Integration:

Identity Basics

TrailHead: <https://trailhead.salesforce.com/content/learn/modules/identity_basics>

<https://trailhead.salesforce.com/content/learn/modules/external-services?trail_id=learn-essential-software-development-skills-for-admins>

* Integration is connecting with 3rd party system or vice versa.
* Data communication will be in xml or Json format
* Outbound:When a request is sent to erp system
* Types:Ui,Business logic,Data

Integration interview Questions:  
<https://www.apexhours.com/salesforce-integration-interview-questions/>

A screenshot of a phone

Description automatically generated

**What is Salesforce Connect?**

[Salesforce Connect](https://www.apexhours.com/salesforce-connect-integration/) is a powerful App Cloud integration service, which enables users of Salesforce applications to seamlessly access and handle data stored in external sources, without leaving the Salesforce native environment. Instead of copying the data into your org, you can use external objects to access the data in real time via web service callouts.

**Interviewer: How can you integrate Salesforce with an external web service using the REST API?**

terviewee: To integrate Salesforce with a web service using the REST API, you need to define an Apex class as a REST resource and annotate it with the **`@RestResource’**annotation. The class methods should be annotated with appropriate HTTP method annotations such as **`@HttpGet`** or **`@HttpPost`**.

@RestResource(urlMapping='/myService/\*')  
global with sharing class MyService {  
 @HttpGet  
 global static void getAccountInfo() {  
 // Implement your logic to retrieve account information  
 }  
}

<https://medium.com/@saurabh.samirs/salesforce-integration-interview-questions-67388001e0e5>

API is an interface with which salesforce communicate with the external system

Authentication:Check whether genuine user or not

Authorization:what all the resources that can be processed by the authenticated user

Multi-factor authentication

Multi-factor authentication (or MFA) adds an extra layer of protection against threats like phishing attacks, increasing security for your business and your customers.

**Single Sign-On**

Single sign-on (SSO) is an authentication method that enables users to access multiple applications with one login and one set of credentials. For example, after users log in to your org, they can automatically access all apps from the App Launcher. You can set up your Salesforce org to trust a third-party identity provider to authenticate users. Or you can configure a third-party app to rely on your org for authentication.

Types of integration:  
User Interface  
Business logic  
Data

Flow of communication:Inbound,Outbound

Basic and local identity and access management:

Modern identity management:credentials will be handled by the identity provider as an external system and will build a trust with application and allow the user to access if he is verified by the identity provider

Types of Authentication:  
Basic authentication:  
API key base authentication:  
O Auth 2  
 Web server flow  
 Username password flow  
JWT(JSON web service token)

No Authentication:  
can used to access the resource without any authentication

API Key based Authentication:  
Based on api key it authenticates the user

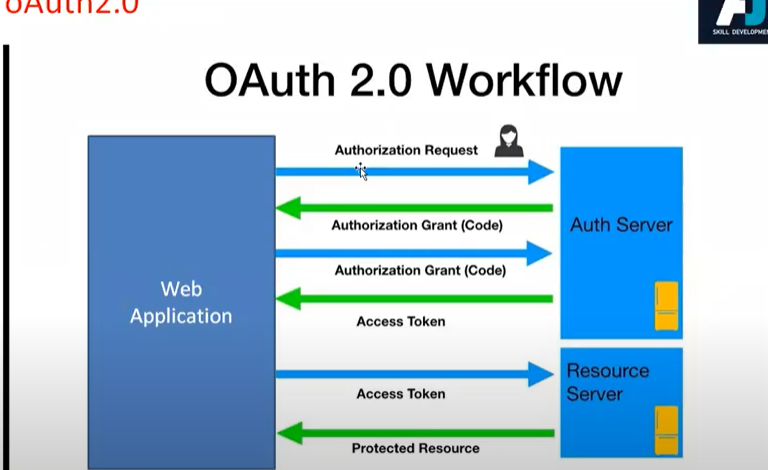
Rest methods:get,put,post,delete

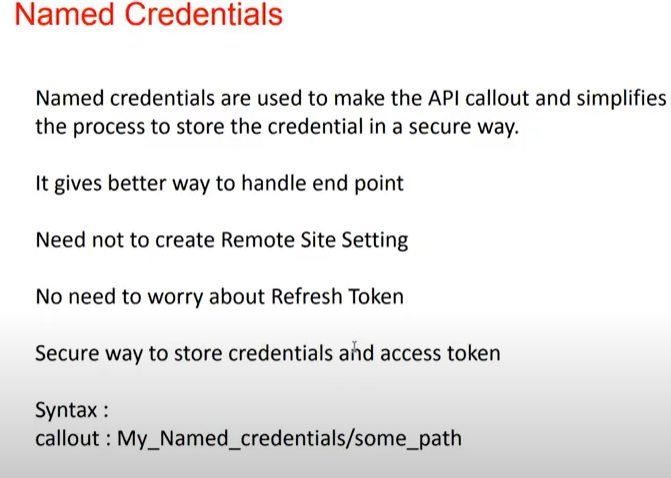
Status:200 Ok  
404:page not found  
500:internal server error

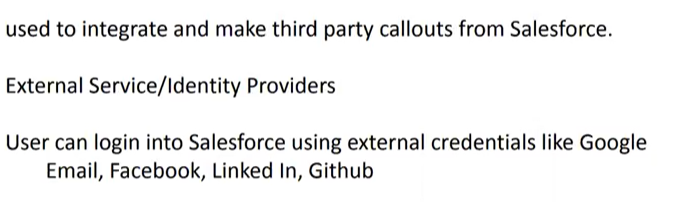
Apex Rest API Callout:  
HttpRequest  
HttpResponse  
Http  
JSON

Hit the endpoint with REST API:  
Create a class  
Add the end point to the remote site settings  
Add the end point to the custom label  
HttpRequest  
Http  
HttpResponse  
JSON

JSON2Apex:  
generate an apex class based on the JSON  
When we do post:

OAUTH 2:  
It has 2 server  
 Auth Server-Only authorization will be handle  
 Resource server-only resorce request will be handled  
  
Receiving a client key and secret key  
with the client key will hit auth server and auth server sends back a code(Secret key)  
Using auth code will do encoding(with client key,secret key) and sends a code to the auth server  
and auth server sends a access token back to the application (can be a bearer access token or refresh token(which can end in 1 hour)) and with the access token we can access the resource server

Named Credential:used to store authentication details like access token,user name ,pwd.After adding we don’t need to add url in remote site settings  
If ur doing rest api callout we can use named credential  
  
In Quick find box:named credentials

Auth Provider:  
  
If ur connecting with third party API with OAUTH authentication then we can use auth provider