Formulas Fields:

To get to formula editor, you will select an object, and then look at 'Fields & Relationships'.
By selecting 'New' and choosing 'formula' as data type, you can create a formula, and once you choose a few settings such as name and expected data to be returned, you will be taken to the editor.

Formula Editor notes:

- Has two types Simple and Advanced, shown as tabs.
- Insert Field button - opens menu of available fields. Will auto generate syntax for access as well.
- Insert Operator button - opens menu of available math and logic operators.
- Function Menu - view/insert formula functions. (the functions are pre-implemented by Salesforce) Some can be used as is. Others need parameters.
- Text Area - where you enter your formula. When writing, keep in mind:
    - Whitespace doesn't matter.
    - Formulas are case sensitive. Watch cap of field and object names
    - Standard order of operation applies with numbers.
- Check Syntax button to ensure that everything is in working order before saving. Syntax checker will alert you to spec problems, similar to Intellisense.

Power of One:
Create a custom formula field on an object: in the formula, just place the number 1. Apparently this can be very useful when dealing with large amounts of data.

Debugging:
Common issues:

- Missing parens: Missing ')'. You will get this same error if you leave out commas between parameters, so if the parens look good, check your commas
- Incorrect parameter type for function … expected Text, received Num.
- Incorrect parameter count for function
- Incompatible with expected data type - If you use pre-implemented functions, be sure you select a return type that aligns with what will be delivered. (num/date/text/etc)
- Field does not Exist. - Usually means check your spelling and capitalization.
- You may also get the above error if you leave out quotation marks around a text literal or hyperlink (for example, in your parameter parens)
- Unknown function - make sure Salesforce supports what you are trying to use, and double check your spelling of the function.

Roll-Up Summary Fields:
These fields calculate values from a set of related records. With these you can create a field that auto displays a value on a master record based on values from all of its detail records. The detail records MUST be directly related to the master via a master-detail relationship.

Defining a Roll-Up Summary Field:
You define a RUS field ON the object that is from the master side of the relationship. (ex: create a RUS on Account object, summarizing related opportunities.

Types of Summaries:

- COUNT - Totals number of all related records
- SUM - Totals values in field (field to aggregate). Only Num, Currency, and % fields avail
- MIN - displays lowest value of field in Fld to Agg for all related records. Num, Cur, %, date, date/time avail.
- MAX - displays highest value of field in Fld to Agg for all related records. Num, Cur, %, date, date/time avail.

Creating a RUS:
Object Mgr, select Obj, choose Fields and Relationships, Click New
Choose Roll-Up Summary as field type, click Next.
On Define page, Summarized Object is the detail object that you want to summarize.
Below, you select your RU type (sum, min,max, count) and the Field to Aggregate from the dropdown. *This is the field on the Summarized Object you want to use the records of.

Validation Rules:
Verifies that data entered by users in records meet the standards you specify before a save occurs. Validation rules can contain formula or expression to evaluate data in one or more fields and return a true/false (boolean). You can also include error messages.

Define a Validation Rule:
Valid rules can be made for objects, fields, campaign members, or case milestones.
“Setup’, Obj Mgr, select obj, left sidebar select ‘Validation Rules’. Click New

On Rule Edit page, Rule Name- use underscores? No empty spaces?
The Error Condition Formula area looks just like the Formula editor(advanced).
Below this, there is an area to enter an error message.