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

The core module in AEM is a important one in the AEM. Because it will act as a basic and foundational of AEM. There are nearly 9 to 10 modules present in AEM. Each and every one act as a independent one. Even if u comment any one module in the code there will be no error. Other module will work fine since each and every module is working independently. These modules are responsible for the basic operation and workflow of the AEM. It helps in integrating with other modules too.

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

We get lots of files and code regarding our project. Our folder is named as myTraining, so all the files mostly contains the name as myTraining. We can get files like core, tests, apps, ui config, frontend, tests and so on. We can also get pom.xml which is an important one. Mainly we get java source code of our project, OSGI components, servlets and filters, jcr, configuration and resource files also.

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

In AEM ui.apps is also one of the component or also called as module. It is very important one. It is responsible for the frontend and the integration of AEM components. Since the name itself says that it is a ui.app (User Interface) it is maily used and helpful for the frontend of our project. In eclipse we have various files for ui.apps like ui.config, ui.content, ui.frontend, ui.tests , ui.apps and so on.

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

In folder we have myTraining.ui.apps.structure. We also have pom.xml file inside it. We also have test file and component configuration files in it. It also have target folder in it. It is used to how the user interact with the component. We also have clientlibs bundle to maintain.

1. Hello World Component:

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

Core:

core/src/main/java/com/mytraining/core/components/HelloWorldComponent.java

ui.apps:

ui.apps/src/main/content/jcr\_root/apps/mytraining/components/hello-world/hello-world.html

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

In java we have getter and setter method to display the Hello World. We also have adaptables=resource.class.

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

Uses HTML to render the Hello Word component dunamically in ui.apps.

• How are properties and dialogs defined for this component?

In ui.apps modules, the dialogs defines the properties that can be configured by the suer. It is done in dialog.xml.

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

There are 10 modules defined in the pom.xml. They are “all, core, ui.frontend, ui.apps, ui.app.structure, ui.config, ui.content, it.tests, dispatcher, ui.tests”. Each and every module has its own property and own uses. But all these modules are work independently. They wont depent on one another.

1. How does Maven build these modules?

These modules are huge and it is is used to create a muti-module projects. So Maven build these types of modules with the help of pom.xml. Because here the modules are declared and used. Also mostly each and every module has its own pom.xml file. Modules like core, ui.app, ui.content and so on has also its own pom.xml file. So this is how the Maven build these modules.

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

The lifecycle of Maven in AEM refers to the various stages involved in developing, testing, etc in the process of AEM-based project. The build lifecycle of Maven can be described in 8 steps or also called as 8 process. They are Validate, Compile, Test, Package, Integration Test, Verify, Install and Deploy. These are the key process in the lifecycle of Maven in context of AEM.

1. How are dependencies managed in pom.xml?

In Maven, the dependencies for the AEM project are managed withing the pom.xml file. This is the central configuration file for a Maven project. All the necessary dependencies for the AEM project will be added to the pom.xml file. The dependencies can include libraries, frameworks and so on. Dependency declaration can be done within the tag of <dependency>….</dependency>

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

Maven is a good tool and it is mostly used in many projects. AEM projects too uses Maven. Maven is a java-based projects which helps to download dependencies which refers to the libraries or JAR files. It provides a standardized build process. AEM projects are structured with multi module and Maven is best suit for managing these types of projects because it has built-in support for multi-module build.

1. What advantages does Maven offer for AEM development?

Maven simplifies the planning process of any specific project. So it is very useful in using in the AEM projects. Maven mainly standardize the build process which enforces a consistent structure across different modules such as core, ui.apps, ui.content. It also reduce the configuration complexity. Maven dependency management is also one of the very useful for the AEM projects. Maven offer a robust solution for the big level, enterprise-level projects.

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

Maven helps in managing the dependencies and plugins in AEM project which is very helpful for the user to code. Maven automatically download and manage the require libraries(dependencies) from a central repository and ensures a consistent versions across the entire project which also provide a framework to execute specific build tasks through plugins. Maven manage the dependencies by centralize configuration through pom.xml file.

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

mvn clean install is one of the Maven command used in AEM project. It is used to delete the build artifacts, which will clear the old and build a completely fresh new one. This will help the user with issues related to leftover field or out-of-date dependencies. It ensures a clean build. While executing this command, the clean phase is executed which is used to remove the previously generated build artifacts includes the compiles code, pacakaged JARs and so on.

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

We have to go to command prompt and check for the maven installation in your system. It can be done by running the comman mvn -version. Once we confirmed the maven is installed the we can change the path to the folder where we want to create the project or package then we have to use the command. The command is

mvn -B org.apache.maven.plugins:maven-archetype-plugin:3.2.1:generate -D archetypeGroupId=com.adobe.aem -D archetypeArtifactId=aem-project-archetype -D archetypeVersion=39 -D appTitle="myTraining" -D appId="myTraining" -D groupId="com.myTraining" -D aemVersion=6.5.5 -D includeExamples=n -D includeErrorHandler=y

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

In AEM projects Maven profiles such as autoInstallPackage, autoInstallBundle is very important and plays a important role. Both plays the role in the automating the deployment process and make the developer to quickly deploy the updates to AEM instance. autoInstallPackage is used to automatically install the AEM packages that ae built during the Maven build process. autoInstallBundle is used to automatically install AEM bundles which are all OSGI-based modules.

1. What is the purpose of dumplibs in AEM?

In AEM dumplibs is a library modules which allows developers to inspect and debug the client-side libraries. It is used to generate a detailed dump of the OSGi bundles and their dependencies within the AEM instance. It help the user or developer to analayze and troubleshoot issues related to the missing or incompatible dependencies. It is useful when dealing with the versioning problems.

1. How can you view client libraries using dumplibs?

In AEM dumplibs allows you to view the client libraries. You can view the client libraries in system/console/bundles or you can also use the website link to view. The website link will be localhost:4502/libs/granite/ui/content/dumplibs.html. You can view all the libraries or modules present in your project

1. Explain how client libraries are structured in AEM.

In AEM all the client libraries are structure in files and folders. Each and everything is categorized according to it. These are organised in clientlibs folder within the project. The structure consist of multiple modules and each containing their set of related resources.