1. Write a program to store multiple products in an array. Each product belongs to a specific category and a vendor. Make sure to avoid data redundancy while story categories and vendors for each product. We need to store following details for products, categories & vendors respectively:
    a. **Products:** id, name, price, vendor_id, category_id
    a. **Categories:** id, name, status (can be active or inactive)
    b. **Vendor:** id, first_name, last_name, contact_number

    Your program should contain following functions:

    I. AddProduct(product)
    Params: "product" is an object having name, price, vendor_id & category_name
        1. **Notes:** Should create a ***random*** id for the product, fetch category_id from categories array by matching category name, store product and return a success message. Function should return an error message if no matching category found

    II. DeleteProductById(productId)
    Params: "productId" is id of the product
    Notes: Delete product if id matches & return success message else return a message saying "No matching Product found"
    f

    III. DeleteProductByName(productName)
    Params: "productName" is name of the product
    Notes: Delete product if name matches & return success message else return a message saying "No matching Product found"

    IV. GetCategoryProducts(categoryName)
    Params: "categoryName" is name of the category
    Notes: Return array of products which belong to the provided category

    V. GetVendorProducts(vendorFirstName, vendorLastName)
    Params:
    "vendorFirstName" is first name of the vendor
    "vendorLastName" is last name of the vendor
    Notes: Return array of products which belong to the provided vendor. To do so, first, you need to grab vendor_id from the vendors array. Grab the vendor_id either first name or last name matches.

    VI. GetCheapProducts(maxPrice)
    Params: "maxPrice" is the number which distinguishes the cheap products. Any product having price less than maxPrice is a cheap product
    Notes: Return array of all cheap products

VII. GetProductById(productId)
Params: “productId” is id of the product
Notes: Return product where id matches else return error message.

2. Write a program to sort array objects on the basis of libraryID. Write a function to sort the array in ascending and descending order. Function should take a parameter to decide sorting in ascending or descending order. Also, write another function to add new books in the array. Use Object construction function.

Example:

```
var library = [
  {
    title:  'The Road Ahead',
    author: 'Bill Gates',
    libraryID: 1254
  },
  {
    title: 'Walter Isaacson',
    author: 'Steve Jobs',
    libraryID: 4264
  },
  {
    title: 'Mockingjay: The Final Book of The Hunger Games',
    author: 'Suzanne Collins',
    libraryID: 3245
  }];
```

3. Write a program to store persons data as below. Use Object construction function.

```
        var persons = [
                {
                        name: “Ali”,
                        city: “Lahore”
                },
                {
                        name: “Obaid”,
                        city: “Karachi”
                },
        ]
```

Now, write a function that will create a new object containing each city as key and an array of person names as value. For example, for the above example out would be as follows:

```
{
        Lahore: [“Ali”],
        Karachi: [Obaid]
}
```