* Service cloud
  + Maintaining customer service in an efficient manner is what keeps customers around
  + This is salesforces customer and support application
  + Have to pay for it normally
  + Revolves around the case object
  + Cases are used to represent a customers product questions or problems
  + Service setup
    - Just a paired down version of the setup menu that is solely related to service cloud
  + Support processes
    - Service cloud automation tool that assists your very busy support agents by ensuring they follow the same steps to resolve a case
    - It is a way to create steps that a support agent should go through to resolve a case
    - The case status is what determines where in the lifecycle it is
    - When you create a record type of case you can associate it with a support process
    - If closing a case isn’t an option on your stairs in the UI you need to close a case in a different way
      * Add an action to a button and add it to the page layout
      * You can also enable a setting to allow this in the picklist value
  + Case comments
    - Sometimes the customer doesn’t know what the agent might need to solve the problem
    - They allow for a dialog between the customer and the agent after you log the case
    - Can be setup up easily
      * Enable the case comment related list into the case page layout
    - We want the customer and the agent to be able to see the comments
  + Case assignment tool
    - Ownership is pivotal to salesforce functionally
    - They determine the default owner of a case when it is first logged based of criteria
    - You can also use queues here for assignment
  + Case teams
    - Groups of users with a diverse set of skills that can work together to resolve a case
    - When your case team gets added to a case, you get additional permissions on the cases object than what you already have in the sharing level of security
    - Similar to granting ownership with the caveat that you can set the permissions they have
      * We can set it to:
        + Private
        + Read only
        + Read/write
  + Case auto-response rules
    - Used to deflect customers from reaching out just to see if you got their case
    - Send an automated email to the customer as soon as they create a case
  + Escalation rules
    - Cases can slip through the cracks
    - When this happens, we need to notify the owner and escalate to someone higher up
    - This time frame is customizable
  + Web to case
    - This allows external users to create cases
    - This can be through an experience site or another site you host is
    - Because cases are native to salesforce and not other software’s, we need to use web to case
    - Provides a form you can copy paste into your html
  + Email to case
    - More extensive to set up
    - Is a way to set up cases generated from an email received from customers
  + Lightning knowledge
    - Is a general FAQ section
    - Helps deflect customers for calling about basic things
    - These have data categories associated with them
* Sales cloud
  + Salesforce made app that provides a lot of sales functionality
  + Salesforces most popular application/product
  + Includes many standard objects like accounts, contacts, opportunities, leads, price books quotes, contracts and orders
  + Also includes lots of automation to help streamline the sales experience
  + Would normally need to purchase this
  + Leads
    - Is a potential customer
    - Lots of similarities to case creation
      * Web to lead
        + Can generate up to 500 leads per day
      * Auto response rules
      * Assignment rules
    - Lead lifecycle
      * We have lead processes: an automation tool that allows us to provide guidance for internal users and determine where the lead is in the recruitment lifecycle
      * Also uses the lead status picklist
    - If the status gets converted (we have confirmed that our criteria is met or that they are in fact going to do business) we can convert this into an account, contact and optionally an opportunity
    - Account teams
      * Identical to case teams
      * Provide more permissions for the people who are working on that account
  + Opportunity teams
    - Same as account teams
    - Have a few additional benefits
    - Enable collaboration on different sales
    - Can enable opportunity splits
      * Is a way of tracking the amount of each opportunity that can be attributed to each member of the opportunity team
  + Sales process
    - Same as lead and case processes but for the opportunity object
    - Uses the stage field for its steps
  + Path
    - Visual tool that is connected with a picklist field
    - Used to create step by step guidance
    - Each step will get its own screen
    - Enable from settings
  + Products and price books
    - Products are things your company sells
    - The price book are how we give prices to our products
      * We have several because we can track many different prices for the same product
    - The first time we want to give an item a price, we must add it to its standard price
      * This is the default price
    - After, you can add it to custom price books
  + Marketing cloud
    - All about marketing and its features
    - Tools can create marketing materials
    - In order to work with marketing features, you must have the marketing user feature license
    - Campaigns are the only marking feature in salescloud
    - Campaigns
      * Used to represent your marketing efforts
        + Email
        + Newsletter
        + Conference
        + Mail
        + Etc
      * Can form a hierarchy
      * To do this, set the parent campaign field on the child
* Governor limits (revisited)
  + Limitations on the org because we are hosted on the cloud with shared resources
  + Limits per transaction
  + SOQL
    - 100 queries synchronously
    - 200 queries asynchronously
  + SOSL
    - 20 queries
  + DML
    - 150 statements
  + Heap size
    - 6 MB when executing synchronously
    - 12 MB when executing asynchronously
  + CPU time
    - 10 seconds synchronously
    - 60 seconds asynchronously
  + Execution time
    - 10 minutes per transaction
    - Pretty hard to go over that one
* System namespace
  + Contains the classes and methods that are responsible for a lot of the functionality that we have seen up to this point
  + This contains all classes related to primitives, our collections, MATH, ENUM, exceptions, triggers, search and more
  + Database class
    - Contains methods to perform DML operations or interact with our database in some way
  + Limits class
    - Allows us to monitor our usage of our governor limits
    - There are a series of paired methods
      * One that will give you your current used amount
      * Other that will give you the entire limit
    - Example
      * SOQL
        + Limits.getQueries();

Num used

* + - * + Limits.getLimitQueries();

Total num

* + - * SOSL
        + Limits.getSOSLQueries();
        + Limits.getLimitSOSLQueries();
      * DML
        + Limits.getDMLStatemenets();
        + Limits.getLimitDMLStatements();
  + System class
    - System.debug();
    - System.assert();
    - System.assertEquals();
    - System.assertNotEquals();
    - System.equals();
    - System.runAs();
  + Test class
    - Test.startTest();
    - Test.stopTest();
    - Test.setCurrentPage();
    - Test.setMock();
  + UserInfo class
    - Returns all sorts of info about the running user
    - Userinfo.getFirstName();
    - Userinfo.getLanguage();
    - Userinfo.getLastName();
    - Userinfo.getProfileID();
    - Userinfo.getUserID();
  + Schema namespace
    - Hold a variety of classes that are used to get information about the configuration of our data model
  + Check governor limits if you put SOQL, SOSL, or DML in a loop
* Security and sharing
  + By default, apex runs in system mode
    - This means it can access anything
  + We may want to enforce our security model
    - We need to do so programmatically
  + Record level security
    - We have the ability for apex to listen to our sharing model by using keywords in the class definition
      * With sharing
        + Enforces record level security
      * Without sharing
        + Does not enforce record level security
        + Is default if not declared
      * Inherited sharing
        + Follows the rules of who called it
        + Can act as system or a user