

## CS348 – Spring 2017 - Project 2

### PL/SQL

Due Sunday March 24, 2019 at 11:59PM on Blackboard.

(There will be a 10% penalty for each late calendar day. After five calendar days, the project will not be accepted.)

In this project, you are asked to complete 5 procedures with PL/SQL.

Notes:

1. The schema definition of database tables and sample test data are provided in **tables.sql** and **data.sql**, respectively. You need to use **droptables.sql** to clean your database before you start this project because test data may be different from the data used in Project 1. Study and acquaint yourself with the schema (in tables.sql) and the data in the tables (in data.sql). This will make it easier to understand the required procedures.
2. You should finish all your work in **answer.sql**. Skeleton code for procedures is already provided in answer.sql. Oracle will give error messages if you do not finish all of the procedures, so you can comment the unfinished ones during development.
3. Please do not change the names of procedures in answer.sql.
4. You can assume that all possible inputs to Procedures 1-4 will be legal input, but we will test illegal input in Procedure 5. So, the exception block is not required for testing Procedures 1-4. Please, refer to the requirement of Procedure 5 for details.
5. Submit your answer via **Blackboard**.
6. Hint: You may want to use the command “**show errors;**” to debug your procedures.

The detailed requirements of each procedure are listed below:

## 1. CarRentalSiteDetail

Create a procedure that shows the detailed information of a specific **CarRentalSite**, given the **CarRentalSiteId** as input. The detailed information should include the following:

- a. CarRentalSite Name
- b. CarRentalSite City
- c. CarRentalSite total rentals in the Rentals table.
- d. The most popular **compact** car and the **number of days rented** in the Rentals table. We define the most popular compact car as the one having the largest number of rental days in the Rentals table. ***You can assume there is only one most popular compact car.***

The output should be in the following format:

**CarRentalSiteDetail(1) ;**

```
CarRentalSite Name: Hertz
CarRentalSite City: Lafayette
CarRentalSite Total Rentals: 6
Most Popular Compact Car: Chevy Spark
Total Days Rented: 15
```

## 2. MonthlyBusinessRentalsReport

Create a procedure that can generate a simple report for the **business** rentals of all months in the database. For each month, you need to list the total number of **business** rentals and the CarRentalSites that have business rentals in that month. Only display the months with **business** rentals. The ordering of months should be from the earliest to the latest and the ordering of CarRentalSites should be by **CarRentalSiteName** attribute.

The output should be in the following format:

### **MonthlyBusinessRentalsReport;**

```
Total Business Rentals in 2018-1: 2
In Car Rental Sites:
- Hertz: 36 days
Total Business Rentals in 2018-2: 2
In Car Rental Sites:
- Alamo: 10 days
- Hertz: 14 days
Total Business Rentals in 2018-4: 1
In Car Rental Sites:
- Enterprise: 2 days
Total Business Rentals in 2018-5: 3
In Car Rental Sites:
- Avis: 2 days
- Budget: 3 days
- Hertz: 25 days
Total Business Rentals in 2018-6: 1
In Car Rental Sites:
- Alamo: 10 days
```

### 3. MostandLeastProfitCarIndiana

Create a procedure that can generate a simple report for the cars with **least average profit** and **most average profit** in each car **category** ONLY for cars provided by car dealers located in **Indiana**. Here, we define the profit of a car as the average of :  
(Rentals.RentalRate - Car.SuggestedDealerRentalPrice)  
since the car appears in different rental entries with different rental rates. Notice that there may be more than one car with the same average profit. Display all the car names and their profit for all categories ordered by CarName.

The output should be in the following format:

**MostandLeastProfitCarIndiana;**

```
Least Profit in compact
- Ford Focus: 4
- Nissan Versa: 4
- Toyota Yaris: 4
Least Profit in luxury
- Porsche: 40
Least Profit in van
- Chrysler: 2
Most Profit in compact
- Chevy Spark: 5
Most Profit in luxury
- Audi: 45
Most Profit in van
- Honda Odyssey: 9
```

#### 4. **BusinessRentalCategory**

Create Procedure **BusinessRentalCategory** that will populate a table named **BusinessRentalCategoryTable** with the most recent information about how many car rentals of each of the following car categories: compact, SUV, and luxury, respectively.

The output after running “**select \* from BusinessRentalCategoryTable;**” should look like:

CARRENTALSITEID	COMPACT	LUXURY	SUV
1	1	0	0
2	1	0	0
3	2	0	0
4	0	1	0
5	0	0	1

## 5. **CarCategoryInventoryInfo**

Given a **CarSiteId** as input, list the **CarRentalSiteNames** and the **number of available cars** in the car inventory. List the names of car rental sites (in alphabetical order) and the number of available cars in the car inventory. You may need to use **Exception** in PL/SQL to prevent your procedure from crashing if given an invalid customer id or an invalid product id.

The output of a valid input should be similar to:

**CarCategoryInventoryInfo (1) ;**

```
CarRentalSiteId: 1  
CarRentalSiteName: Hertz  
CarName: Chevy: 20  
CarName: Chevy Spark: 10  
CarName: Chrysler: 10  
CarName: Honda CRV: 3
```

The output of invalid inputs should be similar to:

**CarCategoryInventoryInfo (111) ;**

```
CarRentalSiteId: 111  
Invalid CarRentalSiteId!
```