Problem Statement

**Snapdeal product search, add product to cart and remove product from cart**

Retrieve the search terms from the excel file, search in snapdeal with conditions specified in details in the website snapdeal.com and write results to the console. Kindly refer to the steps below for specific conditions.

Pre-requisite – Save names of 3 items available in snapdeal to search in excel file.

# Automate the given scenario (with POM, POI and TestNG):

* Read the excel file and get the list of 3 items to an array
* Launch the browser **Chrome/Firefox/Edge**.
* Open the “**Snapdeal**” application using the URL "**https://**[**www.snapdeal.com**/](http://www.snapdeal.com/)".
* Handle the pop-ups if any.
* Search first item in the list , Enter “**Bluetooth Headphone**” (1 item in list) in search textbox and click "**Search**" button
* Verify the results displayed according to search criteria
* Click on the filter "**Sort By**" list box
* Select "**Popularity**" from the list box options
* Select price range between 700 to 3000 and click on "**GO**" button
* Get and save the first 5 Bluetooth earphone’s name and price into Excel sheet
* Add the First item to the cart.
* Verify the item is added to cart or not.
* Click on the “**remove**” button in the cart page and verify the cart is empty
* Similarly repeat for 2nd and 3rd search item
* Write the results of each module (Pass / Fail) to an excel file –   
  in the format   
  “Module name” – “Test result” - “Comments” in 3 different cells  
  “Login” “Pass” “Logged in successfully”.
* Close the browser.

# Key Automation Scope:

* Handling multiple browsers Chrome/Firefox
* Write functions for commonly used items – such as readExcel and writeToTextFile.
* No hardcoding, have generic code, especially for checking search results etc.
  + Instead of 3 items, if the excel file has 6 it should work (max 10)
  + In case the excel file is updated with similar search terms, the code should work without change.
* Using appropriate synchronization technique
* Exception handling
* Keep the excel file in the Java project, so that if the java project is shared, the code should work.
* Implementing POM, POI, TestNG
* Use Modularity and Best practices
* Coding standards
  + Code comments need to be present, so that the reviewer can understand easily.
  + Proper variable / class names to be given.
* The output file written also, should be accessible from the Java project.
* Write all verification results to console