo" el cual imprimirá el nombre, capital, moneda y bandera de cada país en un PDF. OBS: Solo nombre no será provisto por el servicio web.

- f. Crear las siguientes variables:



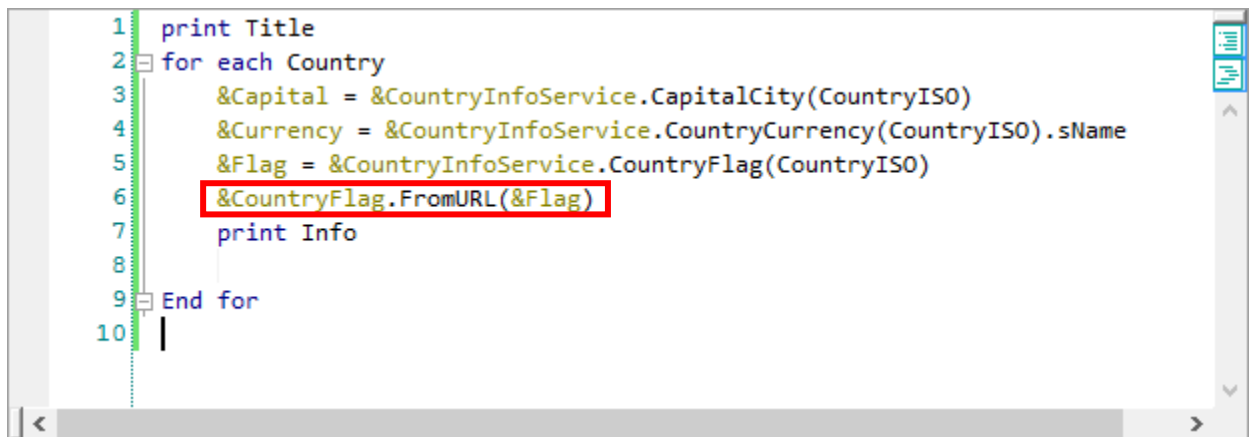
- g. En el Source del procedimiento definir el For Each que recorrerá la tabla Country para proveer los datos a cargar en el layout del procedimiento. La estructura deberá ser la siguiente:

```
print Title
for each Country
    &Capital = &CountryInfoService.CapitalCity(CountryISO)
    &Currency = &CountryInfoService.CountryCurrency(CountryISO).sName
```

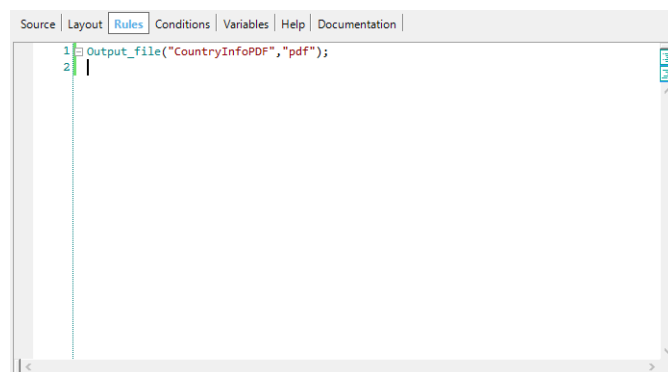
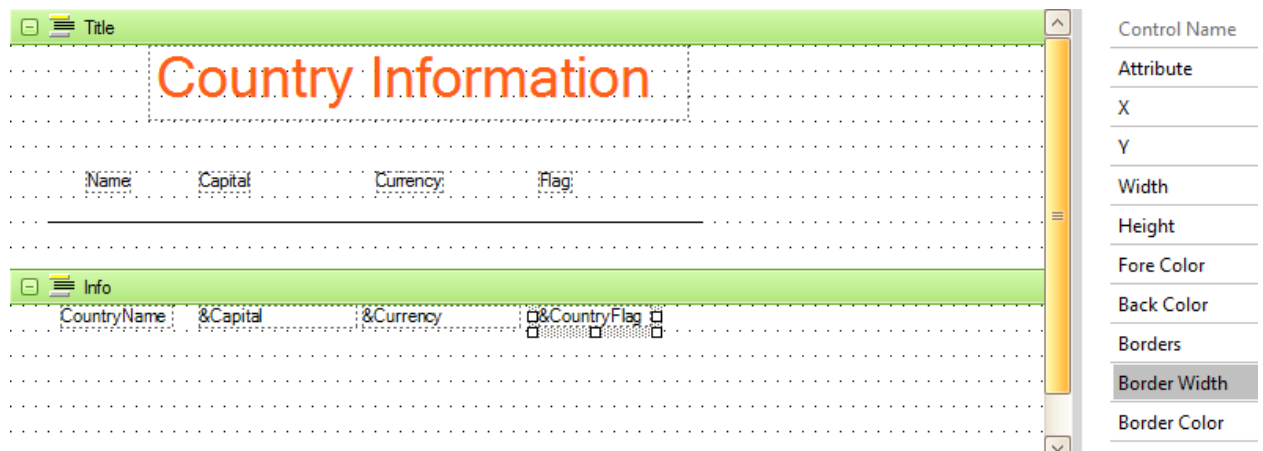
- h. Ya casi están listas las variables. Sin embargo, es necesario agregar la variable que permita obtener la imagen de las banderas. Se observa que el web service retorna la url de la imagen de la bandera (<http://www.oorsprong.org/WebSamples/CountryInfo/Images/Uruguay.jpg>) por lo que se deberá crear una variable &Flag para almacenar la url y otra &CountryFlag del tipo imagen para almacenar la imagen de la bandera propiamente dicha.



- i. Agregar en el source del procedimiento la información que se cargará de la recorrida del For Each



- j. Crear el layout correspondiente



- k. Realizar la siguiente configuración y luego ejecutar el procedimiento (o la aplicación si es necesario crear países)

▼ Procedure: CountryInfo	
Name	CountryInfo
Description	Country Info
Module/Folder	Root Module
Main program	True
Call protocol	HTTP
Execute in new LL	False
Test Case	False
Object To Test	
Qualified Name	CountryInfo
Object Visibility	Public
▼ Web information	
Auto compress	Use Environment property...

- l. La ejecución debería fallar. ¿Qué falló? Si no encuentra el error, diríjase al siguiente [Anexo](#)
- m. Luego de corregido el error, se vuelve a ejecutar y se deberá tener la respuesta del web service lista.
- n. Pero esto no está utilizando la respuesta del Mock creado. Para lograrlo es necesario cambiar la propiedad “Address” en las properties de cada servicio, por la dirección del Mock.

Protocol	SOAP 1.1
Style	Document
Use	Literal
Enable MTOM	False
Address	http://gmanceboNTB:8088/mockCountryInfoServiceSoapBi
Port Type Name	
Port Type Namespace	http://www.oorsprong.org/websamples.countryinfo
Action	
XML Name	CountryFlag
Request Namespace	http://www.oorsprong.org/websamples.countryinfo
Response XML Name	CountryFlagResponse
Response Namespace	http://www.oorsprong.org/websamples.countryinfo
XML Return Element Info	

- o. Solicite apoyo docente para llevar a cabo esta actividad para configurar una respuesta secuencial del mock.
- p. Luego de realizadas las configuraciones vuelva a ejecutar el programa y podrá visualizar el contenido esta vez consumiendo información del Mock.

Anexo:

La transacción Country tiene seteada la propiedad Data Provider en “Yes” y como se cambió la configuración de la transacción agregando nuevos atributos. Esta carga de datos ya no es válida. Se debe cambiar la propiedad populate Data a “False”.