**Adv day3**

**OOP**

1. Create your box object that contains books objects, ensure that you can

* create book object and add it to box object content property.
* count # of books inside box.
* delete any of these books in box according to book name or type.
* use toString() to tell its dimensions and how many books are stored in it.
* implement valueof() so that if there is more than one box object we can get total books in these boxes by adding the i.e. box1 has 5 books while box2 has 2 books, box1 + box2 should return 7

Note:

* there is no inheritance
* using of global variables, is not allowed.
* box object has the following properties: height, width, length, numOfBooks, material, content.
* The content property contains an array of books.
* book object has the following properties: title, numofChapters, author.
* you can define any function needed for both box and book objects.

1. **Use AJAX** to retrieve user data, and display: First Name, Last name and user avatar image (show the user image in an <img> tag) from the following test web API: <https://reqres.in/api/users/1>
   * Make a textbox where the user can enter user ID, and press display and then display the user with the given ID.
   * Use this web API: <https://reqres.in/api/users> to return all users data, and make a dropdown list and fill it with student’s name returning form the API. (Loop on them and display all users).
   * When user selects specific user (onchange event), display his data and image below the dropdown list.

**Cookies**

1. Make your own .js library to create, display and delete cookies, then use it in creating the required cookies to display a greeting message to your visitor with displaying an image as his profile pic referring to his gender, and inform him with his number of visits to the site.

* Display username and number of visits with font color according to his choices.
* Replace the registration page with the profile page using location object
* the library should have the following functions:
  + **getCookie(cookieName):** Retrieves a cookie value based on a cookie name.
  + **setCookie(cookieName,cookieValue,expiryDate):** Sets a cookie based on a cookie name, cookie value, and expiration date.
  + **deleteCookie(cookieName):** Deletes a cookie based on a cookie name.
  + **allCookieList():** returns a list of all stored cookies
  + **hasCookie(cookieName):** Check whether a cookie exists or not

**Note:**

handle any possible wrong call of all implemented function by firing error message.

**e.g there should be an error message if getCookie was called without passing any parameter.**