Salesforce Test - Keobiz

AccountsTrigger :

- Here we have the logic to retrieve only the records were **MissionStatus__c** was **updated** to ***'canceled'***
- Some people/company might rather do this logic **directly in the function** *Accounts.updateRelatedContacts()*
  => but after some **performance tests** in my previous company, we concluded that doing it directly in the trigger is the most efficient and optimal way
- Then we have 2 functions called :
  - **UpdateMissionDate**() : before update, to directly update the Accounts date
  - **UpdateRelatedContacts**() : after update, to update the contacts info and send the api payload
    - Future method : cause we need this function to call an api
    - List<Id> as parameters, cause we can't pass a list of records in a future method

The update of the MissionCanceledDate__c field could also be made in the UpdateRelatedContacts() function, but this would require 2 more DML operations as we would be in an **after trigger,** so I made the decision of splitting it into 2 distinct functions

Accounts.updateRelatedContacts() :

- We retrieve all the **contactList** associated to the updated Accounts
- We retrieve all **AccountContactRelationList**, associated to these **contactList**
  - We have a formula field on AccountContactRelation :
    AccountMissionStatus__c that retrieves the Account Status
- We loop on each record of **AccountContactRelationList** :
  - If one of the Associated Account is still active we had the ContactId it to a list : **ListNoUpdate**
  - This means this contact is not to be updated (as one of his account is still active)
- We then retrieve all the Contacts that are **NOT** in this **ListNoUpdate**
- We finally update those contacts to **IsActive__c = false,** and call the api **updateContactAPI.sendContactInfos()**

This is the most efficient way I found to make the least DML operations in bulk.
I remain available if you have any questions !