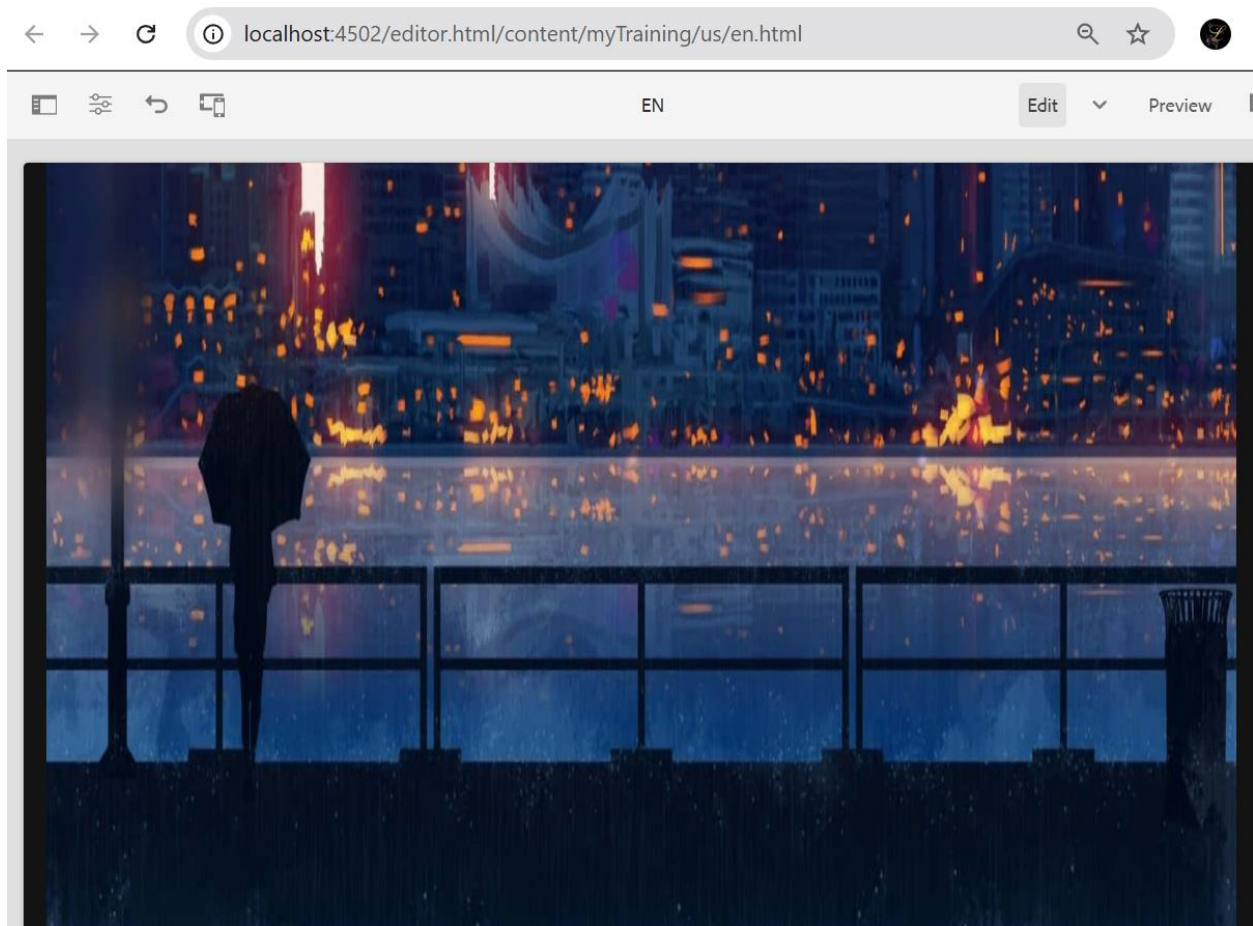


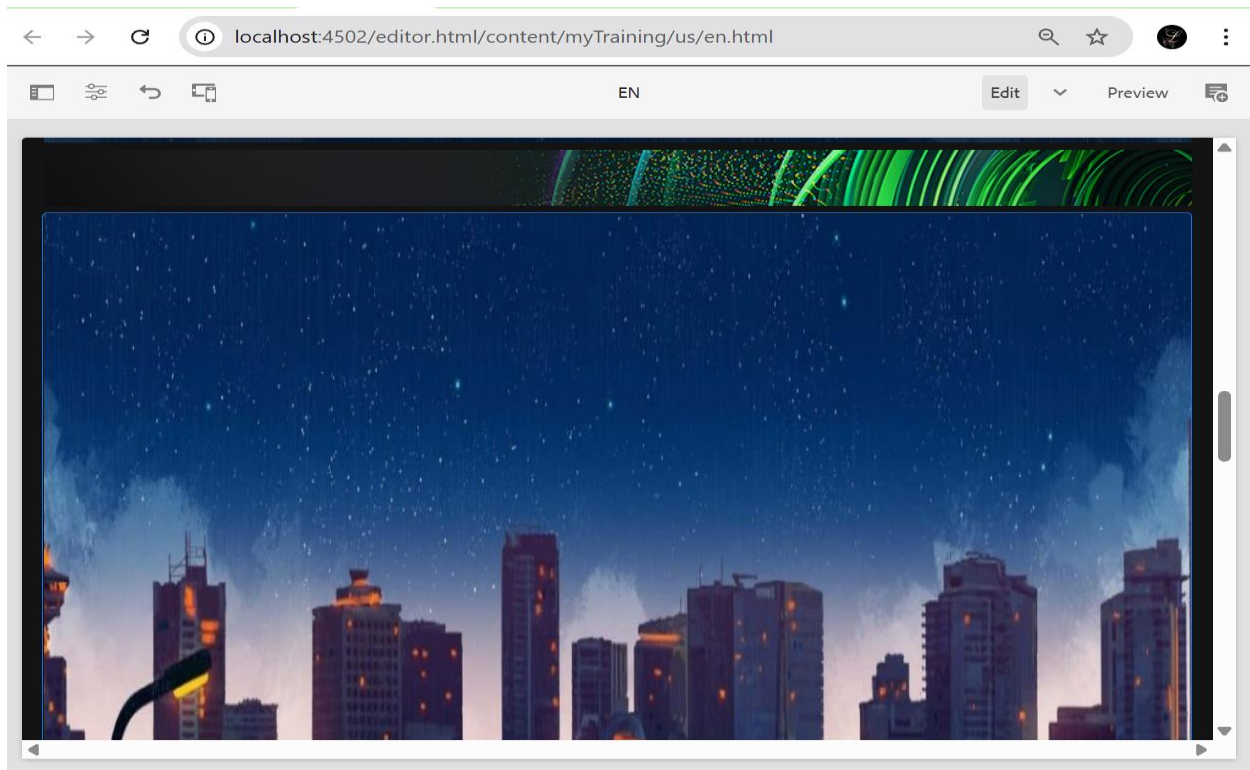
1. What is DAM and why do we use it?

Digital Asset Management (DAM) is a centralized system used to store, organize, retrieve, and manage digital assets like images, videos, and documents. AEM's DAM allows efficient asset management, versioning, metadata handling, and easy integration into websites and applications. It enhances collaboration, ensures consistency, and optimizes content delivery across platforms.

2. Create one folder inside our project folder and follow the path:

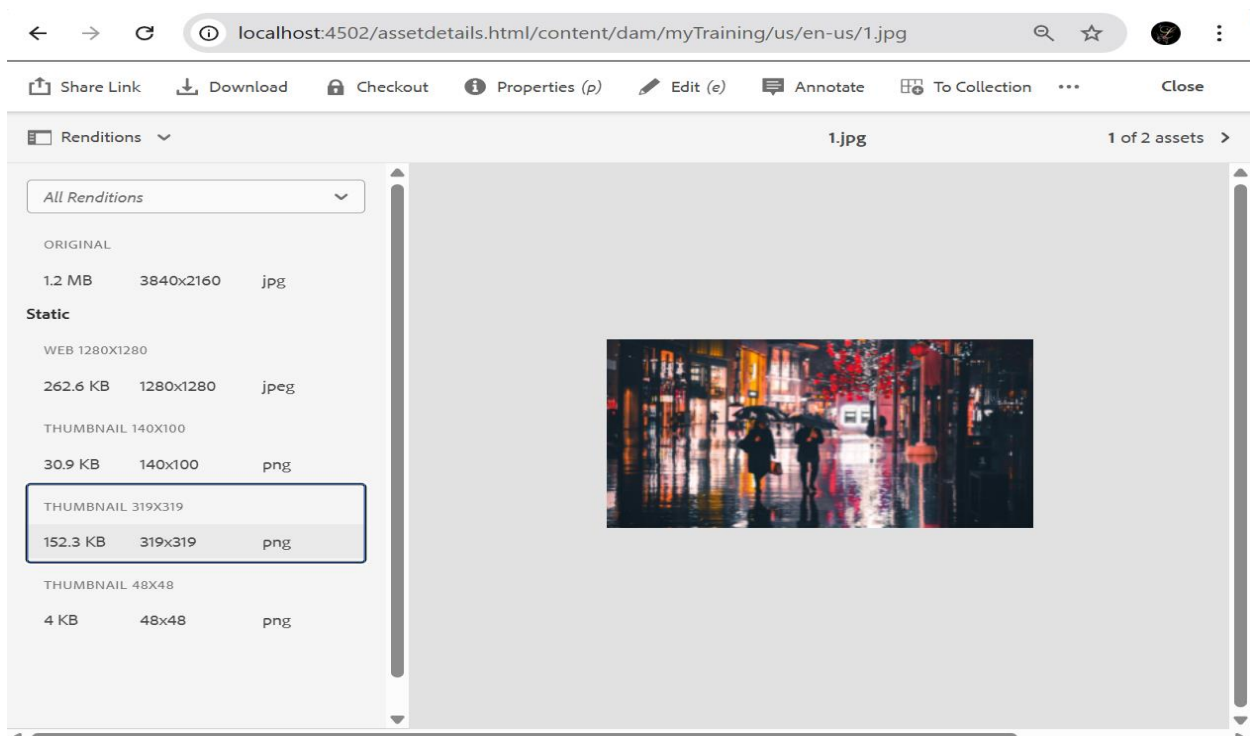
“/content/dam/myTraining/us/en-us” and upload 2 images and author those images on page using image component as we see today.





3. What is renditions? Check the renditions for 2 images which we had uploaded.

Renditions are different versions of an uploaded image generated automatically in AEM DAM. These versions allow for optimized display across different devices and screen resolutions.



4. Add two fields in hello world component with name **FirstName** and **Last Name** and print the values on **hello world.html** component using properties.

The screenshot shows the CRXDE Lite interface. On the left, a tree view displays the component structure. The 'hello world' component is selected, and its 'content' node is expanded, showing 'items' and 'column' nodes. The 'items' node is further expanded, showing 'items' and 'text' nodes. The 'text' node is selected, and its 'properties' tab is active. The 'properties' table shows the following data:

Name	Type	Value	Protected	Mandatory	Multiple	Auto Created	
1 fieldLabel	String	First Name	ntu...	true	true	false	true
2 jcr:primaryType	Name	ntu...	true	true	false	true	true
3 name	String	/fir...	false	false	false	false	false
4 sling:resourceType	String	gra...	false	false	false	false	false

Below the table, there is a form with fields for 'Name', 'Type', 'Value', and 'Multi'. The 'Name' field is empty, 'Type' is 'String', 'Value' is empty, and 'Multi' is 'Add'.

The screenshot shows the 'hello world.html' component in the CRXDE Lite editor. The component is titled 'HELLO WORLD'. The editor displays the following content:

myTraining

Search

Hello World

Hello, Luna Wolf

Hello World Component

Text property:
Welcome to the Twilight Gala!

Model message:
Hello User!

Resource type is: myTraining/components/helloworld
Current page is: /content/myTraining/hello-world

Drag components here

5. Give try using @ValueMapValue annotation on HelloWorldModel sling java model “com.myTraining.core.models.HelloWorldModel”.

```
public class HelloWorldModel {  
    protected String resourceType;  
    private static final Logger LOG = LoggerFactory.getLogger(HelloWorldModel.class); 1 usage  
  
    @ValueMapValue 1 usage  
    @Default(values = "User") // Default placeholder if name is not provided  
    private String userName;  
  
    @Self 2 usages  
    Resource resource;  
  
    @SlingObject 2 usages  
    private ResourceResolver resourceResolver;  
  
    @SlingObject 1 usage  
    private Resource currentResource;  
  
    private static String message; 2 usages  
  
    @PostConstruct no usages  
    protected void init() {  
  
        resourceResolver=resource.getResourceResolver();  
  
        LOG.info( "-----hello world get path---"+resource.getPath());  
  
        PageManager pageManager = resourceResolver.adaptTo(PageManager.class);  
        String currentPagePath = Optional.ofNullable(pageManager) Optional<PageManager>  
    }  
}
```

6. Why we are using package manager and jar create package for the hello world component and 2 images what we have uploaded There should be 2 packages 1 for dam which is images and 2 one is for helloworld component Configure replication agent and publish the page from author to publish env(4502(author)—>4503(publish)) this is my assignment give me everything properly.

Package Manager allows exporting and importing content between AEM environments efficiently. It ensures safe deployment and rollback of configurations, components, and assets.

← → ↻ ⓘ localhost:4502/crx/packmgr/index.jsp#/etc/packages/com.myTraining/dam-image-... 🔍 ☆ ⚙

CRX Package Manager


Reset | **dam** | Create Package | Upload Package | Administrator@crx.default

Sort by

- Last used
- Name
- Last modified
- Installation date
- Recently added

Show

- All packages
- New packages
- Installed/Built



dam-image-package-10.0.0.zip
Version: 10.0.0 | Build: 1 | Last built Mar 19 | admin

Edit | Build | Install | Download | More

Package: dam-image-package

Download: dam-image-package-10.0.0.zip (3.5 MB)

Group: com.myTraining

Filters: /content/dam/myTraining/us/en-us

← → ↻ ⓘ localhost:4502/crx/packmgr/index.jsp#/etc/packages/com.myTraining/dam-image-... 🔍 ☆ ⚙

CRX Package Manager

Reset | **hello** | Create Package | Upload Package | Administrator@crx.default

Sort by


- Last used
- Name
- Last modified
- Installation date
- Recently added

Show

- All packages
- New packages
- Installed/Built

Groups

- All packages (122)
- Adobe (1)
- Adobe (58)
- com.adobe.aem.graphicl (1)
- com.adobe.cq (2)
- com.adobe.cq.inbox (1)
- com.adobe.reef (1)
- com.myTraining (8)
- day (49)
- my_packages (1)
- slings



hello-world-10.0.0.zip
Version: 10.0.0 | Build: 1 | Last built Mar 19 | admin

OK | 7.5 KB