

## AEM TASK -3

### 1. Create a New Component for News (Single News Component)

#### Steps:

##### 1. Navigate to AEM CRXDE Lite:

- a. Open <http://localhost:4502/crx/de>
- b. Go to /apps/your-project/components/
- c. Create a new component folder: /apps/your-project/components/news
- d. Inside, create the following files:
  - i. news.html
  - ii. news.css (for styling)
  - iii. \_cq\_dialog.xml (for dialog configuration)

##### 2. Create the Sling Model (NewsModel.java)

- a. Path: /apps/your-project/core/src/main/java/com/yourproject/models/NewsModel.java

```
@Model(adaptables = Resource.class, defaultInjectionStrategy =
DefaultInjectionStrategy.OPTIONAL)
public class NewsModel {
    @Inject
    private String title;

    @Inject
    private String detail;

    @Inject
    private String publishedDate;

    @Inject
    private String source;

    public String getTitle() {
        return title;
    }
}
```

```
}

public String getDetail() {
    return detail;
}

public String getPublishedDate() {
    return publishedDate;
}

public String getSource() {
    return source;
}
}
```

### 3. Modify news.html

```
<div class="cop-news-component">
    <h2>${newsModel.title}</h2>
    <p>${newsModel.detail}</p>
    <span class="published-date">${newsModel.publishedDate}</span>
    <span class="news-source">${newsModel.source}</span>
</div>
```

### 4. Create news.css for Styling

```
.cop-news-component h2 {
    color: green;
}
.cop-news-component p {
    color: yellow;
}
.cop-news-component .published-date {
    color: black;
}
```

## 2. Create a Multi-field Component for Multiple News

### Steps:

#### 1. Navigate to CRXDE Lite:

- a. Create a new component: /apps/your-project/components/multi-news
- b. Add multi-news.html and \_cq\_dialog.xml

#### 2. Create the Sling Model (MultiNewsModel.java)

```
@Model(adaptables = Resource.class, defaultInjectionStrategy =
DefaultInjectionStrategy.OPTIONAL)
public class MultiNewsModel {
    @ChildResource
    private List<NewsItem> newsList;

    public List<NewsItem> getNewsList() {
        return newsList;
    }

    @Model(adaptables = Resource.class)
    public static class NewsItem {
        @Inject
        private String title;

        @Inject
        private String source;

        public String getTitle() {
            return title;
        }

        public String getSource() {
            return source;
        }
    }
}
```

### 3. Modify multi-news.html

```
<div class="multi-news">
  <ul>
    <sly data-sly-list.newsItem="${multiNewsModel.newsList}">
      <li>
        <h3>${newsItem.title}</h3>
        <span>${newsItem.source}</span>
      </li>
    </sly>
  </ul>
</div>
```

### 3. Create ClientLibs for News Component

#### Steps:

1. Navigate to `/apps/your-project/clientlibs/news`
  - a. Create `css.txt`
  - b. Add `news.css` inside `css` folder
  - c. Modify `css.txt`

`news.css`

2. **Modify `news.css`** (as created in step 1.4)
3. **Link ClientLibs in `news.html`**

```
<sly data-sly-
use.clientlib="/libs/granite/sightly/templates/clientlib.html" />
<sly data-sly-call="${clientlib.css @ categories='yourproject.news'}"
/>
```

### 4. Add Custom Style to Component

1. **Modify `_cq_dialog.xml`**

```
<items jcr:primaryType="cq:WidgetCollection">
    <cq:style cq:styleClass="cop-news-component"/>
</items>
```

## 5. Create a Base Page Component & Metadata for OpenGraph (OG)

### Steps:

1. Navigate to `/apps/your-project/components/page`
2. Create `page.html` and `metadata.html`
3. Modify `metadata.html`

```
<meta property="og:title" content="${pageProperties.ogTitle}" />
<meta property="og:description"
content="${pageProperties.ogDescription}" />
<meta property="og:image" content="${pageProperties.ogImagePath}" />
```

4. Link `metadata.html` in `page.html`

```
<sly data-sly-include="metadata.html" />
```

## 6. Create Global Page Properties

### Steps:

1. Modify `_cq_dialog.xml` of Page Component

```
<jcr:root xmlns:cq="http://www.day.com/jcr/cq/1.0"
    jcr:primaryType="cq:Dialog"
    title="Global Properties"
    xtype="dialog">
    <items jcr:primaryType="cq:WidgetCollection">
        <textfield
            fieldLabel="OG Title"
            name="./ogTitle"/>
```

```

        <textfield
            fieldLabel="OG Description"
            name="./ogDescription"/>
        <textfield
            fieldLabel="OG Image Path"
            name="./ogImagePath"/>
    </items>
</jcr:root>

```

## 7. What is extraClientlibs?

- **extraClientlibs** is used to include additional client-side JavaScript/CSS that is not part of the main **clientlibs** category.
- It's useful for including extra styles/scripts in specific components.

### *Steps to Implement extraClientlibs in Multi-Field Component*

#### 1. Modify **\_cq\_dialog.xml** of Multi-News Component

```

<extraClientlibs jcr:primaryType="nt:unstructured"
    categories="[yourproject.extra-news]" />

```

#### 2. Create **/apps/your-project/clientlibs/extra-news**

- Add **js.txt**

**extra-news.js**

- Add **extra-news.js**

```

console.log("Extra clientlibs for Multi-News Component loaded!");

```

#### 3. Modify **multi-news.html**

```

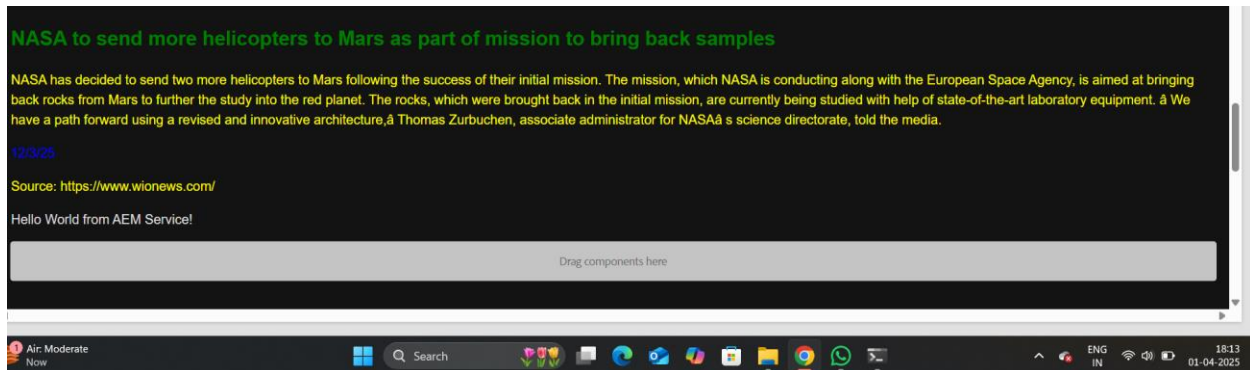
<sly data-sly-
    use.clientlib="/libs/granite/sightly/templates/clientlib.html" />
<sly data-sly-call="${clientlib.js @ categories='yourproject.extra-

```

```
news' }" />
```

## Final Steps

1. Deploy the code using **Maven** (`mvn clean install -PautoInstallPackage`).
2. Open AEM (<http://localhost:4502/>), add the **news component** and **multi-news component** to a page.
3. Ensure styling and metadata properties work correctly.



## News Component



News Title

NASA to send more helicopters to Mars as part of mission to bring back samples

News Detail

NASA has decided to send two more helicopters to Mars following the success of their initial mission. The mission, which NASA is conducting along with the European Space Agency, is aimed at bringing back rocks from Mars to further the study into the red planet. The rocks, which were brought back in the initial mission, are currently being studied with help of state-of-the-art

Published Date

12/3/25

Source

<https://www.wionews.com/>

Cancel

Done