1. **Explain Adobe CQ5?**

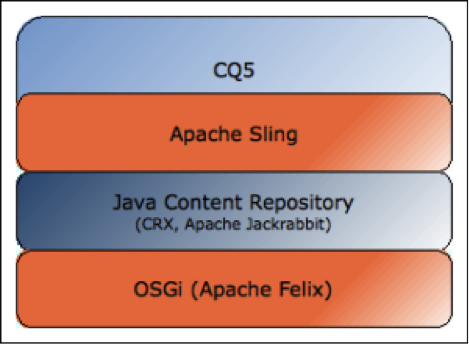
Adobe CQ5(also known as Adobe Experience Manager) is a java based content management system from adobe.

* It is based on a content repository to store the content of a website and use JCR(java content repository) specification to access the content repository.
* It uses RestfulApache Sling framework to map request url to the corresponding node in content repository
* It uses powerful OSGi framework internally to allow modular application development. It means individual pieces of your application (called bundles in terms of OSGi) can be independently started and stopped.
* It uses Apache Felix as the OSGi container. Therefore different parts of cq5 can be independently started and stopped.

1. **What is the technology stack used in cq5 or AEM(Adobe Experience Manager)?**

Adobe CQ5 uses the following technologies :

* JCR – Java specification for accessing a content repository JSR-283 specification jcr 2.0 , cq5 uses its own implementation of jcr called CRX. Apache Jackrabbit is an open-source implementation of jcr 2.0 specification.
* Apache Sling – RESTful framework to access a jcr over http protocol. It maps the request url to the node in jcr.
* OSGi(Apache Felix) – Framework for modular application development using java. Each module called bundle can be independently started and stopped.OSGi container which provides implemention classes for OSGi framework.



1. **Why a content management system is required in CQ?**

Now a days websites are very dynamic in nature, content needs to be updated very frequently, So, it is easier to manage the content of such websites using a CMS.

1. **What are the advantages of CQ5 over other CMS?**

Below are the advantages of CQ5 over other CMS(Content management System):-

* We can easily implement workflows for creating, editing and publishing of content.
* It is easily to manage digital assets like images, documents and integrating them to the websites.
* Usage of search queries to find content no matter where it is stored in your organization.
* Easily setting up the social collaboration blogs, groups.

1. **What is Adobe marketing cloud? Why do i need it?**

Basically it is a collection of 6 adobe marketing solutions, it helps customers to master digital marketing by leveraging these adobe marketing solutions.

* Adobe Experience Manager- helps customer to create a seamless digital appearance, across all platforms and languages. That helps customers to build their brand globally and increase their demand.
* Adobe Social – helps customers to lead in their social media channels like Facebook, twitter, google plus and enables customers to know its community sentiments.
* Adobe Analytics – like google analytics it also collects, analyze and divides visitors into segments that can be plotted on charts to get better understanding and insight about your visitors.
* Adobe Target – It uses Adobe Analytics inputs to deliver targeted or most relevant content to right visitor and right time. It understand customer preferences provides personalized experience to meet each customer need instantly and hence increases overall sale.
* Adobe Media Optimizer – It allows customers to provided targeted ads to the visitors and hence optimize overall money spent on ads.
* Adobe Campaign – It allows customers to provided targeted one to one campaigns to the visitors based on their interest.

1. **What is a Template?**

A CQ template enables you to define a consistent style for the pages in your application. A template comprises of nodes that specify the page structure. Learn How to Create Template in AEM

1. **What is a Component?**

Components are re-usable modules that implement specific application logic to render the content of your web site. You can think of a component as a collection of scripts (for example, JSPs, Java servlets, and so on). Learn How to Create Component in AEM

1. **Why we need to include global.jsp if we are creating a component in jsp?**

by default it declares the Sling, CQ and JSTL taglibs  to make component creation easy in AEM.

It provides most often used Objects such as properties, pageProperties ,pageManager, currentPage, currentStyle, slingRequest, slingResponse, resource, sling. Leveraging these objects in component programming is a great help and saves tons of time from initializing each of them manually.

1. **Which script you should include to display sidekick ?**

init.jsp should be included in our jsp or script file to display sidekick.

1. What is the use of EditConfig node in creating a component?

A cq:EditConfig node is used to define the behavior of the component.

1. What are the basic SCR Annotations used for creating an OSGI component?

Basic SCR Annotation used for developing a component or service in osgi are:-

* @Component – defines the class as a component.
* @Service – defines the service interface that is provided by the component.
* @Reference – injects a service into the component.
* @Property – defines a property that can be used in the class.

1. Difference between Dialog and Design Dialog?

See book

1. How to put multiple files in CRX repository?

To put multiple files we can use many tools that supports WebDav Protocol Like NetDrive. Learn how to use WebDav in Details.

1. What are the different interfaces available in AEM ?

The different interfaces available in CRX are:

* CRX Explorer – <http://localhost:4502/crx/explorer/browser/index.jsp>
* CRX DE Lite – <http://localhost:4502/crx/de/index.jsp>
* Apache Felix – <http://localhost:4502/system/console>
* Site admin – <http://localhost:4502/siteadmin>
* etc/Tools – http://localhost:4502/miscadmin

1. How do you resolve a resource ?

You can resolve using ResourceResolver which you can get from method getResourceResolver() and then use resolve() method to resolve a resource.

1. How do you adapt a resource ?

You can adapt a resource to any another type using adaptTo() method, which accepts class type in which you want to adapt your resource. i.e. Page page = resource.adaptTo(Page.class);

1. What is the purpose of clientlibs?

It is used for adding site specific js and css files to the page. jcr:primaryType of clientlibs folder is cq:ClientLibraryFolder. It takes cares of dependency management, merging files and minification of all js and css files stored under it.

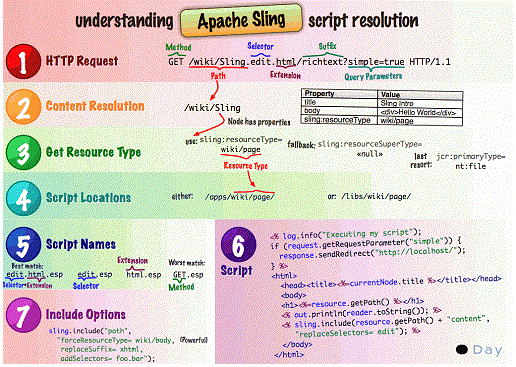
### What is Dependency/Dependencies in client library?

It is a list of other client library categories on which this library folder depends. Suppose I have one clientlib with categories training.a and one more clientlib with categories training.b. so if i want to use some code of training.a in training.b then in dependencies of second clientlib I have to mention training.b as dependency value.

Usually to load JS librariesFor example like jquery, handlebars, bootstrap even angular sometimes

1. **How resource resolution is done in Apache Sling?**

How a URL is resolved and mapped to a resource by Apache sling is described in below image.



Consider the URL

GET – <http://www.mywebsite.com/products/product1.printable.a4.html/a/b?x=12>

Here request type used is HTTP GET request

Lets break it down into its composite parts:

Protocol host content path selector(s) extension   suffix   param(s)

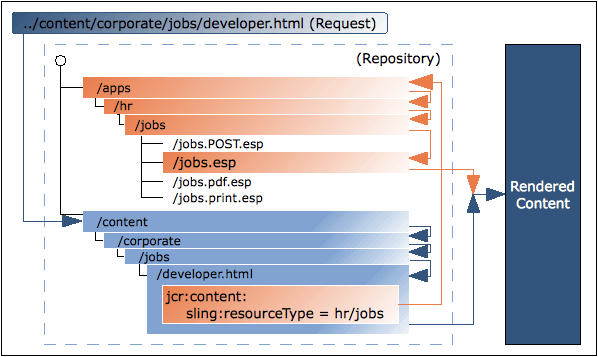
http:// myhost products/product1 .printable.a4. html / a/b ? x=12

From URL to Content and Scripts

Using these principles:

* The mapping uses the content path extracted from the request to locate the resource.
* When the appropriate resource is located, the sling resource type is extracted, and used to locate the script to be used for rendering the content.

The below figure illustrates the mechanism used :-



Mapping the requests to resources

The request is broken down and the necessary information extracted like what is the content Path, method and selector from URL. The repository is searched for the requested resource (content node):

 First Sling checks whether a node exists at the location specified in the request; e.g. ../ content/corporate/jobs/developer.html

If no node is found, the extension is dropped and the search repeated; e.g. ../content/ corporate/jobs/developer

 If no node is found then Sling will return the http code 404 (Not Found).

Note: Sling also allows things other than JCR nodes to be resources, but this is an advanced feature.

Locating the script:

When the appropriate resource (content node) is located, the sling resource type is extracted. This is a path, which locates the script to be used for rendering the content. The path specified by the sling:resourceType can be either:

* Absolute.
* Relative, to a configuration parameter.

Script Resolution Priority –> selection+extn, extn, selector, method, default (same name as component name) Understand Sling Resource Resolution in Details

1. **Explain the role of Dispatcher in CQ5?**

In CQ5 Dispatcher has two main roles:

Caching – It is used to cache as much content as possible in the form of a static html which helps to reduce the continuous functioning of layout engine for frequently generating dynamic content.

For caching dispatcher works as part of HTTP server such as apache and places the cached documents under the root of web server.

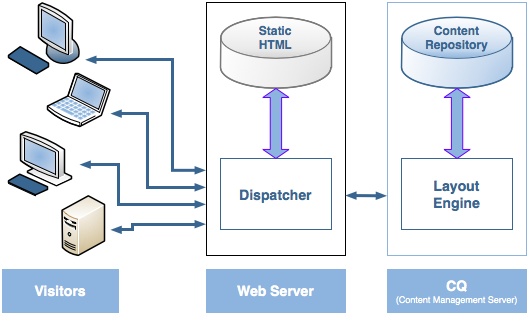
It has two methods to update the dispatcher content when changes are made to the website.

1. It will remove the pages form the dispatcher which have been changed or updated.
2. Second is auto invalidation, which invalidates the pages which are out of date without deleting anything.

Give example /content/geometrix/en

Load-balancing – To increase the performance by load-balancing.

Note:  If there are multiple cq instances configured with a dispatcher, the dispatcher can do a load-balancing and if there is too much load on any cq instance, it can relay the request to another less busy instance.



1. **How Dispatcher performs Load-balancing ?**

* Performance Statistics – Dispatcher keeps statistics on how fast each instance of cq is responding to a particular url. Based on those metrics, dispatcher determines which instance of cq will fetch the quickest response for a request so that relays on that instance for those requests
* Sticky Connections – when a user session is established, then all incoming requests from that user should be served by the same cq instance, because other cq instances cannot recognize the user session and generate personalized pages for him. Dispatcher makes sure all requests for user session are served from the same cq instance.
* Increased fail-safe coverage: If the Dispatcher does not receive responses from an instance, it will automatically relay requests to one of the other instance(s)
* Increased processing power: In practice this means that the Dispatcher shares requests between several instances of CQ. So that each instance has fewer request to process, which results in faster response times.

1. **Can I implement multiple Dispatchers in a setup?**

Yes. In such cases, ensure that both the Dispatchers can access the CQ website directly. A Dispatcher cannot handle requests coming from another Dispatcher.

1. **What are the differences between package and bundle?**

Package:

 A Package is a zip file that displays the content from the repository as an easy-to-use-and-edit representation of files and folders. Packages can include content and project-related data.

Bundle:

* Bundle dynamically loadable collection of classes, jars, and configuration files that explicitly declare their external dependencies (if any).
* OSGi bundles will be having metadata in the jar’s manifest, found at META-INF/MANIFEST.MF which contains version, last installed date etc.
* With OSGi, if the class is public it doesn’t mean you can access it form anywhere. All bundles are included in an export list of package names, and if a package isn’t in the export list, it doesn’t exist to the outside world. Which gives a package-private visibility. A common pattern, for instance, is to put interfaces in one package and implementations in another, and only export the interface package.
* All OSGi bundles are given a version number, so it’s possible for an application to simultaneously access different versions of the same bundle (eg: junit 3.8.1 and junit 4.0.). Since each bundle has it’s own class-loader, both bundles classes can coexist in the same JVM.
* In OSGi bundle, there will be an Activator.java class which is an optional listener class which notifies the bundle start and stop events.

1. **Explain life cycle of OSGI[Open Systems Gateway initiative] bundle?**

OSGi is a framework which allows modular development of applications using java. A large application can be constructed using small reusable components(called bundles in terms of OSGi) each of which can be independently started, stopped, and also can be configured dynamically while running without requiring a restart.

Following are the states of OSGI life cycle:

* Installed – The bundle has been successfully installed.
* Resolved – All Java classes and the dependencies that the bundle needs are available. This state indicates that the bundle is ready to be started.
* Starting – The bundle is being started, the BundleActivator.start method will be called, and this method has not yet returned. When the bundle has an activation policy, the bundle will remain in the STARTING state until the bundle is activated according to its activation policy.
* Active – The bundle has been successfully activated and is running; its Bundle Activator start method has been called and returned.
* Stopping – The bundle is being stopped. The BundleActivator.stop method has been called but the stop method has not yet returned.
* Uninstalled – The bundle has been uninstalled. It cannot move into another state.

1. **What are the advantages of using OSGI?**

* Advantages of using OSGI are stated below:-
* Dynamic module system for Java.
* Universal Middleware Category.
* Helps applications to be constructed from small, reusable and collaborative components.
* OSGi bundles can contain compiled Java code, scripts, or any contents to be loaded in the repository. Helps the bundles to be loaded, installed.
* Reduces the complexity of the system.

1. **What are the differences between parsys and iparsys?**

parsys –

It is a placeholder called “Paragraph System”, where we can drag and drop or add different types of components or scripts at page level.

iparsys –

It is inherited paragraph system, it is similar to parsys except that it allows to inherit parent page “paragraph system”.

1. **What are the differences between Parbase and parsys?**

Below are the differences between Parbase and Parsys.

Parbase

It allows components to inherit attributes from other components, similar to subclasses in object oriented languages such as Java, C++, and so on.

For example, when you open the/libs/foundation/components/text node in the CRXDE Lite, you see that it has a property named sling:resourceSuperType, which references the parbase component. The parbase here defines tree scripts to render images, titles, and so on, so that all components subclassed from this parbase can use this script.

The parbase component is never used explicitly, but only facilitates the rendering of a component when added in the Parsys. If we are creating any new xtype and we want it to be draggable then we give its sling:resourceType as parbase.

Users do not need access to the parbase.

Parsys

The paragraph system (parsys) is a compound component that allows authors to add components of different types to a page and contains all other paragraph components. Each paragraph type is represented as a component. The paragraph system itself is also a component, which contains the other paragr

1. **What is Multi Site Manager (MSM) in AEM ? How to configure MSM in AEM ?**
2. **What do you mean by Live copy and blueprints in AEM ? How to create Live copies and blueprints ?**
3. **Have you worked on workflow model in AEM or CQ? Can you tell me how i can execute a workflow and what are the steps involved in it?**
4. **What are the steps for creating custom components in cq? Steps for creating a custom component in detail.**
5. **What is process step in workflow?**

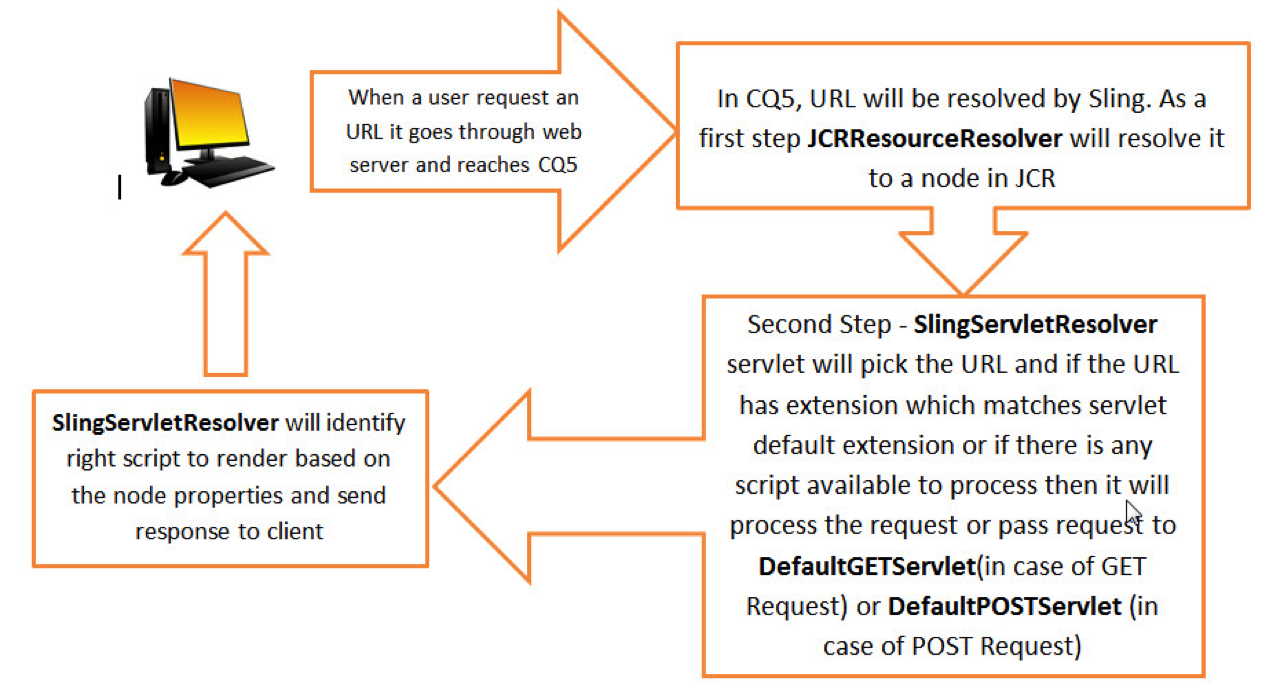
It is a workflow component which can be found at – “/libs/cq/workflow/components/model/process”, this is used for calling a java class in workflow.

### What is Replication? And what are the steps involved in Replication?

Replication is process of activating/publishing the content from Author Instance to Process Instance.

**Steps involved in replication:**

1. The author requests that certain content needs to be be published (activated); this can be initiated by a manual request, or by automatic triggers which have been preconfigured.
2. The request is passed to the appropriate default replication agent; an environment can have several default agents which will always be selected for such actions.
3. The replication agent “packages” the content and places it in the replication queue.
4. The content is lifted from the queue and transported to the publish environment using the configured protocol; usually this is HTTP.
5. A servlet in the publish environment receives the request and publishes the received content; the default servlet is [*http://localhost:4503/bin/receive*](http://localhost:4503/bin/receive).
6. **Classes involved in URL Resolution?**

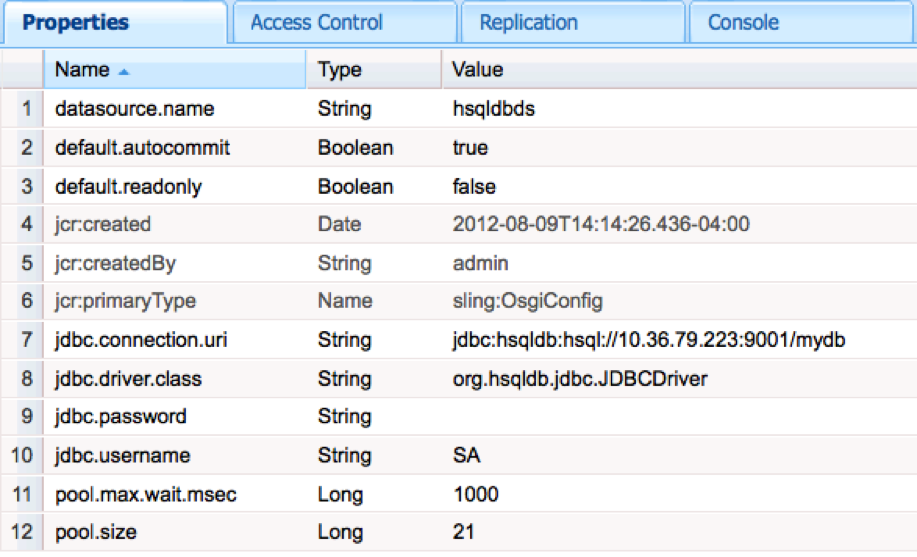


1. **Role of Persistence Manager in CQ5?**

The persistence manager saves the repository content to a permanent storage location, such as the file system or a database. By default, CRX saves repository content to the Tar persistence manager. Following DB’s can be used for storing content – DB2, Oracle, SQL Server, MySQL.

1. **How to connect to external Database?**

To connect to external DB, a connection pool needs to be configured by creating a node of type sling:OsgiConfig. Below screen shot if for connecting to hsqldatabase, but if you want to connect to other DB then properties values needs to be changed accordingly:



Use below code to get connection:

**DataSourcePool dspService = sling.getService(DataSourcePool.class);**

**DataSource ds = (DataSource) dspService.getDataSource(“hsqldbds”);**

### Explain DAM Renditions?

The DAM allows you to create renditions of an image that include different sizes and versions of the same asset. And that can be helpful when you need to create thumbnails or smaller views of large, high-resolution images. When you are creating multi-device websites, this feature becomes even more important.

1. **What is a content repository? What is JCR?**

A Content repository is basically a place where digital content is stored. Generally the structure of the content repository is hierarchial and represented as a tree structure where each node of the tree is used to store content.

Java Content Repository is a specification provided by the java community to access the content repository in a uniform way (platform independent and vendor independent way). The specification was initially released as JSR-170(JCR 1.0) and then later revised version 2 as (JCR-283).The javax.jcr API provides the various classes and interfaces to access a content repository.

### What is REST? What is a RESTful Framework?

REST stands for Representational State Transfer. REST-style architectures consist of clients and servers. Clients initiate requests to servers; servers process requests and return appropriate responses. Requests and responses are built around the transfer of representations of resources. Apache Sling is RESTful framework to access a java content repository over http protocol.

### What is Reverse Replication?

Reverse replication is used to get user content generated on a publish instance back to the author instance. To do this you need a reverse replication agent in the author environment. This act as the active component to collect information from outbox in the publish environment.

### How bundles are loaded and installed in CQ5?

This is managed by the Sling Management Console of CQ5.

### What is the contribution of Servlet Engine in CQ5?

In CQ5, servlet Engine acts as the server within which each CQ (and CRX if used) instance runs as a web application  
Any Servlet Engine supporting the Servlet API version 2.4 (or higher) can be used.  
  
We can always run CQ WCM without an application server, a Servlet Engine is needed. CRX + CQ WCM,

1. **How you can inherit properties of one dialog to another dialog?**

For inheriting properties we have to create two components with unique names in the base component dialog. For eg. If your plan is to have two rich text areas in the dialog of components that inherit from the base, then you must include two rich text areas with unique names in the base component dialog. In any case every input field of a dialog must have a unique name, else they will point to the same property path relative to the jcr:content node of the component when used on a page.

1. **Can we restrict for certain users not to display some digital assets?**

You can always limit who can access certain folders in CQ Digital Assets by making the folder part of a CUG(closed user group).

Steps to make a folder part of a CUG:

In CQ DAM, right-click the folder you want to add closed user group properties for and select Properties.

Click the CUG tab.

Select the Enabled check box to make the folder and its assets available only to a closed user group.  
Browse to the login page, if there is one, to add that information. Add admitted groups by clicking Add item. If necessary, add the realm. Click OK to save your changes.

1. **What is the listener property in AEM?**

The listener property for a component is used to define what happens before or after an action on the component. This is added by using the “cq:listeners” node with a node type of “cq:EditListenersConfig”. The listener property can also be added to any widget in AEM. In order to add a listener to that widget you just need to add a node that is called “listeners” with a type of “nt:unstructured”. You can find a list of events for each widget by searching CQ Widget API documentation.

1. **How do you load digital assets in dam?**

You can add digital assets to your dam by using the graphical user interface or through WebDav access. If you are using the graphical user interface, you would just browse for the selected files you would like to add, and then cq will create the metadata for those assets in the dam folder. You would generally use the WebDav option when you want to upload a large number or assets at once.

1. **What is a CQ5 overlay/override component?**

The overlay/override component is used when you want to use an out of the box component in AEM and you want to add extra features to it and you want to change it across all instances of that component. For example if you want to add features to the out of the box “text” component, but you don’t want to create a new component, you would want to copy the “text” component from “libs/foundation/components” to “apps/foundation/components” and keep the same folder structure that is used. When this is done, any changes that done to this new component will be reflected in the out of the box “text” component, without changing the original code under “libs”. It is generally not a good practice to modify the original code, which is why CQ offers the ability to use overlays.

1. **What is extending a component?**

Extending a component is when you want to create a new component that will not override a base component, but will have the same features as the original so that you can change or build upon that component. In order to extend a component, you must set“sling:resourceSuperType” of the component to the base component you would like to extend. By doing this, you will inherit everything from the base component.

1. **How does sightly differ from other templating systems?**
2. **Can you create a page without a template?**

Pages are usually created by selecting a template for the page when you create them in the Website console. However, a page can be created manually in the crxde by using the same properties that are created when you make a page in the Website console. For example, in crxde under the content folder, create a node with the following type “cq:Page” and save it. Then add a child node with the following type “cq:PageContent”. Then you can add the required properties to show any components you want displayed on the page. It is generally not good practice to create pages manually.

1. **What are xtypes and how are they useful in AEM?**

In the ExtJS language an xtype is a symbolic name given to a class. In AEM these xtypes are widgets that used in the creation of components. AEM comes with a bunch of widgets that are available out of the box. You can also create and define your own xtype to be used in AEM. An example of an xytpe is when you are creating your dialog for your component, and you want the author to be able to enter text, you would add a “cq:Widget” that has an xtype property of “textfield”.

1. **How do you add properties to the Page Properties dialog?**

You can add properties to your page properties dialog by copying the dialog located under “libs/foundation/components/page/dialog” and adding it to your page template. This will allow you to add new tabs and add new properties that you would like the author to chooses from for pages that use that template.  
You could modify the original page dialog as well to get the same result, however it is not recommended.

1. **What is the difference between the Author and Publish environment?**

A production environment usually has two difference instances of AEM running. One is the author instance, and the other is the publish instance. These two instances are usually kept in two different settings. The author instance is where you will enter and manage content for your website. This is where you will administer your site as well. The author environment is usually kept behind a firewall. The publish instance is where you will make your content available for your targeted audience. The publish environment is usually kept in a Demilitarized Zone (DMZ).

1. **What design patterns are used in AEM?**

Since AEM is built using OSGI, many of the design patterns for OSGI are valid. Some of these design patterns are Singleton (Service), Adapter Service, Resource Adapter Service, and Whiteboard. Also, since AEM is modular, you should be able to use any design pattern in your application.

1. **What is Sling? How is it different from other web-development frameworks?**

Apache Sling is RESTful framework to access a java content repository over http protocol.

It is a content driven framework that is it maps the incoming user request based on URI to the corresponding node in the content repository and depending on the type of the request(GET,POST, etc) executes the corresponding dynamic script.

#### **What is Personalization?**

There is an ever-increasing volume of content available today, be it on internet, extranet, or intranet websites. Personalization centers provides the users dynamic content selection according to their specific needs. It can be on the basis of predefined profiles, user selection, or interactive user behavior.  
**Teaser Component used in Personalization**

* Touch UI is based on CORAL UI + GRANITE UI. CORAL UI has been used to provide widgets (CSS+JS) while GRANITE UI provides the foundation components.
* Touch UI is responsive, supports desktop & touches devices and uses mobile first approach.

AEM (5.6+) supports both classic UI & Touch UI. Default UI is set as Touch UI by adobe in AEM 5.6+ but it provides the options to users to switch classic UI as default. Please note that currently AEM provides the option to switch default UI but does not provide configuration support to disable any of the UI(Touch or Classic).

**Differences between Classic UI & Touch UI**

|  |  |
| --- | --- |
| **Classic UI Dialog** | **Touch UI Dialog** |
| Classic UI dialog is based on EXTJs | Touch UI dialog is based on Coral UI + Granite UI |
| Classic UI dialog is not responsive | Touch UI dialog is responsive & uses Mobile first approach |
| dialog is the root node | cq:dialog is the root node |
| Root node uses EXTJs xtypes | Root node uses Granite UI resource Types |
| Classic UI Dialog is rendered on client side | Touch UI dialog is rendered on Server side |

1. **What are the differences between sling:resourceType and sling:resourceSuperType ?**

**sling:resourceSuperType:** It is used to achieve inheritance in cq. When set, it inherits the specified component to this component.

**sling:resourceType:** It is a path, which locates the script to be used for rendering the content. Path used can be absolute or relative.

1. **List Implicit objects of CQ?**  
   slingRequest ,slingResponse ,resource , currentNode, log , properties , pageManager ,component ,designer ,currentDesign ,currentStyle
2. **What are the differences between CRX DE and Ecllipse ? What are the advantages of using CRXDE over Ecllipse for AEM Development?**

**The major difference between Crxde and eclipse**

Crxde comes with AEM and it runs on server side where as eclipse runs on client side. CRXDE contains build in API’s where as in eclipse we need to add extra plugin for supporting CRX API.

**Built in API’s of crxde**

**Java Content Repository API**—a complete JCR 2.0 implementation Content Management Interoperability Services—CMIS 1.0

**WebDAV**—with versioning, access control, and search Windows Network File Share—CIFS/SMB

**RESTful Web API** for JavaScript and Flash/Flex  

1. **How many wcm modes are there?**

* Edit mode
* Design mode
* Preview mode

**Preview mode**: This mode removes all the authoring interface and shows you how the web page would look on the website. It's best to use this mode for checking the end results.

**Edit mode:** Whenever author makes changes or configures components (Example: Adding title to a page by editing the Title component) in this mode the data is stored under the components' node. The path of the configured data will look something like this: /content/your website/path\_to\_the\_page/jcr/component

**Design mode:** This mode is used to configure components that have same data on different pages like the logo or footer. Any changes you make on the component in any one of the pages' will be reflected on all the pages that have the component. The catch is all these pages must have the same design.

1. **What is service?**

A service is *any object* that is registered in the OSGi Service Registry and can be looked up using its interface name(s). The only prerequisite is that a service should implement some interface...