



School of Computing, Creative Technology and Engineering

Module: Database Systems
Academic Year: 2019/20
Level 5: Semester A
Assignment Title: DS Assessment 1 Component 2
Date Due: 15th January 2020.
Tutor: Rohit Raj Pandey.
Student Name: Avhimantu Sapkota
Student ID: c7202323

Table of Contents

Table of Contents.....	2
Introduction	3
Task 1: Implementation and Testing	4
Trigger 1: Code and Test Implementation.....	4
Trigger 2: Code and Test Implementation.....	7
CRUD Triggers.....	10
Task 2: Application Builder.....	18
Application: CARS4U Cars Hire Company.	18
Home Page (with CSS).....	18
Navigation Menu with Icons	19
Dashboard	22
Master Detail Forms.....	23
Insertion Forms for Master Page.....	26
Graphs and Charts.....	32
Calendars	34
Reports of Assignment Component 1.....	36
Reports with Forms.....	39
Multiple User Access	45
Application Administrator	48
Application's Reader – CARS4U Customer.....	49
Application's Contributor – CARS4U Team	51
Bibliography	53
Appendix.....	53

Introduction

This is a report of the assignment of Database Systems, a syllabus curriculum of Leeds Beckett University which is also taught in The British College. Students were supposed to create EERM, ERD, tables with all valid data, views and reports in the first phase of the assignment. This report is based on the second phase of the assignment which is the extension version of the previous report. Thus, this report makes use of all those tables, views and reports created in component 1 of the assignment.

In this particular assignment, we were supposed to choose one option to perform key database solutions. I have chosen the second option i.e. Importance of programming, validations and other Apex features to ensure data quality. However, choosing option two is not quite sufficient to ensure the complete application environment therefore I have also added some features of option one into my assignment.

This report consists of screenshots of the task performed during the creation and implementation of various features of the application. Screenshots of triggers created to perform particular operations, application's home page, dashboard, master detail form, forms to insert data, charts, calendar, reports, other validations and multiple user login process have been included along with a brief description.

Task 1: Implementation and Testing

Trigger 1: Code and Test Implementation.

This trigger has been created for HIRE_INVOICE table during INSERTION or UPDATE process in order to calculate discount for hire. This trigger has various conditions to fulfill before inserting or updating into a table. Those conditions are:

- If the total amount to be paid is higher than 10000 then 10% discount.
- If the total amount to be paid is higher than 15000 then 15% discount.
- If the total amount to be paid is higher than 20000 then 20% discount.
- If the total amount to be paid is higher than 25000 then 25% discount.
- If the total amount to be paid is higher than 30000 then 30% discount.
- Finally, no discount will be offered if the total amount to be paid is less than 10000.

- ❖ Screenshot of trigger code and statement that says trigger created:

The screenshot shows the Oracle APEX interface with the SQL Workshop tab selected. A new trigger named 'ASSIGNMENT_TRIGGER_1' is being created. The code in the editor is as follows:

```
1 DROP TRIGGER UPDATE_DISCOUNT;
2
3 CREATE OR REPLACE TRIGGER UPDATE_DISCOUNT
4 BEFORE INSERT OR UPDATE ON HIRE_INVOICE
5 FOR EACH ROW
6 BEGIN
7 IF (:NEW.TOTAL_AMOUNT>=10000 AND :NEW.TOTAL_AMOUNT<15000)
8 THEN
9   :NEW.LOYALTY_DISCOUNT:=:NEW.TOTAL_AMOUNT*.1;
10  :NEW.TOTAL_AMOUNT:=:NEW.TOTAL_AMOUNT - :NEW.LOYALTY_DISCOUNT;
11  DBMS_OUTPUT.PUT_LINE('YOU HAVE BEEN OFFERED WITH 10% DISCOUNT');
12
13 ELSIF (:NEW.TOTAL_AMOUNT>=15000 AND :NEW.TOTAL_AMOUNT<20000)
14 THEN
15   :NEW.LOYALTY_DISCOUNT:=:NEW.TOTAL_AMOUNT*.15;
16  :NEW.TOTAL_AMOUNT:=:NEW.TOTAL_AMOUNT - :NEW.LOYALTY_DISCOUNT;
17  DBMS_OUTPUT.PUT_LINE('YOU HAVE BEEN OFFERED WITH 15% DISCOUNT');
18
19 ELSIF (:NEW.TOTAL_AMOUNT>=20000 AND :NEW.TOTAL_AMOUNT<25000)
20 THEN
21   :NEW.LOYALTY_DISCOUNT:=:NEW.TOTAL_AMOUNT*.2;
22  :NEW.TOTAL_AMOUNT:=:NEW.TOTAL_AMOUNT - :NEW.LOYALTY_DISCOUNT;
23  DBMS_OUTPUT.PUT_LINE('YOU HAVE BEEN OFFERED WITH 20% DISCOUNT');
24
25 ELSIF (:NEW.TOTAL_AMOUNT>=25000 AND :NEW.TOTAL_AMOUNT<30000)
26 THEN
27   :NEW.LOYALTY_DISCOUNT:=:NEW.TOTAL_AMOUNT*.25;
28  :NEW.TOTAL_AMOUNT:=:NEW.TOTAL_AMOUNT - :NEW.LOYALTY_DISCOUNT;
29  DBMS_OUTPUT.PUT_LINE('YOU HAVE BEEN OFFERED WITH 25% DISCOUNT');
30
31 ELSIF (:NEW.TOTAL_AMOUNT>=30000)
32 THEN
33   :NEW.LOYALTY_DISCOUNT:=:NEW.TOTAL_AMOUNT * 0.3;
34  :NEW.TOTAL_AMOUNT:=:NEW.TOTAL_AMOUNT - :NEW.LOYALTY_DISCOUNT;
35  DBMS_OUTPUT.PUT_LINE('YOU HAVE BEEN OFFERED WITH 30% DISCOUNT');
36
37 ELSE
38   :NEW.LOYALTY_DISCOUNT:= :NEW.TOTAL_AMOUNT * 0;
39  :NEW.TOTAL_AMOUNT:= :NEW.TOTAL_AMOUNT - :NEW.LOYALTY_DISCOUNT;
40  DBMS_OUTPUT.PUT_LINE('YOU HAVE NOT BEEN OFFERED WITH ANY DISCOUNT');
41
42 END IF;
43 END;
44 /
```

Script: **ASSIGNMENT_TRIGGER_1** Status: **Complete**
View: **Summary** Rows: 15 Go

Number ↑	Elapsed	Statement	Feedback	Rows
1	0.05	DROP TRIGGER UPDATE_DISCOUNT	Trigger dropped.	0
2	0.09	CREATE OR REPLACE TRIGGER UPDATE_DISCOUNT BEFORE INSERT OR U	Trigger created.	0

Download row(s) 1 - 2 of 2

Statements Processed: 2 | Successful: 2 | With Errors: 0

- ❖ Screenshot of trigger's implementation test code along with the inserted/updated value:

Rows: 10 Save Run

```
1 UPDATE HIRE_INVOICE  
2 SET INVOICE_DATE='12/20/2019', TOTAL_AMOUNT='18000'  
3 WHERE INVOICE_ID = 5001;
```

Results Explain Describe Saved SQL History

YOU HAVE BEEN OFFERED WITH 15% DISCOUNT
1 row(s) updated.
0.02 seconds

ORACLE APEX App Builder SQL Workshop Team Development App Gallery

SQL Commands Schema DATABASESYSTEMSASSESSA c7202323

Rows 10 Clear Command Find Tables Save Run

1 SELECT * FROM HIRE_INVOICE;

Results Explain Describe Saved SQL History

INVOICE_ID	INVOICE_DATE	TOTAL_AMOUNT	LOYALTY_DISCOUNT	FK1_HIRE_ID
5001	12/20/2019	15300	2700	1
5002	11/21/2019	4800	-	2
5003	11/23/2019	26400	-	3
5004	12/07/2019	9900	-	4
5005	12/17/2019	56400	-	5
5006	12/23/2019	10800	-	6

6 rows returned in 0.01 seconds [Download](#)

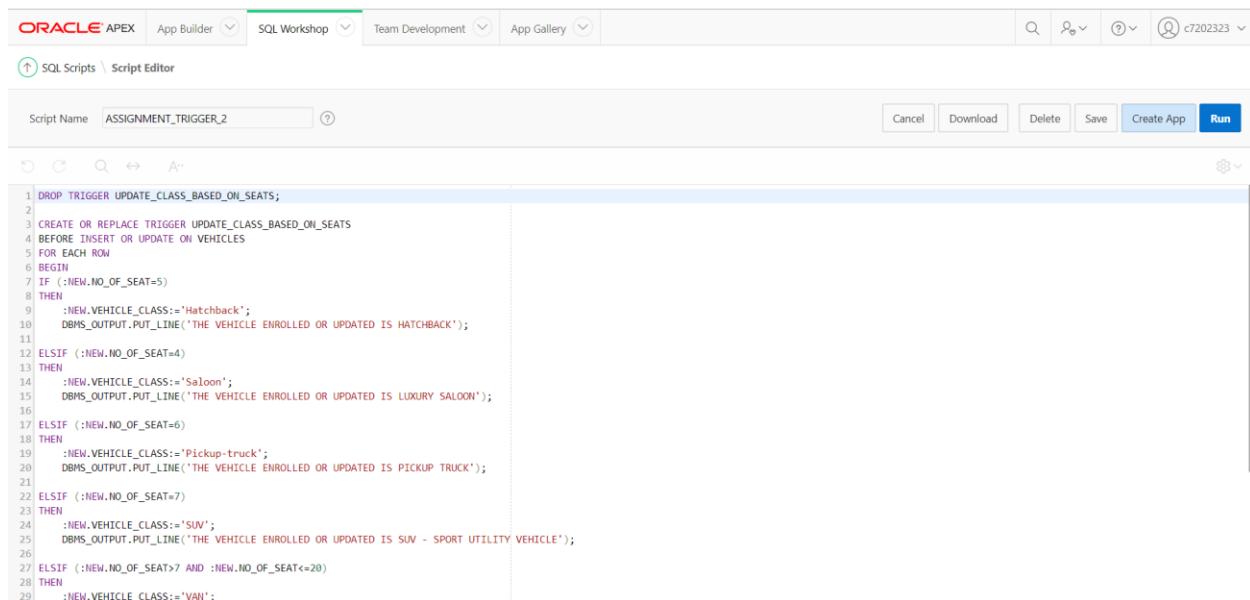
av.sapkota999@gmail.com database_systems_assessment en Copyright © 1999, 2019, Oracle. All rights reserved. Application Express 19.2.0.0.18

Trigger 2: Code and Test Implementation.

This trigger has been created for VEHICLES table during INSERTION or UPDATE process in order to find out the vehicle class by reading the number of seats of that particular vehicle. This trigger has various conditions to fulfill before inserting or updating into a table. Those conditions are:

- If the number of seats in the vehicle is equal to 5 then the vehicle is of hatchback class.
- If the number of seats in the vehicle is equal to 4 then the vehicle is of Luxury Saloon class.
- If the number of seats in the vehicle is equal to 6 then the vehicle is of Pickup - Truck class.
- If the number of seats in the vehicle is equal to 7 then the vehicle is of SUV (Sport Utility Vehicle) class.
- If the number of seats in the vehicle is greater than 7 and less than or equal to 20 then the vehicle is of Van class.
- If the number of seats in the vehicle is greater than 20 and less than or equal to 55 then the vehicle is of Bus class.
- Finally, if the number of seats does not fall under above mentioned conditions then error is occurred.

❖ Screenshot of trigger code and statement that says trigger created:



The screenshot shows the Oracle APEX interface with the SQL Workshop tab selected. A new trigger named 'ASSIGNMENT_TRIGGER_2' is being created. The code in the editor is as follows:

```
1 DROP TRIGGER UPDATE_CLASS_BASED_ON_SEATS;
2
3 CREATE OR REPLACE TRIGGER UPDATE_CLASS_BASED_ON_SEATS
4 BEFORE INSERT OR UPDATE ON VEHICLES
5 FOR EACH ROW
6 BEGIN
7 IF (:NEW.NO_OF_SEAT=5)
8 THEN
9 :NEW.VEHICLE_CLASS:='Hatchback';
10 DBMS_OUTPUT.PUT_LINE('THE VEHICLE ENROLLED OR UPDATED IS HATCHBACK');
11
12 ELSIF (:NEW.NO_OF_SEAT=4)
13 THEN
14 :NEW.VEHICLE_CLASS:='Saloon';
15 DBMS_OUTPUT.PUT_LINE('THE VEHICLE ENROLLED OR UPDATED IS LUXURY SALOON');
16
17 ELSIF (:NEW.NO_OF_SEAT=6)
18 THEN
19 :NEW.VEHICLE_CLASS:='Pickup-truck';
20 DBMS_OUTPUT.PUT_LINE('THE VEHICLE ENROLLED OR UPDATED IS PICKUP TRUCK');
21
22 ELSIF (:NEW.NO_OF_SEAT=7)
23 THEN
24 :NEW.VEHICLE_CLASS:='SUV';
25 DBMS_OUTPUT.PUT_LINE('THE VEHICLE ENROLLED OR UPDATED IS SUV - SPORT UTILITY VEHICLE');
26
27 ELSIF (:NEW.NO_OF_SEAT>7 AND :NEW.NO_OF_SEAT<=20)
28 THEN
29 :NEW.VEHICLE_CLASS:='VAN';
```

```

30|     DBMS_OUTPUT.PUT_LINE('THE VEHICLE ENROLLED OR UPDATED IS VAN');
31|
32|ELSIF (:NEW.NO_OF_SEAT>20 AND :NEW.NO_OF_SEAT<=55)
33|THEN
34|    :NEW.VEHICLE_CLASS:='Bus';
35|    DBMS_OUTPUT.PUT_LINE('THE VEHICLE ENROLLED OR UPDATED IS BUS');
36|
37|ELSE
38|    DBMS_OUTPUT.PUT_LINE('ERROR !!! THERE IS NO SUCH VEHICLE CLASS FOR THE ENTERED NUMBER OF SEAT !!!');
39|    RAISE_APPLICATION_ERROR(-12345, 'THE INSERT WILL NOT BE EXECUTED');
40|
41|END IF;
42|END;

```

av.sapkota999@gmail.com database_systems_assessment en Copyright © 1999, 2019, Oracle. All rights reserved. Application Express 19.2.0.0.18

The screenshot shows the Oracle APEX interface with the SQL Workshop tab selected. A script named "ASSIGNMENT_TRIGGER_2" is running, indicated by a progress bar. The status is "Complete". The results table shows two rows processed:

Number	Elapsed	Statement	Feedback	Rows
1	0.06	DROP TRIGGER UPDATE_CLASS_BASED_ON_SEATS	Trigger dropped.	0
2	0.10	CREATE OR REPLACE TRIGGER UPDATE_CLASS_BASED_ON_SEATS BEFORE	Trigger created.	0

Below the table, it says "Download" and "row(s) 1 - 2 of 2". At the bottom, there are three summary counts: "Statements Processed" (2), "Successful" (2), and "With Errors" (0).

av.sapkota999@gmail.com database_systems_assessment en Copyright © 1999, 2019, Oracle. All rights reserved. Application Express 19.2.0.0.18

- ❖ Screenshot of trigger's implementation test code along with the inserted/updated value:

The screenshot shows the Oracle APEX interface with the SQL Workshop tab selected. A command is being run in the SQL Commands editor:

```

1 UPDATE VEHICLES
2 SET NO_OF_SEAT = 18, VEHICLE_CLASS = 'BUS'
3 WHERE VEHICLE_CLASS_SURR_ID=910;

```

The results show the output of the update statement:

THE VEHICLE ENROLLED OR UPDATED IS VAN
1 row(s) updated.
0.00 seconds

av.sapkota999@gmail.com database_systems_assessment en Copyright © 1999, 2019, Oracle. All rights reserved. Application Express 19.2.0.0.18

ORACLE APEX App Builder SQL Workshop Team Development App Gallery

SQL Commands Schema DATABASESYSTEMSASSESSA c7202323

Rows 50 Clear Command Find Tables Save Run

SELECT * FROM VEHICLES;

Results Explain Describe Saved SQL History

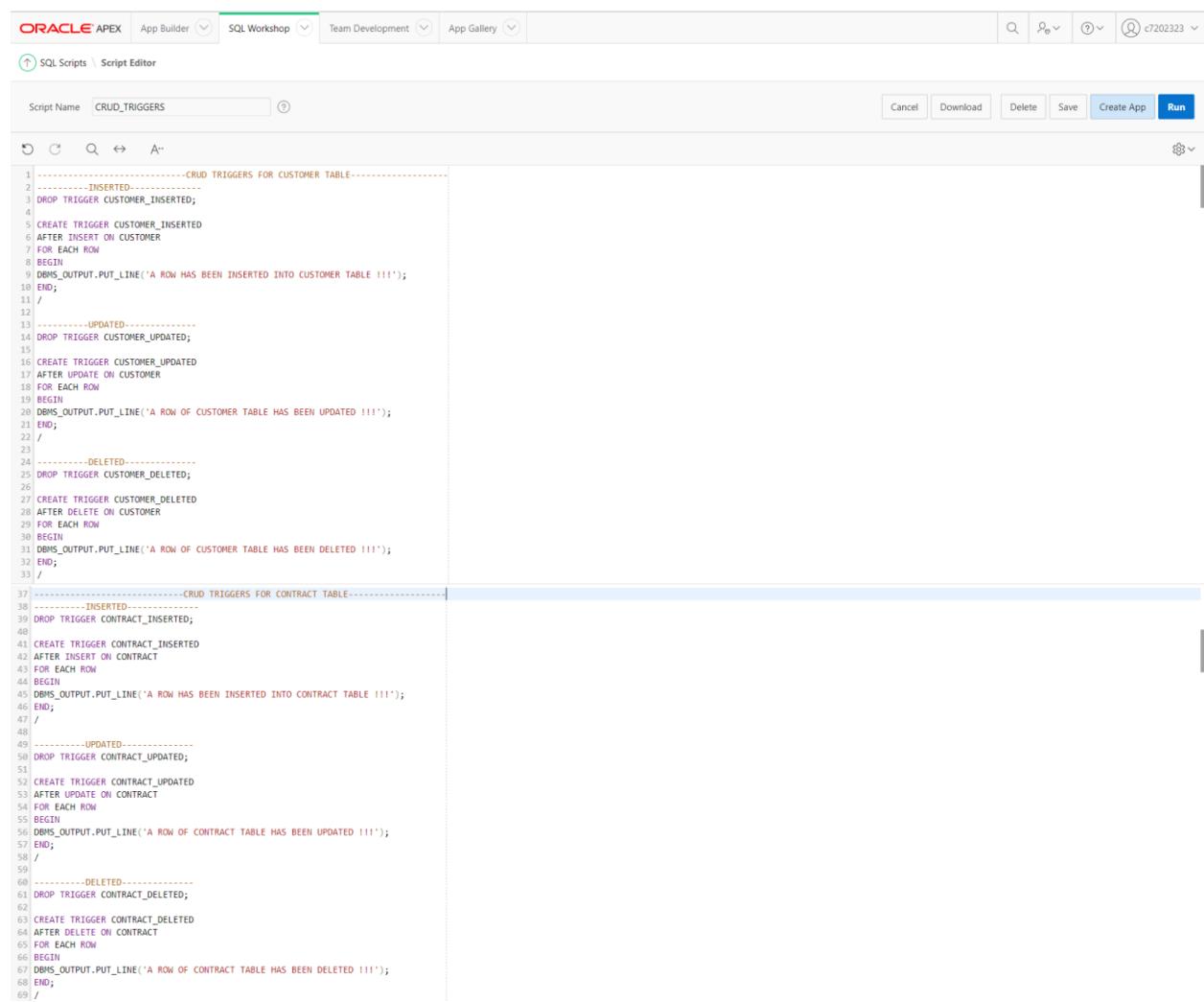
VEHICLE_CLASS_SURR_ID	VEHICLE_ID	VEHICLE_NAME	VEHICLE_MAKYEAR	NO_OF_SEAT	VEHICLE_CLASS	VEHICLE_DESCRIPTION
910	2389	Tata Winger	2018	18	VAN	This is a van that is luxurious for 18 people and are good for long drives.
901	2370	Ford Fiesta 1.1	2013	5	Hatchback	A car of ford. It is a hatchback whose ground clearance is high
902	6623	Fiat Panda	2010	5	Hatchback	A car of Fiat. It is a hatchback good for city use
903	7023	Hyundai i20	2018	5	Hatchback	A car of Hyundai. It is a hatchback that is good in highway
904	2980	Nissan Sunny	2017	4	Saloon	A car of Nissan. It is a luxury sedan who's good for business hire
905	3790	Honda City	2014	4	Saloon	A car of Honda. It is a sedan which is luxurious in city
906	6379	Suzuki Swift D	2016	4	Saloon	A car of Suzuki. It is a sedan which is good for family purposes

au.sapkota999@gmail.com database_systems_assessment en Copyright © 1999, 2019, Oracle. All rights reserved. Application Express 19.2.0.0.18

CRUD Triggers

In this part, triggers have been created for all the tables of CARS4U. These triggers print the message when a CRUD (Create, Update, Delete) operation is performed in the tables.

Codes for CRUD Triggers:



The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes 'ORACLE APEX', 'App Builder', 'SQL Workshop' (selected), 'Team Development', 'App Gallery', and various search and configuration icons. The main area is a 'Script Editor' titled 'CRUD_TRIGGERS'. The script contains PL/SQL code for creating triggers on the 'CUSTOMER' and 'CONTRACT' tables. The code is organized into sections for INSERTED, UPDATED, and DELETED operations on each table. It uses the DBMS_OUTPUT.PUT_LINE procedure to log messages ('A ROW HAS BEEN INSERTED INTO CUSTOMER TABLE !!!', 'A ROW HAS BEEN UPDATED !!!', 'A ROW HAS BEEN DELETED !!!') whenever a trigger is fired. The code is numbered from 1 to 69.

```
1 -----CRUD TRIGGERS FOR CUSTOMER TABLE-----
2 -----INSERTED-----
3 DROP TRIGGER CUSTOMER_INSERTED;
4 
5 CREATE TRIGGER CUSTOMER_INSERTED
6 AFTER INSERT ON CUSTOMER
7 FOR EACH ROW
8 BEGIN
9 DBMS_OUTPUT.PUT_LINE('A ROW HAS BEEN INSERTED INTO CUSTOMER TABLE !!!');
10 END;
11 /
12 
13 -----UPDATED-----
14 DROP TRIGGER CUSTOMER_UPDATED;
15 
16 CREATE TRIGGER CUSTOMER_UPDATED
17 AFTER UPDATE ON CUSTOMER
18 FOR EACH ROW
19 BEGIN
20 DBMS_OUTPUT.PUT_LINE('A ROW OF CUSTOMER TABLE HAS BEEN UPDATED !!!');
21 END;
22 /
23 
24 -----DELETED-----
25 DROP TRIGGER CUSTOMER_DELETED;
26 
27 CREATE TRIGGER CUSTOMER_DELETED
28 AFTER DELETE ON CUSTOMER
29 FOR EACH ROW
30 BEGIN
31 DBMS_OUTPUT.PUT_LINE('A ROW OF CUSTOMER TABLE HAS BEEN DELETED !!!');
32 END;
33 /
34 
35 -----CRUD TRIGGERS FOR CONTRACT TABLE-----
36 -----INSERTED-----
37 DROP TRIGGER CONTRACT_INSERTED;
38 
39 CREATE TRIGGER CONTRACT_INSERTED
40 AFTER INSERT ON CONTRACT
41 FOR EACH ROW
42 BEGIN
43 DBMS_OUTPUT.PUT_LINE('A ROW HAS BEEN INSERTED INTO CONTRACT TABLE !!!');
44 END;
45 /
46 
47 -----UPDATED-----
48 DROP TRIGGER CONTRACT_UPDATED;
49 
50 CREATE TRIGGER CONTRACT_UPDATED
51 AFTER UPDATE ON CONTRACT
52 FOR EACH ROW
53 BEGIN
54 DBMS_OUTPUT.PUT_LINE('A ROW OF CONTRACT TABLE HAS BEEN UPDATED !!!');
55 END;
56 /
57 
58 -----DELETED-----
59 DROP TRIGGER CONTRACT_DELETED;
60 
61 CREATE TRIGGER CONTRACT_DELETED
62 AFTER DELETE ON CONTRACT
63 FOR EACH ROW
64 BEGIN
65 DBMS_OUTPUT.PUT_LINE('A ROW OF CONTRACT TABLE HAS BEEN DELETED !!!');
66 END;
67 /
68 
69 /
```

```

74 -----CRUD TRIGGERS FOR INSURANCE TABLE-----| |
75 DROP TRIGGER INSURANCE_INSERTED;
76
77 CREATE TRIGGER INSURANCE_INSERTED
78 AFTER INSERT ON INSURANCE
79 FOR EACH ROW
80 BEGIN
81 DBMS_OUTPUT.PUT_LINE('A ROW HAS BEEN INSERTED INTO INSURANCE TABLE !!!');
82 END;
83 /
84
85 -----UPDATED-----
86 DROP TRIGGER INSURANCE_UPDATED;
87
88 CREATE TRIGGER INSURANCE_UPDATED
89 AFTER UPDATE ON INSURANCE
90 FOR EACH ROW
91 BEGIN
92 DBMS_OUTPUT.PUT_LINE('A ROW OF INSURANCE TABLE HAS BEEN UPDATED !!!');
93 END;
94 /
95
96 -----DELETED-----
97 DROP TRIGGER INSURANCE_DELETED;
98
99 CREATE TRIGGER INSURANCE_DELETED
100 AFTER DELETE ON INSURANCE
101 FOR EACH ROW
102 BEGIN
103 DBMS_OUTPUT.PUT_LINE('A ROW OF INSURANCE TABLE HAS BEEN DELETED !!!');
104 END;
105 /
106
107 -----CRUD TRIGGERS FOR ACCESSORIES TABLE-----| |
108
109 -----INSERTED-----
110 DROP TRIGGER ACCESSORIES_INSERTED;
111
112 CREATE TRIGGER ACCESSORIES_INSERTED
113 AFTER INSERT ON ACCESSORIES
114 FOR EACH ROW
115 BEGIN
116 DBMS_OUTPUT.PUT_LINE('A ROW HAS BEEN INSERTED INTO ACCESSORIES TABLE !!!');
117 END;
118 /
119
120 -----UPDATED-----
121 DROP TRIGGER ACCESSORIES_UPDATED;
122
123 CREATE TRIGGER ACCESSORIES_UPDATED
124 AFTER UPDATE ON ACCESSORIES
125 FOR EACH ROW
126 BEGIN
127 DBMS_OUTPUT.PUT_LINE('A ROW OF ACCESSORIES TABLE HAS BEEN UPDATED !!!');
128 END;
129 /
130
131 -----DELETED-----
132 DROP TRIGGER ACCESSORIES_DELETED;
133
134 CREATE TRIGGER ACCESSORIES_DELETED
135 AFTER DELETE ON ACCESSORIES
136 FOR EACH ROW
137 BEGIN
138 DBMS_OUTPUT.PUT_LINE('A ROW OF ACCESSORIES TABLE HAS BEEN DELETED !!!');
139 END;
140 /
141
142 -----CRUD TRIGGERS FOR HIRE_ARRANGEMENT TABLE-----| |
143
144 -----INSERTED-----
145 DROP TRIGGER HIRE_ARRANGEMENT_INSERTED;
146
147 CREATE TRIGGER HIRE_ARRANGEMENT_INSERTED
148 AFTER INSERT ON HIRE_ARRANGEMENT
149 FOR EACH ROW
150 BEGIN
151 DBMS_OUTPUT.PUT_LINE('A ROW HAS BEEN INSERTED INTO HIRE_ARRANGEMENT TABLE !!!');
152 END;
153 /
154
155 -----UPDATED-----
156 DROP TRIGGER HIRE_ARRANGEMENT_UPDATED;
157
158 CREATE TRIGGER HIRE_ARRANGEMENT_UPDATED
159 AFTER UPDATE ON HIRE_ARRANGEMENT
160 FOR EACH ROW
161 BEGIN
162 DBMS_OUTPUT.PUT_LINE('A ROW OF HIRE_ARRANGEMENT TABLE HAS BEEN UPDATED !!!');
163 END;
164 /
165
166 -----DELETED-----
167 DROP TRIGGER HIRE_ARRANGEMENT_DELETED;
168
169 CREATE TRIGGER HIRE_ARRANGEMENT_DELETED
170 AFTER DELETE ON HIRE_ARRANGEMENT
171 FOR EACH ROW
172 BEGIN
173 DBMS_OUTPUT.PUT_LINE('A ROW OF HIRE_ARRANGEMENT TABLE HAS BEEN DELETED !!!');
174 END;
175 /
176
177 / |

```

```

181 -----CRUD TRIGGERS FOR HIRE_INVOICE TABLE-----|-----|
182 -----INSERTED-----|-----|
183 DROP TRIGGER HIRE_INVOICE_INSERTED;
184
185 CREATE TRIGGER HIRE_INVOICE_INSERTED
186 AFTER INSERT ON HIRE_INVOICE
187 FOR EACH ROW
188 BEGIN
189 DBMS_OUTPUT.PUT_LINE('A ROW HAS BEEN INSERTED INTO HIRE_INVOICE TABLE !!!');
190 END;
191 /
192
193 -----UPDATED-----|-----|
194 DROP TRIGGER HIRE_INVOICE_UPDATED;
195
196 CREATE TRIGGER HIRE_INVOICE_UPDATED
197 AFTER UPDATE ON HIRE_INVOICE
198 FOR EACH ROW
199 BEGIN
200 DBMS_OUTPUT.PUT_LINE('A ROW OF HIRE_INVOICE TABLE HAS BEEN UPDATED !!!');
201 END;
202 /
203
204 -----DELETED-----|-----|
205 DROP TRIGGER HIRE_INVOICE_DELETED;
206
207 CREATE TRIGGER HIRE_INVOICE_DELETED
208 AFTER DELETE ON HIRE_INVOICE
209 FOR EACH ROW
210 BEGIN
211 DBMS_OUTPUT.PUT_LINE('A ROW OF HIRE_INVOICE TABLE HAS BEEN DELETED !!!');
212 END;
213 /
214 -----CRUD TRIGGERS FOR CENTER TABLE-----|-----|
215 -----INSERTED-----|-----|
216 DROP TRIGGER CENTER_INSERTED;
217
218 CREATE TRIGGER CENTER_INSERTED
219 AFTER INSERT ON CENTER
220 FOR EACH ROW
221 BEGIN
222 DBMS_OUTPUT.PUT_LINE('A ROW HAS BEEN INSERTED INTO CENTER TABLE !!!');
223 END;
224 /
225
226 -----UPDATED-----|-----|
227 DROP TRIGGER CENTER_UPDATED;
228
229 CREATE TRIGGER CENTER_UPDATED
230 AFTER UPDATE ON CENTER
231 FOR EACH ROW
232 BEGIN
233 DBMS_OUTPUT.PUT_LINE('A ROW OF CENTER TABLE HAS BEEN UPDATED !!!');
234 END;
235 /
236
237 -----DELETED-----|-----|
238 DROP TRIGGER CENTER_DELETED;
239
240 CREATE TRIGGER CENTER_DELETED
241 AFTER DELETE ON CENTER
242 FOR EACH ROW
243 BEGIN
244 DBMS_OUTPUT.PUT_LINE('A ROW OF CENTER TABLE HAS BEEN DELETED !!!');
245 END;
246 /
247
248 -----DELETED-----|-----|
249 /

```

```

253 -----CRUD TRIGGERS FOR CENTER_CLASSIFICATION_PRICE TABLE-----
254 -----INSERTED-----
255 DROP TRIGGER CENTER_CLASSIFICATION_PRICE_INSERTED;
256
257 CREATE TRIGGER CENTER_CLASSIFICATION_PRICE_INSERTED
258 AFTER INSERT ON CENTER_CLASSIFICATION_PRICE
259 FOR EACH ROW
260 BEGIN
261 DBMS_OUTPUT.PUT_LINE('A ROW HAS BEEN INSERTED INTO CENTER_CLASSIFICATION_PRICE TABLE !!!');
262 END;
263 /
264
265 -----UPDATED-----
266 DROP TRIGGER CENTER_CLASSIFICATION_PRICE_UPDATED;
267
268 CREATE TRIGGER CENTER_CLASSIFICATION_PRICE_UPDATED
269 AFTER UPDATE ON CENTER_CLASSIFICATION_PRICE
270 FOR EACH ROW
271 BEGIN
272 DBMS_OUTPUT.PUT_LINE('A ROW OF CENTER_CLASSIFICATION_PRICE TABLE HAS BEEN UPDATED !!!');
273 END;
274 /
275
276 -----DELETED-----
277 DROP TRIGGER CENTER_CLASSIFICATION_PRICE_DELETED;
278
279 CREATE TRIGGER CENTER_CLASSIFICATION_PRICE_DELETED
280 AFTER DELETE ON CENTER_CLASSIFICATION_PRICE
281 FOR EACH ROW
282 BEGIN
283 DBMS_OUTPUT.PUT_LINE('A ROW OF CENTER_CLASSIFICATION_PRICE TABLE HAS BEEN DELETED !!!');
284 END;
285 /
286 -----CRUD TRIGGERS FOR VEHICLES TABLE-----
287 -----INSERTED-----
288 DROP TRIGGER VEHICLES_INSERTED;
289
290 CREATE TRIGGER VEHICLES_INSERTED
291 AFTER INSERT ON VEHICLES
292 FOR EACH ROW
293 BEGIN
294 DBMS_OUTPUT.PUT_LINE('A ROW HAS BEEN INSERTED INTO VEHICLES TABLE !!!');
295 END;
296 /
297
298 -----UPDATED-----
299 DROP TRIGGER VEHICLES_UPDATED;
300
301 CREATE TRIGGER VEHICLES_UPDATED
302 AFTER UPDATE ON VEHICLES
303 FOR EACH ROW
304 BEGIN
305 DBMS_OUTPUT.PUT_LINE('A ROW OF VEHICLES TABLE HAS BEEN UPDATED !!!');
306 END;
307 /
308
309 -----DELETED-----
310 DROP TRIGGER VEHICLES_DELETED;
311
312 CREATE TRIGGER VEHICLES_DELETED
313 AFTER DELETE ON VEHICLES
314 FOR EACH ROW
315 BEGIN
316 DBMS_OUTPUT.PUT_LINE('A ROW OF VEHICLES TABLE HAS BEEN DELETED !!!');
317 END;
318 /
319

```

avhimanhusapkota99@gmail.com

[database_systems_assessment](#)

[en](#)

Copyright © 1999, 2019, Oracle. All rights reserved.

Application Express 19.2.0.00.18

CRUD Triggers Code Implementation:

SQL Scripts \ Results				
Script: CRUD.TRIGGERS Status: Complete		Statement	Feedback	Rows
Number ↑	Elapsed			
1	0.05	DROP TRIGGER CUSTOMER_INSERTED	Trigger dropped.	0
2	0.05	CREATE TRIGGER CUSTOMER_INSERTED AFTER INSERT ON CUSTOMER FOR	Trigger created.	0
3	0.02	DROP TRIGGER CUSTOMER_UPDATED	Trigger dropped.	0
4	0.03	CREATE TRIGGER CUSTOMER_UPDATED AFTER UPDATE ON CUSTOMER FOR	Trigger created.	0
5	0.02	DROP TRIGGER CUSTOMER_DELETED	Trigger dropped.	0
6	0.04	CREATE TRIGGER CUSTOMER_DELETED AFTER DELETE ON CUSTOMER FOR	Trigger created.	0
7	0.03	DROP TRIGGER CONTRACT_INSERTED	Trigger dropped.	0
8	0.04	CREATE TRIGGER CONTRACT_INSERTED AFTER INSERT ON CONTRACT FOR	Trigger created.	0
9	0.04	DROP TRIGGER CONTRACT_UPDATED	Trigger dropped.	0
10	0.05	CREATE TRIGGER CONTRACT_UPDATED AFTER UPDATE ON CONTRACT FOR	Trigger created.	0
11	0.02	DROP TRIGGER CONTRACT_DELETED	Trigger dropped.	0
12	0.05	CREATE TRIGGER CONTRACT_DELETED AFTER DELETE ON CONTRACT FOR	Trigger created.	0
13	0.02	DROP TRIGGER INSURANCE_INSERTED	Trigger dropped.	0
14	0.04	CREATE TRIGGER INSURANCE_INSERTED AFTER INSERT ON INSURANCE	Trigger created.	0
15	0.03	DROP TRIGGER INSURANCE_UPDATED	Trigger dropped.	0
16	0.05	CREATE TRIGGER INSURANCE_UPDATED AFTER UPDATE ON INSURANCE FOR	Trigger created.	0
17	0.03	DROP TRIGGER INSURANCE_DELETED	Trigger dropped.	0
18	0.04	CREATE TRIGGER INSURANCE_DELETED AFTER DELETE ON INSURANCE FOR	Trigger created.	0
19	0.02	DROP TRIGGER ACCESSORIES_INSERTED	Trigger dropped.	0
20	0.05	CREATE TRIGGER ACCESSORIES_INSERTED AFTER INSERT ON ACCESSORIES	Trigger created.	0
21	0.02	DROP TRIGGER ACCESSORIES_UPDATED	Trigger dropped.	0
22	0.04	CREATE TRIGGER ACCESSORIES_UPDATED AFTER UPDATE ON ACCESSORIES	Trigger created.	0
23	0.02	DROP TRIGGER ACCESSORIES_DELETED	Trigger dropped.	0
24	0.04	CREATE TRIGGER ACCESSORIES_DELETED AFTER DELETE ON ACCESSORIES	Trigger created.	0
25	0.02	DROP TRIGGER HIRE_ARRANGEMENT_INSERTED	Trigger dropped.	0
26	0.04	CREATE TRIGGER HIRE_ARRANGEMENT_INSERTED AFTER INSERT ON HIRE	Trigger created.	0
27	0.02	DROP TRIGGER HIRE_ARRANGEMENT_UPDATED	Trigger dropped.	0
28	0.04	CREATE TRIGGER HIRE_ARRANGEMENT_UPDATED AFTER UPDATE ON HIRE	Trigger created.	0
29	0.02	DROP TRIGGER HIRE_ARRANGEMENT_DELETED	Trigger dropped.	0
30	0.04	CREATE TRIGGER HIRE_ARRANGEMENT_DELETED AFTER DELETE ON HIRE	Trigger created.	0
31	0.04	DROP TRIGGER HIRE_INVOICE_INSERTED	Trigger dropped.	0
32	0.05	CREATE TRIGGER HIRE_INVOICE_INSERTED AFTER INSERT ON HIRE_INVOICE	Trigger created.	0
33	0.02	DROP TRIGGER HIRE_INVOICE_UPDATED	Trigger dropped.	0
34	0.05	CREATE TRIGGER HIRE_INVOICE_UPDATED AFTER UPDATE ON HIRE_INVOICE	Trigger created.	0
35	0.02	DROP TRIGGER HIRE_INVOICE_DELETED	Trigger dropped.	0
36	0.05	CREATE TRIGGER HIRE_INVOICE_DELETED AFTER DELETE ON HIRE_INVOICE	Trigger created.	0
37	0.02	DROP TRIGGER CENTER_INSERTED	Trigger dropped.	0
38	0.04	CREATE TRIGGER CENTER_INSERTED AFTER INSERT ON CENTER FOR EA	Trigger created.	0
39	0.03	DROP TRIGGER CENTER_UPDATED	Trigger dropped.	0

40	0.04	CREATE TRIGGER CENTER_UPDATED AFTER UPDATE ON CENTER FOR EACH ROW	Trigger created.	0
41	0.02	DROP TRIGGER CENTER_DELETED	Trigger dropped.	0
42	0.04	CREATE TRIGGER CENTER_DELETED AFTER DELETE ON CENTER FOR EACH ROW	Trigger created.	0
43	0.22	DROP TRIGGER CENTER_CLASSIFICATION_PRICE_INSERTED	Trigger dropped.	0
44	0.14	CREATE TRIGGER CENTER_CLASSIFICATION_PRICE_INSERTED AFTER INSERT	Trigger created.	0
45	0.02	DROP TRIGGER CENTER_CLASSIFICATION_PRICE_UPDATED	Trigger dropped.	0
46	0.04	CREATE TRIGGER CENTER_CLASSIFICATION_PRICE_UPDATED AFTER UPDATE	Trigger created.	0
47	0.02	DROP TRIGGER CENTER_CLASSIFICATION_PRICE_DELETED	Trigger dropped.	0
48	0.03	CREATE TRIGGER CENTER_CLASSIFICATION_PRICE_DELETED AFTER DELETE	Trigger created.	0
49	0.02	DROP TRIGGER VEHICLES_INSERTED	Trigger dropped.	0
50	0.03	CREATE TRIGGER VEHICLES_INSERTED AFTER INSERT ON VEHICLES FOR EACH ROW	Trigger created.	0
51	0.02	DROP TRIGGER VEHICLES_UPDATED	Trigger dropped.	0
52	0.02	CREATE TRIGGER VEHICLES_UPDATED AFTER UPDATE ON VEHICLES FOR EACH ROW	Trigger created.	0
53	0.02	DROP TRIGGER VEHICLES_DELETED	Trigger dropped.	0
54	0.02	CREATE TRIGGER VEHICLES_DELETED AFTER DELETE ON VEHICLES FOR EACH ROW	Trigger created.	0

[Download](#)

row(s) 1 - 54 of 54

54

54

0

Statements Processed

Successful

With Errors

av.sapkota999@gmail.com

database_systems_assessment

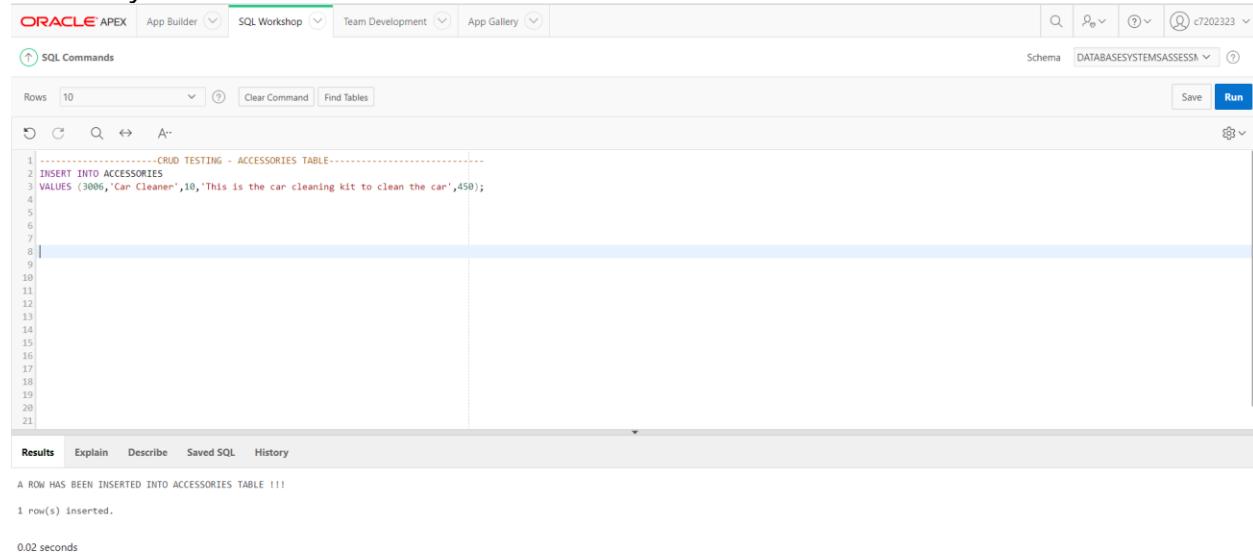
en

Copyright © 1999, 2019, Oracle. All rights reserved.

Application Express 19.2.0.0.18

CRUD Triggers Code Test:

The test has been done using ACCESSORIES table only. However, the triggers work for every table.



ORACLE APEX App Builder SQL Workshop Team Development App Gallery

SQL Commands

Rows 10 Clear Command Find Tables

Save Run

Schema DATABASESYSTEMSASSESSM

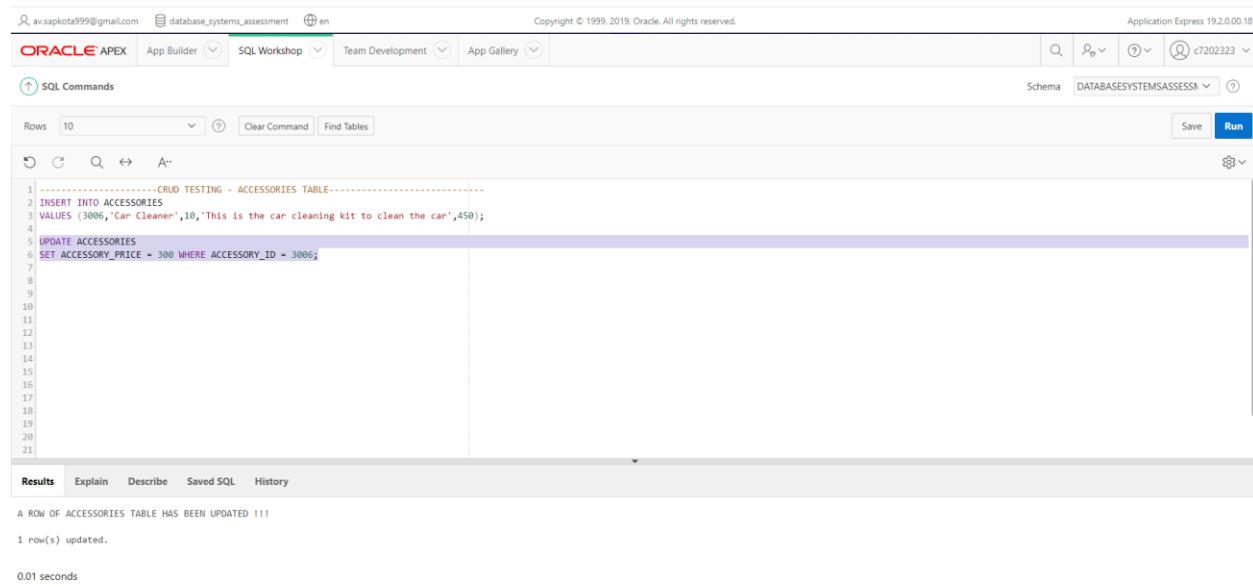
Results Explain Describe Saved SQL History

```
1 -----CRUD TESTING - ACCESSORIES TABLE-----
2 INSERT INTO ACCESSORIES
3 VALUES (3006,'Car Cleaner',10,'This is the car cleaning kit to clean the car',450);
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
```

A ROW HAS BEEN INSERTED INTO ACCESSORIES TABLE !!!

1 row(s) inserted.

0.02 seconds



avsapkota999@gmail.com database_systems_assessment en Copyright © 1999-2019, Oracle. All rights reserved. Application Express 19.2.0.0.18

ORACLE APEX App Builder SQL Workshop Team Development App Gallery

SQL Commands

Rows 10 Clear Command Find Tables

Save Run

Schema DATABASESYSTEMSASSESSM

Results Explain Describe Saved SQL History

```
1 -----CRUD TESTING - ACCESSORIES TABLE-----
2 INSERT INTO ACCESSORIES
3 VALUES (3006,'Car Cleaner',10,'This is the car cleaning kit to clean the car',450);
4
5 UPDATE ACCESSORIES
6 SET ACCESSORY_PRICE = 300 WHERE ACCESSORY_ID = 3006;
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
```

A ROW OF ACCESSORIES TABLE HAS BEEN UPDATED !!!

1 row(s) updated.

0.01 seconds

The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes links for ORACLE APEX, App Builder, SQL Workshop (selected), Team Development, App Gallery, and various user icons. The SQL Workshop tab is active, displaying a query editor with the following SQL code:

```
1 --CRUD TESTING - ACCESSORIES TABLE-----
2 INSERT INTO ACCESSORIES
3 VALUES (3006,'Car Cleaner',10,'This is the car cleaning kit to clean the car',450);
4
5 UPDATE ACCESSORIES
6 SET ACCESSORY_PRICE = 300 WHERE ACCESSORY_ID = 3006;
7
8 DELETE FROM ACCESSORIES
9 WHERE ACCESSORY_ID = 3006;
```

The code is annotated with comments: line 1 starts a multi-line comment, lines 2-4 perform an insert, lines 5-7 perform an update, and lines 8-9 perform a delete. The results section below the code editor shows the message "A ROW OF ACCESSORIES TABLE HAS BEEN DELETED !!!" and "1 row(s) deleted." The execution time is listed as "0.03 seconds".

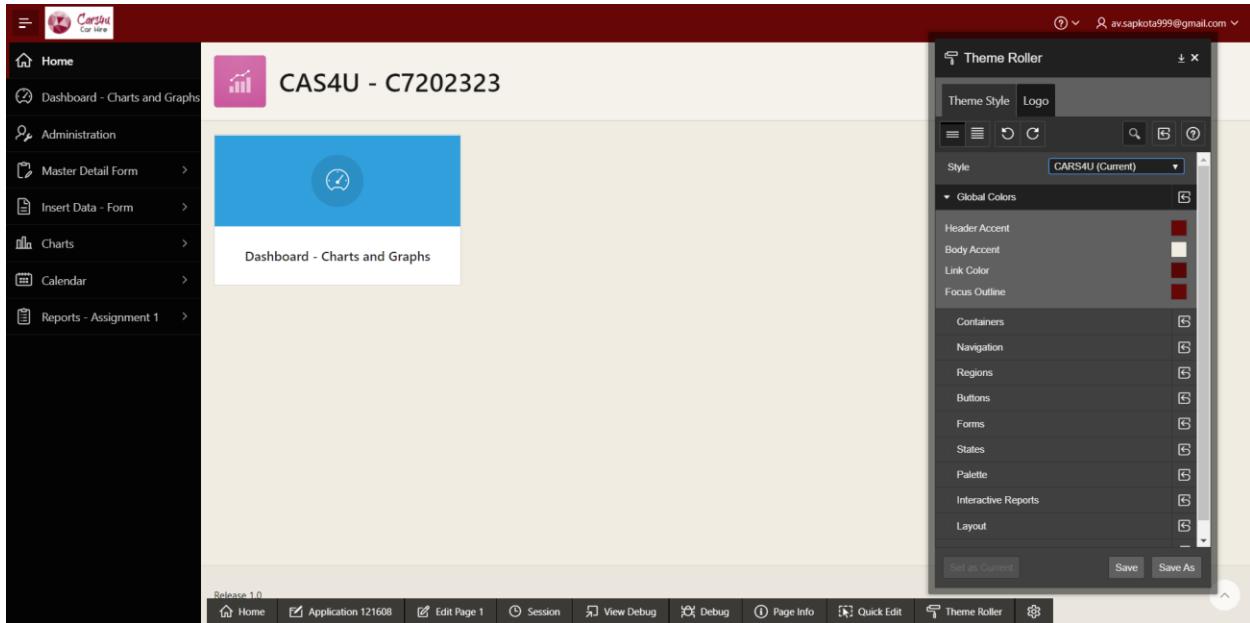
Task 2: Application Builder

Application: CARS4U Cars Hire Company.

A desktop application has been built for CARS4U Car Hire Company. The application consists of various functions and features. Some of the features of the CARS4U Car Hire Company's Applications have been presented below.

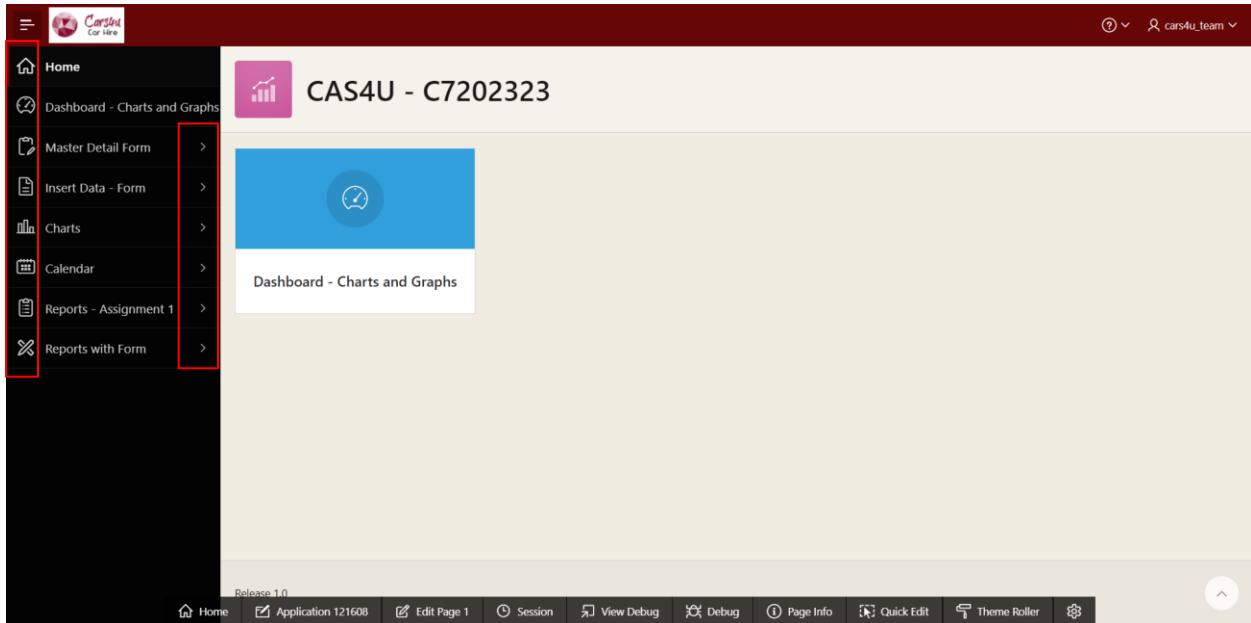
Home Page (with CSS)

The application's appearance including colors and some other CSS has been beautifully designed.



Navigation Menu with Icons

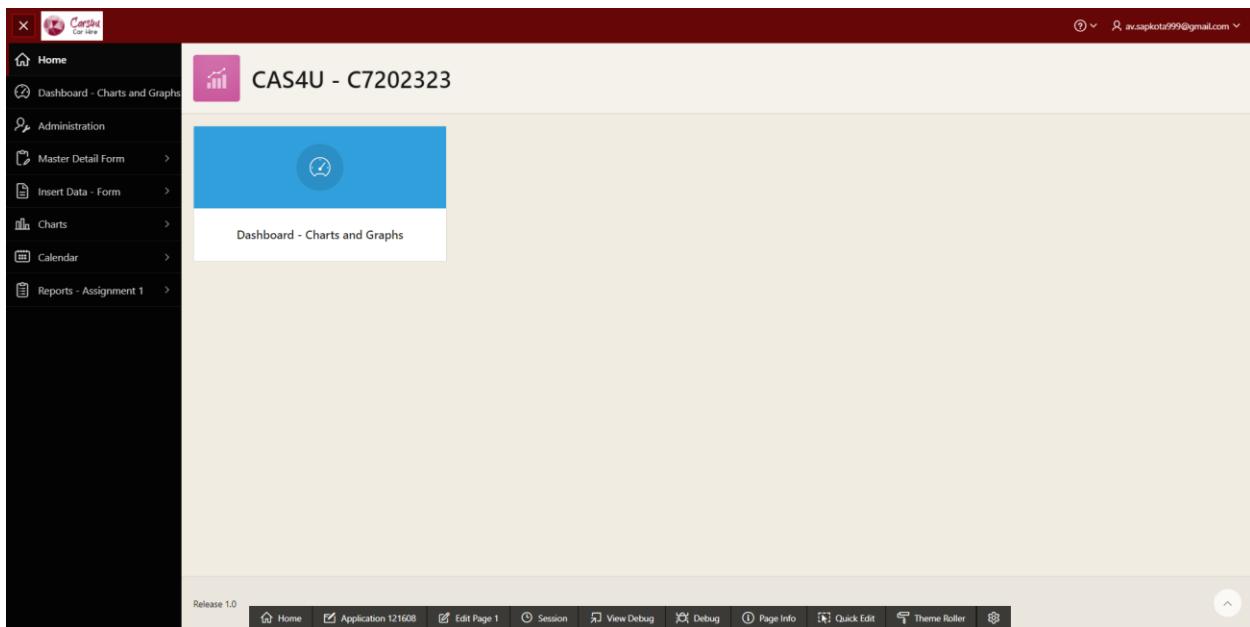
The navigation menu has been customized with drop-down menus and creative icons.



A screenshot of the Oracle APEX App Builder interface, specifically the 'List Entry' configuration screen. The top navigation bar includes 'ORACLE APEX', 'App Builder', 'SQL Workshop', 'Team Development', 'App Gallery', and search/filter buttons. The main content area shows the 'List Entry' configuration for a 'Desktop Navigation Menu'. The 'Entry' tab is selected, displaying fields for 'List' (set to 'Desktop Navigation Menu'), 'Parent List Entry' (set to '- No Parent List Item -'), 'Sequence' (set to '10010'), 'Image/Class' (set to 'fa-clipboard-edit'), 'Attributes' (empty), 'Alt Attribute' (empty), and 'List Entry Label' (set to 'Master Detail Form'). The 'Target' tab is also visible, showing 'Target type' set to 'Page in this Application' with a dropdown menu listing various page items. A sidebar on the right provides links for 'List Entry', 'Tasks', 'Change Parent List', 'Developer Resources', 'Font APEX Icons', and a detailed list of 'List Entries' with their IDs and names.

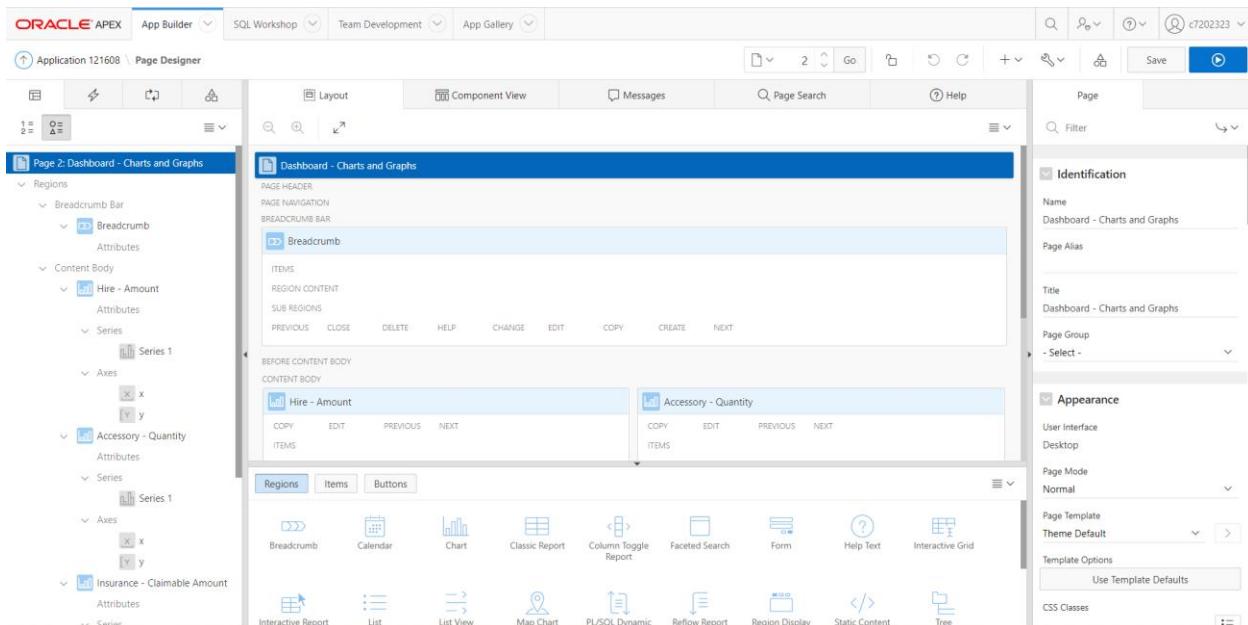
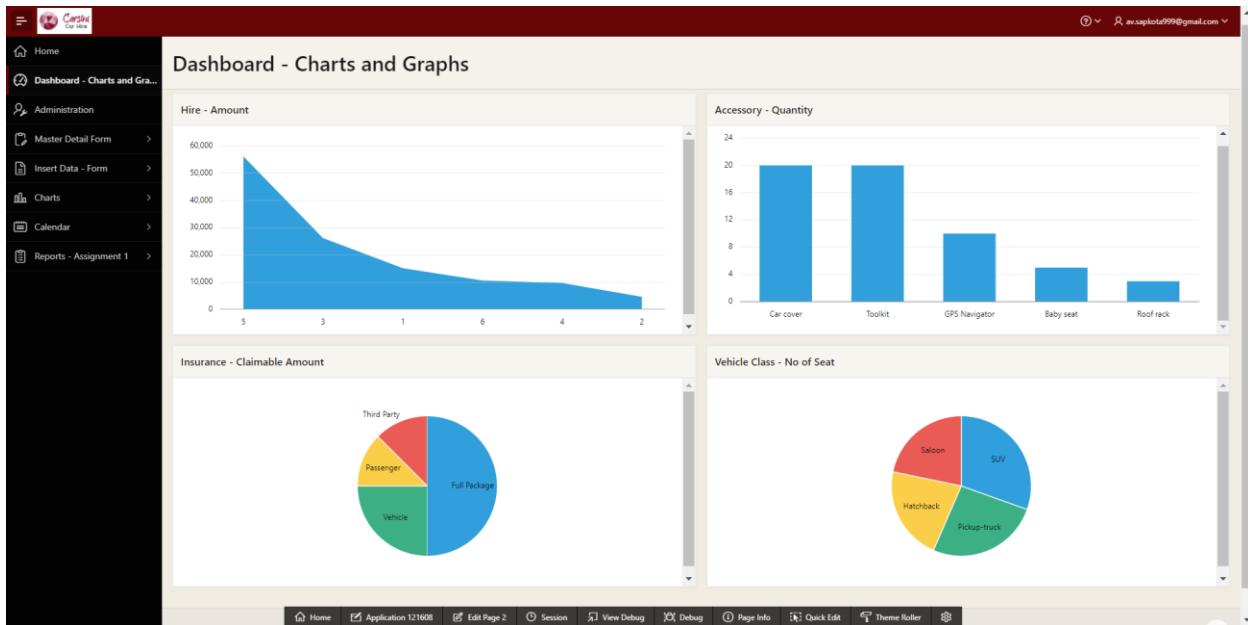
List Details									List Entries	
Sequence	Name	Parent Entry	Target	Conditional	Updated	Level	Authorization Scheme	Copy		
10	Home	-	f?p=&APP_ID.:1:&APP_SESSION.:&DEBUG:	-	9 hours ago	1	-			
20	Dashboard - Charts and Graphs	-	f?p=&APP_ID.:2:&APP_SESSION.:&DEBUG:	-	9 hours ago	1	-			
30	Customer - Hire Details	Master Detail Form	f?p=&APP_ID.:3:&SESSION.:&DEBUG:	-	9 hours ago	2	-			
40	Hire - Invoice Details	Master Detail Form	f?p=&APP_ID.:5:&SESSION.:&DEBUG:	-	9 hours ago	2	-			
80	Accessory - Price Chart	Charts	f?p=&APP_ID.:15:&SESSION.:&DEBUG:	-	9 hours ago	2	-			
90	Hire - Price Chart	Charts	f?p=&APP_ID.:16:&SESSION.:&DEBUG:	-	9 hours ago	2	-			
100	Contract Dates	Calendar	f?p=&APP_ID.:17:&SESSION.:&DEBUG:	-	9 hours ago	2	-			
110	Customer Hire Dates	Calendar	f?p=&APP_ID.:19:&SESSION.:&DEBUG:	-	9 hours ago	2	-			
120	Customer Hire Details	Reports - Assignment 1	f?p=&APP_ID.:20:&SESSION.:&DEBUG:	-	8 hours ago	2	-			
130	SUV Price List	Reports - Assignment 1	f?p=&APP_ID.:21:&SESSION.:&DEBUG:	-	8 hours ago	2	-			
140	Accessory Hire Details	Reports - Assignment 1	f?p=&APP_ID.:23:&SESSION.:&DEBUG:	-	8 hours ago	2	-			
150	Customer Price List	Reports - Assignment 1	f?p=&APP_ID.:24:&SESSION.:&DEBUG:	-	8 hours ago	2	-			
160	Saloon and Hatchback Vehicles	Reports - Assignment 1	f?p=&APP_ID.:25:&SESSION.:&DEBUG:	-	8 hours ago	2	-			
170	Manage Customer	Insert Data - Form	f?p=&APP_ID.:29:&SESSION.:&DEBUG:	-	5 hours ago	2	-			
180	Manage Hire	Insert Data - Form	f?p=&APP_ID.:30:&SESSION.:&DEBUG:	-	5 hours ago	2	-			
10000	Administration	-	f?p=&APP_ID.:10000:&APP_SESSION.:&DEBUG:	-	9 hours ago	1	Administration Rights			
10010	Master Detail Form	-	-	-	9 hours ago	1	-			

The application has Home Page that consists of Dashboard.



Dashboard

The dashboard consists of four graphs/charts based on the tables or views created in the first component of the assessment.



Master Detail Forms

Two master-detail forms 'Customer-Hire Details' and 'Hire-Invoice Details' have been created to view the data in detail. Each master detail form consists of Buttons for CRUD functionality (Create, Update or Delete).

Master Detail Form 1:

The screenshot shows the Oracle APEX application interface for 'Customer - Hire Details'. At the top, there's a navigation bar with the application logo, user name 'av.sapkota999@gmail.com', and a search bar. Below the header, the main content area is divided into two sections: 'Customer' and 'Hire Arrangement'. The 'Customer' section displays a list of records with columns for Name, Email, Address, and Phone number. The 'Hire Arrangement' section contains a table with columns for Hire Id, Hire Date, Hire Noofdays, Fk2 Accessory Id, and Fk3 Vehicle Center Id. At the bottom, the page footer includes links for Home, Application 121608, Edit Page 3, Session, View Debug, Page Info, Quick Edit, Theme Roller, and a search bar. The left sidebar shows the page structure with regions like 'Regions', 'Content Body', 'Buttons', 'Page Items', and 'Computations'. On the right, there are various configuration options for the page template, media type, and navigation.

Customer - Hire Details

Customer Name	Customer Email	Customer Address	Customer Phoneno	Hire Arrangement	Fk2 Insurance Policyno	Fk3 Vehicle Center Id
Anita Rijal arijal@gmail.com	Rohit Raj Pandey rpandey@gmail.com	Chabahil, Kathmandu	9800232876	Hire Id 3	802	208
Joey Foster jfoster@gmail.com						1 - 1
John Jensen jjenien@gmail.com						
Justin Gurung jgurung@yahoo.com						
Nepal Bank nepalbank@gmail.com						
Nepal Bank nepalbank@gmail.com						
Nepal Bank nepalbank@gmail.com						
Rohit Raj Pandey rpandey@gmail.com						

Customer

Customer Name: Rohit Raj Pandey

Customer Email: rpandey@gmail.com

Customer Address: Chabahil, Kathmandu

Customer Phoneno: 9800232876

Hire Arrangement

Hire Id: 3

Fk2 Insurance Policyno: 802

Fk3 Vehicle Center Id: 208

1 - 1

Cancel Apply Changes

Release 1.0 Home Application 121608 Edit Page 4 Session View Debug Debug Page Info Quick Edit Theme Roller Help

Master Detail Form 2:

Hire					
<input type="button" value="New Hire"/>					
<input type="button" value="Reset"/>					
<input type="button" value="Search: All Text Columns"/>					
<input type="button" value="Go"/>					
<input type="button" value="Actions"/>					
<input type="button" value="Edit"/>					
<input type="button" value="Add Row"/>					
	Hire Date	Hire Noofdays	Fk1 Customer Id	Fk2 Accessory Id	Fk3 Vehicle Center Id
<input checked="" type="checkbox"/>	11/15/2019	5	1005	3001	234
<input type="checkbox"/>	11/19/2019	2	1008	3003	206
<input type="checkbox"/>	11/20/2019	3	1002	3001	208
<input type="checkbox"/>	12/4/2019	3	1004	3004	244
<input type="checkbox"/>	12/11/2019	6	1001	3001	228
<input type="checkbox"/>	12/20/2019	3	1006	3001	244

1 rows selected Total 6

Invoice			
<input type="button" value="New Hire"/>			
<input type="button" value="Reset"/>			
<input type="button" value="Search: All Text Columns"/>			
<input type="button" value="Go"/>			
<input type="button" value="Actions"/>			
<input type="button" value="Edit"/>			
<input type="button" value="Add Row"/>			
	Invoice Date	Total Amount	Loyalty Discount
<input checked="" type="checkbox"/>	12/20/2019	15300	2700

1 rows selected Total 1

Home Application 121608 Edit Page 6 Session View Debug Debug Page Info Quick Edit Theme Roller Help

ORACLE APEX App Builder SQL Workshop Team Development App Gallery

Application 121608 \ Page Designer

Page 6: Hire - Invoice Details

Regions

- Content Body
 - Hire
 - Invoice
- Buttons
- Page Items
- Computations
- Processes
- Branches

Page Rendering

- Regions
- Content Body
 - 20 Hire Interactive Grid
 - 30 Invoice Interactive Grid
- Buttons
 - 10 New Hire Go To Page 30
- Page Items
- Computations
- Processes
- Dynamic Actions

Page Processing

- Computations
- Validations
- Processes
- Processing
 - 10 Hire - Save Interactive Grid Data Interactive Grid - Automatic Row...
 - 20 Invoice - Save Interactive Grid Data Interactive Grid - Automatic Ro...
- Branches

Identification

- Name: Hire - Invoice Details
- Page Alias
- Title: Hire - Invoice Details
- Page Group: - Select -

Appearance

- User Interface: Desktop
- Page Mode: Normal
- Page Template: Theme Default
- Template Options: Use Template Defaults
- CSS Classes

CarShil Car Hire

Hire

Hire

Hire Date		Hire Noofdays	Fk1 Customer Id	Fk2 Accessory Id	Fk3 Vehicle Center Id
11/15/2019		5	1005	3001	234
<input style="width: 100px; height: 25px; border: none; background-color: #f0f0f0; padding: 2px 5px; font-size: 10px; margin-right: 10px;" type="button" value="November 2019"/> < >		2	1008	3003	206
		3	1002	3001	208
		3	1004	3004	244
		6	1001	3001	228
		3	1006	3001	244
					Total 6

Invoice

Invoice Date		Total Amount	Loyalty Discount
12/20/2019		15300	2700
			Total 1

1 rows selected

Home Application 121608 Edit Page Session View Debug Page Info Quick Edit Theme Roller

Insertion Forms for Master Page

The application consists of forms i.e. Manage Customer and Manage Hire. These forms are executed when the buttons in the master-detail form is clicked. These forms have been validated and use List of Values function to fulfill foreign key attributes to facilitate user to insert foreign keys.

Form for Customer – Hire Master Detail (with validations and list of values):

The screenshot shows a web-based application interface for managing customers. On the left, there's a sidebar menu with options like Home, Dashboard - Charts and Graphs, Administration, Master Detail Form, Insert Data - Form (selected), Manage Customer, Manage Hire, Charts, Calendar, Reports - Assignment 1, and Reports with Form. The main content area is titled 'Manage Customer' and contains several input fields: Customer ID, Customer Name, Customer Email, Customer Address, Customer Phoneno, Headquarter Id, and FK2 Insurance Policyno. Below these fields is a 'Cancel' button and a prominent red 'Create' button. At the bottom of the page, there's a toolbar with various icons and a status bar indicating 'Release 1.0' and other application details.

This screenshot shows the Oracle APEX Page Designer interface for the 'Page 29: Manage Customer' page. The left sidebar lists components: Regions, Content Body (Manage Customer), Buttons (Manage Customer, CANCEL, DELETE, SAVE, CREATE), and Page Items (Manage Customer, P29_CUSTOMER_ID, P29_CUSTOMER_NAME, P29_CUSTOMER_EMAIL, P29_CUSTOMER_ADDRESS, P29_CUSTOMER_PHONENO, P29_HEADQUARTER_ID, P29_FK2_INSURANCE_POLICYNO). The main workspace is divided into three sections: 'Page Rendering' (Regions, Content Body, Buttons), 'Page Processing' (Computations, Validations, Processes, Branches, After Processing), and 'Page Items' (Region, Customer ID, Customer Name, Customer Email, Customer Address, Customer Phoneno, Headquarter Id, FK2 Insurance Policyno). On the right, there's a 'Page Item' panel with settings for 'List of Values' (INSURANCE_NO) and 'Advanced' options. The top navigation bar includes links for ORACLE APEX, App Builder, SQL Workshop, Team Development, App Gallery, and various developer tools.

Manage Customer

Customer ID

Customer Name

Please fill out this field.

Customer Email

Customer Address

Please fill out this field.

Customer Phoneno

Please fill out this field.

Headquarter Id

Fk2 Insurance Policyno

Please select an item in the list.

[Cancel](#) [Create](#)

[Home](#) [Application 121608](#) [Edit Page 29](#) [Session](#) [View Debug](#) [Debug](#) [Page Info](#) [Quick Edit](#) [Theme Roller](#) [⚙️](#)

av.sapkota999@gmail.com

Manage Customer

Customer ID

Customer Name

Customer Email

Customer Address

Customer Phoneno

Headquarter Id

Fk2 Insurance Policyno

No Policy
Third Party
Vehicle
Passenger
Full Package

[Release 1.0](#) [Home](#) [Application 121608](#) [Edit Page 29](#) [Session](#) [View Debug](#) [Debug](#) [Page Info](#) [Quick Edit](#) [Theme Roller](#) [⚙️](#)

av.sapkota999@gmail.com

List of Values: INSURANCE_NO

	Name	Subscription	Source	Comments
<input type="button" value="Show All"/>				<input type="button" value="Cancel"/> <input type="button" value="Delete"/> <input type="button" value="Apply Changes"/>
Data Source Local Database				
* Source Type	<input checked="" type="radio"/> Table <input type="radio"/> SQL Query <input type="radio"/> Function Returning SQL ?			
* Table / View Owner	DATABASESYSTEMSASSESSMENT ?			
* Table / View Name	INSURANCE ?			
Where Clause	<input type="text"/>			
Column Mapping				
* Return	INSURANCE_POLICYNO ?			
Display	INSURANCE_NAME ?			
Default Sort	INSURANCE_POLICYNO ?			
Sort Direction	Ascending (Nulls Last) ?			
Group	- Not Defined - ?			
Group Sort Direction	Ascending (Nulls Last) ?			
Icon	- Not Defined - ?			
Oracle Text Column	- Not Defined - ?			

Form for Hire - Invoice Master Detail (with validations and list of values):

The screenshot shows the 'Manage Hire' page from an Oracle APEX application. The left sidebar contains navigation links for Home, Dashboard, Administration, Master Detail Form, Insert Data - Form (selected), Manage Customer, and Manage Hire. The main content area is titled 'Manage Hire' and contains the following fields:

- Hire ID
- Hire Date
- Hire Noofdays
- Fk1 Customer Id
- Fk2 Accessory Id
- Fk3 Vehicle Center Id

Below the fields are a 'Cancel' button and a prominent red 'Create' button. At the bottom of the page, there is a toolbar with various icons and a status bar indicating 'Release 1.0'.

The screenshot shows the Oracle APEX App Builder Page Designer for the 'Page 30: Manage Hire' page. The left sidebar lists components such as Regions, Content Body, Buttons, and Page Items. The right panel is divided into several sections:

- Page Rendering:** Shows the regions and content body for the 'Manage Hire' page.
- Page Processing:** Contains sections for Computations, Validations, Processes, and a detailed view of the 'Process' section for the 'Manage Hire' region.
- List of Values:** Configured for the 'VEHICLE_CLASS_ID' field, showing options for Shared Component, List of Values, and various display settings.

Manage Hire

Hire ID

Hire Date

Please fill out this field.

Hire Noofdays

Please fill out this field.

Fk1 Customer Id

Please select an item in the list.

Fk2 Accessory Id

Please select an item in the list.

Fk3 Vehicle Center Id

Please select an item in the list.

OK

Create

Edit List of Values | Page Designer | Manage Hire | Results

apex.oracle.com/pls/apex/f?p=121608:30:704758391180002:NOW...

YouTube | Mariana's Corner | Blackboard Learn | Login Page | Web Applications a... | Reading List | Web App Tech | Database Sample ... | Assignment List | EDC Warehouse

Home | Application 121608 | Edit Page 30 | Session | View Debug | Debug | Page Info | Quick Edit | Theme Roller |

Manage Hire

Hire ID

Hire Date

Hire Noofdays

Fk1 Customer Id

Fk2 Accessory Id

Roof rack
Baby seat
Car cover
Toolkit
GPS Navigator

Release 1.0

List of Values: ASSESSORY_ID

Name	Subscription	Source	Comments

Source

Data Source: Local Database

Source Type: Table

Table / View Owner: DATABASESYSTEMSASSESSMENT

Table / View Name: ACCESSORIES

Where Clause:

Column Mapping

Return: ACCESSORY_ID
Display: ACCESSORY_NAME
Default Sort: ACCESSORY_ID
Sort Direction: Ascending (Nulls Last)
Group: - Not Defined -
Group Sort Direction: Ascending (Nulls Last)
Icon: - Not Defined -

ORACLE APEX App Builder SQL Workshop Team Development App Gallery

Application 121608 \ Shared Components \ Lists of Values

Name	Type	Updated	Entry Count	Subscribed From	Subscribers
ACCESS_ROLES	Local	31 hours ago			
ASSESSORY_ID	Local	5 hours ago			
CUSTOMER_ID	Local	5 hours ago			
DESKTOP_THEME_STYLES	Local	31 hours ago			
EMAIL_USERNAME_FORMAT	Static	31 hours ago	1		
INSURANCE_NO	Local	5 hours ago			
LOGIN_REMEMBER_USERNAME	Static	31 hours ago	1		
TIMEREFAME (4 WEEKS)	Local	31 hours ago			
USER_THEME_PREFERENCE	Static	31 hours ago	1		
VEHICLE_CLASS_ID	Local	5 hours ago			
VIEW_AS_REPORT_CHART	Static	31 hours ago	2		

1 - 11

A List of Values can be referenced by page items as well as report fields. It controls the values displayed and limits the user's selection. Lists of Values can be static (based on values you enter), or dynamic (based on different data sources of either Local Database, REST Enabled SQL or Web Source Module).

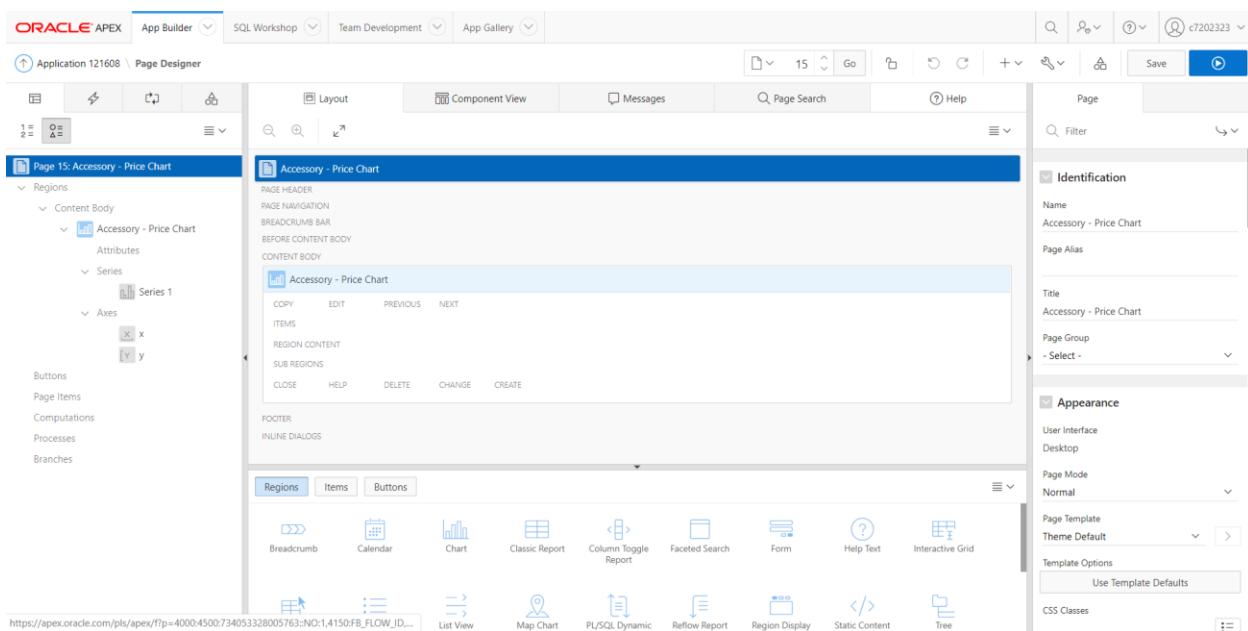
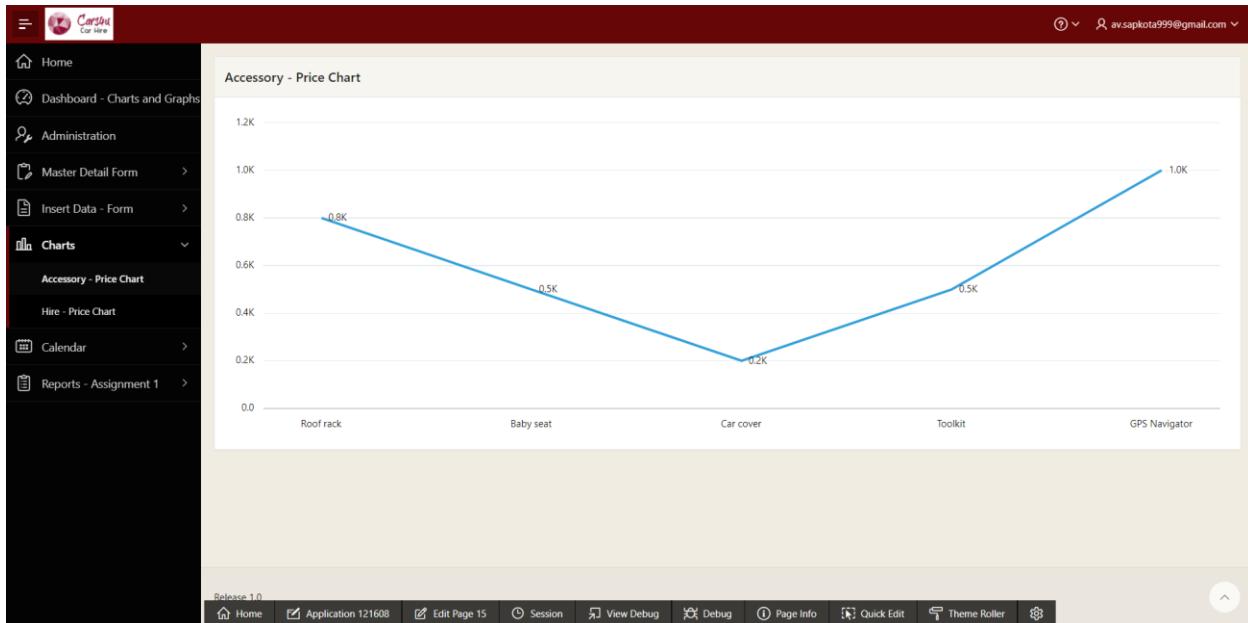
Recently Edited

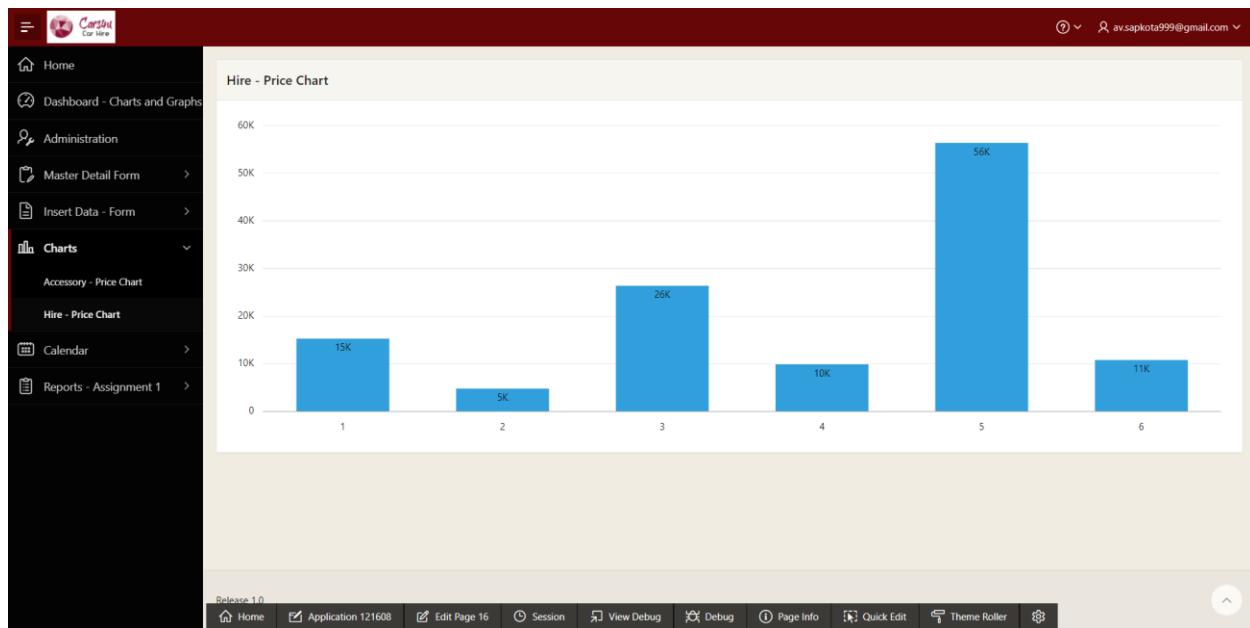
- Vehicle Class Id
- Insurance No
- Customer Id
- Assesory Id
- Email Username Format
- View As Report Chart
- Access Roles
- Desktop Theme Styles
- Timeframe (4 Weeks)
- User Theme Preference

av.sapkota999@gmail.com database_systems_assessment en Copyright © 1999, 2019, Oracle. All rights reserved. Application Express 19.2.0.0.18

Graphs and Charts

In addition, the application also has charts 'Accessory – Price Chart' and 'Hire – Price Chart' which makes the interface more attractive and easier to read progress values.





Page 16: Hire - Price Chart

Identification

- Name: Hire - Price Chart
- Page Alias:

Appearance

- User Interface: Desktop
- Page Mode: Normal

Page Template

- Theme Default

Template Options

- Use Template Defaults

CSS Classes

Calendars

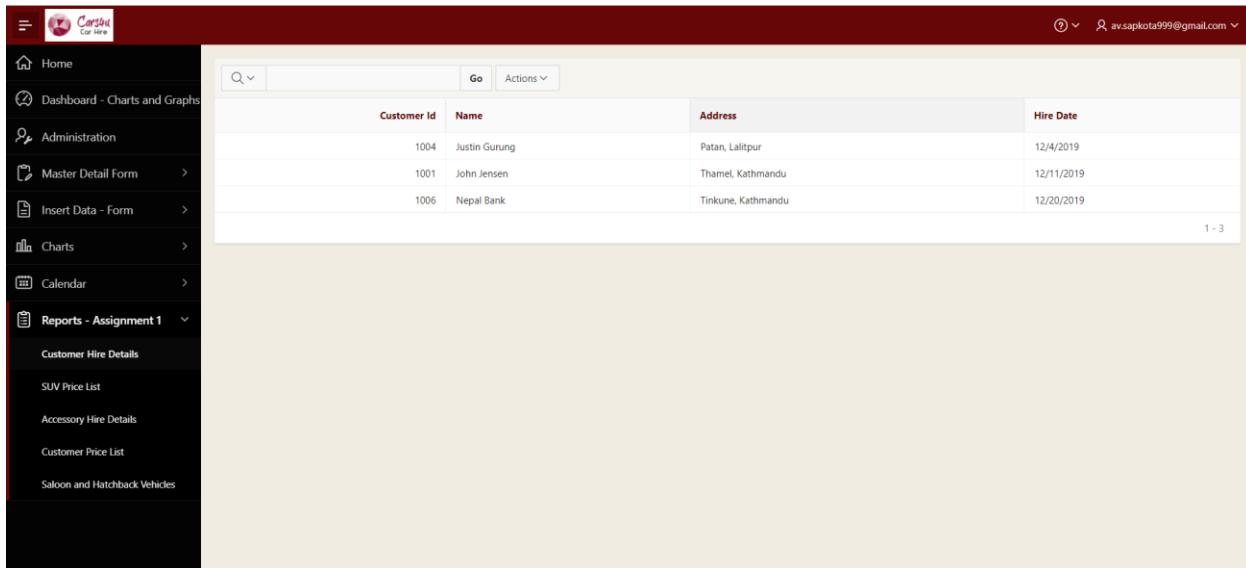
The application also has a calendar features for 'Contract Dates' during contract with user and 'Customer Hire Dates' to easily point out the hire dates of various users.

This screenshot shows the 'Contract Dates' page from an Oracle APEX application. The left sidebar navigation includes 'Home', 'Dashboard - Charts and Graphs', 'Administration', 'Master Detail Form', 'Insert Data - Form', 'Charts', 'Calendar' (selected), 'Contract Dates' (selected), 'Customer Hire Dates', and 'Reports - Assignment 1'. The main content area displays a calendar for October 2019. Blue bars highlight specific dates: October 10, 15, 23, and 25. The bottom navigation bar includes links for Home, Application 121608, Edit Page 17, Session, View Debug, Debug, Page Info, Quick Edit, Theme Roller, and a help icon.

This screenshot shows the 'Contract Dates' page in the Oracle APEX App Builder Page Designer. The left sidebar lists page items like Regions, Content Body, Buttons, Page Items, Computations, Processes, and Branches. The main workspace shows the 'Contract Dates' page structure with regions for Page Header, Page Navigation, Breadcrumb Bar, Before Content Body, Content Body, and Footer. The Content Body region contains a table with columns for COPY, EDIT, PREVIOUS, and NEXT. The right sidebar shows Identification (Name: Contract Dates, Page Alias), Appearance (User Interface: Desktop, Page Mode: Normal), and Page Template (Theme Default). The bottom navigation bar includes links for Regions, Items, Buttons, Breadcrumb, Calendar, Chart, Classic Report, Column Toggle Report, Faceted Search, Form, Help Text, Interactive Grid, List View, Map Chart, PL/SQL Dynamic, Reflow Report, Region Display, Static Content, and Tree.

Reports of Assignment Component 1

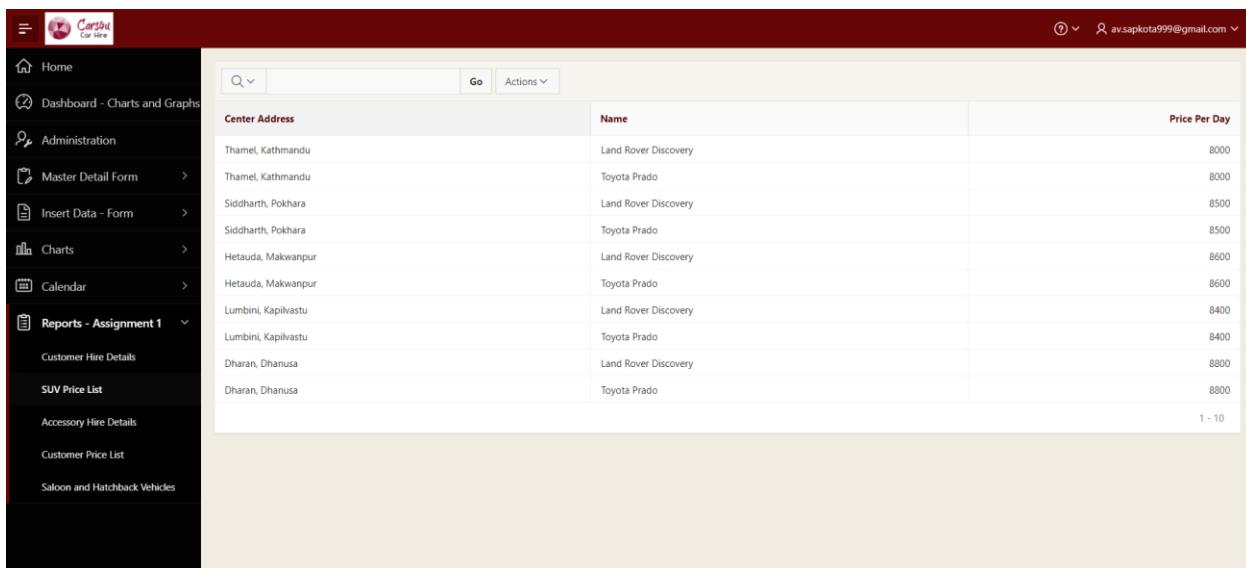
Furthermore, the application features with some reports, apparently the reports of component 1, that provides the data that customers and application users may frequently use during the use of this application.



The screenshot shows the CarShu application interface. The left sidebar contains navigation links: Home, Dashboard - Charts and Graphs, Administration, Master Detail Form, Insert Data - Form, Charts, Calendar, Reports - Assignment 1 (which is expanded), Customer Hire Details, SUV Price List, Accessory Hire Details, Customer Price List, and Saloon and Hatchback Vehicles. The main content area displays a table titled "Customer Hire Details" with columns: Customer Id, Name, Address, and Hire Date. The data in the table is as follows:

Customer Id	Name	Address	Hire Date
1004	Justin Gurung	Patan, Lalitpur	12/4/2019
1001	John Jensen	Thamel, Kathmandu	12/11/2019
1006	Nepal Bank	Tinkune, Kathmandu	12/20/2019

Page number 1 - 3 is visible at the bottom right of the table.



The screenshot shows the CarShu application interface, identical to the previous one but with a different report selected. The left sidebar shows the same navigation links. The main content area displays a table titled "Center Address" with columns: Center Address, Name, and Price Per Day. The data in the table is as follows:

Center Address	Name	Price Per Day
Thamel, Kathmandu	Land Rover Discovery	8000
Thamel, Kathmandu	Toyota Prado	8000
Siddharth, Pokhara	Land Rover Discovery	8500
Siddharth, Pokhara	Toyota Prado	8500
Hetauda, Makwanpur	Land Rover Discovery	8600
Hetauda, Makwanpur	Toyota Prado	8600
Lumbini, Kapilbastu	Land Rover Discovery	8400
Lumbini, Kapilbastu	Toyota Prado	8400
Dharan, Dhanusa	Land Rover Discovery	8800
Dharan, Dhanusa	Toyota Prado	8800

Page number 1 - 10 is visible at the bottom right of the table.

The screenshot shows a web-based application interface for managing vehicle accessories. On the left, there is a dark sidebar menu with various options like Home, Dashboard - Charts and Graphs, Administration, Master Detail Form, Insert Data - Form, Charts, Calendar, and Reports - Assignment 1. Under Reports - Assignment 1, the 'Accessory Hire Details' option is selected. The main content area displays a table with columns: Customer ID, Customer Name, Accessory, and Description. The table contains six rows of data.

Customer ID	Customer Name	Accessory	Description
1001	John Jensen	Roof rack	This is the carrier roof that vehicle needs
1002	Rohit Raj Pandey	Roof rack	This is the carrier roof that vehicle needs
1004	Justin Gurung	Toolkit	This helps in maintaining the vehicle
1005	Joey Foster	Roof rack	This is the carrier roof that vehicle needs
1006	Nepal Bank	Roof rack	This is the carrier roof that vehicle needs
1008	Nepal Bank	Car cover	This is a cover that covers vehicle

1 - 6

The screenshot shows a web-based application interface for managing customer hires. On the left, there is a dark sidebar menu with various options like Home, Dashboard - Charts and Graphs, Administration, Master Detail Form, Insert Data - Form, Charts, Calendar, and Reports - Assignment 1. Under Reports - Assignment 1, the 'Customer Price List' option is selected. The main content area displays a table with columns: Customer, Hire No Of Days, Hire Date, and Total Amount. The table contains five rows of data.

Customer	Hire No Of Days	Hire Date	Total Amount
John Jensen	6	12/11/2019	56400
Rohit Raj Pandey	3	11/20/2019	26400
Joey Foster	5	11/15/2019	15300
Nepal Bank	3	12/20/2019	10800
Justin Gurung	3	12/4/2019	9900

1 - 5

The screenshot shows a web-based application interface for 'Carzil Car Hire'. The left sidebar contains a navigation menu with items like Home, Dashboard - Charts and Graphs, Administration, Master Detail Form, Insert Data - Form, Charts, Calendar, and Reports - Assignment 1. Under Reports - Assignment 1, there are links for Customer Hire Details, SUV Price List, Accessory Hire Details, Customer Price List, and Saloon and Hatchback Vehicles. The main content area displays a table titled 'Vehicle Name' with columns for 'Vehicle Id' and 'Vehicle Name'. The table lists six vehicles: Ford Fiesta 1.1 (Vehicle Id 2370), Fiat Panda (Vehicle Id 6623), Hyundai i20 (Vehicle Id 7023), Nissan Sunny (Vehicle Id 2980), Honda City (Vehicle Id 3790), and Suzuki Swift D (Vehicle Id 6379). At the bottom of the page, there is a footer bar with links for Home, Application 121608, Edit Page 25, Session, View Debug, Debug, Page Info, Quick Edit, Theme Roller, and a gear icon. The footer also indicates 'Release 1.0'.

Vehicle Id	Vehicle Name
2370	Ford Fiesta 1.1
6623	Fiat Panda
7023	Hyundai i20
2980	Nissan Sunny
3790	Honda City
6379	Suzuki Swift D

Reports with Forms

The application also includes reports with form which allows user to read the report and make some edit if necessary, using forms. These reports are the reports of tables where triggers of task 1 has been inserted or updated.

Report with Form 1:

The screenshot displays two views of an Oracle APEX application. The top half shows a report page titled "Vehicle Details" with a table listing vehicle information. The bottom half shows the "Page Designer" interface for creating the same page.

Report Page (Top):

Vehicle Class Surr Id	Vehicle Id	Vehicle Name	Vehicle Makeyear	No Of Seat	Vehicle Class	Vehicle Description
910	2389	Tata Winger	2018	18	VAN	This is a van that is luxurious for 18 people and are good for long drives.
901	2370	Ford Fiesta 1.1	2013	5	Hatchback	A car of ford. It is a hatchback whose ground clearance is high
902	6623	Fiat Panda	2010	5	Hatchback	A car of Fiat. It is a hatchback good for city use
903	7023	Hyundai i20	2018	5	Hatchback	A car of Hyundai. It is a hatchback that is good in highway
904	2980	Nissan Sunny	2017	4	Saloon	A car of Nissan. It is a luxury sedan who's good for business hire
905	3790	Honda City	2014	4	Saloon	A car of Honda. It is a sedan which is luxurious in city
906	6379	Suzuki Swift D	2016	4	Saloon	A car of Suzuki. It is a sedan which is good for family purposes
907	2989	Toyota Hilux	2015	6	Pickup-truck	A car of Toyota. It is a pickup truck and has high power and more space
908	6000	Land Rover Discovery	2018	7	SUV	It is a luxury adventure SUV which is good on off-road and on highway
909	7093	Toyota Prado	2018	7	SUV	A car of Toyota. It is a SUV which is awesome for all purposes

Page Designer (Bottom):

The page designer shows the structure of "Page 8: Vehicle Details". It includes regions like "Content Body" containing "Report 1", "Buttons" with a "CREATE" button, and various page items like Breadcrumb, Calendar, Chart, etc. The right panel shows settings for "Identification" (Name: Vehicle Details, Page Alias), "Appearance" (User Interface: Desktop, Page Mode: Normal), and "Template Options" (Theme Default, Use Template Defaults).

The screenshot displays two views of an Oracle APEX application:

- Top View (Live Application):** Shows the "Manage Vehicles" page. The vehicle details are as follows:
 - VEHICLE_CLASS_SURR_ID: 909
 - Vehicle Id: 7093
 - Vehicle Name: Toyota Prado
 - Vehicle Makeyear: 2018
 - No Of Seat: 7
 - Vehicle Class: SUV
 - Vehicle Description: A car of Toyota. It is a SUV which is awesome for all purposes
 Buttons at the bottom include "Delete" and "Apply Changes".
- Bottom View (Page Designer):** Shows the "Page 9: Manage Vehicles" configuration. The page includes:
 - Regions:** Content Body, Buttons, Page Items.
 - Buttons:** Manage Vehicles (with CANCEL, DELETE, SAVE, CREATE icons).
 - Page Items:** P9_VEHICLE_CLASS_SURR_ID, P9_VEHICLE_ID, P9_VEHICLE_NAME, P9_VEHICLE_MAKEYEAR, P9_NO_OF_SEAT, P9_VEHICLE_CLASS, P9_VEHICLE_DESCRIPTION.
 - Content Body:** Displays the vehicle details from the live application.
 - Regions (Toolbar):** Breadcrumb, Calendar, Chart, Classic Report, Column Toggle Report, Faceted Search, Form, Help Text, Interactive Grid, Interactive Report, List, List View, Map Chart, PL/SQL Dynamic, Reflow Report, Region Display.
 - Page Properties:** Identification (Name: Manage Vehicles, Page Alias), Appearance (User Interface: Desktop, Page Mode: Normal, Page Template: Theme Default), and Template Options (Use Template Defaults, CSS Classes).

Report with Form 2:

The screenshot displays two views of an Oracle APEX application. The top view shows a user interface with a sidebar menu and a main content area containing a table of accessory details. The bottom view shows the Page Designer interface where the same report page is being edited.

Report Page Content (Top Screenshot):

Accessory Id	Accessory Name	Total Quantity	Accessory Description	Accessory Price
3001	Roof rack	3	This is the carrier roof that vehicle needs	800
3002	Baby seat	5	This is an extrenal seat for baby	500
3003	Car cover	20	This is a cover that covers vehicle	200
3004	Toolkit	20	This helps in maintaining the vehicle	500
3005	GPS Navigator	10	This navigates user through the route	1000

Page Designer (Bottom Screenshot):

The Page Designer shows the structure of the "Accessories Details" page. It includes regions like PAGE HEADER, PAGE NAVIGATION, BREADCRUMB BAR, BEFORE CONTENT BODY, and CONTENT BODY. A report item named "Report 1" is placed in the CONTENT BODY region. The right side of the designer provides page-level settings such as Name (Accessories Details), Title (Accessories Details), Page Group (Select), Appearance (User Interface, Desktop), and Page Mode (Normal). A toolbar at the bottom contains various report and form components.

The screenshot displays two views of an Oracle APEX application:

- Top View (Live Application):** Shows the "Manage Accessories" page with a form for editing accessory details. The form fields include ACCESSORY_ID (3005), ACCESSORY_NAME (GPS Navigator), TOTAL_QUANTITY (10), ACCESSORY_DESCRIPTION (This navigates user through the route), and ACCESSORY_PRICE (1000). Buttons for "Delete" and "Apply Changes" are at the bottom.
- Bottom View (Page Designer):** Shows the "Page Designer" interface for "Page 11: Manage Accessories". The left sidebar lists components like Content Body, Buttons, and Page Items. The main area shows the "Manage Accessories" page with its layout, items (P11_ACCESSORY_ID, P11_ACCESSORY_NAME, P11_TOTAL_QUANTITY, P11_ACCESSORY_DESCRIPTION), and buttons (CANCEL, DELETE, SAVE, CREATE). The right sidebar contains sections for Identification (Name: Manage Accessories, Page Alias), Appearance (User Interface: Desktop, Page Mode: Normal), and Page Template (Theme Default).

Report with Form 3:

The screenshot displays two views of an Oracle APEX application. The top view shows a report page titled "Hire Invoice Details" with a table of invoice data. The bottom view shows the "Page Designer" interface for creating the same page.

Report Page (Top):

	Invoice Id	Invoice Date	Total Amount	Loyalty Discount	Fkt Hire Id
5001	12/20/2019	15300	2700	1	
5002	11/21/2019	4800	-	2	
5003	11/23/2019	26400	-	3	
5004	12/7/2019	9900	-	4	
5005	12/17/2019	39200	16800	5	
5006	12/23/2019	15300	2700	6	

Page Designer (Bottom):

The Page Designer shows the "Hire Invoice Details" page structure. It includes regions like "Content Body" and "Buttons". The "Buttons" region contains a "Report 1" button. The right panel displays page properties for "Identification" (Name: Hire Invoice Details, Page Alias: Hire Invoice Details) and "Appearance" (User Interface: Desktop, Page Mode: Normal, Page Template: Theme Default). Below the page properties are "Template Options" and "CSS Classes".

The screenshot displays two views of the Oracle APEX application 'Manage Hire Invoice'.

Top View (Live Application):

- Page Title:** Manage Hire Invoice
- Form Fields:**
 - Invoice ID: 5005
 - Invoice Date: 12/17/2019
 - Total Amount: 39200
 - Loyalty Discount: 16800
 - Rk1 Hire Id: 5
- Buttons:** Cancel, Delete, Apply Changes

Bottom View (Page Designer):

- Page Title:** Page 13: Manage Hire Invoice
- Regions:**
 - Content Body: Manage Hire Invoice
 - Buttons: CANCEL, DELETE, SAVE, CREATE
 - Page Items: P13_INVOICE_ID, P13_INVOICE_DATE, P13_TOTAL_AMOUNT, P13_LOYALTY_DISCOUNT, P13_FK1_HIRE_ID
 - Computations
 - Processes: Initialize form Manage Hire Invoice
- Component View:** Shows the visual representation of the page with various items like Breadcrumb, Calendar, Chart, Classic Report, etc.
- Identification:** Page Name: Manage Hire Invoice, Page Alias: Manage Hire Invoice, Title: Manage Hire Invoice, Page Group: - Select -, Appearance: User Interface, Desktop, Page Mode: Normal, Page Template: Theme Default, Template Options: Use Template Defaults, CSS Classes.

Multiple User Access

Finally, the application allows multiple user access which facilitates administration facility, application operator and application reader. Each user type can only view application's features which has been allowed to access, for that particular user type.

The screenshot shows the Oracle APEX Application Access Control interface. At the top, there are tabs for App Builder, SQL Workshop, Team Development, and App Gallery. Below the tabs, the URL is Application 121608 \ Shared Components \ Application Access Control. The main area is titled "Application Access Control". It has three tabs: "Show All", "Roles", and "User Role Assignments". The "Roles" tab is selected, displaying a table with three rows:

Role	Static Identifier	Users	Description
Administrator	ADMINISTRATOR	1	Role assigned to application administrators.
Contributor	CONTRIBUTOR	1	Role assigned to application contributors.
Reader	READER	1	Role assigned to application readers.

To the right of the table, there is a sidebar titled "Access Control" with the following information:

- Access Control requires that developers define application users and roles. Since roles are applied to users, you must create roles before adding users. You can assign users to one or more roles.
- To control access to application pages and components, you need to create an Authorization Scheme.
- Roles and users defined for applications can be reviewed using the following views:
APEX_APPL_ACL_USERS
APEX_APPL_ACL_USER_ROLES
APEX_APPL_ACL_ROLES
- Use the following API to manage access control settings:
APEX_ACL

At the bottom of the page, there is a link: <https://apex.oracle.com/pls/apex/f?p=4000:2300:706221534988083::NO::RP:2300:#R8464873...>

Application Access Control

Role	Static Identifier	Users	Description
Administrator	ADMINISTRATOR	1	Role assigned to application administrators.
Contributor	CONTRIBUTOR		
Reader	READER		

User Role Assignments

User Name ↑
AV.SAPKOTA999@GMAIL.COM
CARS4U_CUSTOMER
CARS4U_TEAM

User Assignment

About Application Users

Application users are not exported as part of your application. When you deploy your application you will need to manually manage your user to role assignments. Roles are exported as part of an application export and imported with application imports.

* User Name **CARS4U_CUSTOMER**

Application Role Administrator Contributor Reader

Cancel **Delete** **Save** 1 - 3

User Assignment

About Application Users

Application users are not exported as part of your application. When you deploy your application you will need to manually manage your user to role assignments. Roles are exported as part of an application export and imported with application imports.

* User Name **CARS4U_TEAM**

Application Role Administrator Contributor Reader

Cancel **Delete** **Save** 1 - 3

Access Control requires that developers define application users and roles. Since roles are applied to users, you must create roles before adding users. You can assign users to one or more roles.

To control access to application pages and components, you need to create an Authorization Scheme.

Roles and users defined for applications can be reviewed using the following views:
 APEX_APPL_ACL_USERS
 APEX_APPL_ACL_USER_ROLES
 APEX_APPL_ACL_ROLES

Use the following API to manage access control settings:
 APEX_ACL

[Learn More ...](#)

ORACLE APEX App Builder SQL Workshop Team Development App Gallery

Manage Users and Groups

Users Groups Group Assignments

User	Email	Account Type	Locked	Builder Last Login	Created
AV.SAPKOTA999@GMAIL.COM	av.sapkota999@gmail.com	Workspace Administrator	No	11 hours ago	3 weeks ago
CARS4U_CUSTOMER	CARS4U_CUSTOMER@CARS4U.COM	End User	No	-	28 hours ago
CARS4U_TEAM	CARS4U_TEAM@CARS4U.COM	End User	No	-	28 hours ago

1 - 3

Manage Users

Use this page to control access for Application Express workspace administrators, application developers, and end users.

Workspace administrators can create and modify applications and database objects as well as manage user accounts, groups, and development services.

Developers can create and modify applications and database objects.

End users have no development privileges and can only access applications that do not use an external authentication scheme.

[Learn More ...](#)

Tasks

- Change My Password >
- About Application Express >
- Monitor Activity >
- Manage Service >
- Manage User Groups >
- Create User >
- Create Multiple Users >

User: CARS4U_CUSTOMER

Show All Edit User Account Privileges Password Group Assignments

Username: CARS4U_CUSTOMER

Email Address: CARS4U_CUSTOMER@CARS4U.COM

First Name:

Last Name:

Description:

Default Date Format:

Cancel Delete User **Apply Changes**

authenticate against the username and password stored in these accounts.

If you want to change a user's password, enter a new password. Otherwise, leave the password columns null and the current password will not be changed.

Manage Users

Use this page to control access for Application Express workspace administrators, application developers, and end users.

Workspace administrators can create and modify applications and database objects as well as manage user accounts, groups, and development services.

Developers can create and modify applications and database objects.

End users have no development privileges and can only access applications that do not use an external authentication scheme.

[Learn More ...](#)

Account Privileges

Default Schema: DATABASESYSTEMASSESSMENT

Accessible Schemas (null for all):

User is a workspace administrator: Yes No

User is a developer: Yes No

App Builder Access: No

SQL Workshop Access: No

Team Development Access: Yes

Account Availability: Unlocked

Application Administrator

Application's administrator can access all the features of the application.

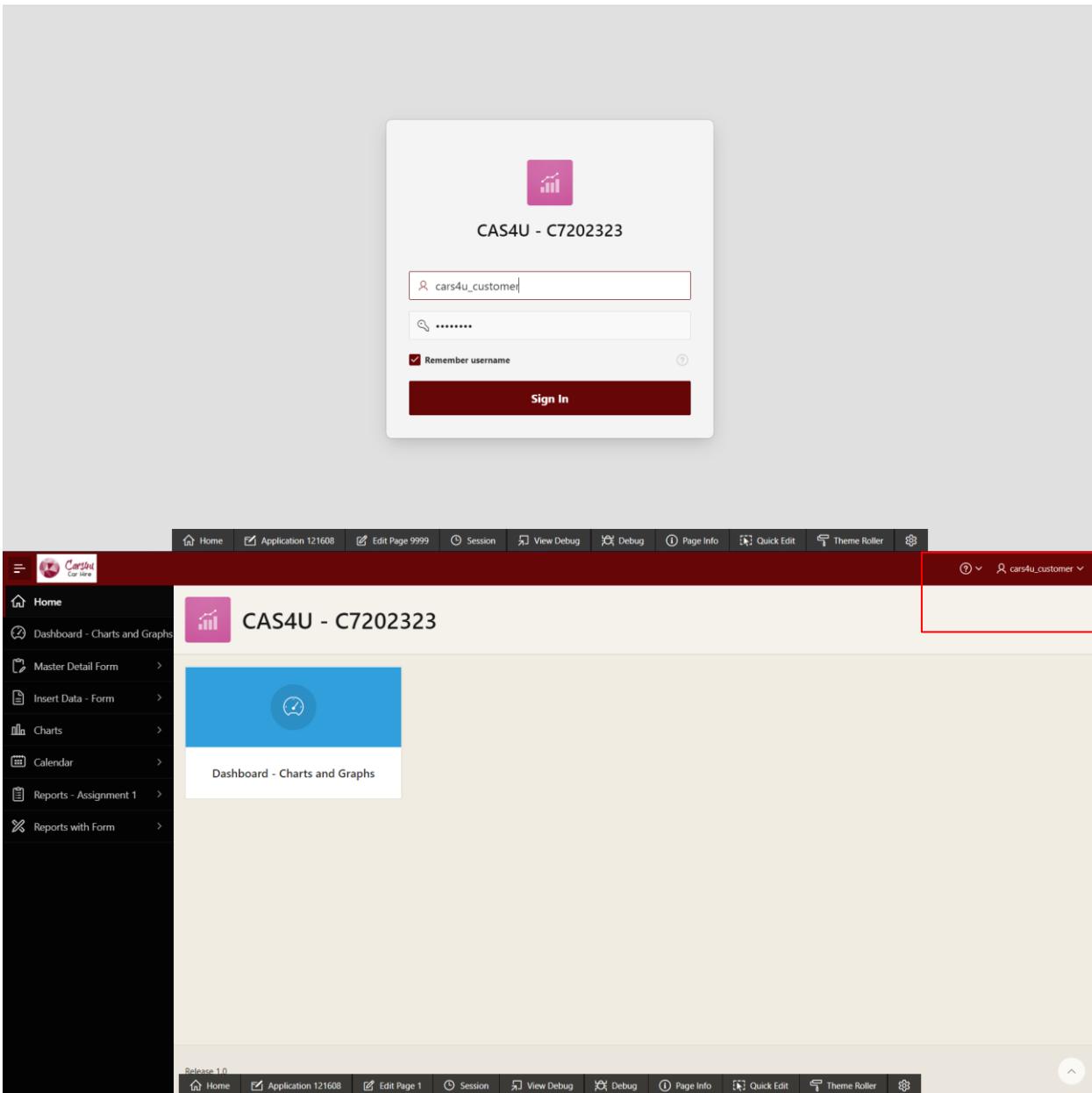
The screenshot shows the 'Administration' section of the CarShed application. On the left, there is a navigation sidebar with links such as Home, Dashboard - Charts and Graphs, Administration, Master Detail Form, Insert Data - Form, Charts, Calendar, Reports - Assignment 1, and Reports with Form. The main content area is titled 'Administration' and contains several sections:

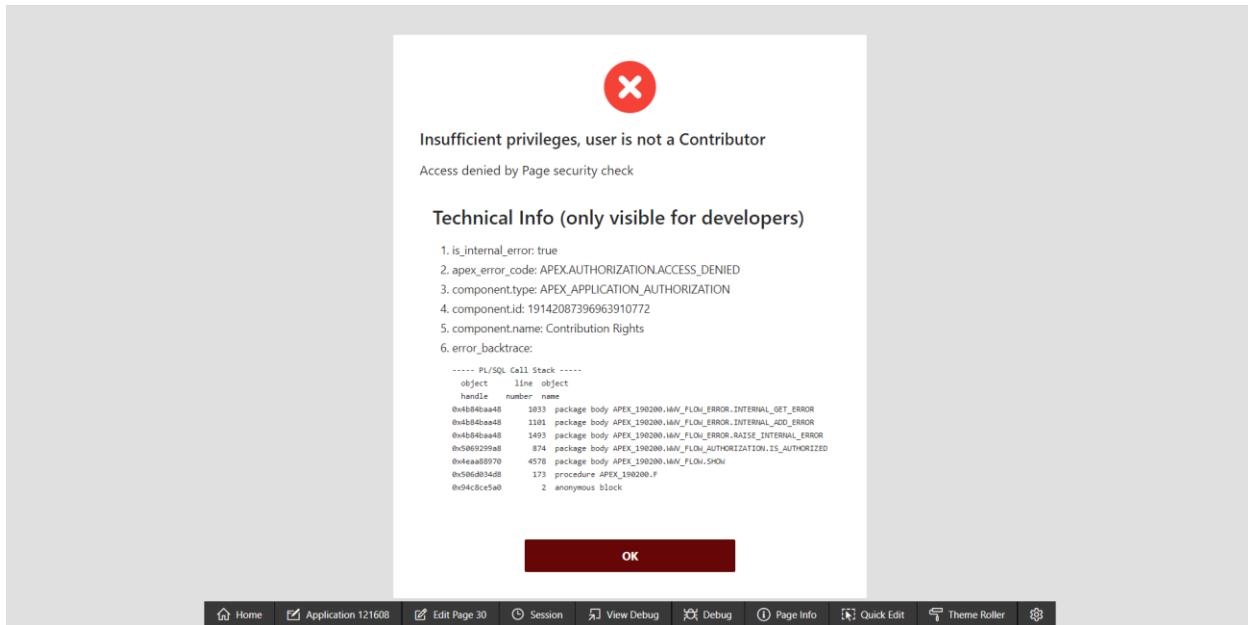
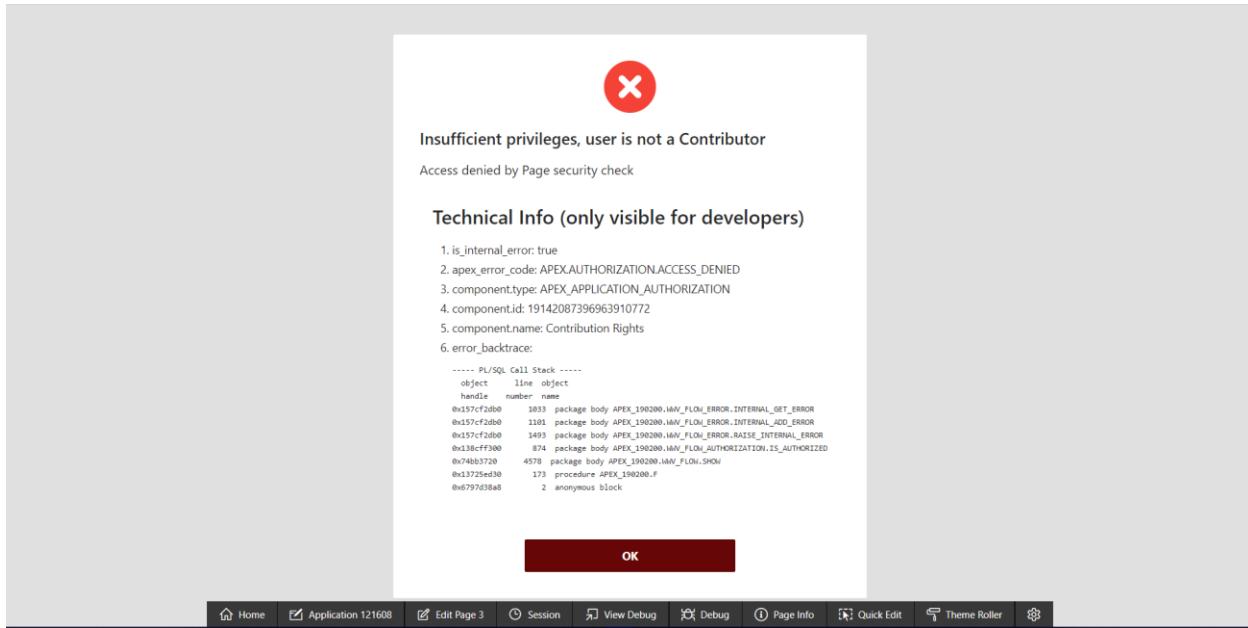
- Configuration Options**: Enable or disable application features.
- User Interface**: Theme Style Selection (Set the default application look and feel).
- Activity Reports**:
 - Dashboard (View application activity metrics)
 - Top Users (Report of page views aggregated by user)
 - Application Error Log (Report of errors logged by this application)
 - Page Performance (Report of activity and performance by application page)
 - Page Views (Report of each page view by user including date of access and elapsed time)
- Access Control**:
 - Only users defined in the application access control list may access this application
 - Administrator (1)
 - Contributor (1)
 - Reader (1)
 - Users**: Change access control settings and disable access control.
 - Access Control**: Set level of access for authenticated users of this application

At the bottom of the screen, there are several navigation icons: Session, View Debug, Debug, Page Info, Quick Edit, Theme Roller, and a gear icon. The URL in the address bar is: javascript:apex.navigation.dialog('f?p=121600:10010:704758391188002::NO::10010:j0026p_d...').

Application's Reader – CARS4U Customer

Application's reader can access home page, dashboard, charts, calendar and report from the application. However, they cannot access master detail forms, insertion forms and reports with forms as user are not meant to make changes into the system according to the CARS4U point of view (assumed).





Application's Contributor – CARS4U Team

Application's contributor can access home page, dashboard, master-detail forms, insertion forms, charts, calendar, reports and reports with form from the application. As CARS4U team is the one who is going to use this application and update tables and records accordingly (assumed). However, they do not have access to the backend of the application (only administrator can use those functionalities).

The screenshot displays the CARS4U application interface. At the top, there is a sign-in dialog box with a pink header containing a bar chart icon. The text "CAS4U - C7202323" is displayed below the header. The sign-in form includes fields for "cars4u_team" (username) and a password (represented by dots), a "Remember username" checkbox, and a "Sign In" button. Below the sign-in box is the main application interface. The header bar is dark red with various navigation icons and links. On the left, a black sidebar menu lists several options: Home, Dashboard - Charts and Graphs, Master Detail Form, Insert Data - Form, Charts, Calendar, Reports - Assignment 1, and Reports with Form. The central content area shows a dashboard titled "CAS4U - C7202323". This dashboard features a large blue section with a circular loading icon and the text "Dashboard - Charts and Graphs". A red rectangular box highlights the search bar in the top right corner of the dashboard area. The footer is a light gray bar with standard application links like Home, Application 121608, Edit Page, Session, View Debug, Debug, Page Info, Quick Edit, Theme Roller, and a gear icon.

The screenshot shows a web-based application interface for managing customer and hire details. The top navigation bar includes a logo for 'Cars4u Car Hire', a search bar with placeholder text 'Search...', and a dropdown menu for 'cars4u_team'. A red box highlights the 'Add Customer' button in the top right corner.

Customer - Hire Details

Customer

Anita Rijal arijal@gmail.com	Customer Name: Nepal Bank
Joey Foster jfoster@gmail.com	Customer Email: nepalbank@gmail.com
John Jensen jjensen@gmail.com	Customer Address: Chabahil, Kathmandu
Justin Gurung jgurung@yahoo.com	Customer Phoneno: 14402324
Nepal Bank nepalbank@gmail.com	Headquarter Id: 1006
Nepal Bank nepalbank@gmail.com	Fk2 Insurance Policyno: 800

Hire Arrangement

No data found.

Release 1.0

Home | Application 121608 | Edit Page 3 | Session | View Debug | Debug | Page Info | Quick Edit | Theme Roller | ☰

Bibliography

- SQL Statements. (2019). docs.oracle.com [online] Available at: https://www.docs.oracle.com/cd/E11882_01/timesten.112/e21642/state.htm [Accessed 12,13,14 Jan. 2020].
- Oracle Learning Library (2009). apexapps.oracle.com [online] Available at: https://apexapps.oracle.com/pls/apex/f?p=44785:2:0:FORCE_QUERY::2,RIR,CIR:P2_GROUP_ID,P2_PRODUCT_ID:1000%2C2039 [Accessed 12,13,14 Jan. 2020].
- Oracle Database Server Messages. docs.oracle.com [online] Available at: http://docs.oracle.com/cd/B10500_01/server.920/a96525/rdbms_pa.htm [Accessed 13,14 Jan. 2020].
- PL/SQL challenge website. Plsqlchallenge.oracle.com [online] Available at: <https://plsqlchallenge.oracle.com/pls/apex/f?p=10000:8:::::> [Accessed 12 Jan. 2020].

Appendix

- DDL and DML codes have been attached into the script folder.
- SQL for reports and PL/SQL triggers has also been attached into the script folder.
- Screenshots have been attached with the assignment inside a screenshots folder.