

# 10X Already Powerful Formulas

So many new possibilities!

- Search contacts by their account's website
  - No more formula character limits
  - Give access to an opportunity based on its campaign's net revenue
  - Trigger a flow annually on a contact's birthday
  - Much, much more. See how in remaining slides.
-

# Search contacts by account website

Or search any object by any field on its parent (lookup) object

- 1 Create a formula field on child (searched) object (e.g. Contact) pointing to field on the Account (e.g. Account.Website)
- 2 Create **regular** text field on child object. Index field if there is a large number of records.
- 3 Create a Formula Mapping. (See slide 6 for more details).
  - Object Name = Child object API Name (e.g. Contact)
  - Formula Field Name = The formula field API Name
  - Static Field Name = The text field API Name
- 4 Add the **regular text** field to the search layout

# No more formula character limits

- 1 Creatively split your intended formula into two formulas that can be joined together by a third formula.

e.g. `CASE(Purch__c, "Printer", "toner cartridges", "Camera", "Memory cards", "")` can be split into

`Formula1 = CASE(Purch__c, "Printer", "toner cartridges", "")`

`Formula2 = CASE(Purch__c, "Camera", "Memory cards", "")`

`Formula3 = Formula1__c & Formula2__c`

- 2 Create **regular** text fields to correspond to `Formula1` and `Formula2` and map them (see slide 6).

- 3 Your final formula field will be **Regular1\_\_c & Regular2\_\_c** (note: using `Formula1__c & Formula2__c` wont work)

# Give access to an opportunity based on its campaign's net revenue

Or use any formula (even if they include a field on a parent object) to set sharing criteria

- 1 Create formula number field on child/Opportunity **Campaign.AmountAllOpportunities-Campaign.ActualCost**, and create **regular** currency field.
- 2 Map the fields together (see slide 6).
- 3 Create a sharing rule using the **regular** (mapped) **currency** field for the criteria

# Trigger a flow annually on a contact's birthday

- 1 Create a formula field of type checkbox on Contact with the formula `MONTH(Birthdate) == MONTH(TODAY()) && DAY(Birthdate) == DAY(TODAY())`
- 2 Create a **regular** checkbox field and map it together (see next slide).
- 3 Trigger a flow (or workflow rule or process builder or apex trigger) when the **regular** checkbox is TRUE

# SETUP INSTRUCTIONS - 1

- 1 Find the "Formula Mappings" tab in the "Unleash Your Formula" app.
- 2 Simply hit **New** and create new **Formula Mapping** record.
- 3 Populate the fields on the left hand side. The right hand side will indicate that the mapping was synced successfully.

**Active** = TRUE

**Object Name** = The API Name of the Object that the formula resides in

**Formula Field Name** = The API Name of the Formula Field

**Static Field Name** = The API Name of the "Static" (i.e. non-Formula) field that we are using in the search, sharing rules and flow trigger criteria

# SETUP INSTRUCTIONS - 2

- 1 Find the "Setup" tab in the "Unleash Your Formula" app.
- 2 Click "Set Hourly Schedule" to set the sync to run on the hour, every hour
- 3 If you wish to set a different sync schedule, you can schedule the `unleashformula.ScheduleSync` class.

For example, the following apex executed anonymously, will sync the formulas (additionally) on minute 30 every hour.

```
String CRON = '0 30 * * * ?';
```

```
String jobId = system.schedule('30 Minute Sync', CRON, new  
unleashformula.ScheduleSync());
```



Email me with any questions!

[support@unleashyourformula.app](mailto:support@unleashyourformula.app)