

Tip – Usar Model Con Escenarios

En el siguiente tip mostraremos cómo usar modelos con los distintos tipos de escenarios (scenario y escenarios outline), basándonos en el reto # 1 para el patrón Screenplay.

Como ya sabemos, para usar el paquete model y hacer una asignación de datos podemos hacer uso de la funcionalidad que nos brinda Cucumber y Java en función de una lista (List).

```
# language:es
#Author: yarias@choucairtesting.com

> Característica: Automation Demo Site
  Como usuario
    Quiero realizar el registro en Automation Demo Site
    A través del diligenciamiento del formulario
  @casol
  > Escenario: Relizar el registro en la pagina Automation Demo Site
    Dado que Carlos quiere acceder al Sitio Automation Demo Site
    Cuando el realiza el registro en la página
      |first_name | last_name | address | mail | phone | languages | skills | conuntry | select_conuntry | date_birth_year |
      |Yeison | Arias | Calle 1 N 1 1 | yfarias2@correo.co | 3013696970 | Arabic | C++ | Colombia | Australia | 1981 |
    Entonces el verifica que se carga la pantalla con texto Double Click on Edit Icon to EDIT the Table Row
```

Al tener la información en una tabla, podemos asignar dicha información a una lista del tipo de un objeto específico. Es decir, toda la información de la tabla anterior podríamos agregarla a una lista de la siguiente forma:

Creamos una clase e nuestro paquete “model”, para el ejemplo, la clase se llama “RegistroAutomationDemo” y definimos sus atributos con el mismo nombre de los títulos de los encabezados que le asignamos en la tabla de nuestro feature:

```

public class RegistroAutomationDemo {
    private String first_name ;
    private String last_name ;
    private String address ;
    private String mail ;
    private String phone ;
    private String languages ;
    private String skills ;
    private String conuntry ;
    private String select_conuntry ;
    private String date_birth_year ;
    private String date_birth_month ;
    private String date_birth_day ;
    private String password ;
    private String confirm_password ;
}

```

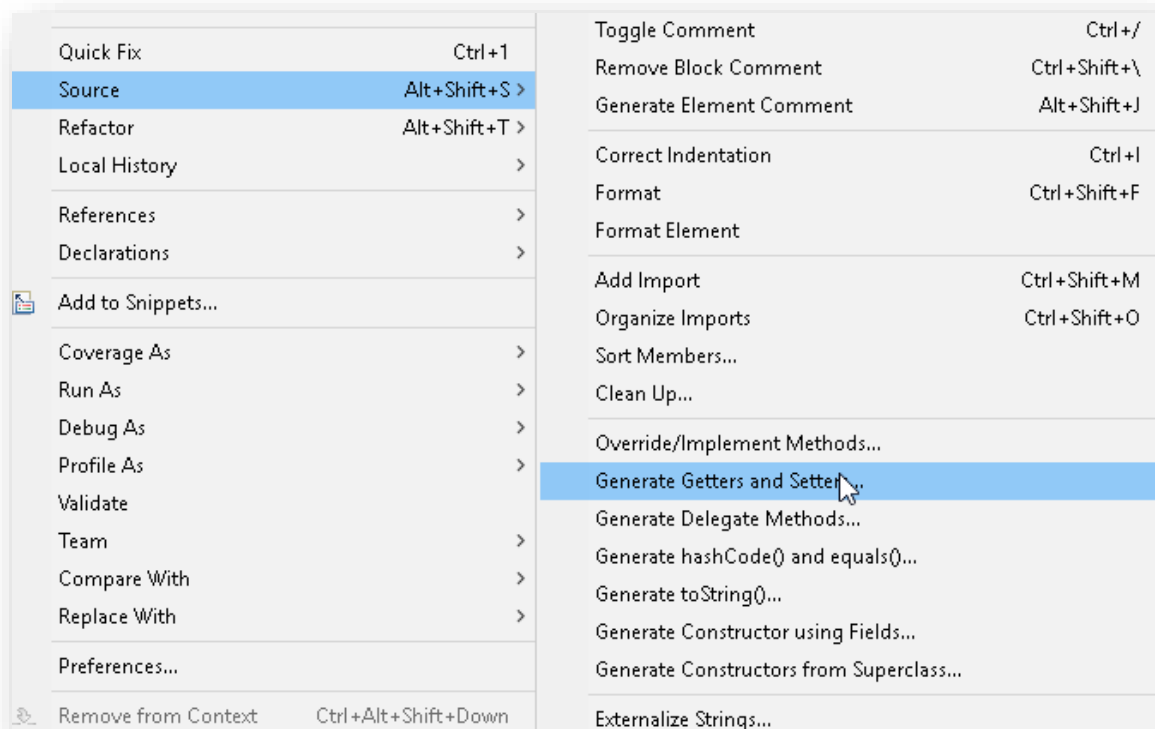
```

first_name | last_name | address |
veilson | Arias | Calle 1 N 1 1 | y
tonces el verifica que se carga la

```

Una vez definidos los atributos, realizamos los getter y los setters de la siguiente forma:

Damos click derecho en la clase y vamos donde dice “Source” y luego a “Generate Getters and Setters”.



Por último, en nuestro StepDefinition agregamos como parámetro una lista que contenga elementos del tipo RegistroAutomationDemo.

```
@Cuando("^el realiza el registro en la página$")
public void elRealizaElRegistroEnLaPágina(List<RegistroAutomationDemo> registroAutomationDemo) {
    carlos.attemptsTo(RealizarElRegistro.EnLaPaginaAutomationDemo(registroAutomationDemo));
}
```

De esta forma tendremos en esta lista toda la información que se tenía en la tabla del feature. Por último, cuando tengas un escenario outline y desees aplicar estos conceptos, podrás hacerlo armando la estructura de la tabla desde el feature como se muestra a continuación:

```
# language:es
#Author: yarias@choucairtesting.com

Característica: Automation Demo Site
  Como usuario
  Quiero realizar el registro en Automation Demo Site
  A través del diligenciamiento del formulario
@casol
Esquema del escenario: Relizar el registro en la pagina Automation Demo Site
  Dado que Carlos quiere acceder al Sitio Automation Demo Site
  Cuando el realiza el registro en la página
  |first_name| last_name| address| mail| phone| languages| skills| conuntry| select_conuntry| date_birth_year|
  |<first_name>| <last_name>| <address>| <mail>| <phone>| <languages>| <skills>| <conuntry>| <select_conuntry>|
  Entonces el verifica que se carga la pantalla con texto Double Click on Edit Icon to EDIT the Table Row

Ejemplos:
|first_name| last_name| address| mail| phone| languages| skills| conuntry| select_conuntry| date_birth_year| |
|Yeison| Arias| Calle 1 N 1 1| yfarias2@correo.co| 3013696970| Arabic| C++| Colombia| Australia| 1981| February|
|Yeison| Arias| Calle 1 N 1 1| yfarias2@correo.co| 3013696970| Arabic| C++| Colombia| Australia| 1981| February|
```

¡Ahora puedes implementarlo en tus proyectos!