**Dynamically Populate Drop Down Values in AEM**

There are so many places in AEM project where we need to dynamically populate

drop down for instance populate the list of pages, populate the country list and so on.

There are two types of use cases which are being used in each and every project.

* Dynamically populate drop down based on datasource (Build on logic)
* Dynamically Populate drop down based on values available statically using ACS Commons Generic list. (No logic)

### Dynamically populate drop down based on datasource (Build on logic)

### Problem Statement: Let suppose I have a requirement of pulling all the children pages of a root path in drop down. Solution:

**Step 1:** Make a drop down widget and create a datasource node under it like this:

### https://lh5.googleusercontent.com/xVqn3AsG8Jqr9WBqz-zKLxQWx0GQxM26OLWGjX-ffoKua5nGvo4EXq-TXvJR-fXo4esEeLJVCaARtGl7MQbA6Fhst1ZKP82vA-Afr4P9yXClm_RezV7-5Xx2TmhbUUuOex9Xzybw

### Step 2: Add a property “resourceType” in datasource which can be mapped to “paths” and “resourceTypes” of the servlet.

### https://lh3.googleusercontent.com/v4ZDGOcKIFR-5sT_qNdT-unIMeusmDhOV9dFcWfDAvD2Gp5IkaxR04P7cgZXcwv5ZMHYKz89fXxMdjmre74n-u0h7QCcAC-oeo-ADPf_kbCy2BRlnEE0NmXplyIaeYBul-ggaqbz

**Step3:** Write a java class mapped to datasource and write a logic to dynamically populate values in the drop down.

**sling:resourceType** of datasource = **sling.servlet.paths** of servlet

OR

**sling:resourceType** of datasource = **sling.servlet.resourceTypes** of servlet

**Problem statement:** In a project there can be situations where in a component A, you want to pull drop down from Root X and in the component B, you want to pull drop down from Root Y.

**Solution:** you can add a property may be rootPath in datasource node and fetch the value of that rootPath in the servlet and show values.

**Problem statement:** There can be a situation that in component A you want to pull list of pages but in component B you want to pull list of countries.

**Solution:** You can add a property may be **selector** as “pageListing” for fetching all the pages or “countryList” to fetch all the countries list and in the servlet call different functions to dynamically populate different values.

|  |
| --- |
| https://lh4.googleusercontent.com/8nZdXwxmzd3ulj5FQNoTK209lyLvhUGV9gj6n90K5KL9QHZUexB733Fmz5YrfgHRZKuPi8_t2ucoM3LCfeVISkMwYI8LfMhxyStjxEQGY5igbwIvctTNBDbDrL-GwmztgnNCrD0Y |

The complete code for this can be found in the below attached doc.

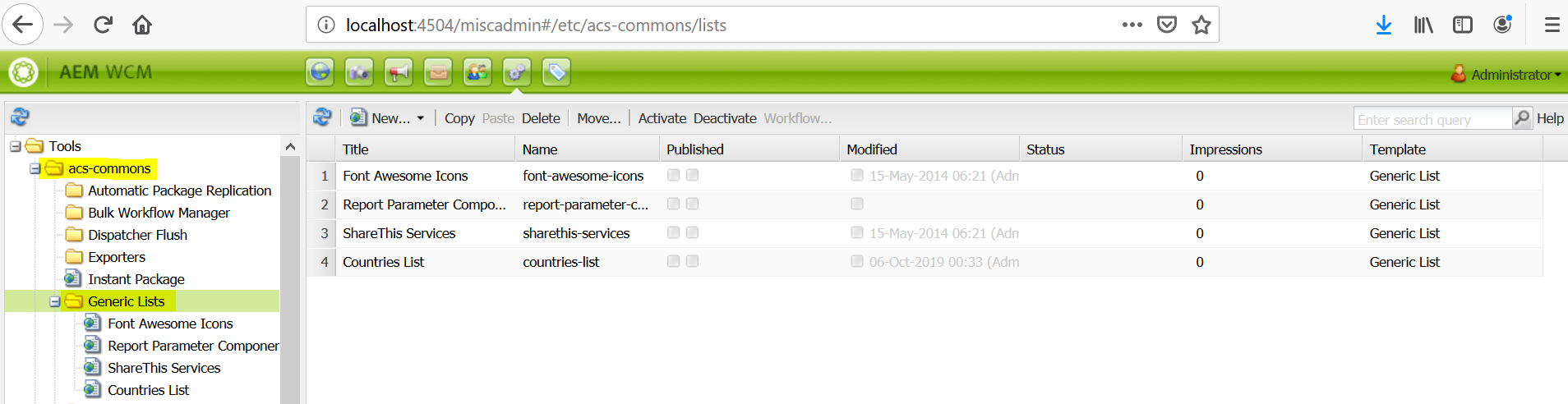


### Dynamically Populate drop down based on values available statically using ACS Commons Generic list. (No logic)

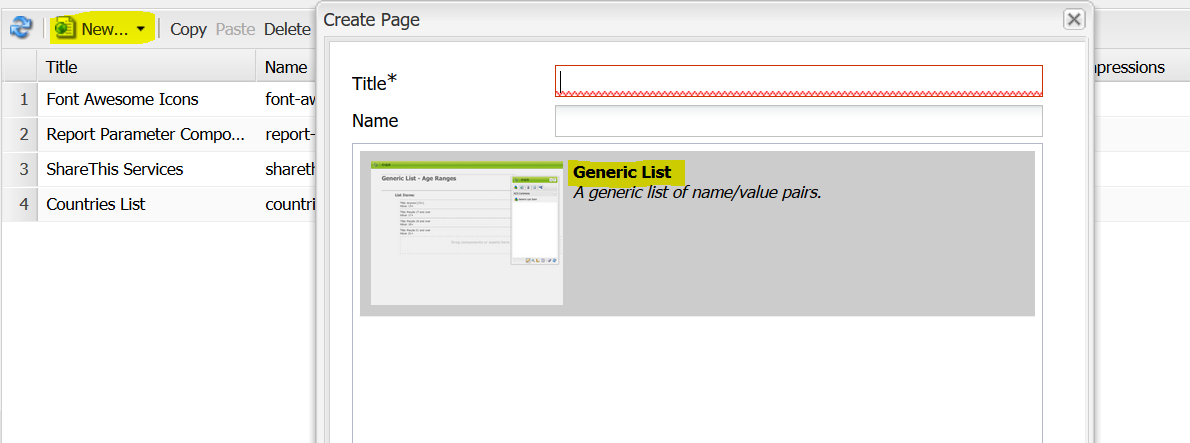
Using generic lists, if you have some values and want to populate these values dynamically in drop down you can do. The benefit is if one day author want to add one more value, he can add one more value by himself.

**Prerequisites**: Your AEM instance must have acs commons package.

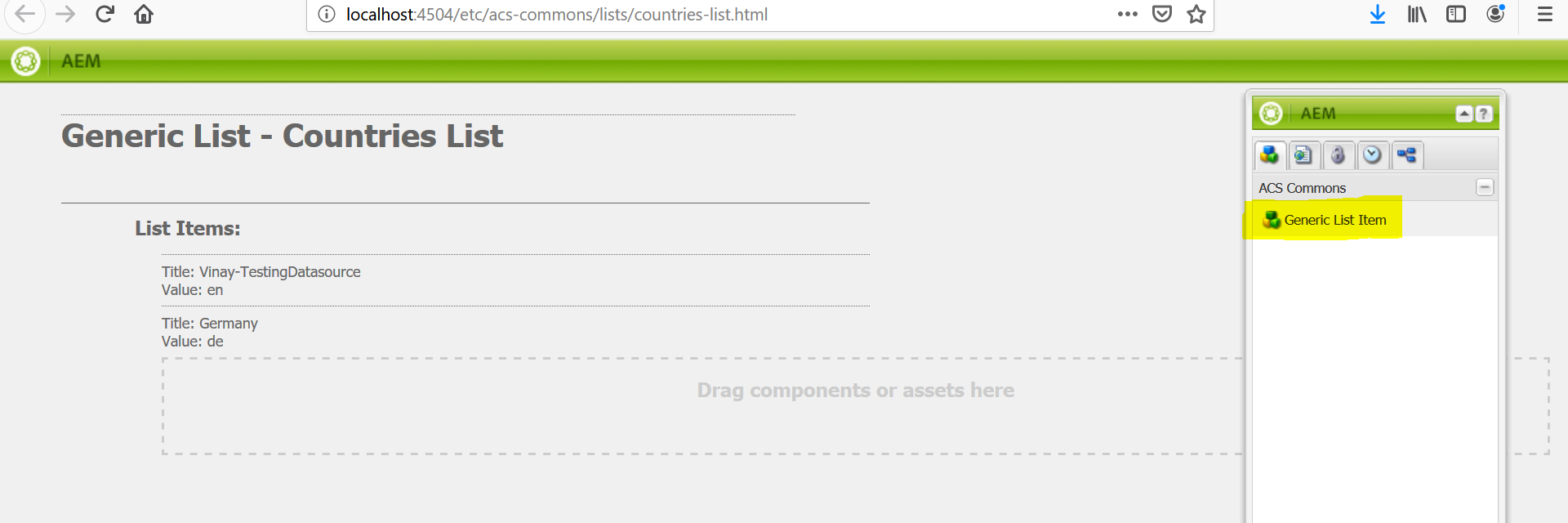
**Step1**: Go to [misadmin](http://localhost:4502/miscadmin#/etc/acs-commons/lists) (<http://localhost:4504/miscadmin>) and select Generic Lists under acs commons.



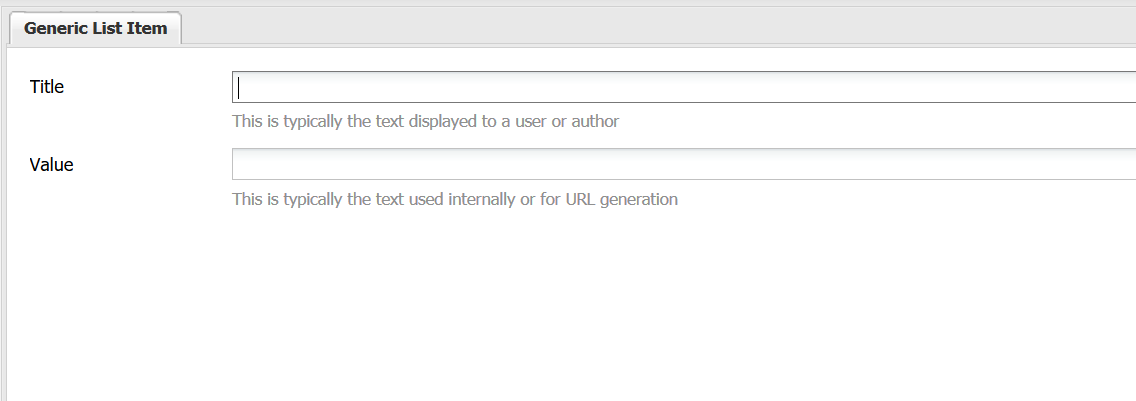
**Step2**: Click on New.  
**Step3**: Select generic list template and click on Create (provide title and name for this list).

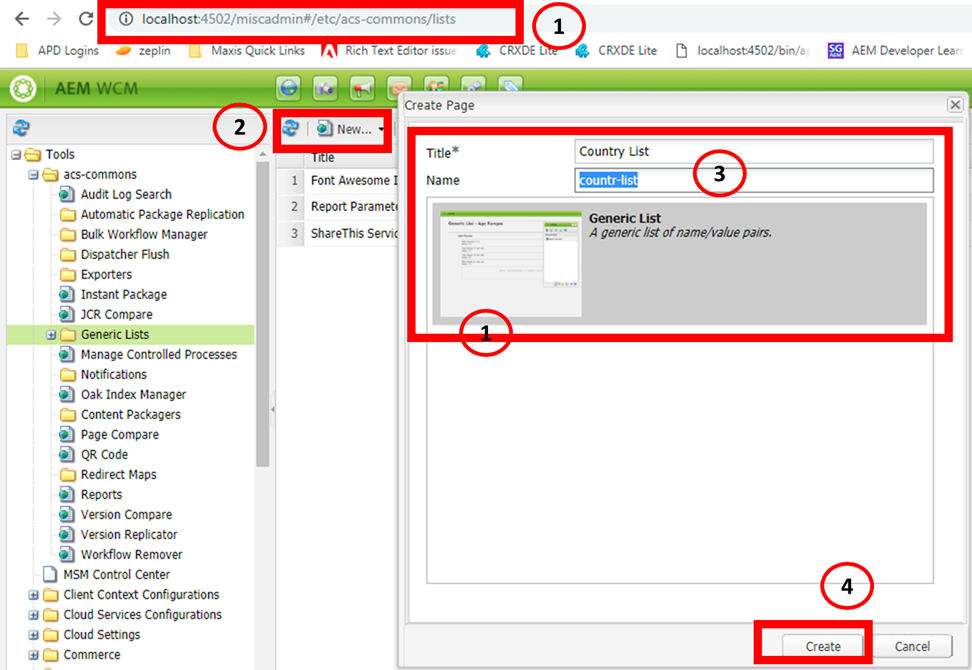


**Step4**:Open the page and drag and drop a component named Generic List Item



To add new entry, drag and drop the Generic List Item component again and provide the values.





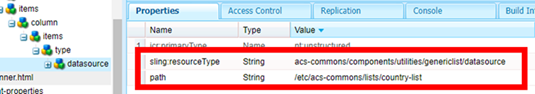
**Step5**: Add the Title and Value in the Generic List Item component.



**Title**: The title which will be visible to the author in drop down.

**Value**: Value is what gets stored corresponding to what text you selected from drop down. Now values are ready to populate in drop down component.

* Make a drop down widget and create a datasource node.
* Add following values in the datasource node.



**1.** sling:resourceType="acs-commons/components/utilities/genericlist/datasource"   
**2.** path="/etc/acs-commons/lists/country-list" //This path is the page path you created earlier and added generic list item.

Final result in the component dialog looks like below

