**What is the purpose of the core module in AEM?**

The core module in AEM is responsible for the backend logic. It contains the Java code for OSGi services, Sling models, and servlets which process business logic and interact with the JCR repository.

**What kind of files and code can be found in the core folder?**

The core folder typically contains backend logic, OSGi components, Sling models(to fetch and structure the data), and Servlets.

**Explain the role of ui.apps in AEM projects.**

The ui.apps module is responsible for the frontend of the AEM project. It consists of Components, Client Libraries(styling lib) and Content Definitions(template that defines how the content to be structured in AEM)

**How are components structured in the ui.apps folder?**

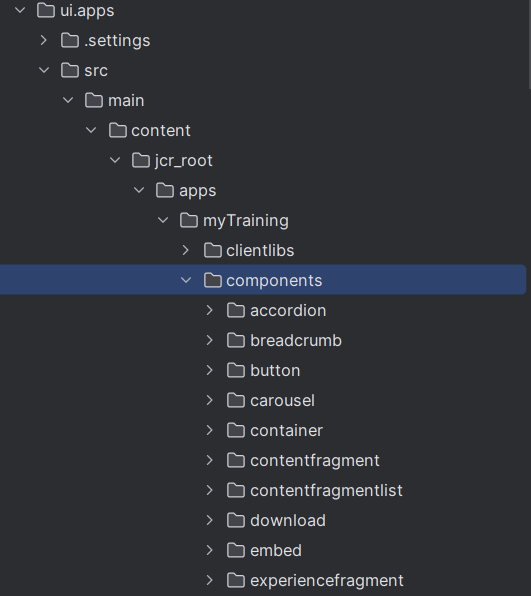
In ui.apps folder, components are structured as folders under: ui.apps/src/main/content/jcr\_root/apps/myTraining/components

Each component typically contains:

component.html – Defines the component's HTML structure.

\_cq\_dialog.xml – Configures the authoring dialog.

.content.xml – Stores component metadata.

****

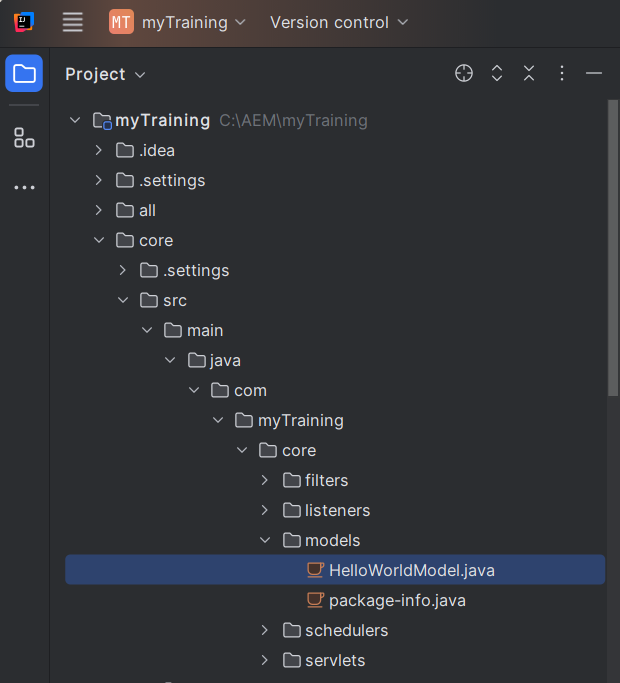
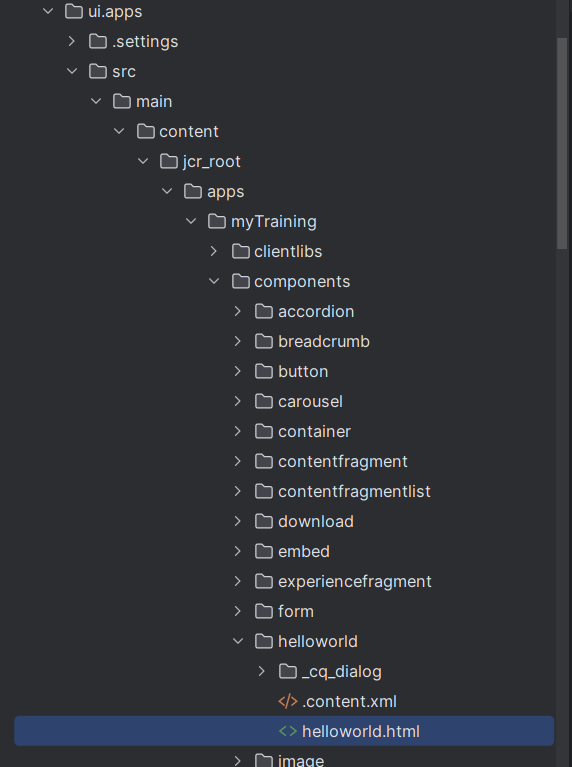
**Hello World Component:**

* **Where is the Hello World component located in both core and ui.apps?**

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

In ui.apps- ui.apps/src/main/content/jcr\_root/apps/myTraining/components/helloworld/helloworld.html

**core ui.apps**

* **Explain the Java class (in core) for the Hello World component.**

This Java class is an AEM Sling model that retrieves the resource type and current page details.

It initializes a message in the @PostConstruct method, and provides it via the getMessage() method for frontend display.

* **How does the HTL script work in ui.apps for Hello World?**

HTL script work is to render the data from the Sling model.

* **How are properties and dialogs defined for this component?**

Properties and dialogs are defined in .content.xml for component settings and \_cq\_dialog.xml for author input fields using Granite UI and stored in JCR.

**What are the different types of AEM modules (core, ui.apps, ui.content, etc.)?**

core – Backend logic (OSGi services, Sling models, Servlets).

ui.apps – Frontend (components, dialogs, client libraries).

ui.content – Stores site-specific content (pages, assets).

ui.config – Manages OSGi configurations.

all – Bundles all modules for deployment.

dispatcher – Handles caching and security (AEM Cloud).

**How does Maven build these modules?**

Maven builds each module separately but manages them together through a parent POM. It compiles Java (core), packages components (ui.apps), and creates a deployable AEM package (.zip).

**Explain the build lifecycle of Maven in the context of AEM**.

Maven phases includes:

* Clean – Removes old build files.
* Compile – Builds Java classes.
* Package – Creates JARs for core and ZIPs for AEM.
* Install – Installs the package for local testing.
* Deploy – Deploys to AEM.

**How are dependencies managed in pom.xml?**

Dependencies are defined in the pom.xml file,Maven automatically downloads and manages them.

**Why is Maven used instead of other build tools?**

Maven is widely used in AEM because:

* Automated Dependency Management
* Provides structured build phases.
* Easily manages core, ui.apps, and other modules.

**What advantages does Maven offer for AEM development?**

* Standardized build and deployment.
* Automatic dependency resolution.
* Supports multi-module AEM projects.

**How does Maven help in managing dependencies and plugins in AEM projects?**

Maven defines dependencies in POM files, downloads them automatically, and uses plugins like:

* maven-bundle-plugin (for OSGi bundles).
* content-package-maven-plugin (for AEM packages).

**What does** mvn clean install **do in an AEM project?**

**mvn clean install** in an AEM project cleans old builds, compiles Java code, packages AEM components, and installs the package locally.

**How to deploy packages directly to AEM using Maven commands?**

Command to deploy packages directly to AEM using Maven commands-

mvn clean install -PautoInstallPackage

**Explain the purpose of different Maven profiles in AEM (autoInstallPackage, autoInstallBundle).**

autoInstallPackage (mvn clean install -PautoInstallPackage)– Deploys full AEM package.

autoInstallBundle(mvn clean install -PautoInstallBundle) – Deploys only the core JAR (OSGi bundle).

**What is the purpose of dumplibs in AEM?**

dumplibs is used to debug client libraries by listing CSS/JS files and dependencies.

**How can you view client libraries using dumplibs?**

You can view client libraries in AEM using dumplibs by visiting:  
http://localhost:4502/libs/granite/ui/content/dumplibs.html

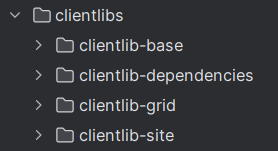
**Explain how client libraries are structured in AEM.**

In AEM, client libraries (clientlibs) are structured as folders under:

ui.apps/src/main/content/jcr\_root/apps/myTraining/clientlibs/

Each client library contains:

* CSS and JS files
* resources folder
* .content.xml file (Defines properties and configurations)
* cq:ClientLibraryFolder node(dependencies)

****