JAVA CORE – (CLASSES & OBJECTS, CONSTRUCTOR, PACKAGE, INHERITANCE, POLYMORPHISM, ABSTRACTION, INTERFACE)

#1: Assignment

WAP to create a class "Artist". Declare variables to store uniqueid and name of artist. Define functions createNewArtist() and ask user to enter artist details, showArtist() to show artist details.

Create objects of a class to handle records of 5 artists. Perform following operations on class objects

1. findDuplicates()
   Check that which are duplicates (same artistid and name)
2. showArrayObject()
   using Enhanced loop iterate each class object inside array and show "Artist" information. Function has argument of type "Object Array"

#2: Assignment

WAP to create following classes with following requirements,

1. **Auther**
   varaibles : auther_id, auther_name
   funtions :
   a. createNewAuther(int auther_id, String auther_name)
      function set values to class varaibles
   b. toString()
      function returns formatted string contains auther_id, auther_name using String.format() function
2. **Publication**
   varialbes : publication_id, publication_name
   funtions :
   a. createNewPublication(int publication_id, String publication_name)
      function set values to class varaibles
   b. toString()
      function returns formatted string contains publication_id, publication_name using String.format() function
3. **Book**
   variables : bookid, bookname, auther_obj,publication_obj
   functions :
   a. addBook(int bookid, String bookname, Auther a, Publication p)
      function set values to class Variables
   b. toString()
      function returns formatted string contains bookid, bookname, auther and publication details using String.format() function

Create records of 4 books using object array of class **Book.**

In main function Do the following bulk operation on class object as follow

1. searchBookByAuther()
   function takes argument as "Book" class's Object Array and auther name. function show the output as records of that books which match with auther_name
2. sortBookByName()
   write your own logic to sort books by their name. *[without using any library function]*

#3: Assignment

Following is abstract classes. Inherit and using parent class object solve the requirements

```
abstract class Login
{
String person_name, username, password;



        abstract void createLogin (String person_name,String username, String password);
        public boolean isValid()
        {
            // define a logic to meet the policy of username and password. [see end of the question]
        }
}
```

Define child classes as follow

```
class Facebook
{
      // Facebook class should show output after calling createLogin() as "Hello <person_name> Facebook Successfully
      Created your profile."

      public void requestFriend()
      {
            // Ask friend name here and prompt "Facebook friend <friend_name> requested. Will notify you once
            accepted
      }
}

class Google
{
      // Google class should show output after calling createLogin() as "Hi <person_name>, Google welcomes you."

      public void sendEmail()
      {
            // Ask email id and prompt "Your email to <emailid> sent successfully. "
      }
}
```

***Username** : should contain only small alphabates, minimum 6 and maximum 18 characters*

***Password :** should contains atleast one symbol, atleast one Capital alphabate, atleast one digit. Password length should be minimum 8 characters*

Sample Output:

Hello **Priyanka** Facebook Successfully Created your profile.

Enter Name of Friend to send friend request : **Anuja**

Facebook friend **Anuja** requested. Will notify you once accepted

Hi **Shreyas**, Google welcomes you.

Enter Email Id of recipient : aditya.kapor@gmail.com

Your email to aditya.kapor@gmail.com sent successfully.

#4: Assignment

WAP to create following classes with **default** and **parameterized** constructor

1. class "Labor"
   variables : laborname, age, town, contactno *[variables should be private]*
2. class "Lorry"
   variables : lorry_name, vehicle_number, weight_of_goods_in_tons *[variables should be private]*
3. class "Contracts"
   variables : contract_id,contract_person_name,labor_obj, lorry_obj *[variables should be protected]*

Define **toString() method** in class "Labor" , "Lorry" and "

Contactracts"which returns formatted output of String.format() method

Define main() function as follow,

1. create 5 "Labor" class object and must call parameterized constructor for 4 times and default constructor for 1 time
2. create 3 "Lorry" class object and must call only parameterized constructor
3. create 6 objects of class "Contracts" and call parameterized constructor at least for 4 times

Sample Output:

```
Labor
 John            37              california       +1 7878987565
 Smith           41              moscow           +7 9099828394
 Peter           29              bijing           +86 9987364664
 Michael         28              Francisco        +1 9388475773
 #Unkwon_name    0               #unkown_town     #unkwon_contactno

Lorry
 Panel truck     83774           20
 Flatbed truck   9929            65
 Dump truck      10084           105

Contracts
 91            James          Smith            Panel truck        83774
 92            Robert         Peter            Panel truck        83774
 93            Richard        Smith            Dump truck         10084
 94            Paul           Michael          Flatbed truck      9929
 0             0              #unkown_labor    #unkwon_lorry      0
 0             0              #unknown_labor   #unknown_lorry     0
```

#5: Assignment

WAP to create package and their classes as follow,

1. package **"Ticket"**
   Define following interface and classes with given requirements

   interface **"Ticket"**

   varialbes : price = 4.50 *[distance per km]*

   functions: setTicketDetails(int ticketno, String date, int distance, int qtys), float calculateTicketPrice()

2. package **"Vehicle"**
   Define flowing interface and classes with given requirements

   interface **"Driver"**

   variable : driver_min_age = 20, driver_max_age=60

   function: getDriverName(), getDriverAge(), getDriverCity()

   interface **"Efficiency"**

   variables : fuel_cap=20.00, avg_speed=21.00

   functions : String getVehicleName(), String getVehicleNumber()

   class **"Car"** *(implements "Driver" and "Efficiency", "Ticket")*

   variables : carname, carno, driver_name, age, driver_city, ticketno, ticket_date, distance, qtys
   [variables should be private]

   functions : constructor to read carname, carno, driver_name, age, driver_city

   class **"Bus"** *(implements "Driver" and "Efficiency")*

   variables : busname, busno, depot_city, pincode , driver_name, age, driver_city, ticketno, ticket_date, distance, qtys *[variables should be private]*

   functions : constructor to read busname, busno, depot_city, pincode , driver_name, age, driver_city

3. package **"projectmain"**
   define class **"Journey"** which contains main() method

Define main() function as follow,

1. create 3 **"Car"** class objects and with calling class methods set Driver, efficiency and ticket details
2. create 2 **"Bus"** class objects and with calling class methods set Driver, efficiency and ticket details

Sample Output:

Car1

```
 carname :mini-cooper        carno: MH 01  DS 9983     driver name : M. A. Sharma
 ticket no : 1033            date : 02/04/2015         passengers : 2              distance : 19 Km
 [ ======== Price per km : Rs. 4.50 ========      ======== Total Bill Amount : Rs. 171.00 ======== ]
```

Car2

```
 carname : acura             carno: GJ 23 AA 9983      driver name : C. S. Bhatt
 ticket no : 1034            date : 02/04/2015         passengers : 4              distance : 25 Km
```

```
  [ ======== Price per km : Rs. 4.50 ========       ======== Total Bill Amount : Rs. 450.00 ======== ]

Car3
  carname : acura              carno: MH 11 DE 1122     driver name : F, T. Gupta
  ticket no : 1035             date : 11/05/2016        passengers : 3              distance : 11 Km
  [ ======== Price per km : Rs. 4.50 ========       ======== Total Bill Amount : Rs. 148.50 ======== ]

Bus1
  busname : Wally Swift        busno: WB 07 DE 1027     depot city : kolkata              driver name : A. A. Roy
  ticket no : 45887            date : 19/02/2017        passengers : 25             distance : 120 Km
  [ ======== Price per km : Rs. 4.50 ========       ======== Total Bill Amount : Rs. 13500.00 ======== ]

Bus2
  busname : Henry Blofeld      busno: HP 07 AA 9874     depot city : shimla               driver name : V. T. Cruz
  ticket no : 45887            date : 19/02/2017        passengers : 15             distance : 200 Km
  [ ======== Price per km : Rs. 4.50 ========       ======== Total Bill Amount : Rs. 13500.00 ======== ]
```