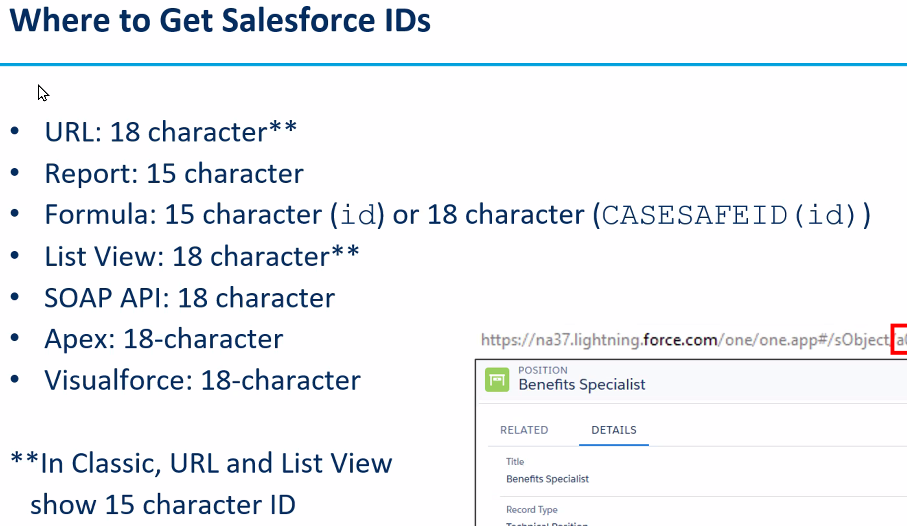
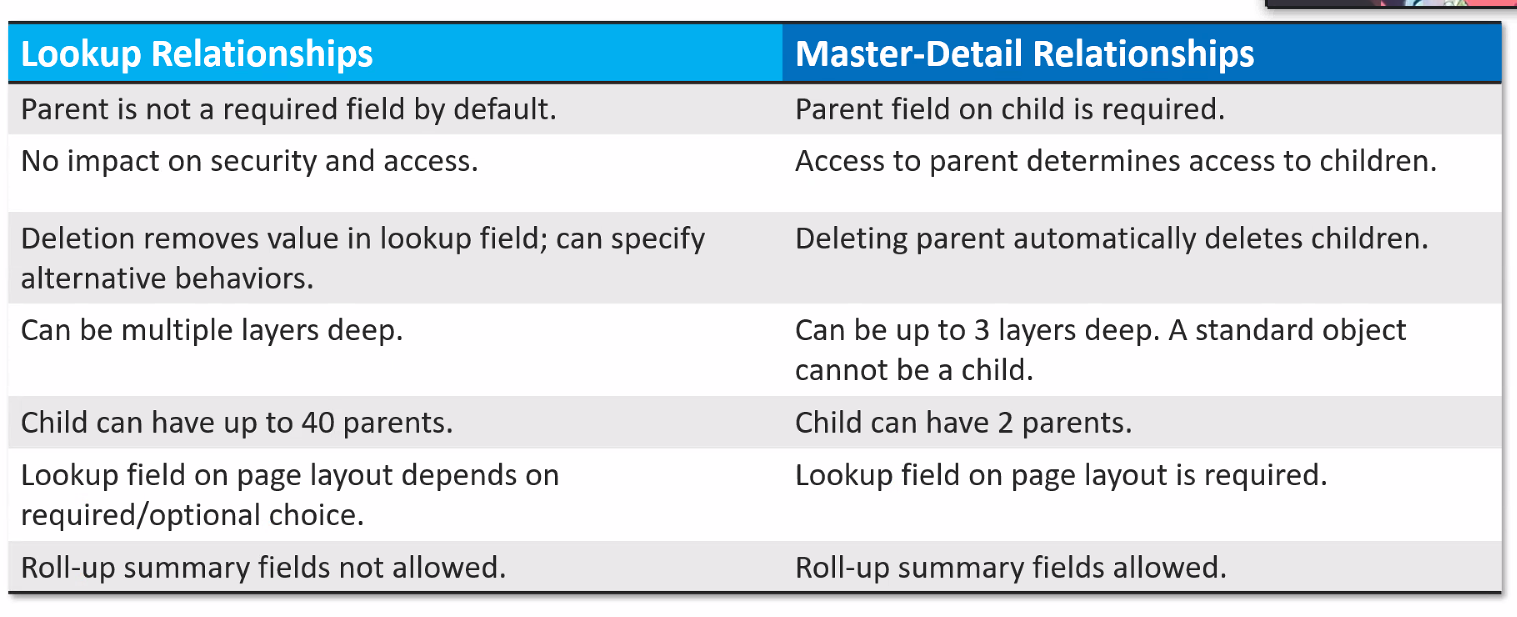
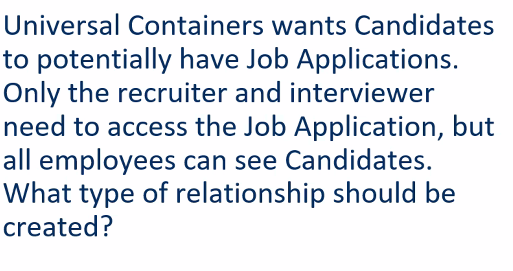
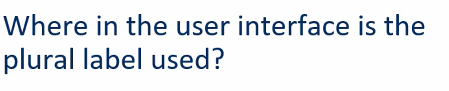
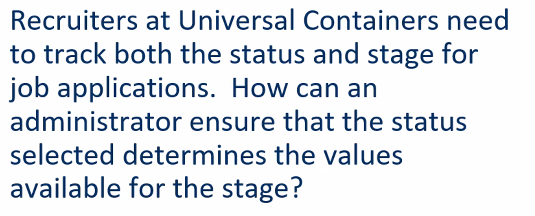
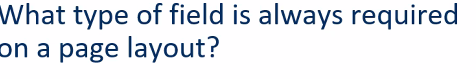
* Objects
  + Standard fields exist in every single object in salesforce
    - Created by
      * Points to user
      * Used for reference later on if needed
    - Currency
      * Used in case of multiple currencies
    - Customer story name
      * Also called name
      * Can be text or autogenerated
      * Indexed
      * A short description to identify a record’s content
      * Is usually unique
      * Autogenerated can overflow but will not replace, but will no longer be unique
      * Label of the field can change based on what makes sense for that specific object
    - Last modified by
    - Owner
      * Points to user or queue
    - ID field
      * Not able to be viewed in the record
      * Does not have to be unique but should be
      * Viewable in the URL
      * Read only
      * Unique across all salesforce organizations
      * Indexed
      * Comes in 2 forms
        + 15-character unique case sensitive
        + 18-character unique case sensitive
      * Where to find
        + URL
        + Report (15 character)
        + Formula
        + List view (18 character)
        + SOAP API (18 character)
        + Apex (18 character)
        + Visualforce (18 character)
        + 
  + A queue is a group of users
  + The owner can grant additional privileges
  + Custom fields
    - What we use to represent columns or information to track
    - Help text is what is shown when you hover over it
    - When you add one, you are not required to modify all records to account for it
      * Next time you modify the record, you will be required to update this data
  + Field dependencies
    - Think drop down for state then city in another drop down
    - Control field is a check box or picklist only
    - The dependent field is always a picklist
      * These both allow for multipicklists
    - To build one, create a field dependency
      * Follow setup wizard
  + Global picklists
    - May be reused on different objects
    - Enforces consistency and scalability
  + Relationships between objects
    - Foreign key points to primary key
      * Relates child to parent
    - No one to one relationship
    - One to many and many to one
      * Also many to many through a junction object
    - Self-relationships are from one object back to itself
      * Think manager and employee, a manager is still an employee
    - Types of relationship fields
      * Master-Detail
        + Tightly coupled relationships
        + Child cannot exist without a parent
        + Access to the detail record is controlled by the master record
        + Detail is automatically deleted when the master is deleted

Called cascade on delete

* + - * + Can add a lookup filter

This means that objects with relationship are returned in a search

* + - * + Only 2 masters per one child
        + Can reparent a child
        + Lookup
        + Hierarchy
        + Self-Relationship
        + With external objects we use indirect and external
      * Lookup relationship
        + The child record and parent record have independent sharing
        + Lookup field on the child can be optional or required
      * 
      * Lookup filters
        + A way to filter the different types of records
      * A self-relationship is a lookup relationship
      * Hierarchical relationship
        + Self-relationship on the user object
      * Junction object
        + Can link many position records to many job posting site records
        + One position can be posted to many sites
        + 2 master detail relationships
        + Allows for a many to many relationship
* Salesforce connect is how you can access stuff outside of salesforce
  + You need to create an external data source
  + Endpoint
  + Authentication information
  + Can relate to objects within salesforce
  + Metadata are stored in salesforce, but data are not stored in salesforce
  + You can create relationships between standard/custom objects and external objects
* Schema Builder
  + Not just visualization
  + Easy way to drag and drop things
* Review
  + 
    - Lookup
    - This is for security since childs inherit the security of the master
  + 
    - Tabs
    - Object creation
    - Related lists
  + 
    - Field dependencies
  + 
    - To interact with data outside of salesforce
    - Legacy purposes
  + 
    - Master detail
    - Need to be closely related to the master
* Security
  + 4 levels of security
    - Org security
      * Restricting access to your organization
      * Who can login
      * Where they can login from
      * What features of salesforce could they have access ti
    - Object security
      * CRUD operations on objects
        + Create, read, update, delete
      * This is for records they own
    - Field security
      * Of objects you have access to, what fields can you see/edit
    - Record level security
      * Of objects you have access to, whose records can you see
  + Some users only need read vs write permissions
  + User creation
    - When you create a user, someone that can login to your salesforce org, you need to do a few things
    - They need a
      * Name
      * Email
      * Username
      * Alias
      * License
        + A tool that is used to grant access to features in your org
        + Many types
        + Can only have 1

Can purchase more

* + - * + Types

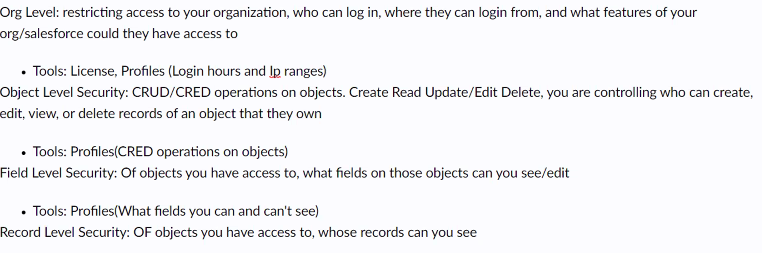
Salesforce

The one we will want to use for everything

Platform user

* + - * + Chatter free

Can only access chatter

* + - * Profile
      * Role (kinda)
    - Login access policies
      * Force relogin
  + States of a user
    - Active
      * Can login, business as usual
    - Frozen
      * Cannot login but still have license
      * Usually for someone who left but is coming back
      * Someone leaves the company, but things depend on that user existing
        + You keep them alive until you can fix the dependency issue
    - Locked
      * Tried too many times to login and must wait X time or contact your admin
    - Deactivated
      * Cannot login and no longer have a license
      * Meant for someone who leaves the company
      * This is in place of deletion
  + 
  + Every user will have a profile assigned to them
    - A profile defines what access to objects and data a user has
    - Three categories
      * Permissions
        + App permissions
        + System permissions
      * Access to org and objects
        + Field level security
        + Login hours and IP ranges
        + Standard/Custom object CRUD
      * User interface
        + Page layouts
        + Record types
        + Tabs
        + Applications
    - Permissions cascade
    - If you can’t access the org, you can’t access objects, fields or records
    - If you cannot read an object, you cannot read the fields
    - If you cannot edit an object, you cannot edit any fields or anyones records
    - Very rarely do you change licenses
      * Usually just change the profile