19/03/2025 - TASKS

Table of Contents

---

1. What is DAM and Why We Use It?

Digital Asset Management (DAM) in AEM

DAM (**Digital Asset Management**) is used to store, organize, retrieve, and manage digital assets like images, videos, and documents within Adobe Experience Manager (AEM).

Why Do We Use DAM?

- Centralized storage for digital assets
- Metadata management for organizing assets
- Image renditions for automatic optimization
- Version control for tracking changes
- Workflow integration for approvals before publishing

Location in AEM

All assets are stored inside:

/content/dam

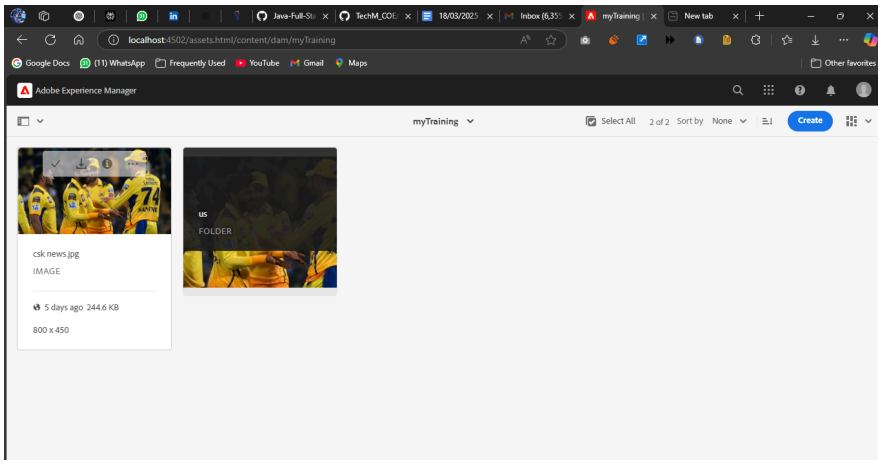
2. Create Folder and Upload Images

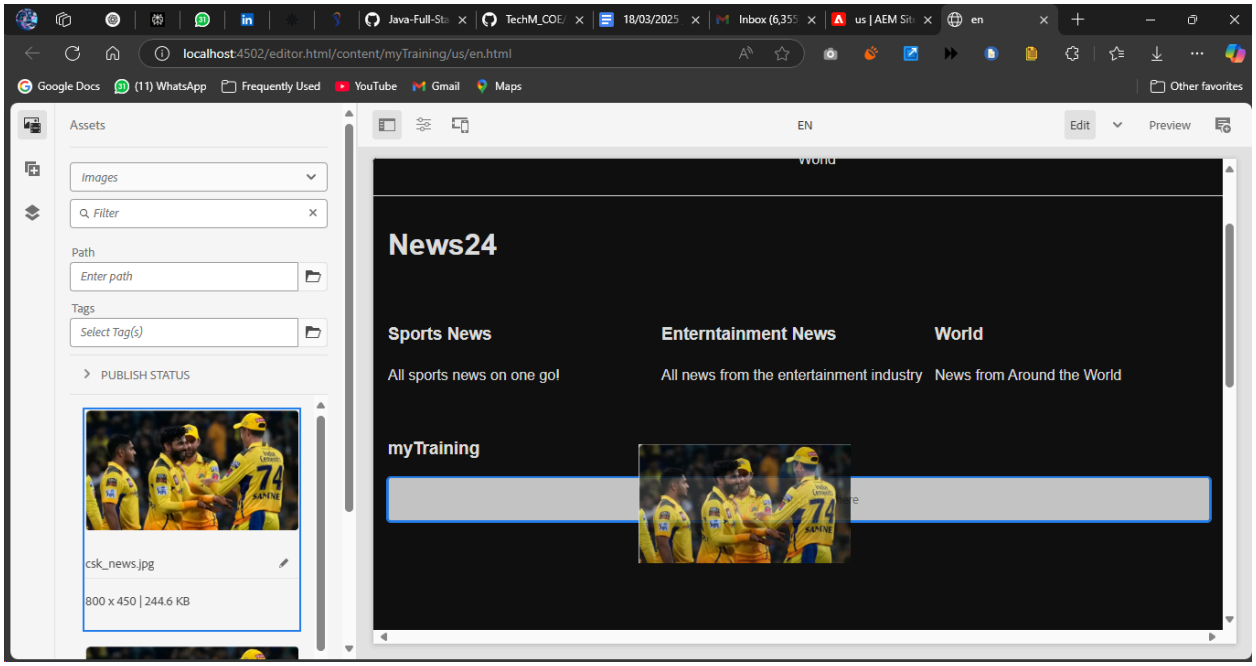
Steps to Create Folder & Upload Images

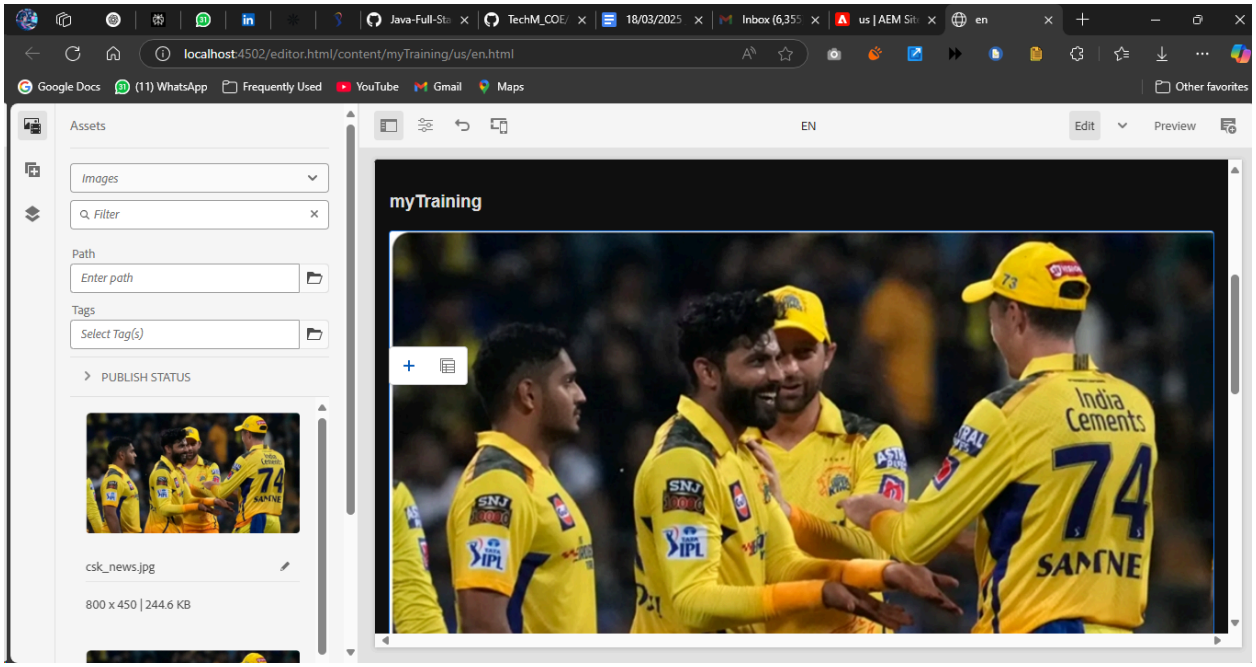
Open AEM **DAM Console**

http://localhost:4502/assets.html/content/dam

1. Create a Folder
    - Click **Create** → **Folder**
    - Enter **Folder Name**: `myTraining`
    - Inside `myTraining`, create another folder: `us/en-us`
2. Upload Images
    - Open `us/en-us` folder
    - Click **Upload** → Select 2 images from your system
    - Click **Save**
3. Add Image to a Page







Open **Sites Console**:

http://localhost:4502/sites.html/content/myTraining

- Edit your page and drag **Image Component** from the Side Panel
- Select one of the uploaded images and **Save the Page**

3. Check Renditions in AEM

What are Renditions?

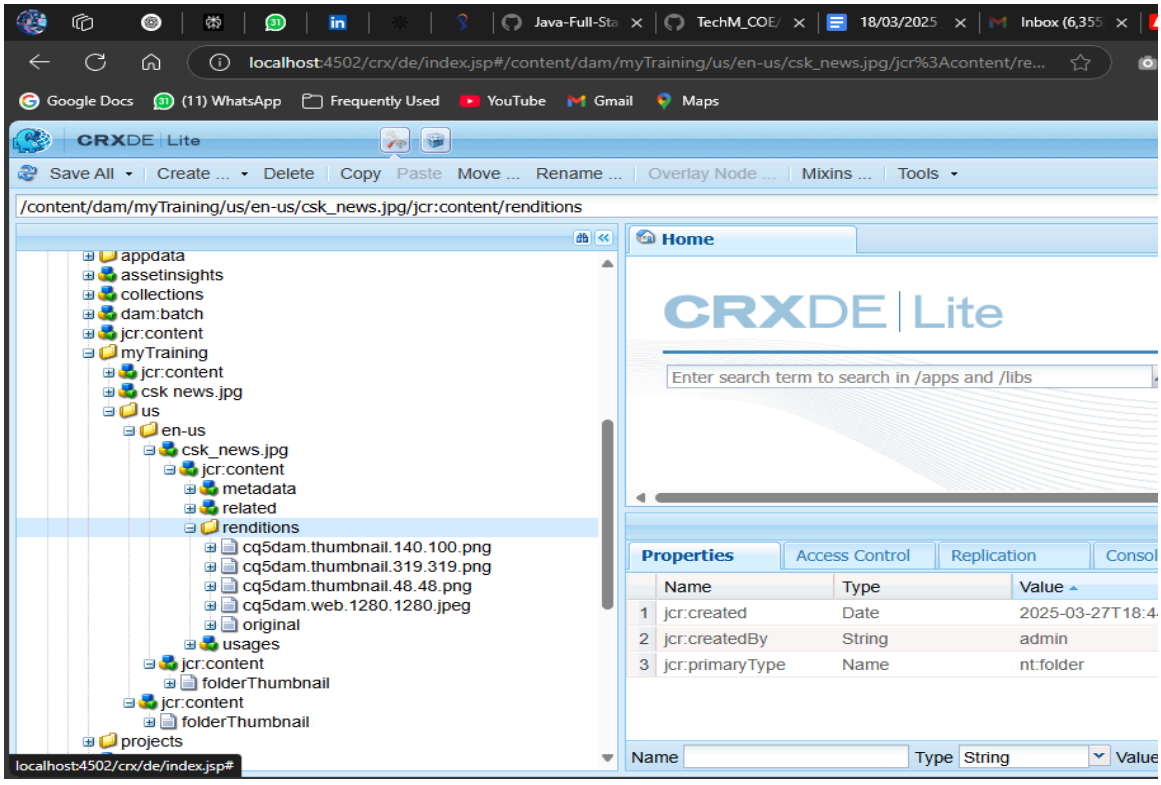
Renditions are different versions of an image automatically created when you upload an asset to DAM. These help optimize images for different screen sizes and devices.

Steps to Check Renditions

Open **DAM Console**:

http://localhost:4502/assets.html/content/dam/myTraining/us/en-us

1. Click on an **uploaded image**
2. Click **View Properties → Renditions Tab**
3. You will see multiple renditions such as:
    - `cq5dam.thumbnail.48.48.png` (48x48 px thumbnail)
    - `cq5dam.web.1280.1280.jpeg` (Web-optimized version)



4. Modify HelloWorld Component

Steps to Add FirstName & LastName Fields

Open **CRXDE Lite**

http://localhost:4502/crx/de/

Navigate to:

/apps/myTraining/components/helloworld

1. Update `cq:dialog` to add input fields:

```
<firstName
  jcr:primaryType="nt:unstructured"
  sling:resourceType="granite/ui/components/coral/foundation/form/textfield"
  fieldLabel="First Name"
  name="./firstName"/>

<lastName
  jcr:primaryType="nt:unstructured"
  sling:resourceType="granite/ui/components/coral/foundation/form/textfield"
  fieldLabel="Last Name"
  name="./lastName"/>
```

4. Update `helloWorld.html` to display values:

```
<div>
  <h2>Hello World Component</h2>
  <p>First Name: ${properties.firstName}</p>
  <p>Last Name: ${properties.lastName}</p>
</div>
```

5. Save and Deploy the Component

5. Use @ValueMapValue in HelloWorldModel

Open Java Model File:

/core/src/main/java/com/myTraining/core/models/HelloWorldModel.java

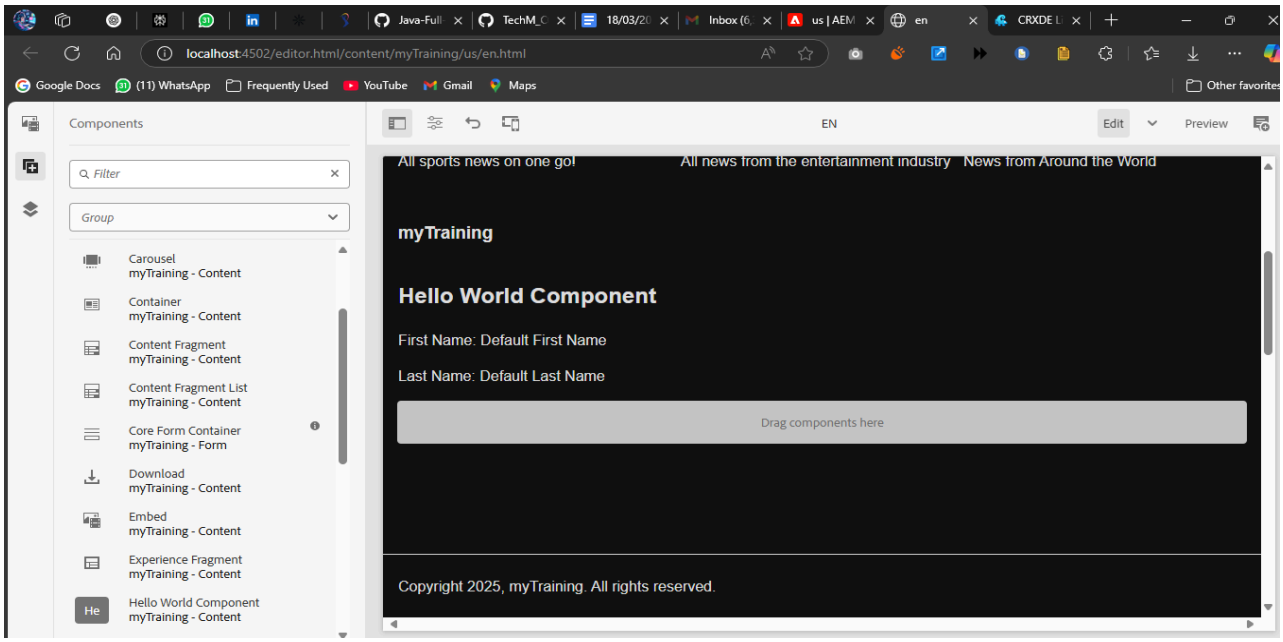
1. Update `HelloWorldModel.java` to fetch values:

```
package com.myTraining.core.models;
import org.apache.sling.api.resource.Resource;
import org.apache.sling.models.annotations.DefaultInjectionStrategy;
import org.apache.sling.models.annotations.Model;
import org.apache.sling.models.annotations.injectorspecific.ValueMapValue;

@Model(adaptables = Resource.class, defaultInjectionStrategy =
DefaultInjectionStrategy.OPTIONAL)
public class HelloWorldModel {

   @ValueMapValue
   private String firstName;

   @ValueMapValue
   private String lastName;

   public String getFirstName() {
      return firstName;
   }

   public String getLastName() {
      return lastName;
   }
}
```

3. Update `helloWorld.html` to use Sling Model:

```
<div data-sly-use.model="com.myTraining.core.models.HelloWorldModel">
  <h2>Hello World Component</h2>
  <p>First Name: ${model.firstName}</p>
  <p>Last Name: ${model.lastName}</p>
</div>
```

4. Save & Build the Project

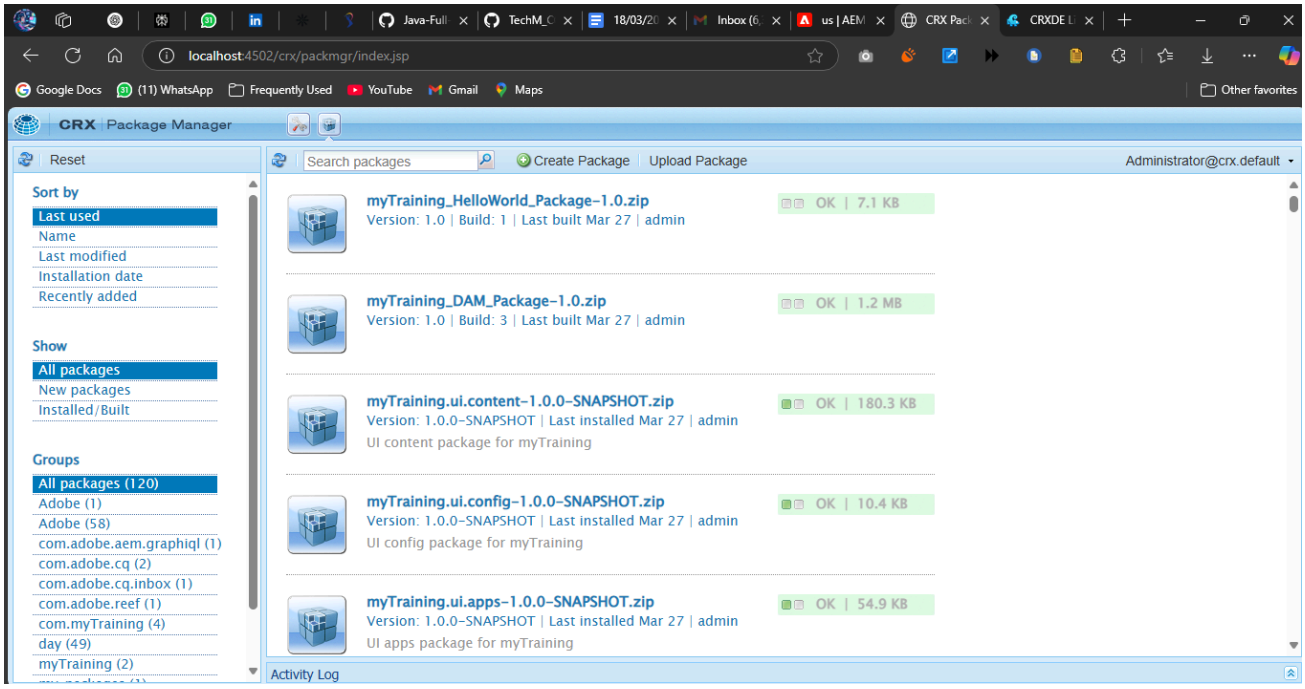


6. Create AEM Packages Using Package Manager

Open **Package Manager**

http://localhost:4502/crx/packmgr/index.jsp

1. Create DAM Package
    - Name: `myTraining_DAM_Package`
    - Group: `myTraining`
    - Path: `/content/dam/myTraining/us/en-us`
    - Click **Build & Download**
2. Create HelloWorld Package
    - Name: `myTraining_HelloWorld_Package`
    - Group: `myTraining`
    - Path: `/apps/myTraining/components/helloworld`
    - Click **Build & Download**



7. Configure Replication Agent & Publish Page

Open **Replication Agents Console**

http://localhost:4502/etc/replication/agents.author.html

1. Edit **Default Replication Agent**
    - **Name:** Default Agent
    - **Transport URI:** `http://localhost:4503/bin/receive`
    - **User:** `admin`
    - **Password:** `admin`
    - Click **Test Connection** → Should show **Successful**
2. Publish Page to 4503

Open **Sites Console**:

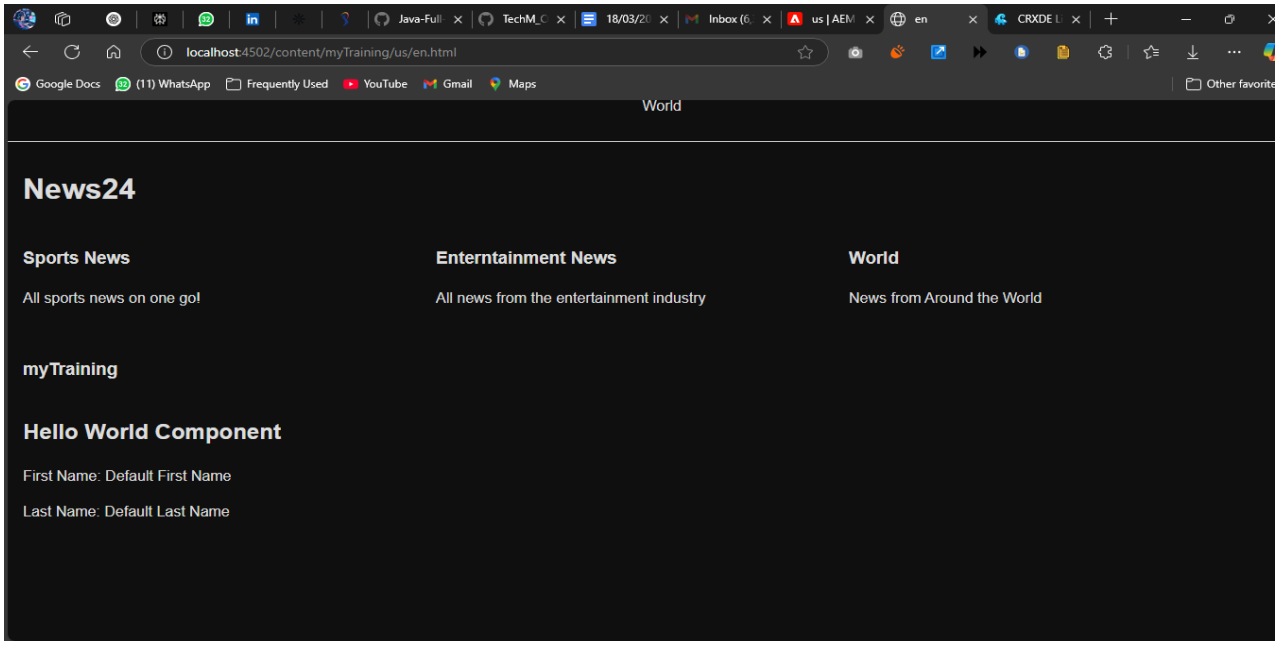
http://localhost:4502/sites.html/content

- Select your page → Click **Publish**

Verify the page in the **Publish Instance (4503)**:

http://localhost:4503/content/myTraining/us/en.html

AUTHOR INSTANCE:



PUBLISH INSTANCE: