```
dealer_name_list = ["-1"]  # creating a list for dealer names
dealer_country_list = ["-1"]  # creating a list for dealer countries
dealer_rating_list = ["-1"]  # creating a list for dealer ratings
dealer_contact_no_list = ["-1"]  # creating a list for dealer contact numbers


def add_record():  # creating function for adding records
    dealer_name_list.append(input("Enter the Dealer name  :"))
    # taking dealer name as input and appending to list
    dealer_country_list.append(input("Enter the  Dealer Country name  :"))
    # taking dealer country as input and appending to list
    dealer_rating_list.append(input("Enter the Rating of Dealer :"))
    # taking dealer rating as input and appending to list
    dealer_contact_no_list.append(input("Enter the Contact No  :"))
    # taking dealer contact no as input and appending to list


def view_record(sno):  # creating function for viewing records
    if sno <= len(dealer_name_list):  # making sure that the serial no is in the list limit
        print("Serial No           :", sno)  # printing the serial no of the record
        print("Dealer Name         :", dealer_name_list[sno])  # printing the dealer name of the record
        print("Dealer Country Name :", dealer_country_list[sno])  # printing the dealer Country name of the record
        print("Rating of Dealer    :", dealer_rating_list[sno])  # printing the dealer Rating of the record
        print("Contact No          :", dealer_contact_no_list[sno])  # printing the dealer Contact No of the record
        print("------------------------------------------------------------")  # separating different outputs
    else:  # if record is not available
        print("Record not available")  # printing "record not available" to tell the user


def update_record(sno):  # creating function for updating records
    if sno <= len(dealer_name_list):  # making sure that the serial no is in the list limit
        dealer_name_list[sno] = input("Enter the Dealer name  :")
        # taking dealer name as input and assigning in the list at the given serial no
        dealer_country_list[sno] = input("Enter the  Dealer Country name  :")
        # taking dealer country as input and  assigning in the list at the given serial no
        dealer_rating_list[sno] = input("Enter the Rating of Dealer :")
        # taking dealer rating as input and  assigning in the list at the given serial no
        dealer_contact_no_list[sno] = input("Enter the Contact No  :")
        # taking dealer contact no as input and assigning in the list at the given serial no
        print("Updated record:")  # making a heading of the updated record
        view_record(sno)  # printing the updated record
    else:  # if record is not available
        print("Record with this serial number is not available")  # telling the user that this record is not
available


def search_record():  # creating function for searching records
    choice = 1  # making a variable so that loop may start
    while choice != 5:  # making a loop that will continue till user enters 5
        print("Enter the perimeter you want to use to search:\n1 = Dealer name\n2 = Dealer Country name\n"
              "3 = Rating of Dealer\n4 = Contact No\n5 = Exit")  # printing the available searching options
        choice = input("Enter your choice :")  # taking the choice of the user as input
        if choice == "1":  # if the user inputs 1 searching using dealer name
            key = input("Enter the name of the dealer you want to find :")
            # taking an input for name of the dealer the user wants to find
            if key in dealer_name_list:  # checking if the required record is available
                for index in range(1, len(dealer_name_list)):  # making a loop to scan every entity in the list
                    if dealer_name_list[index] == key:  # if a record is found ..
                        view_record(index)  # print the record
            else:  # if record is not available
                print("No Record Found")  # telling the user that no record was found
        elif choice == "2":  # if the user inputs 2 searching using dealer country
            key = input("Enter the Country name of the dealer you want to find :")
            # taking an input for country of the dealer the user wants to find
            if key in dealer_country_list:  # checking if the required record is available
                for index in range(1, len(dealer_country_list)):  # making a loop to scan every entity in the list
                    if dealer_country_list[index] == key:
```

```
                    view_record(index)  # print the record
            else:  # if record is not available
                print("No Record Found")  # telling the user that no record was found
        elif choice == "3":   # if the user inputs 3  searching using dealer rating
            key = input("Enter the Rating of the dealer you want to find :")
            # taking an input for Rating of the dealer the user wants to find
            if key in dealer_rating_list:  # checking if the required record is available
                for index in range(1, len(dealer_rating_list)):  # making a loop to scan every entity in the list
                    if dealer_rating_list[index] == key:  # if a record is found ..
                        view_record(index)  # print the record
            else:  # if record is not available
                print("No Record Found")  # telling the user that no record was found
        elif choice == "4":   # if the user inputs 4 searching using dealer contact no
            key = input("Enter rhe Contact No of the dealer you want to find :")
            # taking an input for Contact No of the dealer the user wants to find
            if key in dealer_contact_no_list:  # checking if the required record is available
                for index in range(1, len(dealer_contact_no_list)):  # making a loop to scan every entity in the
list
                    if dealer_contact_no_list[index] == key:  # if a record is found ..
                        view_record(index)   # print the record
            else:  # if record is not available
                print("No Record Found")  # telling the user that no record was found
        elif choice == "5":  # if the user inputs 5
            break  # ending the loop
        else:  # if the user inputs an unavailable choice
            print("Select a proper choice")  # telling the user tto enter a proper choice


def main_menu():  # creating function for main menu
    check_1 = 0  # making a variable so that the loop initiates
    while check_1 != "5":  # making a loop that will continue till the user inputs 5
        print("What do you want to do\n1 = Add record\n2 = View Record\n3 = Update Record\n4 = Search Record\n5 =
Exit")
        # telling the user about the available options
        check_1 = input("Enter your choice number :")  # taking input from the user
        if check_1 == "1":  # if the user selects 1
            check_2 = "Y"  # making a variable so that the loop initiates
            while check_2 == "Y":  # making a loop that will continue till the user inputs something other than Y
                add_record()  # calling the add record function
                check_2 = input("If you want to add another record, enter Y :")
                # asking the user if he wants to enter another record
        elif check_1 == "2":  # if the user selects 2
            check_3 = 0  # making a variable so that the loop initiates
            while check_3 != -2:  # making a loop that will continue till the user inputs -2
                print("Enter the Serial number of the Record you want to View, select -1 to view all, select"
                      " -2 to Exit")
                # telling the user about the available options
                check_3 = eval(input("Enter your choice :"))  # taking input from the user
                if check_3 == -1:  # if the user selects -1
                    for num in range(1, len(dealer_name_list)):  # unpacking the list from index 1 to max
                        view_record(num)  # printing all available records
                elif check_3 == -2:  # if the user inputs -2
                    break  # ending the loop
                elif 0 < check_3:  # if the user inputs a serial number
                    view_record(check_3)  # printing the record at the requested serial no
                else:  # if the user inputs an unavailable option
                    print("Please select a proper choice")  # telling the user that this option is not available
        elif check_1 == "3":  # if the user inputs 3
            while True:  # making an infinite loop
                num = eval(input("Enter the serial no of the record you want to update :"))
                # asking the user for the serial no of the record he wants to update
                if 0 < num:  # making sure that the serial no is greater than 0
                    update_record(num)  # calling the update record function
                    break  # ending the loop
                else:  # if the entered serial number is less than or equal to 0
                    print("Serial no must ne an integer greater than 0")
                    # telling the user that serial no must be greater than 0
```

```
        elif check_1 == "4":  # if the user enters 4
            search_record()  # calling the search function
        elif check_1 == "5":  # if the user enters 5
            print("-------------Program terminated here-----------------")
            # telling the user that the program is terminated
            break  # ending the loop
        else:   # if the user enters an option other than those available
            print("Please Select a proper choice")   # telling the user to enter a proper choice


def main():  # creating function for main program
    choice = input("Press Y to enter menu or any other key to exit :")
    # asking the user if he wants to enter the main menu
    if choice == "Y":  # if the user inputs Y
        main_menu()  # calling the main menu function
    else:  # if the ser enters something other than Y
        print("-------------Program terminated here-----------------")
        # telling the user that the program is terminated


main()  # calling main function
```