***Dealer.h***

```
#import <Foundation/Foundation.h>
#import "Department.h"



@interface Dealer : NSObject<NSCoding>

@property(assign,nonatomic,readwrite) NSString *name;
@property(assign,nonatomic,readwrite) NSString *city;
@property(assign,nonatomic,readwrite) NSString *address;
@property(assign,nonatomic,readwrite) NSString *phone;
@property NSMutableArray<Department* > *departments;

-(void)addDepartment:(Department *) department;

@end
```

***Dealer.m***

```
#import "Dealer.h"
#import "Department.h"
@implementation Dealer

-(void)addDepartment:(Department *) department{
    if(_departments==nil){
        _departments=[[NSMutableArray alloc]init];
    }
    [_departments addObject:department];


}

- (id)initWithCoder:(NSCoder *)coder {
    self = [self init];
    self.city = [coder decodeObjectForKey:@"city"];
    self.name    = [coder decodeObjectForKey:@"name"];
    self.address = [coder decodeObjectForKey:@"address"];
    self.phone = [coder decodeObjectForKey:@"phone"];
    [[NSUserDefaults standardUserDefaults] setObject:[NSKeyedArchiver
archivedDataWithRootObject:_departments] forKey:@"departments"];
    return self;
}
```

```
- (void)encodeWithCoder:(NSCoder *)aCoder {
    [aCoder encodeObject:_city forKey:@"city"];
    [aCoder encodeObject:_name forKey:@"name"];
    [aCoder encodeObject:_address forKey:@"address"];
    [aCoder encodeObject:_phone forKey:@"phone"];
    [aCoder encodeObject:_departments forKey:@"departments"];
}



@end
```

***Employee.h***

```
#import <Foundation/Foundation.h>
#import "Department.h"

@interface Employee : NSObject <NSCoding>

@property NSString *firstName;
@property NSString *LastName;
@property NSString *city;
@property NSString *address;
@property NSString *phone;
@property NSString *zip;
@property NSString *ssn;
@property(assign,nonatomic,readwrite) Department *department;
@end
```

***Employee.m***

```
#import "Employee.h"

@implementation Employee

- (id)initWithCoder:(NSCoder *)coder {
    self = [self init];
    self.firstName = [coder decodeObjectForKey:@"firstName"];
    self.LastName    = [coder decodeObjectForKey:@"LastName"];
    self.city = [coder decodeObjectForKey:@"city"];
    self.address = [coder decodeObjectForKey:@"address"];
    self.phone    = [coder decodeObjectForKey:@"phone"];
    self.zip = [coder decodeObjectForKey:@"zip"];
    self.department = [coder decodeObjectForKey:@"department"];
```

```
    return self;
}

- (void)encodeWithCoder:(NSCoder *)aCoder {
    [aCoder encodeObject:_firstName forKey:@"firstName"];
    [aCoder encodeObject:_LastName forKey:@"LastName"];
    [aCoder encodeObject:_address forKey:@"address"];
    [aCoder encodeObject:_phone forKey:@"phone"];
    [aCoder encodeObject:_city forKey:@"city"];
    [aCoder encodeObject:_zip forKey:@"zip"];
    [aCoder encodeObject:_ssn forKey:@"department"];
}

@end
```

***Department.h***

```
#import <Foundation/Foundation.h>


@class Dealer;
@class Vehicle;
@class Employee;
@interface Department : NSObject<NSCoding>

@property(assign,nonatomic,readwrite) NSString *name;
@property(assign,nonatomic,readwrite) NSString *phone;
@property(assign,nonatomic,readwrite) Dealer *dealer;
@property NSMutableArray<Vehicle* > *vehicles;
@property NSMutableArray<Employee* > *employees;

-(void)addVehicle:(Vehicle *) vehicle;
-(void)listVehicles;

-(void)addEmployee:(Employee *) employee;
-(void)listEmployees;

-(NSMutableArray<Vehicle* > *) searchVehicles:(NSString *)
vehicleModel;

-(Vehicle *) modifyVehicle:(NSString *) vehicleName;

@end
```

Department.m

```
#import "Department.h"
#import "Dealer.h"
```

```
#import "Vehicle.h"
#import "Employee.h"

@implementation Department

-(void)addVehicle:(Vehicle *) vehicle{
    if(_vehicles==nil){
        _vehicles=[[NSMutableArray alloc]init];
    }
    [_vehicles addObject:vehicle];
}

-(void)listVehicles{
    for(Vehicle *v in _vehicles){
        NSLog(@"Inside here");
        NSLog(@"Vehicle Make %@",v.make);
        NSLog(@"Vehicle Model %@",v.model);
        NSLog(@"Vehicle Type %@",v.type);
        NSLog(@"Vehicle Price%@",v.price);
        NSLog(@"Vehicle Rating%li",v.rating);
        NSLog(@"Vehicle Date%@",v.year);


    }

}


-(NSMutableArray<Vehicle* > *) searchVehicles:(NSString *)
vehicleModel{
    NSMutableArray<Vehicle* >
*listOfVehicles=[[NSMutableArray<Vehicle* > alloc]init];
       for(Vehicle *v in _vehicles){
           if([v.model isEqualToString:vehicleModel]){
               [listOfVehicles addObject:v];
           }


       }

    return listOfVehicles;
}

-(Vehicle *) modifyVehicle:(NSString *) vehicleName{

    Vehicle *returnVehicle=nil;
    for(Vehicle *v in _vehicles){
        if([v.make isEqualToString:vehicleName]){
            returnVehicle=v;
        }
```

```
    }
        return returnVehicle;
}

-(void)addEmployee:(Employee *)employee{
    if (_employees==nil) {
        _employees=[[NSMutableArray alloc] init];
    }
    [_employees addObject:employee];
}

-(void)listEmployees{
    for(Employee *e in _employees){

        NSLog(@"Employee First Name %@",e.firstName);
        NSLog(@"Employee Last Name%@",e.LastName);
        NSLog(@"Employee City%@",e.city);
        NSLog(@"Employee Address%@",e.address);
        NSLog(@"Employee Phone%@",e.phone);
        NSLog(@"Employee Zipcode%@",e.zip);
        NSLog(@"Employee SSN%@",e.ssn);

    }

}


- (id)initWithCoder:(NSCoder *)coder {
    self = [self init];
    self.name = [coder decodeObjectForKey:@"name"];
    self.phone    = [coder decodeObjectForKey:@"phone"];
    self.dealer = [coder decodeObjectForKey:@"dealer"];
    [[NSUserDefaults standardUserDefaults] setObject:[NSKeyedArchiver
archivedDataWithRootObject:_vehicles] forKey:@"vehicles"];
    [[NSUserDefaults standardUserDefaults] setObject:[NSKeyedArchiver
archivedDataWithRootObject:_employees] forKey:@"employees"];
    return self;
}

- (void)encodeWithCoder:(NSCoder *)aCoder {
    [aCoder encodeObject:_name forKey:@"name"];
    [aCoder encodeObject:_phone forKey:@"phone"];
    [aCoder encodeObject:_dealer forKey:@"dealer"];
    [aCoder encodeObject:_vehicles forKey:@"vehicles"];
    [aCoder encodeObject:_employees forKey:@"employees"];

}

@end
```

***Vehicle.h***

```
#import <Foundation/Foundation.h>
#import "Department.h"

@interface Vehicle : NSObject<NSCoding>


@property NSString *make;
@property NSDecimalNumber *miles;
@property(readonly) NSString *model;
@property NSString *photo;
@property NSDecimalNumber *price;
@property NSInteger rating;
@property NSString *type;
@property NSDate *year;
@property Department *department;

- (id)initWithVehicleModel:(NSString *) model;

@end
```

**//Code Section for Second Part of Assigment**

***Vehicle.m***

```
#import "Vehicle.h"

@interface Vehicle ()

@property(readwrite) NSString *model;

@end

@implementation Vehicle


@synthesize model = _model;

- (id)initWithVehicleModel:(NSString *) model {
    self = [super init];
    if (self) {
        // This WILL work because of the extension.
        [self setModel:model];
    }
    return self;
}
```

```
- (id)initWithCoder:(NSCoder *)coder {
    self = [self init];
    self.make = [coder decodeObjectForKey:@"make"];
    self.miles    = [coder decodeObjectForKey:@"miles"];
    self.model = [coder decodeObjectForKey:@"model"];
    self.price = [coder decodeObjectForKey:@"price"];
    self.rating = [coder decodeIntegerForKey:@"rating"];
    self.type = [coder decodeObjectForKey:@"type"];
    self.year = [coder decodeObjectForKey:@"year"];
    self.department = [coder decodeObjectForKey:@"department"];


    return self;
}

- (void)encodeWithCoder:(NSCoder *)aCoder {
    [aCoder encodeObject:_make forKey:@"make"];
    [aCoder encodeObject:_miles forKey:@"miles"];
    [aCoder encodeObject:_model forKey:@"model"];
    [aCoder encodeObject:_price forKey:@"price"];
    [aCoder encodeInteger:_rating forKey:@"rating"];
    [aCoder encodeObject:_type forKey:@"type"];
    [aCoder encodeObject:_year forKey:@"year"];
    [aCoder encodeObject:_department forKey:@"department"];
}

@end

// Code Snippet for the first section of Assignment
```

***NSString+NSStringCategory.h***

```
#import <Foundation/Foundation.h>

@interface NSString (NSStringCategory)
-(NSString *)formatString:(NSString *) stringToFormat;
-(NSString *)formatNumberString:(NSString *) stringToFormat;
@end
```

***NSString+NSStringCategory.m***

```
#import "NSString+NSStringCategory.h"

@implementation NSString (NSStringCategory)

-(NSString *)formatString:(NSString *) stringToFormat{

    NSString *trimmedString = nil;
```

```
    NSCharacterSet *numbersSet = [NSCharacterSet
characterSetWithCharactersInString:@"0123456789!@#$%^&*() "];
    trimmedString = [stringToFormat
stringByTrimmingCharactersInSet:numbersSet];

    return trimmedString;

}

-(NSString *)formatNumberString:(NSString *) stringToFormat;{

    NSString *trimmedString = nil;
    NSCharacterSet *numbersSet = [NSCharacterSet
characterSetWithCharactersInString:@"abcdefghijklmnopqrstuvwxyz!@#$%^&
*() "];
    trimmedString = [stringToFormat
stringByTrimmingCharactersInSet:numbersSet];

    return trimmedString;
}

@end
```

***Main.m***

```
#import <Foundation/Foundation.h>
#import "Department.h"
#import "Employee.h"
#import "Dealer.h"
#import "Vehicle.h"
#import "NSString+NSStringCategory.h"



int alphabeticalCheck(NSString *str);
int phoneNumberCheck(NSString *phoneNo);
int alphaNumericCheck(NSString *str1);
void saveDatatoDisk(Dealer *dealer);
void loadDataFromDisk();


int main(int argc, const char * argv[]) {
    @autoreleasepool {


        char dName[2048]={ 0} ,dPhone[2048]={ 0 } ;
        char deaName[2048]={ 0},deaPhone[2048]={ 0 },deaCity[2048]={ 0
},deaAddress[2048]={ 0 },vMake[2048]={ 0 },
        vModel[2048]={ 0 },veType[2048]={ 0 };
```

```
        char eFirstName[2048]={0}, eLastName[2048]={0},
eCity[2048]={0}, eAddress[2048]={0}, ePhone[2048]={0},
        eZip[2048]={0}, eSSN[2048]={0};

        //Get the Dealer Details

        NSLog(@"Enter the Dealer Details");
        NSLog(@"Dealer Name(Only Alphabets):");
        scanf("%s",deaName);
        NSLog(@"Dealer Phone No(Only Numbers up to 10 digits");
        scanf("%s",deaPhone);
        NSLog(@"Dealer City(Only Alphabets):");
        scanf("%s",deaCity);
        NSLog(@"Dealer Address(Only Alphanumeric");
        scanf("%s",deaAddress);

        NSString *dealerName = [NSString
stringWithUTF8String:deaName];
        NSString *dealerPhone= [NSString
stringWithUTF8String:deaPhone];
        NSString *dealerCity = [NSString
stringWithUTF8String:deaCity];
        NSString *dealerAddress= [NSString
stringWithUTF8String:deaAddress];

        dealerName=[dealerName formatString:dealerName];
        dealerPhone=[dealerPhone formatNumberString:dealerPhone];
        dealerCity=[dealerCity formatString:dealerCity];


        if(alphabeticalCheck(dealerName)==1){
            NSLog(@" Error !! Enter only alphabets for Department
Name");
            exit(0);
        }
        if(phoneNumberCheck(dealerPhone)==1){
            NSLog(@"Error !! Enter only 10 digit Nos for Phone no");
            exit(0);
        }
        if(alphabeticalCheck(dealerCity)==1){
            NSLog(@" Error !! Enter only alphabets for Dealer City");
            exit(0);
        }

        if(alphaNumericCheck(dealerAddress)==1){
            NSLog(@" Error !! Enter only Number and alphabets for
Department Name");
            exit(0);
        }
```

```
        Dealer *dealer=[[Dealer alloc]init];
        dealer.name=dealerName;
        dealer.address=dealerAddress;
        dealer.city=dealerCity;
        dealer.phone=dealerPhone;

        //Get the Department Details

        NSLog(@"Enter the Department Details");
        NSLog(@"Department Name(Only Alphabets):");
        scanf("%s",dName);
        NSLog(@"Phone No(Only Numbers up to 10 digits");
        scanf("%s",dPhone);


        NSString *deptName = [NSString stringWithUTF8String:dName];
        NSString *deptPhone= [NSString stringWithUTF8String:dPhone];

        deptPhone=[deptPhone formatNumberString:deptPhone];
        deptName=[deptName formatString:deptName];


        if(alphabeticalCheck(deptName)==1){
            NSLog(@" Error !! Enter only alphabets for Department
Name");
            exit(0);
        }
        if(phoneNumberCheck(deptPhone)==1){
            NSLog(@"Error !! Enter only 10 digit Nos for Phone no");
            exit(0);
        }

        Department *dept=[[Department alloc]init];
        dept.name=deptName;
        dept.phone=deptPhone;
        dept.dealer=dealer;

        [dealer addDepartment:dept];



        double miles,price;
        NSInteger vRating;
```

```
        BOOL value=YES;
        while(value){

            NSLog(@"1: Add a Vehicle  2: List Vehicle  3: Search
Vehicle 4: Add Employee 5: List Employees 6: Modify a Vehicle 7: Load
Data from File");

            int option;
            scanf("%d",&option);

            switch(option){
                case 1:
                {
                    //Get the Vehicle Details

                    NSLog(@"Enter the Vehicle Details");
                    NSLog(@"Vehicle Make(Only Alphabets):");
                    scanf("%s",vMake);
                    NSLog(@"Vehicle Model(Only Alphabets):");
                    scanf("%s",vModel);
                    NSLog(@"Vehicle Type(Only Alphabets):");
                    scanf("%s",veType);
                    NSLog(@"Miles(Only Decimals):");
                    scanf("%lf",&miles);
                    NSLog(@"Price(Only Decimals):");
                    scanf("%lf",&price);
                    NSLog(@"Vehicle Rating");
                    scanf("%li",&vRating);


                    NSString *vmake=[NSString
stringWithUTF8String:vMake];
                    NSString *vmodel=[NSString
stringWithUTF8String:vModel];
                    NSString *vtype=[NSString
stringWithUTF8String:veType];

                    Vehicle *vehicle=[[Vehicle
alloc]initWithVehicleModel:vmodel];
                    vehicle.miles=[NSDecimalNumber
decimalNumberWithDecimal:[[NSNumber
numberWithDouble:miles]decimalValue]];
                    vehicle.price=[NSDecimalNumber
decimalNumberWithDecimal:[[NSNumber
numberWithDouble:price]decimalValue]];
                    vehicle.year=[NSDate date];
                    vehicle.rating=vRating;
                    vehicle.department=dept;
                    vehicle.make=[vmake formatString:vmake];
                    vehicle.type=[vtype formatString:vtype];
```

```
                    [dept addVehicle:vehicle];
                    saveDatatoDisk(dealer);

                    break;
                }
                case 2:
                {
                    //List the Vehicles

                    [dept listVehicles];
                    break;
                }

                case 3:
                {
                    //Search the Vehicles

                    NSString *searchModel;
                    char model[2048]={ 0 };
                    NSLog(@"Enter the Vehicle Model");
                    scanf("%s",model);
                    searchModel=[NSString stringWithUTF8String:model];
                    NSMutableArray<Vehicle* >
*searchedVehicles=[[NSMutableArray<Vehicle* > alloc]init];
                    searchedVehicles=[dept
searchVehicles:searchModel];
                    if(searchedVehicles!=nil){
                    for(Vehicle *veh in searchedVehicles){
                        NSLog(@"Vehicle Make %@",veh.make);
                        NSLog(@"Vehicle Model %@",veh.model);
                        NSLog(@"Vehicle Type %@",veh.type);
                        NSLog(@"%@",veh.price);
                        NSLog(@"%li",veh.rating);
                    }
                    }else{
                        NSLog(@"%@ Model Not Present",searchModel);
                    }
                    break;
                }

                case 4:
                {
                    //Get the Employee Details

                    NSLog(@"Enter the Employee Details");
                    NSLog(@"First Name(Only Alphabets):");
                    scanf("%s",eFirstName);
                    NSLog(@"Last Name(Only Alphabets):");
                    scanf("%s",eLastName);
                    NSLog(@"City(Only Alphabets):");
```

```
                    scanf("%s",eCity);
                    NSLog(@"Address(Only Alphabets):");
                    scanf("%s",eAddress);
                    NSLog(@"Phone(Only Numeric):");
                    scanf("%s",ePhone);
                    NSLog(@"Employee Zip(Only Numeric):");
                    scanf("%s",eZip);
                    NSLog(@"Employee SSN(Only Numeric):");
                    scanf("%s",eSSN);

                    Employee *employee = [Employee alloc];
                    NSString *efirstName = [NSString
stringWithUTF8String:eFirstName];
                    NSString *elastName = [NSString
stringWithUTF8String:eLastName];
                    NSString *ecity = [NSString
stringWithUTF8String:eCity];
                    NSString *eaddress = [NSString
stringWithUTF8String:eAddress];
                    NSString *ephone = [NSString
stringWithUTF8String:ePhone];
                    NSString *ezip = [NSString
stringWithUTF8String:eZip];
                    NSString *essn = [NSString
stringWithUTF8String:eSSN];

                    employee.firstName=[efirstName
formatString:efirstName];
                    employee.LastName=[elastName
formatString:elastName];
                    employee.city=[ecity formatString:ecity];
                    employee.address=eaddress;
                    employee.phone=[ephone formatNumberString:ephone];
                    employee.zip=[ezip formatNumberString:ezip];
                    employee.ssn=[essn formatNumberString:essn];


                    [dept addEmployee:employee];
                    saveDatatoDisk(dealer);
                    break;
                }

                case 5:
                {
                    //List the Employees

                    [dept listEmployees];
                    break;
                }
```

```
                case 6:
                {
                    NSString *searchVehicleName,*vehicleModel;
                    char name[2048]={ 0 },model[2048]={ 0 };
                    NSLog(@"Enter the Vehicle Name whose Model is to
be modified");
                    scanf("%s",name);
                    searchVehicleName=[NSString
stringWithUTF8String:name];
                    Vehicle *v=[dept modifyVehicle:searchVehicleName];
                    searchVehicleName=[NSString
stringWithUTF8String:name];
                    NSLog(@"The Model of the vehicle is %@",v.model);

                    NSLog(@"Enter the New Vehicle Model");
                    scanf("%s",model);

                    vehicleModel=[NSString
stringWithUTF8String:model];
                    [v initWithVehicleModel:vehicleModel];
                    break;
                }

                case 7:
                {
                    loadDataFromDisk();
                    break;
                }

                default:
                {
                    NSLog(@"Incorrect Details entered");
                    break;
                }
         }

            int c;
            NSLog(@"Do you wish to continue 1-CONTINUE/0 - EXIT");
            scanf("%d",&c);
            NSLog(@"%d",c);
            if(c == 1)
                value=YES;
            else
                value=NO;

        }
```

```
    }
    return 0;
}

//Validation for Alphabets

int alphabeticalCheck(NSString *str){

    NSString *myregex = @"^[A-Za-z]+$";
    int i=0;
    NSPredicate *deptNameTest = [NSPredicate
predicateWithFormat:@"SELF MATCHES %@", myregex];
    if (![deptNameTest evaluateWithObject:str])  {
        i=1;
    }
    return i;

}

//Validation for Phone Number

int phoneNumberCheck(NSString *phoneNo){

    NSString *phoneRegex = @"[0-9]{10}";
    NSPredicate *test = [NSPredicate predicateWithFormat:@"SELF
MATCHES %@", phoneRegex];
    BOOL matches = [test evaluateWithObject:phoneNo];
    int i=0;
    if(!matches){
        i=1;
    }
    return i;
}

//Validation for Alphanumeric

int alphaNumericCheck(NSString *str1){

    NSString *myregex = @"^[A-Za-z0-9 ]+$";
    int i=0;
    NSPredicate *deptNameTest = [NSPredicate
predicateWithFormat:@"SELF MATCHES %@", myregex];
    if (![deptNameTest evaluateWithObject:str1])  {
        i=1;
    }
    return i;
}
```

```
//Important Code Section for the last part of the assignment

void saveDatatoDisk(Dealer *dealer){


    [NSKeyedArchiver archiveRootObject:dealer
toFile:@"/Users/kuzhandaivel.n/Documents/dealer.plist"];


}

void loadDataFromDisk(){

    Dealer *dealer=[[NSKeyedUnarchiver
unarchiveObjectWithFile:@"/Users/kuzhandaivel.n/Documents/dealer.plist
"] retain];
    NSLog(@"Dealer Details");
    NSLog(@"Dealer Name : %@ | Dealer City : %@ | Dealer Address : %@
| Dealer Phone : %@" ,
dealer.name,dealer.city,dealer.address,dealer.phone);

NSLog(@"==========================================================
==========================================================
");

    for(Department *department in dealer.departments){
        NSLog(@"Department Details");
        NSLog(@"Department Name : %@ | Department Phone : %@" ,
department.name,department.phone);

NSLog(@"==========================================================
==========================================================
");
        for(Vehicle *vehicle in department.vehicles){
            NSLog(@"Vehicle Details");
            NSLog(@"Vehicle Make : %@ | Vehicle Model : %@ | Vehicle
Type : %@ | Vehicle Rating : %ld" ,
vehicle.make,vehicle.model,vehicle.type,(long)vehicle.rating);
            NSLog(@"Vehicle Price : %@ | Vehicle Year : %@ | Vehicle
Department : %@" ,
vehicle.price,vehicle.year,vehicle.department.name);

NSLog(@"==========================================================
==========================================================
");

        }

        for(Employee *employee in department.employees){
```

```
            NSLog(@"Employee Details");
            NSLog(@"Employee firstname : %@ | Employee  LastName : %@
| Employee city : %@ | Employee address : %@" ,
employee.firstName,employee.LastName,employee.city,employee.address);
            NSLog(@"Employee ssn : %@ | Employee zip : %@ | Employee
Department : %@" ,
employee.ssn,employee.zip,employee.department.name);

NSLog(@"=============================================================
=============================================================
");

        }

    }

}
```

***OUTPUT SCREENSHOTS:***

***Details Added***

```
2016-10-23 02:44:12.810 Assignment3[8951:268579] Enter the Dealer Details
2016-10-23 02:44:12.811 Assignment3[8951:268579] Dealer Name(Only Alphabets):
hotwire
2016-10-23 02:44:22.274 Assignment3[8951:268579] Dealer Phone No(Only Numbers up to 10 digits
9876541230
2016-10-23 02:44:25.641 Assignment3[8951:268579] Dealer City(Only Alphabets):
boston
2016-10-23 02:44:28.929 Assignment3[8951:268579] Dealer Address(Only Alphanumeric
75saintalphonsus
2016-10-23 02:44:34.093 Assignment3[8951:268579] Enter the Department Details
2016-10-23 02:44:34.093 Assignment3[8951:268579] Department Name(Only Alphabets):
sales
2016-10-23 02:44:52.456 Assignment3[8951:268579] Phone No(Only Numbers up to 10 digits
9876541230
2016-10-23 02:44:57.545 Assignment3[8951:268579] 1: Add a Vehicle  2: List Vehicle  3: Search Vehicle 4:
Add Employee 5: List Employees 6: Modify a Vehicle 7: Load Data from File
1
2016-10-23 02:45:19.584 Assignment3[8951:268579] Enter the Vehicle Details
2016-10-23 02:45:19.584 Assignment3[8951:268579] Vehicle Make(Only Alphabets):
Honda
2016-10-23 02:45:26.336 Assignment3[8951:268579] Vehicle Model(Only Alphabets):
civic
2016-10-23 02:45:29.352 Assignment3[8951:268579] Vehicle Type(Only Alphabets):
fourwheeler
2016-10-23 02:45:37.543 Assignment3[8951:268579] Miles(Only Decimals):
98
2016-10-23 02:45:39.976 Assignment3[8951:268579] Price(Only Decimals):
6789
2016-10-23 02:45:48.959 Assignment3[8951:268579] Vehicle Rating
9
2016-10-23 02:45:50.865 Assignment3[8951:268579] Do you wish to continue 1-CONTINUE/0 - EXIT
1
2016-10-23 02:46:54.478 Assignment3[8951:268579] 1
2016-10-23 02:46:54.478 Assignment3[8951:268579] 1: Add a Vehicle  2: List Vehicle  3: Search Vehicle 4:
Add Employee 5: List Employees 6: Modify a Vehicle 7: Load Data from File
4
2016-10-23 02:47:00.206 Assignment3[8951:268579] Enter the Employee Details
2016-10-23 02:47:00.206 Assignment3[8951:268579] First Name(Only Alphabets):
nirmal
2016-10-23 02:47:04.286 Assignment3[8951:268579] Last Name(Only Alphabets):
anand
2016-10-23 02:47:05.582 Assignment3[8951:268579] City(Only Alphabets):
chennai
2016-10-23 02:47:09.030 Assignment3[8951:268579] Address(Only Alphabets):
34alwarpetstreet
2016-10-23 02:47:20.589 Assignment3[8951:268579] Phone(Only Numeric):
9876541230
2016-10-23 02:47:23.909 Assignment3[8951:268579] Employee Zip(Only Numeric):
02120
2016-10-23 02:47:26.941 Assignment3[8951:268579] Employee SSN(Only Numeric):
20136958
2016-10-23 02:47:40.910 Assignment3[8951:268579] Do you wish to continue 1-CONTINUE/0 - EXIT
1
2016-10-23 02:47:43.629 Assignment3[8951:268579] 1
2016-10-23 02:47:43.629 Assignment3[8951:268579] 1: Add a Vehicle  2: List Vehicle  3: Search Vehicle 4:
Add Employee 5: List Employees 6: Modify a Vehicle 7: Load Data from File
2
2016-10-23 02:47:51.237 Assignment3[8951:268579] Inside here
2016-10-23 02:47:51.237 Assignment3[8951:268579] Vehicle Make Honda
2016-10-23 02:47:51.237 Assignment3[8951:268579] Vehicle Model civic
2016-10-23 02:47:51.237 Assignment3[8951:268579] Vehicle Type fourwheeler
2016-10-23 02:47:51.237 Assignment3[8951:268579] Vehicle Price6789
2016-10-23 02:47:51.237 Assignment3[8951:268579] Vehicle Rating9
2016-10-23 02:47:51.242 Assignment3[8951:268579] Vehicle Date2016-10-23 06:45:50 +0000
2016-10-23 02:47:51.242 Assignment3[8951:268579] Do you wish to continue 1-CONTINUE/0 - EXIT
1
2016-10-23 02:48:04.924 Assignment3[8951:268579] 1
2016-10-23 02:48:04.924 Assignment3[8951:268579] 1: Add a Vehicle  2: List Vehicle  3: Search Vehicle 4:
Add Employee 5: List Employees 6: Modify a Vehicle 7: Load Data from File
```

***The Generated Files Now***

dealer.plist 〉 No Selection

| Key | Type | Value |
|---|---|---|
| ▼ Root | Dictionary | (4 items) |
| $version | Number | 100,000 |
| ▼ $objects | Array | (34 items) |
| Item 0 | String | $null |
| ▶ Item 1 | Dictionary | (0 items) |
| Item 2 | String | boston |
| Item 3 | String | hotwire |
| Item 4 | String | 75saintalphonsus |
| Item 5 | String | 9876541230 |
| ▶ Item 6 | Dictionary | (1 item) |
| ▶ Item 7 | Dictionary | (0 items) |
| Item 8 | String | sales |
| Item 9 | String | 9876541230 |
| ▶ Item 10 | Dictionary | (1 item) |
| ▶ Item 11 | Dictionary | (1 item) |
| Item 12 | String | Honda |
| ▶ Item 13 | Dictionary | (6 items) |
| ▶ Item 14 | Dictionary | (2 items) |
| Item 15 | String | civic |
| ▶ Item 16 | Dictionary | (6 items) |
| Item 17 | String | fourwheeler |
| ▶ Item 18 | Dictionary | (1 item) |
| ▶ Item 19 | Dictionary | (2 items) |
| ▶ Item 20 | Dictionary | (2 items) |
| ▶ Item 21 | Dictionary | (2 items) |
| ▶ Item 22 | Dictionary | (1 item) |
| ▶ Item 23 | Dictionary | (0 items) |
| Item 24 | String | nirmal |
| Item 25 | String | anand |
| Item 26 | String | 34alwarpetstreet |
| Item 27 | String | 9876541230 |
| Item 28 | String | chennai |
| Item 29 | String | 02120 |
| Item 30 | String | 20136958 |
| ▶ Item 31 | Dictionary | (2 items) |
| ▶ Item 32 | Dictionary | (2 items) |
| ▶ Item 33 | Dictionary | (2 items) |
| $archiver | String | NSKeyedArchiver |
| ▶ $top | Dictionary | (0 items) |

## ***Changing the model now. Vehicle Details Modified***

```
2016-10-23 02:48:04.924 Assignment3[8951:268579] 1
2016-10-23 02:48:04.924 Assignment3[8951:268579] 1: Add a Vehicle  2: List Vehicle  3: Search Vehicle
4: Add Employee 5: List Employees 6: Modify a Vehicle 7: Load Data from File
6
2016-10-23 02:51:39.766 Assignment3[8951:268579] Enter the Vehicle Name whose Model is to be modified
Honda
2016-10-23 02:51:47.096 Assignment3[8951:268579] The Model of the vehicle is civic
2016-10-23 02:51:47.096 Assignment3[8951:268579] Enter the New Vehicle Model
Accord
2016-10-23 02:51:52.640 Assignment3[8951:268579] Do you wish to continue 1-CONTINUE/0 - EXIT
1
2016-10-23 02:51:54.600 Assignment3[8951:268579] 1
2016-10-23 02:51:54.600 Assignment3[8951:268579] 1: Add a Vehicle  2: List Vehicle  3: Search Vehicle
4: Add Employee 5: List Employees 6: Modify a Vehicle 7: Load Data from File
```

## ***Now the Plist has changed***

dealer.plist 〉 No Selection

| Key | Type | Value |
|---|---|---|
| ▼ Root | Dictionary | (4 items) |
| $version | Number | 100,000 |
| ▼ $objects | Array | (34 items) |
| Item 0 | String | $null |
| ▶ Item 1 | Dictionary | (0 items) |
| Item 2 | String | boston |
| Item 3 | String | hotwire |
| Item 4 | String | 75saintalphonsus |
| Item 5 | String | 9876541230 |
| ▶ Item 6 | Dictionary | (1 item) |
| ▶ Item 7 | Dictionary | (0 items) |
| Item 8 | String | sales |
| Item 9 | String | 9876541230 |
| ▶ Item 10 | Dictionary | (1 item) |
| ▶ Item 11 | Dictionary | (1 item) |
| Item 12 | String | Honda |
| ▶ Item 13 | Dictionary | (6 items) |
| ▶ Item 14 | Dictionary | (2 items) |
| Item 15 | String | Accord |
| ▶ Item 16 | Dictionary | (6 items) |
| Item 17 | String | fourwheeler |
| ▶ Item 18 | Dictionary | (1 item) |
| ▶ Item 19 | Dictionary | (2 items) |
| ▶ Item 20 | Dictionary | (2 items) |
| ▶ Item 21 | Dictionary | (2 items) |
| ▶ Item 22 | Dictionary | (1 item) |
| ▶ Item 23 | Dictionary | (0 items) |
| Item 24 | String | nirmal |
| Item 25 | String | anand |
| Item 26 | String | 34alwarpetstreet |
| Item 27 | String | 9876541230 |
| Item 28 | String | chennai |
| Item 29 | String | 02120 |
| Item 30 | String | 20136958 |
| ▶ Item 31 | Dictionary | (2 items) |
| ▶ Item 32 | Dictionary | (2 items) |
| ▶ Item 33 | Dictionary | (2 items) |
| $archiver | String | NSKeyedArchiver |
| ▶ $top | Dictionary | (0 items) |

## ***NSString + NSCategory Output***

Added the below details but the Extension class NSCategory removed the unwanted characters from the output before saving the file .

```
2016-10-23 02:54:52.970 Assignment3[9126:273947] Enter the Dealer Details
2016-10-23 02:54:52.970 Assignment3[9126:273947] Dealer Name(Only Alphabets):
hotwire1234
2016-10-23 02:54:57.099 Assignment3[9126:273947] Dealer Phone No(Only Numbers up to 10 digits
9876541230a
2016-10-23 02:55:01.107 Assignment3[9126:273947] Dealer City(Only Alphabets):
boston12
2016-10-23 02:55:04.595 Assignment3[9126:273947] Dealer Address(Only Alphanumeric
75saintalphonsus
2016-10-23 02:55:10.238 Assignment3[9126:273947] Enter the Department Details
2016-10-23 02:55:10.238 Assignment3[9126:273947] Department Name(Only Alphabets):
department1
2016-10-23 02:55:16.403 Assignment3[9126:273947] Phone No(Only Numbers up to 10 digits
9876541230
2016-10-23 02:55:20.850 Assignment3[9126:273947] 1: Add a Vehicle  2: List Vehicle  3: Search Vehicle 4:
Add Employee 5: List Employees 6: Modify a Vehicle 7: Load Data from File
1
2016-10-23 02:55:32.570 Assignment3[9126:273947] Enter the Vehicle Details
2016-10-23 02:55:32.570 Assignment3[9126:273947] Vehicle Make(Only Alphabets):
honda
2016-10-23 02:55:35.706 Assignment3[9126:273947] Vehicle Model(Only Alphabets):
civic
2016-10-23 02:55:37.418 Assignment3[9126:273947] Vehicle Type(Only Alphabets):
fourwheeler
2016-10-23 02:55:40.738 Assignment3[9126:273947] Miles(Only Decimals):
98
2016-10-23 02:55:43.722 Assignment3[9126:273947] Price(Only Decimals):
9876
2016-10-23 02:55:44.954 Assignment3[9126:273947] Vehicle Rating
9
2016-10-23 02:55:46.268 Assignment3[9126:273947] Do you wish to continue 1-CONTINUE/0 - EXIT
```

No Selection

dealer.plist

dealer.plist 〉 No Selection

| Key | Type | Value |
|---|---|---|
| Root | Dictionary | (4 items) |
| $version | Number | 100,000 |
| $objects | Array | (24 items) |
| Item 0 | String | $null |
| Item 1 | Dictionary | (0 items) |
| Item 2 | String | boston |
| Item 3 | String | hotwire |
| Item 4 | String | 75saintalphonsus |
| Item 5 | String | 9876541230 |
| Item 6 | Dictionary | (1 item) |
| Item 7 | Dictionary | (0 items) |
| Item 8 | String | department |
| Item 9 | String | 9876541230 |
| Item 10 | Dictionary | (1 item) |
| Item 11 | Dictionary | (1 item) |
| Item 12 | String | honda |
| Item 13 | Dictionary | (6 items) |
| Item 14 | Dictionary | (2 items) |
| Item 15 | String | civic |
| Item 16 | Dictionary | (6 items) |
| Item 17 | String | fourwheeler |
| Item 18 | Dictionary | (1 item) |
| Item 19 | Dictionary | (2 items) |
| Item 20 | Dictionary | (2 items) |
| Item 21 | Dictionary | (2 items) |
| Item 22 | Dictionary | (2 items) |
| Item 23 | Dictionary | (2 items) |
| $archiver | String | NSKeyedArchiver |
| $top | Dictionary | (0 items) |