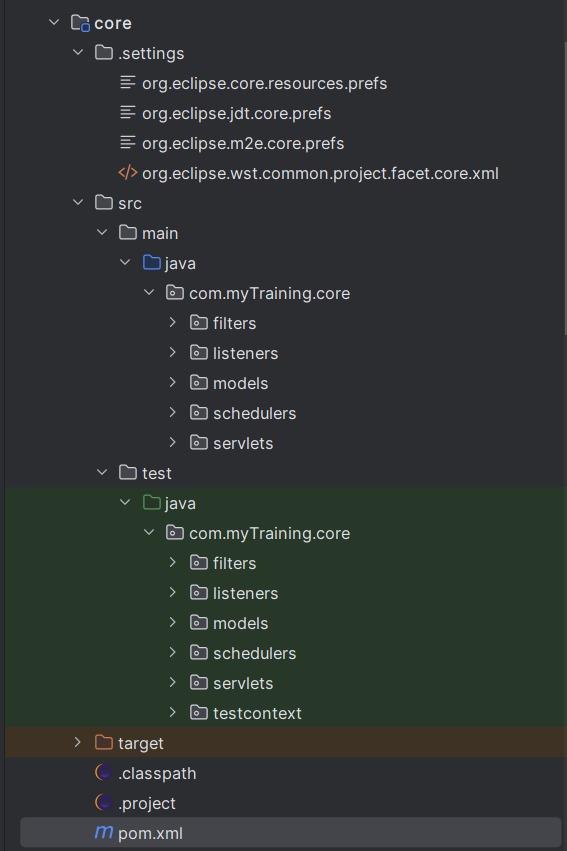
**4/3/25 Day 2 - Learnings Summary Questions**

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

The core module is responsible for the backend logic of the project. It contains Java classes that handle data processing and business logic. It includes Sling Models, OSGi Services, and Servlets to fetch, process, and deliver data to the frontend.

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

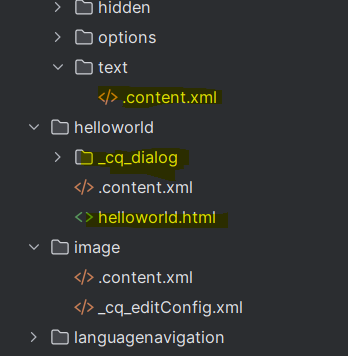
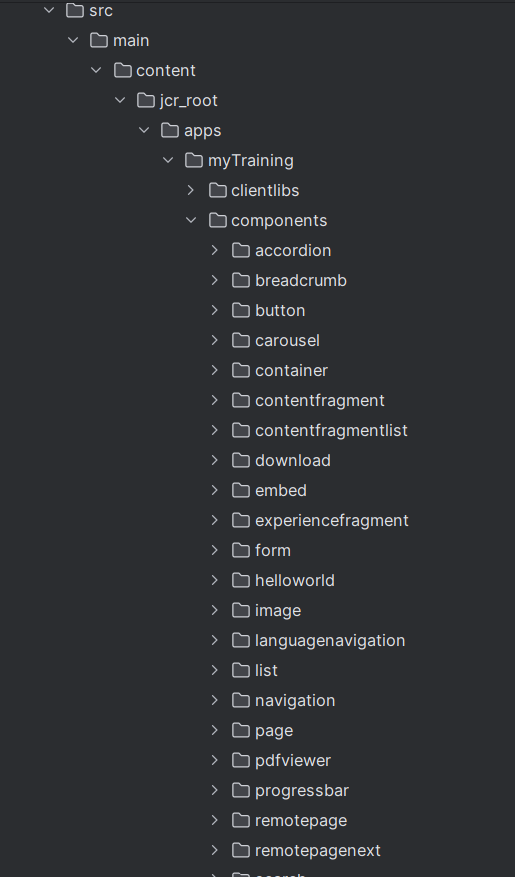


It includes Sling Models, OSGi ,Servlets ,Filters ,Listeners and Schedulers .These files ensure proper data flow between AEM’s repository and the frontend.

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

The ui.apps is responsible for the frontend of the project. It contains components, templates, dialogs, and client libraries that define how the website looks and functions. This module ensures that authors can create and manage content easily using AEM’s drag-and-drop interface.

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



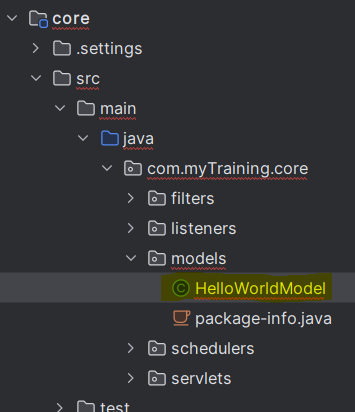
In ui.apps folder, components are stored inside jcr\_root/apps/myTraining/components, with each component having its own folder. Each component contains HTL for display, \_cq\_dialog for editing, and .content.xml for configuration. This setup makes components reusable and easy to manage in AEM.

**5)Hello World Component:**

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

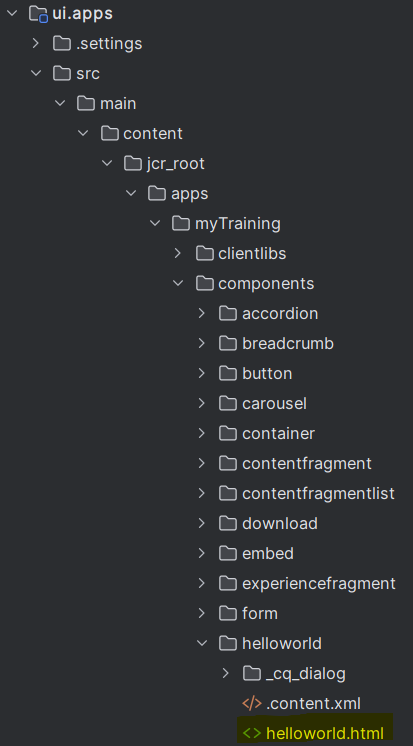
**Core**

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

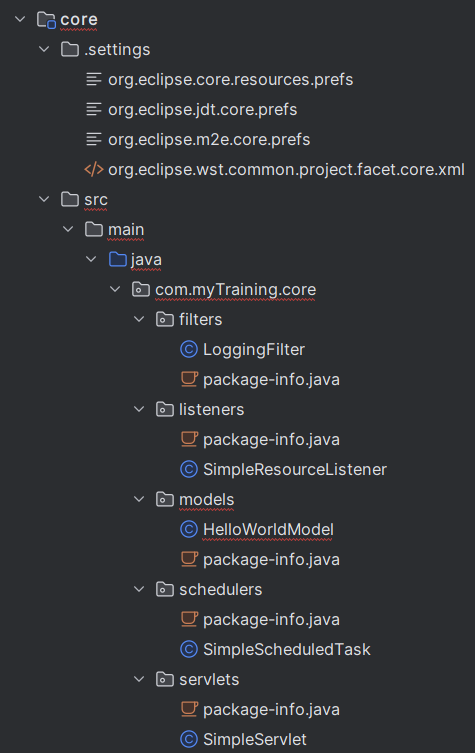
****

**ui.apps**

**ui.apps/src/main/content/jcr\_root/apps/myTraining/components/helloworld**



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



The HelloWorldModel.java in the core module is a Sling Model that fetches data from AEM’s repository and makes it available to the frontend. It is used in the HTL (helloworld.html) file to display dynamic content in the Hello World component.

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

The HTL script (helloworld.html) in ui.apps fetches data from the HelloWorldModel.java in core module and displays it on the frontend. It connects the backend and frontend to show dynamic content in AEM.

* How are properties and dialogs defined for this component?

Properties are defined in .content.xml, while dialogs are created using \_cq\_dialog.xml to allow authors to edit content. The dialog contains fields like text, images, or dropdowns for customization in AEM.

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

core – Backend logic (Sling Models, Servlets, OSGi Services).

ui.apps – Frontend components, dialogs, and client libraries.

ui.content – Stores actual site content and structure.

dispatcher – Manages caching and request handling.

all – Combines everything into one deployable package.

1. **How does Maven build these modules?**

Maven compiles Java code, packages UI files, and creates a deployable ZIP for AEM.

Command

**mvn clean install -PautoInstallPackage -Dmaven.test.skip=true**

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

Compile – Converts Java code into .class files.

Package – Creates JAR (backend) and ZIP (frontend) packages.

Install – Deploys the package to AEM.

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

Dependencies (libraries required for the project) are added in pom.xml using <dependencies> tag .

Maven automatically downloads and manages them.

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

Maven automates builds, manages dependencies efficiently, and ensures smooth integration with AEM.

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

* Manages the dependencies automatically
* Build automation and deployment
* Organizes projects into modules

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

It automatically downloads and maintains the required libraries and APIs through its central repository. It uses the pom.xml file to define dependencies, ensuring consistency across different environments

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

**mvn clean install command** Compiles Java code, Packages frontend components, Deploys the package to AEM

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

To deploy packages directly to AEM use

- **mvn clean install -PautoInstallPackage------** It automatically installs the project inside AEM

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

Maven profiles in AEM helps to automate deployments based on what is needed to be installed.

mvn clean install -PautoInstallPackage ------ Installs everything (backend & frontend).

mvn clean install -PautoInstallBundle ------ Installs only OSGi bundle (backend).

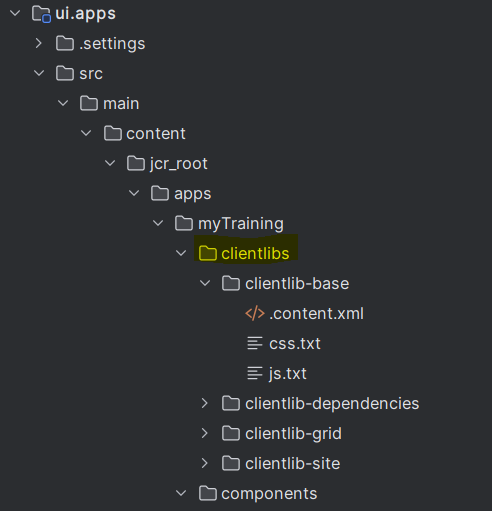
**16) What is the purpose of dumplibs in AEM?**

dumplibs is a tool used to debug and analyze client libraries (clientlibs). It checks how CSS and JavaScript files are loading in AEM. It helps to find issues if styles or scripts are not working properly

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

can view client libraries using dumplibs by accessing:  
 <http://localhost:4502/libs/granite/ui/content/dumplibs.html>

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

****

client libraries (clientlibs) in AEM are structured inside **/apps/myTraining/clientlibs**.

Each folder (like clientlib-base, clientlib-site) contains:

* css.txt – Lists CSS files for styling.
* js.txt – Lists JavaScript files for functionality.
* .content.xml – Defines the clientlib properties.