Q : Difference between v. and c.?

A :

- These are known as value providers for a component.
- v (View) helps to access component attribute's value in markup.
- c (Controller) helps to link with event handlers and action for the component.

Q : How can we deploy lightning components?

A :

- Lightning components can be deployed like any other component using change set, ANT migration tool, Gearset, Copado or other migration tool.

Q : How to get current record id in lightning component?

A :

- implement force:hasReocrdId interface.
- Create an attribute named as recordId which will fetch the current record id automatically if component is placed on record page.

Q : What is Lightning Data Service?

A :

- LDS helps to load, create, edit or delete a record in component without requiring Apex code.
- LDS handles sharing rules and FLS.
- Components:
  - force:recordData
  - force:recordEdit
  - force:recordView
  - lightning:recordForm
  - lightning:recordEditForm
  - lightning:recordViewForm

Q : Capture Phase?

A :

- The event is captured and trickles down from the application root to the source component.
- The event can be handled by a component in the containment hierarchy that receives the captured event.
- Event handlers are invoked in order from the application root, down to the source component that fired the event.

Q : Bubble Phase?

A :

- The component that fired the event can handle it.
- The event then bubbles up from the source component to the application root.
- The event can be handled by a component in the containment hierarchy that receives the bubbled event.
- Event handlers are invoked in order from the source component that fired the event up to the application root.

Q : What is Event Propagation?

A :

- The framework supports **capture** and **bubble** phases for the propagation of component or application events.
- These phases provide an opportunity for interested components to interact with an event and potentially control the behavior for subsequent handlers.
- The component that fires an event is known as the source component.
- The framework allows you to handle the event in different phases.
- These phases give you flexibility for how to best process the event for your application.

Q : What is Application Event?

A :

- An application event is fired from an instance of a component.
- All components that provide a handler for the event are notified.
- It goes to application first and then application broadcast it to each component in application and can handle that event in any component.
- Steps:
  - Create Custom Application Event
  - Register Application Event
  - Fire Application Event
  - Handle Application Event

Q : Important about Component Event?

A :

- Here Child Component will pass parameters to Event Component's attributes.
- Then Parent Component will fetch those attributes from Event Component's attributes.
- Component Event works in a parent child hierarchy only.

Q : What is Component Event?

A :

- A component event is fired from an instance of a component.
- A component event can be handled by the component that fired the event or by a component in the containment hierarchy that receives the event.
- Steps are:
  - Create Custom Component Event
  - Register Component Event
  - Fire Component Event
  - Handle Component Event

Q : Types of Event?

A :

- Browser Event
  - onchange, onclick, onblur etc.
- System Event
  - init, render, locationChange etc.
- Custom Event
  - Component, Application

Q : What is Event?

A :

- Event-driven programming is used in many languages and frameworks, such as javascript and Java Swing.
- In lightning a component registers that it may fire an event in its markup.
- Events are fired from JavaScript controller actions that are typically triggered by a user interacting with the user interface.

Q : Aura Method Vs Aura.Action?

A :

- Aura Method
  - Method is defined in child component
  - Aura method's attribute receives data from parent component.
- Aura.Action
  - Method is defined in parent component
  - Parent component's method receives data from child component.

Q : Where we can place lightning component?

A :

- Lightning Experience
- Mobile App
- Lightning Tab
- Lightning Page
- Lightning Record Page
- Lightning Quick Action
- Experience Builder Site Page
- Visualforce Page
- Flow Screens

Q : How to pass data from parent to child component?

A :

Unbounded Expression:

```
<aura:component>
    <aura:attribute name="parentAttr" type="String" default="parent attribute"/>
    <c:child childAttr = "{#v.parentAttr}"/>
</aura:component/>
```

Bounded Expression:

```
<aura:component>
    <aura:attribute name="parentAttr" type="String" default="parent attribute"/>
    <c:child childAttr = "{!v.parentAttr}"/>
</aura:component/>
```

Q : Can we include external JS or CSS libraries in component ?

A :

- Yes, we can use multiple libraries in our lightning component like JQuery, Bootstrap, custom CSS and custom JavaScript libraries using static resource.

Q : How can we call Apex method in component?

A :

- We can call Apex class method through JavaScript controller.
- Also remember to use the name of apex class in component's view file.

Q : What is the use of @AuraEnabled annotation?

A :

- AuraEnabled annotation enables lightning component to access Apex methods and properties.

Q : How to call Controller function in component?

A :

```
<aura:handler name="init" value="{!this}" action="{!c.myAction}"/>

myAction : function(component, event, helper){
    //block of code

}
```

Q : What is the use of init handler?

A :

- through init handler we can execute the code on load of a component.

Q : How to write expression in component?

A :

```
<aura:attribute name="FirstName" type="String" default="Sanjay"/>

<h1>
    {!v.FirstName}
</h1>
```

Q : What is attribute?

A :

- Attribute are similar to variables created in component's view.
- We can get and set values into attributes in JavaScript controller.

Example:

```
<aura:attribute name="FirstName" type="String"/>
```

Q : How to implement Iteration?

A :

```
<aura:iteration items="{!v.list}" var="item">
    block of code
</aura:iteration>
```

Q : How to perform conditional rendering?

A :

```
<aura:if isTrue="{!v.flag}">
    true block
    <aura:set attribute="else">
        false block
    </aura:set>
</aura:if>
```

Q : Can we include a component in another?

A :

- Yes, we can build parent-child hierarchy while building components.

Q : Message passing between components?

A :

- Parent component to Child component
  - Attribute
  - Aura Method
- Child component to Parent component
  - Component Event
  - Aura.Action

Q : Controller Vs Helper?

A :

- Use controllers to listen to user events and other events like component and application events. But delegate business logic to helper.
- Do similar delegation in all Rendere functions (like render, rerender and so on)
- Whenever you need to call one controller function from another controller function, move that code to helper.

Q : Benefits of Helper?

A :

- Can reuse functions defined in helper.
- Use helper for defining data processing task and queueing service-side actions.
- Helper functions are local to a component.
- Move the code from controller to helper whenever possible.
- Helper function can be called from any javascript code in a component's bundle such as
  - Client-side controller or,
  - Renderer
- Even helper functions can be called within helper itself by another function.
- Helper functions are similar to client-side controller functions in shape.

Q : Lightning Design System?

A :

- The Salesforce Lightning Design System includes the resources to create user interfaces consistent with the Salesforce Lightning principles, design language, and best practices.
- Rather than focusing on pixels, developers can focus on application logic, while designers can focus on user experience, interactions and flows.

Q : Which interfaces are used in Lightning Component?

A :

- *force:appHostable* for Lightning Tabs
- *flexipage:availableForAllPageTypes* for Lightning Page
- *flexipage:availableForRecordHome* for Lightning Record Page
- *force:hasRecordId* for Lightning Record Page to fetch current recordId
- *forceCommunity:availableForAllPageTypes* for Experience Builder Site Page
- *force:lightningQuickAction* for Lightning Quick Action

Q : Lightning Component Bundle includes?

A :

Resource	Resource Name	Usage
Component or Application	sample.cmp or sample.app	The only required resource in a bundle. Contains markup for the component or app. Each bundle contains only one component or app resource.
CSS Styles	sample.css	Styles for the component.
Controller	sampleController.js	Client-side controller methods to handle events in the component.
Design	sample.design	Required for components used in the Lightning App Builder or Lightning Pages.
Helper	sampleHelper.js	JavaScript functions that can be called from any JavaScript code in a component's bundle.
Documentation	sample.auradoc	A description, sample code, and one or multiple references to example components.
Renderer	sampleRenderer.js	Client-side renderer to override default rendering for a component.
SVG	sample.svg	Custom icon resource for components used in the Lightning App Builder.

Q : Difference b/w VF & Lightning Component?

A :

Lightning	Visualforce
Client Side UI Generation	Server Side UI Generation
API Centric Model	Page Centric Model
Component Based Framework	MVC Framework
JavaScript Framework	Tag Based Language
Designed for Lightning Experience and Mobile App	Designed for Salesforce Classic

Q : What is Aura?

A :

- Aura is an open-source UI framework used for developing dynamic web apps for mobile and desktop devices.
- To build lightning components in salesforce lightning experience we use the lightning component framework.
- Lightning components are build on top of Aura.

Q : What is Lightning Component?

A :

- A lightning component is a bundle that includes a definition resource written in markup and may include additional resources like a controller, helper, renderer, style etc.

Q : What is Lightning Component Framework?

A :

- The lightning component framework is a UI framework for developing web apps for mobile and desktop devices.
- It is used for building single-page applications with dynamic, responsive user interfaces for lightning platform apps.
- It uses JavaScript on the client side and Apex on the service side.

Q : What is Lightning Experience?

A :

- Lightning means you can create a user experience that improves productivity, makes jobs easier and more intuitive.
- It is fast, beautiful and unique to each sales and service user.
- It is a simpler user experience, designed to help sales team sell faster, with personalized alerts and an interactive assistant to keep sales reps focused on what's important.
- Helps service reps support customers faster and provides problem solving environment.