

# **System Manual of Attendance Software**

Biman Bangladesh Airlines Ltd.

Developed & Designed by

General Automation Ltd.

House # Halcyon heights (2<sup>nd</sup> Floor) 2/3, Block-A,  
Mirpur Road, Mohammadpur, Dhaka-1207, Bangladesh.

E-mail: [ga@bdmail.net](mailto:ga@bdmail.net)

# INDEX

SERIAL	TITLE	PAGE
1.	<b>Data Flow Diagram</b>	7
2.	<b>DESCRIPTION OF DATABASE</b>	8
	2.A Used Object in "ATT_IN" Schema	8
	2.B Used Constraint in "ATT_IN" Schema	13
3.	2.C Description of User Role	14
4.	Object wise description which are used in "ATT_IN" schema	19
	<b>A) Tables</b>	19
	1. TBL_4_SHIFT_ROSTER	19
	2. TBL_BUSINESSDIVISION	19
	3. TBL_CHECK	19
	4. TBL_COMPANY_INFO	20
	5. TBL_DEPARTMENT	20
	6. TBL_DESIGNATION_SETUP	20
	7. TBL_EMP_OT_APPROVE	21
	8. TBL_EMP_OT_APPROVE_HISTORY	21
	9. TBL_EMP_PERSONAL_INFO	21
	10.TBL_EMP_PERSONAL_INFO_HISTORY	22
	11.TBL_FINANCIAL_YEAR	24
	12.TBL_FORM	24
	13.TBL_GOVТ_HOLIDAY_LIST	24
	14.TBL_LABOUR	25
	15.TBL_LEAVE_APPLY	25
	16. TBL_LEAVECODE	25
	17.TBL_LEAVE_APPLY_HISTORY	25
	18.TBL_LEAVE_TYPE	26
	19.TBL_MACHINE_INFO	27
	20.TBL_OSD_SETUP	27
	21.TBL_OSD_SETUP_HISTORY	27
	22.TBL_OT_EDIT	28
	23.TBL_OT_ENTRY	28
	24.TBL_PLANTFACTORY	29
	25.TBL_PROCESSED_DATA	29
	26.TBL_RAW_DATA	30
	27.TBL_RAW_DATA_CMIS	30
	28.TBL_RAW_DATA_ERROR	30
	29.TBL_RAW_DATA_HISTORY	31

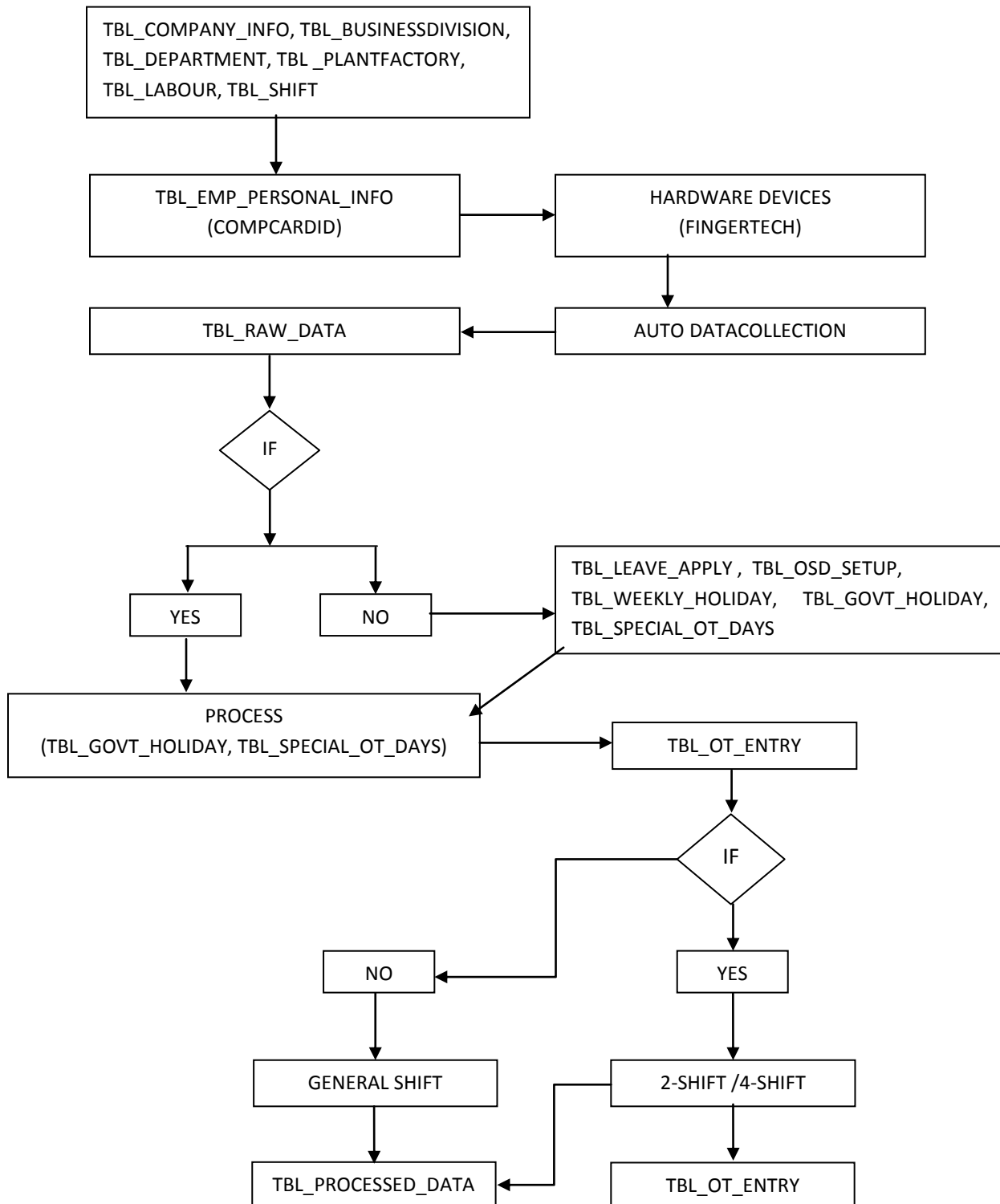
<b>30.TBL_SEASON_SETUP</b>	31
<b>31.TBL_SHIFT_SETUP</b>	32
<b>32.TBL_SHIFT_TYPE</b>	32
<b>33.TBL_SHIFTNAMESETTINGS</b>	32
<b>34.TBL_SPECIAL_OT_DAYS</b>	32
<b>35.TBL_USER</b>	33
<b>36.TBL_USER_HISTORY</b>	33
<b>37.TBL_USER_ROLE_PERMISSION</b>	34
<b>38.TBL_USERTYPE</b>	34
<b>39.TBL_WEEK_DAY</b>	34
<b>40.TBL_WEEKLY_HOLIDAY</b>	34
<b>B) VIEWS</b>	35
<b>1. V_ATTANDANCE_REPORT</b>	35
<b>2. V_EMP_OTLIST</b>	36
<b>3. V_EMP_REGULAR_STATUS</b>	36
<b>4. V_EMPLOYEE_ALL_INFO</b>	37
<b>5. V_OT_EDIT</b>	38
<b>C) FUNCTIONS</b>	39
<b>1. DATEDIFF</b>	39
<b>2. FN_MINUTE_TO_TIME_SHORTREST_OT</b>	39
<b>3. FN_MINUTE_TO_TIME_SPECIAL_OT</b>	40
<b>4. FN_MINUTE_TO_TIME_TEXT</b>	40
<b>5. FN_MINUTE_TO_TIME_TEXT_REPORT</b>	41
<b>6. MAKEMINUTE_FROMVALUE</b>	41
<b>7. MAKETIMEFROMMINUTE</b>	42
<b>8. MAKETIMEFROMMINUTE_FORLATE</b>	42
<b>9. FN_OT_HOUR_14082013_TXTFILE</b>	43
<b>10.FN_PAYBLE_OT_HOUR_14082013</b>	59
<b>11.FN_SHORTREST_SUM</b>	76
<b>12.FN_TOOTHOURJOBCARD</b>	77
<b>13.FN_TOOTHOURJOBCARD_TEXT</b>	79
<b>14.FN_TOTALLATEHOUR</b>	82
<b>15.FN_TOTALLATEHOUR_TEXT</b>	82
<b>16.FN_LEAVECODE</b>	83
<b>17. FN_LEAVECODE1</b>	85
<b>18. FN_LEAVECODE2</b>	86
<b>19. FN_LEAVEDAYS1</b>	87
<b>20. FN_LEAVEDAYS2</b>	88
<b>21.FN_LEAVE_OSD_COUNT</b>	90
<b>22.FN_SHIFTDAYS</b>	90
<b>23.FN_MEALALLOWANCE</b>	91
<b>24.FN_TOSHORTRESTJOBCARD4SHIFT</b>	92
<b>25. FN_TOOTHOURJOBCARD4SHIF</b>	93
<b>D) TRIGGERS</b>	94
<b>1. DELETE_UPDATE_LEAVE_APPLY</b>	94
<b>2. DELETE_UPDATE_TBL_EMP_INFO</b>	95
<b>3. T_COMPFINYID_INCR</b>	96

4. T_COMPID_INCR	96
5. T_EMP_OT_APPROVE_HISTORY	97
6. T_DEPTIDINCREMENT	97
7. T_FORM_ID_INCR	97
8. T_GOVTHOLIID_INCR	97
9. T_LEAVEAPPLYID_INCR	98
10. T_LEAVEYPEID_INCR	98
11. T_MACHINEID_INCR	98
12. T_OSDID_INCR	98
13. T_SEASONID_INCR	99
14. T_SHIFTID_INCR	99
15. T_SHIFTSETUPID_INCR	99
16. T_USER_HISTORY	99
17. T_USER_TYPE	100
18. TBL_RAW_DATA_HISTORY	100
19. T_UNRESTID_INCR	100
<b>E) PROCEDURES</b>	101
1. SP_COMPANY_ADD_UPDATE	101
2. SP_DATA_PROCESS_BIMAN_CMIS	101
3. SP_DESIGNATION_ADD_UPDATE	103
4. SP_EMPLOYEE_ADD_UPDATE_BIMAN	108
5. SP_FINANTIALYEAR_ADD_UPDATE	109
6. SP_FORM_ADD_UPDATE	112
7. SP_FOUR_SHIFT_2014_CMIS	113
8. SP_GENERAL_SHIFT	115
9. SP_GOVTHOLIDAY_ADD_UPDATE	150
10. SP_INSERT_ROWDATA	163
11. SP_LEAVE_APPLIED_ADD_UPDATE	164
12. SP_LEAVE_TYPE_ADD_UPDATE	165
13. SP_MANUAL_INSERT	167
14. SP_NIGHT_SHIFT_1	169
15. SP_NIGHT_SHIFT_2	173
16. SP_OSD_ADD_UPDATE	177
17. SP_OT_APPROVE_ADD_UPDATE	181
18. SP_OT_CALCULATION	183
19. SP_PLANTFACTORY_ADD_UPDATE	184
20. SP_ROLE_ADD_UPDATE	191
21. SP_SEASON_ADD_UPDATE	182
22. SP_SHIFTNAME_ADD_UPDATE	194
23. SP_SHIFT_ADD_UPDATE	195
24. SP_SHORT_REST_CALCULATION	198
25. SP_SPECIAL_OT_CALCULATION	223
26. SP_TWO_SHIFT_OT_TEST_TANIA	234
<b>F) SEQUENCES</b>	277
1. SEQ_COMPFINYID	277
2. SEQ_COMPID	277
3. SEQ_DEPTID	277
4. SEQ_EMPID	277

	5. SEQ_FORM_ID	277
	6. SEQ_GOVTHOLID	277
	7. SEQ_LEAVEAPPLYID	277
	8. SEQ_LEAVEYPEID	277
	9. SEQ_OSDID	277
	10. SEQ_PLANT_ID	278
	11. SEQ_SEASONID	278
	12. SEQ_SHIFTID	278
	13. SEQ_SHIFTSETUPID	278
	14. SEQ_USER_TYPE_ID	278
	15. SEQ_USERID	278
	16. SEQ_WEEKLYHOLID	278
5.	<b>Description of Application</b>	279
6.	<b>Technical Explanations of Forms used in the application</b>	279
7.	<b>Menu Form</b>	279
8. 9.	<b>Submenu Items under main Menu</b> 1. File 2. Initial Setup 3. Employee Setup 4. Data 5. Attendance Report 6. User 7. Utility	279
	<b>DETAILS SUBMENU ITEMS</b>  1. File	282
	2. Initial Setup i. FMB_COMPANY_SETUP ii. FMB_FINANCIAL_YEAR iii. FRB_SHIFT iv. FMB_SHIFT_TIME_SETUP v. FMB_LEAVE_TYPE_ENTRY vi. FMB_LABOUR_ENTRY vii. FMB_WEEKLY_HOLIDAY_SETUP viii. FMB_GOVTHD_ENTRY_FORM ix. FMB_SPECIAL_OT_DAYS x. FMB_OSD_ENTRY_FORM xi. FMB_CHECK	282
	<b>3. Employee Setup</b> i. FMB_NEW_EMPLOYEE_ENTRY ii. FMB_LEAVE_APPLY iii. FMB_2SHIFTOT_ENTRY iv. FMB_4SHIFTOT_ENTRY	295

	v. FMB_OT_APPROVED vi. FMB_OT_EDIT	
	<b>4. Data</b> i. FMB_DATA_COLLECTION ii. FMB_MANUAL_ENTRY iii. FMB_TEXT_FILE_GENERATE	302
	<b>5. User</b> i. FMB_CREATE_USER_TYPE ii. FMB_CREATE_USER iii. FMB_USER_ROLE iv. FMB_USER_ROLE_UPDATE v. FMB_FORM_ENTRY vi. USER_ROLE vii. USER_PASSWORD_CHANGE	305

## Data Flow Diagram



# Description of Database

## A) Used Object in "ATT\_IN" Schema

SL No	Object Name	Object Type	Used For
1.	TBL_4_SHIFT_ROSTER	TABLE	Information of "4-SHIFT " Roster Name
2.	TBL_BUSINESSDIVISION	TABLE	Information of Business division
3.	TBL_CHECK	TABLE	Used to handle exceptional Condition
4.	TBL_COMPANY_INFO	TABLE	Information of Company
5.	TBL_DEPARTMENT	TABLE	Information of Departments
6.	TBL_DESIGNATION_SETUP	TABLE	Information of Designation
7.	TBL_EMP_OT_APPROVE	TABLE	To give approval of OT for an employee
8.	TBL_EMP_OT_APPROVE_HISTORY	TABLE	To take history of OT for an employee
9.	TBL_EMP_PERSONAL_INFO	TABLE	Detail information of an employee
10.	TBL_EMP_PERSONAL_INFO_HISTORY	TABLE	To take history about Employee
11.	TBL_FINANCIAL_YEAR	TABLE	To entry the financial year for a company
12.	TBL_FORM	TABLE	To entry the Form name of the Attendance module
13.	TBL_GOVT_HOLIDAY_LIST	TABLE	
14.	TBL_LABOUR	TABLE	To input Labor Code Information
15.	TBL_LEAVE_APPLY	TABLE	To apply leave for an employee
16.	TBL_LEAVECODE	TABLE	Generate Special Leave code for Monthly report and text file
17.	TBL_LEAVE_APPLY_HISTORY	TABLE	To keep the leave history of an employee
18.	TBL_LEAVE_TYPE	TABLE	To input category of leave
19.	TBL_MACHINE_INFO	TABLE	To input information about attendance machine of Biman
20.	TBL_OSD_SETUP	TABLE	To input information about employees out station official work
21.	TBL_OSD_SETUP_HISTORY	TABLE	To keep information about employees out station official work
22.	TBL_OT_EDIT	TABLE	Used to Edit OT
23.	TBL_OT_ENTRY	TABLE	To entry 2Shift or 4Shift Roster for an employee
24.	TBL_PLANTFACTORY	TABLE	To input Location Information
25.	TBL_PROCESSED_DATA	TABLE	After process data is insert in this table
26.	TBL_RAW_DATA	TABLE	From machine data is load in this table



27.	TBL_RAW_DATA_CMIS	TABLE	When we process data is taken from this table
28.	TBL_RAW_DATA_ERROR	TABLE	Duplicate data is insert in this table
29.	TBL_RAW_DATA_HISTORY	TABLE	When data is delete then keep history using this table
30.	TBL_SEASON_SETUP	TABLE	To input any new season
31.	TBL_SHIFT_SETUP	TABLE	
32.	TBL_SHIFT_TYPE	TABLE	Category of Shift is input in this table
33.	TBL_SHIFTNAMESETTINGS	TABLE	Shift Name is insert in this table
34.	TBL_SPECIAL_OT_DAYS	TABLE	Special Holiday like May Day etc are input in this table
35.	TBL_USER	TABLE	To input User Information
36.	TBL_USER_HISTORY	TABLE	To keep user Information History
37.	TBL_USER_ROLE_PERMISSION	TABLE	To give insert, update, delete or select permission about form to user
38.	TBL_USERTYPE	TABLE	To input category of user
39.	TBL_WEEK_DAY	TABLE	To input days of a week
40.	TBL_WEEKLY_HOLIDAY	TABLE	To input Weekly Holiday for general shift
41.	V_ATTANDANCE_REPORT	VIEW	Used for job card repot ,monthly report and many others functions and procedure
42.	V_EMP_OTLIST	VIEW	Used to create a list who avail OT
43.	V_EMP_REGULAR_STATUS	VIEW	Used for V_ATTANDANCE_REPORT
44.	V_EMPLOYEE_ALL_INFO	VIEW	Used for V_EMP_REGULAR_STATUS
45.	V_OT_EDIT	VIEW	Used to Edit OT
46.	DATEDIFF	FUNCTION	This function is Used for calculate time between two date using dual table
47.	FN_MINUTE_TO_TIME_SHORTREST_OT	FUNCTION	This function is used for convert "Short Rest" minute to specific time format.
48.	FN_MINUTE_TO_TIME_SPECIAL_OT	FUNCTION	This function is used for convert "Special OT" minute to specific time Format.
49.	FN_MINUTE_TO_TIME_TEXT	FUNCTION	This function is used for convert "Payable OT" minute to specific time Format for Text file.
50.	FN_MINUTE_TO_TIME_TEXT_REPORT	FUNCTION	This function is used for convert "Payable OT" minute to specific time Format for Monthly Report.
51.	MAKEMINUTE_FROMVALUE	FUNCTION	This function is used for make time to sum of minute from given value.
52.	MAKETIMEFROMMINUTE	FUNCTION	This function is used for make minute to a specific time from given value.
53.	MAKETIMEFROMMINUTE_FORLATE		This function is used for make time to sum of late minute from given value
54.	FN_OT_HOUR_14082013_TXTFILE	FUNCTION	This function is used to calculate payable OT weekly for Text File between given From date and To date.
55.	FN_PAYBLE_OT_HOUR_14082013	FUNCTION	This function is used to calculate payable OT weekly for Monthly Report between given From date and To date.
56.	FN_SHORTREST_SUM	FUNCTION	This function is used to calculate ShortRest Hour for Monthly Report between given From date and To date.
57.	FN_TOOTHOURJOBCARD	FUNCTION	This function is used to calculate Actual OT Hour for Job Card Report of general shift Employees and Monthly Report for all employees between given From date and To date.

58.	FN_TOOTHOURJOBCARD_TEXT	FUNCTION	This function is used to calculate Actual OT Hour for Text File between given From date and To date.
59.	FN_TOTALLATEHOUR	FUNCTION	This function is used to calculate total Late Hour for Monthly Report between given From date and To date.
60.	FN_TOTALLATEHOUR_TEXT	FUNCTION	This function is used to calculate total Late Hour for Text File between given From date and To date.
61.	FN_LEAVECODE	FUNCTION	This function is used to return more than one type of Leave Category for Monthly Report between given From date and To date.
62.	FN_LEAVECODE1	FUNCTION	This function used in text file generate leave code 1
63.	FN_LEAVECODE2	FUNCTION	This function used in text file generate leave code 2
64.	FN_LEAVEDAYS1	FUNCTION	This function used in text file return count 1 days leave code 1
65.	FN_LEAVEDAYS2	FUNCTION	This function used in text file return count 2 days leave code 2
66.	FN_LEAVE_OSD_COUNT	FUNCTION	This function is used to verify Employee's applied leave or OCS in procedure SP_FOUR_SHIFT_2014_CMIS between given From date and To date
67.	FN_SHIFTDAYS	FUNCTION	This function is used to return for shifted employees shifting days for Monthly Report and Text File between given From date and To date.
68.	FN_MEALALLOWANCE	FUNCTION	This function is used to calculate total Short Rest Hour for Job Card Report of 2Shift /4 Shift Employees between given From date and To date.
69.	FN_TOSHORTRESTJOBCARD4SHIFT	FUNCTION	This function is used to calculate Actual OT Hour for Job Card Report of 2Shift /4 Shift Employees between given From date and To date.
70.	FN_TOOTHOURJOBCARD4SHIF	FUNCTION	
71.	DELETE_UPDATE_LEAVE_APPLY	TRIGGER	When delete or update TBL_LEAVE_APPLY then insert update value TBL_LEAVE_APPLY_HISTORY
72.	DELETE_UPDATE_TBL_EMP_INFO	TRIGGER	When delete or update TBL_EMP_PERSONAL_INFO then insert update value TBL_EMP_PERSONAL_INFO_HISTORY
73.	T_COMPFINYID_INCR	TRIGGER	
74.	T_COMPID_INCR	TRIGGER	
75.	T_EMP_OT_APPROVE_HISTORY	TRIGGER	
76.	T_DEPTIDINCREMENT	TRIGGER	
77.	T_FORM_ID_INCR	TRIGGER	
78.	T_GOVTHOLIID_INCR	TRIGGER	
79.	T_LEAVEAPPLYID_INCR	TRIGGER	
80.	T_LEAVEYPEID_INCR	TRIGGER	
81.	T_MACHINEID_INCR	TRIGGER	
82.	T_OSDID_INCR	TRIGGER	
83.	T_SEASONID_INCR	TRIGGER	
84.	T_SHIFTID_INCR	TRIGGER	

85.	T_SHIFTSETUPID_INCR	TRIGGER	
86.	T_USER_HISTORY	TRIGGER	
87.	T_USER_TYPE	TRIGGER	
88.	TBL_RAW_DATA_HISTORY	TRIGGER	
89.	T_UNRESTID_INCR	TRIGGER	
90.	UPDATE_DELETE_OCS	TRIGGER	
91.	SP_COMPANY_ADD_UPDATE	PROCEDURE	
92.	SP_DATA_PROCESS_BIMAN_CMIS	PROCEDURE	
93.	SP_EMPLOYEE_ADD_UPDATE_BIMAN	PROCEDURE	
94.	SP_FINANTIALYEAR_ADD_UPDATE	PROCEDURE	
95.	FOUR_SHIFT_AUTO_ENTRY	PROCEDURE	
96.	SP_FOUR_SHIFT_2014_CMIS	PROCEDURE	
97.	SP_FORM_ADD_UPDATE	PROCEDURE	
98.	SP_GENERAL_SHIFT	PROCEDURE	
99.	SP_GOVTHOLIDAY_ADD_UPDATE	PROCEDURE	
100.	SP_INSERT_ROWDATA	PROCEDURE	
101.	SP_LEAVE_APPLIED_ADD_UPDATE	PROCEDURE	
102.	SP_LEAVE_TYPE_ADD_UPDATE	PROCEDURE	
103.	SP_MANUAL_INSERT	PROCEDURE	
104.	SP_NIGHT_SHIFT_1	PROCEDURE	
105.	SP_NIGHT_SHIFT_2	PROCEDURE	
106.	SP_OSD_ADD_UPDATE	PROCEDURE	
107.	SP_OT_CALCULATION	PROCEDURE	
108.	SP_SHIFTNAME_ADD_UPDATE	PROCEDURE	
109.	SP_SHIFT_ADD_UPDATE	PROCEDURE	
110.	SP_SHORT_REST_CALCULATION	PROCEDURE	
111.	SP_SPECIAL_OT_CALCULATION	PROCEDURE	
112.	SP_SPECIAL_OT_GENERAL	PROCEDURE	
113.	SP_TWO_SHIFT_OT_TEST_TANIA	PROCEDURE	
114.	SP_USERCREATEINSERT	PROCEDURE	
115.	SP_USERTYPECREATE	PROCEDURE	
116.	SEQ_COMPFINYID	SEQUENCE	Used on TBL_FINANCIAL_YEAR at column COMPFINYID
117.	SEQ_COMPID	SEQUENCE	Used on TBL_COMPANY_INFO at column COMPID
118.	SEQ_DEPTID	SEQUENCE	Used on TBL_COMPANY_INFO at column COMPID
119.	SEQ_EMPID	SEQUENCE	Used on TBL_EMP_PERSONAL_INFO at column EMPID
120.	SEQ_FORM_ID	SEQUENCE	Used on TBL_FORM at column at column FORMID
121.	SEQ_GOVTHOLIID	SEQUENCE	Used on TBL_GOVTHOLIDAY_LIST at column GOVTHOLIID
122.	SEQ_LEAVEAPPLYID	SEQUENCE	Used on TBL_LEAVE_APPLY at column LEAVEAPPLYID
123.	SEQ_LEAVEYPEID	SEQUENCE	Used on TBL_LEAVE_TYPE at column LEAVEYPEID

124.	SEQ_OSDID	SEQUENCE	Used on TBL_OSD_SETUP at column OSDID
125.	SEQ_PLANT_ID	SEQUENCE	
126.	SEQ_SEASONID	SEQUENCE	Used on TBL_SEASON_SETUP at column SEASONID
127.	SEQ_SHIFTID	SEQUENCE	Used on TBL_SHIFTNAMESETTINGS at column SHIFTID
128.	SEQ_SHIFTSETUPID	SEQUENCE	Used on TBL_SHIFT_SETUP at column SHIFTSETUPID
129.	SEQ_USER_TYPE_ID	SEQUENCE	Used on TBL_USERTYPE at column USERTYPEID
130.	SEQ_USERID	SEQUENCE	Used on TBL_USER at column USERID
131.	SEQ_WEEKLYHOLIID	SEQUENCE	Used on TBL_WEEKLY_HOLIDAY at column WEEKLYHOLIID
132.	TBL_LEAVE_APPLY_INDEX	INDEX	Used on TBL_LEAVE_APPLY at Columns(LEAVEAPPLYID, EMPID, LEAVETYPEID, FROMDATE, TODATE)
133.	TBL_OSD_SETUP_INDEX	INDEX	Used on TBL_OSD_SETUP at Columns (DEPTID, PLANT_ID, OSDID, EMPID, OSDSTARTDATE, OSDENDDATE)
134.	TBL_OT_ENTRY_INDEX	INDEX	Used on TBL_OT_ENTRY at Columns (EMPID, OT_OR_REG_DATE, R_SHIFT_ID, R_IN_TIME_PUNCH, R_OUT_TIME_PUNCH, OT_IN_TIME_PUNCH, OT_OUT_TIME_PUNCH)
135.	TBL_PROCESSED_DATA_INDEX	INDEX	Used on TBL_PROCESSED_DATA at Columns(EMPID,PUNCHDATE, SHIFTID, TIMEIN, TIMEOUT)
136.	TBL_RAW_DATA_CMIS_INDEX	INDEX	Used on TBL_RAW_DATA_CMIS at Columns (COMPCARDID, PUNCHDATE, PUNCHTIME, ROW_ID)
137.	TBL_RAWDATA_INDEX	INDEX	Used on TBL_RAW_DATA at Columns (COMPCARDID, PUNCHDATE, PUNCHTIME)
138.	WEBUTIL_DB	PACKAGE	Used to download data from Machine text file to database.

## B) Used Constraint in "ATT\_IN" Schema

Name	Type	Table Name	Column Name	Reference
PK_TBL_OT_ENTRY	Primary Key	TBL_OT_ENTRY	EMPCODE,EMPID,OT_OR_REG_DATE	
TBL_BUSINESSDIVISION_PK	Primary Key	TBL_BUSINESSDIVISION	BD_ID	
TBL_COMPANY_INFO_PK	Primary Key	TBL_COMPANY_INFO	COMPID	
TBL_COSTCENTER_PK	Primary Key	TBL_COSTCENTER	COST_CENTER_ID	
TBL_DEPARTMENT_PK	Primary Key	TBL_DEPARTMENT	DEPTID	
TBL_DESIGNATION_SETUP_PK	Primary Key	TBL_DESIGNATION_SETUP	DESIGNATIONID	
TBL_EMP_OT_APPROVE_PK	Primary Key	TBL_EMP_OT_APPROVE	EMPID,OTDATE	
EMP_PERSONAL_INFO_PK	Primary Key	TBL_EMP_PERSONAL_INFO	EMPID	
TBL_FINANCIAL_YEAR_PK	Primary Key	TBL_FINANCIAL_YEAR	COMPFINYID	
TBL_GOVHOLIDAY_LIST_PK	Primary Key	TBL_GOVHOLIDAY_LIST	GOVTHOLIID	
TBL_GRADENAME_PK	Primary Key	TBL_GRADENAME	GREADID	
TBL_LABOUR_PK	Primary Key	TBL_LABOUR	LABOURCODE	
TBL_LEAVE_APPLY_PK	Primary Key	TBL_LEAVE_APPLY	LEAVEAPPLYID	
TBL_LEAVE_TYPE_PK	Primary Key	TBL_LEAVE_TYPE	LEAVETYPEID	
TBL_OSD_SETUP_PK	Primary Key	TBL_OSD_SETUP	OSDID	
TBL_OT_EDIT_PK	Primary Key	TBL_OT_EDIT	EMPID,PUNCHDATE	
TBL_PROCESSED_DATA_PK	Primary Key	TBL_PROCESSED_DATA	EMPID,PUNCHDATE	
TBL_RAW_DATA_PK	Primary Key	TBL_RAW_DATA	PUNCHDATE,COMPCARDID,PUNCHTIME	
TBL_SEASON_SETUP_PK	Primary Key	TBL_SEASON_SETUP	SEASONID	
TBL_SHIFT_SETUP_PK	Primary Key	TBL_SHIFT_SETUP	SHIFTSETUPID	
TBL_SHIFTNAMESSETTINGS_PK	Primary Key	TBL_SHIFTNAMESSETTINGS	SHIFTID, SHIFTTYPE	
TBL_SHIFTROLLING_SETUP_PK	Primary Key	TBL_SHIFTROLLING_SETUP	TID	
TBL_USER_PK	Primary Key	TBL_USER	USERID	
TBL_WORKCENTER_PK	Primary Key	TBL_WORKCENTER	WORKCENTER_ID	
TBL_BDID_FK	Foreign Key	TBL_EMP_PERSONAL_INFO	BD_ID	
TBL_DEPTID_FK	Foreign Key	TBL_EMP_PERSONAL_INFO	DEPTID	
LABOURCODE_FK	Foreign Key	TBL_EMP_PERSONAL_INFO	LABOURCODE	
TBL_PLANTFACTORY_FK	Foreign Key	TBL_EMP_PERSONAL_INFO	PLANT_ID	

## C) Role Name

Role Name: SUPERADMIN

```
GRANT EXECUTE ON ATT_IN.DATEDIFF TO SUPERADMIN;
GRANT EXECUTE ON ATT_IN.FN_LEAVECODE TO SUPERADMIN;
GRANT EXECUTE ON ATT_IN.FN_MEALALLOWANCE TO SUPERADMIN;
GRANT EXECUTE ON ATT_IN.FN_MINUTE_TO_TIME_SPECIAL_OT TO SUPERADMIN;
GRANT EXECUTE ON ATT_IN.FN_MINUTE_TO_TIME_TEXT TO SUPERADMIN;
GRANT EXECUTE ON ATT_IN.FN_MINUTE_TO_TIME_TEXT_REPORT TO SUPERADMIN;
GRANT EXECUTE ON ATT_IN.FN_OT_HOUR_14082013_TXTFILE TO SUPERADMIN;
GRANT EXECUTE ON ATT_IN.FN_PAYBLE_OT_HOUR_14082013 TO SUPERADMIN;
GRANT EXECUTE ON ATT_IN.FN_SHIFTDAYS TO SUPERADMIN;
GRANT EXECUTE ON ATT_IN.FN_SHORTREST_SUM TO SUPERADMIN;
GRANT EXECUTE ON ATT_IN.FN_TOOTHOURJOBCARD TO SUPERADMIN;
GRANT EXECUTE ON ATT_IN.FN_TOOTHOURJOBCARD2SHIFT TO SUPERADMIN;
GRANT EXECUTE ON ATT_IN.FN_TOOTHOURJOBCARD4SHIFT TO SUPERADMIN;
GRANT EXECUTE ON ATT_IN.FN_TOOTHOURJOBCARD_TEXT TO SUPERADMIN;
GRANT EXECUTE ON ATT_IN.FN_TOSHORTRESTJOBCARD4SHIFT TO SUPERADMIN;
GRANT EXECUTE ON ATT_IN.FN_TOTALLATEHOUR TO SUPERADMIN;
GRANT EXECUTE ON ATT_IN.FN_TOTALLATEHOUR_TEXT TO SUPERADMIN;
GRANT EXECUTE ON ATT_IN.FOUR_SHIFT_AUTO_ENTRY TO SUPERADMIN;
GRANT EXECUTE ON ATT_IN.LATE_CALCULATION TO SUPERADMIN;
GRANT EXECUTE ON ATT_IN.MAKEMINUTE_FROMVALUE TO SUPERADMIN;
GRANT EXECUTE ON ATT_IN.MAKETIMEFROMMINUTE TO SUPERADMIN;
GRANT SELECT ON ATT_IN.SEQ_COMPID TO SUPERADMIN;
GRANT ALTER, SELECT ON ATT_IN.SEQ_DEPTID TO SUPERADMIN;
GRANT ALTER, SELECT ON ATT_IN.SEQ_EMPID TO SUPERADMIN;
GRANT EXECUTE ON ATT_IN.SP_DATA_PROCESS_BIMAN TO SUPERADMIN;
GRANT EXECUTE ON ATT_IN.SP_DATA_PROCESS_BIMAN_02_02 TO SUPERADMIN;
GRANT EXECUTE ON ATT_IN.SP_DATA_PROCESS_BIMAN_CMIS TO SUPERADMIN;
GRANT EXECUTE ON ATT_IN.SP_EMPLOYEE_ADD_UPDATE_BIMAN TO SUPERADMIN;
GRANT EXECUTE ON ATT_IN.SP_FORM_ADD_UPDATE TO SUPERADMIN;
GRANT EXECUTE ON ATT_IN.SP_FOURSHIFT_LOG_ADD_UPDATE TO SUPERADMIN;
GRANT EXECUTE ON ATT_IN.SP_FOUR_SHIFT_18_07_13_OFF TO SUPERADMIN;
GRANT EXECUTE ON ATT_IN.SP_FOUR_SHIFT_2014 TO SUPERADMIN;
GRANT EXECUTE ON ATT_IN.SP_FOUR_SHIFT_2014_CMIS TO SUPERADMIN;
GRANT EXECUTE ON ATT_IN.SP_GENERAL_SHIFT TO SUPERADMIN;
GRANT EXECUTE ON ATT_IN.SP_GOVTHOLIDAY_ADD_UPDATE TO SUPERADMIN;
GRANT EXECUTE ON ATT_IN.SP_INSERT_ROWDATA TO SUPERADMIN;
GRANT EXECUTE ON ATT_IN.SP_LABOURINSERTUPDATE TO SUPERADMIN;
GRANT EXECUTE ON ATT_IN.SP_LEAVE_APPLIED_ADD_UPDATE TO SUPERADMIN;
GRANT EXECUTE ON ATT_IN.SP_LEAVE_TYPE_ADD_UPDATE TO SUPERADMIN;
GRANT EXECUTE ON ATT_IN.SP_MANUAL_INSERT TO SUPERADMIN;
GRANT EXECUTE ON ATT_IN.SP_NIGHT_SHIFT_1 TO SUPERADMIN;
GRANT EXECUTE ON ATT_IN.SP_NIGHT_SHIFT_2 TO SUPERADMIN;
GRANT EXECUTE ON ATT_IN.SP_NIGHT_SHIFT_TANIA TO SUPERADMIN;
GRANT EXECUTE ON ATT_IN.SP_OSD_ADD_UPDATE TO SUPERADMIN;
GRANT EXECUTE ON ATT_IN.SP_OT_APPROVE_ADD_UPDATE TO SUPERADMIN;
GRANT EXECUTE ON ATT_IN.SP_OT_APPROVE_ADD_UPDATE_NEW TO SUPERADMIN;
GRANT EXECUTE ON ATT_IN.SP_OT_CALCULATION TO SUPERADMIN;
GRANT EXECUTE ON ATT_IN.SP_OT_EDIT TO SUPERADMIN;
GRANT EXECUTE ON ATT_IN.SP_PROCESSNORMALNIGHT TO SUPERADMIN;
GRANT EXECUTE ON ATT_IN.SP_PROCESSSHIFTNIGHT TO SUPERADMIN;
GRANT EXECUTE ON ATT_IN.SP_SHIFTNAME ADD_UPDATE TO SUPERADMIN;
```

```

GRANT EXECUTE ON ATT_IN.SP_SHIFT_ADD_UPDATE TO SUPERADMIN;
GRANT EXECUTE ON ATT_IN.SP_SPECIAL_OT_CALCULATION TO SUPERADMIN;
GRANT EXECUTE ON ATT_IN.SP_SPECIAL_OT_GENERAL TO SUPERADMIN;
GRANT EXECUTE ON ATT_IN.SP_TWO_SHIFT_OT_TEST_TANIA TO SUPERADMIN;
GRANT EXECUTE ON ATT_IN.SP_USERCREATEINSERT TO SUPERADMIN;
GRANT EXECUTE ON ATT_IN.SP_USERTYPECREATE TO SUPERADMIN;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.TBL_4_SHIFT_ROSTER TO SUPERADMIN;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.TBL_BUSINESSDIVISION TO SUPERADMIN;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.TBL_CHECK TO SUPERADMIN;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.TBL_COMPANY_INFO TO SUPERADMIN;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.TBL_DEPARTMENT TO SUPERADMIN;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.TBL_EMP_OT_APPROVE TO SUPERADMIN;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.TBL_FINANCIAL_YEAR TO SUPERADMIN;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.TBL_FORM TO SUPERADMIN;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.TBL_GOVHOLIDAY_LIST TO SUPERADMIN;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.TBL_LABOUR TO SUPERADMIN;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.TBL_LEAVE_APPLY TO SUPERADMIN;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.TBL_LEAVE_TYPE TO SUPERADMIN;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.TBL_MACHINE_INFO TO SUPERADMIN;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.TBL_OSD_SETUP TO SUPERADMIN;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.TBL_OT_EDIT TO SUPERADMIN;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.TBL_OT_ENTRY TO SUPERADMIN;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.TBL_PLANTFACTORY TO SUPERADMIN;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.TBL_PROCESSED_DATA TO SUPERADMIN;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.TBL_RAW_DATA TO SUPERADMIN;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.TBL_RAW_DATA_CMIS TO SUPERADMIN;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.TBL_SEASON_SETUP TO SUPERADMIN;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.TBL_SHIFTNAMESETTINGS TO SUPERADMIN;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.TBL_SHIFT_SETUP TO SUPERADMIN;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.TBL_SHIFT_TYPE TO SUPERADMIN;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.TBL_SPECIAL_OT_DAYS TO SUPERADMIN;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.TBL_USER TO SUPERADMIN;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.TBL_USERTYPE TO SUPERADMIN;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.TBL_USER_ROLE_PERMISSION TO SUPERADMIN;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.TBL_WEEKLY_HOLIDAY TO SUPERADMIN;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.V_ATTANDANCE_REPORT TO SUPERADMIN;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.V_EMPLOYEE_ALL_INFO TO SUPERADMIN;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.V_EMP_OTLIST TO SUPERADMIN;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.V_EMP_REGULAR_STATUS TO SUPERADMIN;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.V_OT_EDIT TO SUPERADMIN;
GRANT CREATE SESSION TO SUPERADMIN;

```

## Role Name: ADMIN

```

GRANT EXECUTE ON ATT_IN.DATEDIFF TO ADMIN;
GRANT EXECUTE ON ATT_IN.FN_LEAVECODE TO ADMIN;
GRANT EXECUTE ON ATT_IN.FN_MEALALLOWANCE TO ADMIN;
GRANT EXECUTE ON ATT_IN.FN_MINUTE_TO_TIME_SPECIAL_OT TO ADMIN;
GRANT EXECUTE ON ATT_IN.FN_MINUTE_TO_TIME_TEXT TO ADMIN;
GRANT EXECUTE ON ATT_IN.FN_MINUTE_TO_TIME_TEXT_REPORT TO ADMIN;
GRANT EXECUTE ON ATT_IN.FN_OT_HOUR_14082013_TXTFILE TO ADMIN;
GRANT EXECUTE ON ATT_IN.FN_PAYBLE_OT_HOUR_14082013 TO ADMIN;
GRANT EXECUTE ON ATT_IN.FN_SHIFTDAYS TO ADMIN;
GRANT EXECUTE ON ATT_IN.FN_SHORTREST_SUM TO ADMIN;
GRANT EXECUTE ON ATT_IN.FN_TOOTHOURJOBCARD TO ADMIN;
GRANT EXECUTE ON ATT_IN.FN_TOOTHOURJOBCARD2SHIFT TO ADMIN;

```

```

GRANT EXECUTE ON ATT_IN.FN_TOOTHOURJOBCARD4SHIFT TO ADMIN;
GRANT EXECUTE ON ATT_IN.FN_TOSHORTRESTJOBCARD4SHIFT TO ADMIN;
GRANT EXECUTE ON ATT_IN.FN_TOTALLATEHOUR TO ADMIN;
GRANT EXECUTE ON ATT_IN.FOUR_SHIFT_AUTO_ENTRY TO ADMIN;
GRANT EXECUTE ON ATT_IN.LATE_CALCULATION TO ADMIN;
GRANT EXECUTE ON ATT_IN.MAKEMINUTE_FROMVALUE TO ADMIN;
GRANT EXECUTE ON ATT_IN.MAKETIMEFROMMINUTE TO ADMIN;
GRANT SELECT ON ATT_IN.SEQ_COMPID TO ADMIN;
GRANT SELECT ON ATT_IN.SEQ_DEPTID TO ADMIN;
GRANT SELECT ON ATT_IN.SEQ_EMPID TO ADMIN;
GRANT EXECUTE ON ATT_IN.SP_DATA_PROCESS_BIMAN TO ADMIN;
GRANT EXECUTE ON ATT_IN.SP_DATA_PROCESS_BIMAN_02_02 TO ADMIN;
GRANT EXECUTE ON ATT_IN.SP_DATA_PROCESS_BIMAN_CMIS TO ADMIN;
GRANT EXECUTE ON ATT_IN.SP_EMPLOYEE_ADD_UPDATE_BIMAN TO ADMIN;
GRANT EXECUTE ON ATT_IN.SP_FOURSHIFT_LOG_ADD_UPDATE TO ADMIN;
GRANT EXECUTE ON ATT_IN.SP_FOUR_SHIFT_18_07_13_OFF TO ADMIN;
GRANT EXECUTE ON ATT_IN.SP_FOUR_SHIFT_2014 TO ADMIN;
GRANT EXECUTE ON ATT_IN.SP_FOUR_SHIFT_2014_CMIS TO ADMIN;
GRANT EXECUTE ON ATT_IN.SP_GENERAL_SHIFT TO ADMIN;
GRANT EXECUTE ON ATT_IN.SP_INSERT_ROWDATA TO ADMIN;
GRANT EXECUTE ON ATT_IN.SP_LEAVE_APPLIED_ADD_UPDATE TO ADMIN;
GRANT EXECUTE ON ATT_IN.SP_MANUAL_INSERT TO ADMIN;
GRANT EXECUTE ON ATT_IN.SP_NIGHT_SHIFT_1 TO ADMIN;
GRANT EXECUTE ON ATT_IN.SP_NIGHT_SHIFT_2 TO ADMIN;
GRANT EXECUTE ON ATT_IN.SP_NIGHT_SHIFT_TANIA TO ADMIN;
GRANT EXECUTE ON ATT_IN.SP_OSD_ADD_UPDATE TO ADMIN;
GRANT EXECUTE ON ATT_IN.SP_OT_APPROVE_ADD_UPDATE_NEW TO ADMIN;
GRANT EXECUTE ON ATT_IN.SP_OT_CALCULATION TO ADMIN;
GRANT EXECUTE ON ATT_IN.SP_OT_EDIT TO ADMIN;
GRANT EXECUTE ON ATT_IN.SP_PROCESSNORMALNIGHT TO ADMIN;
GRANT EXECUTE ON ATT_IN.SP_PROCESSSHIFTNIGHT TO ADMIN;
GRANT EXECUTE ON ATT_IN.SP_SHIFT_ADD_UPDATE TO ADMIN;
GRANT EXECUTE ON ATT_IN.SP_SPECIAL_OT_CALCULATION TO ADMIN;
GRANT EXECUTE ON ATT_IN.SP_SPECIAL_OT_GENERAL TO ADMIN;
GRANT EXECUTE ON ATT_IN.SP_TWO_SHIFT_OT_TEST_TANIA TO ADMIN;
GRANT EXECUTE ON ATT_IN.SP_USERCREATEINSERT TO ADMIN;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.TBL_4_SHIFT_ROSTER TO ADMIN;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.TBL_BUSINESSDIVISION TO ADMIN;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.TBL_CHECK TO ADMIN;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.TBL_COMPANY_INFO TO ADMIN;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.TBL_DEPARTMENT TO ADMIN;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.TBL_EMP_OT_APPROVE TO ADMIN;
GRANT INSERT, SELECT, UPDATE ON ATT_IN.TBL_EMP_OT_APPROVE_HISTORY TO ADMIN;
GRANT INSERT, SELECT, UPDATE ON ATT_IN.TBL_EMP_PERSONAL_INFO_HISTORY TO ADMIN;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.TBL_FINANCIAL_YEAR TO ADMIN;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.TBL_FORM TO ADMIN;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.TBL_GOVHOLIDAY_LIST TO ADMIN;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.TBL_LABOUR TO ADMIN;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.TBL_LEAVE_APPLY TO ADMIN;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.TBL_LEAVE_APPLY_HISTORY TO ADMIN;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.TBL_LEAVE_TYPE TO ADMIN;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.TBL_OSD_SETUP TO ADMIN;
GRANT INSERT, SELECT, UPDATE ON ATT_IN.TBL_OSD_SETUP_HISTORY TO ADMIN;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.TBL_OT_EDIT TO ADMIN;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.TBL_OT_ENTRY TO ADMIN;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.TBL_PLANTFACTORY TO ADMIN;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.TBL_PROCESSED_DATA TO ADMIN;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.TBL_RAW_DATA TO ADMIN;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.TBL_RAW_DATA_CMIS TO ADMIN;

```



```

GRANT INSERT, SELECT, UPDATE ON ATT_IN.TBL_RAW_DATA_HISTORY TO ADMIN;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.TBL_SEASON_SETUP TO ADMIN;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.TBL_SHIFTNAMESETTINGS TO ADMIN;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.TBL_SHIFT_SETUP TO ADMIN;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.TBL_SHIFT_TYPE TO ADMIN;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.TBL_USER TO ADMIN;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.TBL_USERTYPE TO ADMIN;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.TBL_USER_ROLE_PERMISSION TO ADMIN;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.TBL_WEEKLY_HOLIDAY TO ADMIN;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.V_ATTANDANCE_REPORT TO ADMIN;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.V_EMPLOYEE_ALL_INFO TO ADMIN;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.V_EMP_OTLIST TO ADMIN;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.V_EMP_REGULAR_STATUS TO ADMIN;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.V_OT_EDIT TO ADMIN;
GRANT EXECUTE ON ATT_IN.WEBUTIL_DB TO ADMIN;

```

## Role Name: USERROLR

```

GRANT EXECUTE ON ATT_IN.DATEDIFF TO USERROLE;
GRANT EXECUTE ON ATT_IN.FN_LEAVECODE TO USERROLE;
GRANT EXECUTE ON ATT_IN.FN_MEALALLOWANCE TO USERROLE;
GRANT EXECUTE ON ATT_IN.FN_MINUTE_TO_TIME_SPECIAL_OT TO USERROLE;
GRANT EXECUTE ON ATT_IN.FN_MINUTE_TO_TIME_TEXT TO USERROLE;
GRANT EXECUTE ON ATT_IN.FN_MINUTE_TO_TIME_TEXT_REPORT TO USERROLE;
GRANT EXECUTE ON ATT_IN.FN_OT_HOUR_14082013_TXTFILE TO USERROLE;
GRANT EXECUTE ON ATT_IN.FN_PAYBLE_OT_HOUR_14082013 TO USERROLE;
GRANT EXECUTE ON ATT_IN.FN_SHIFTDAYS TO USERROLE;
GRANT EXECUTE ON ATT_IN.FN_SHORTREST_SUM TO USERROLE;
GRANT EXECUTE ON ATT_IN.FN_TOOTHOURJOBCARD TO USERROLE;
GRANT EXECUTE ON ATT_IN.FN_TOOTHOURJOBCARD2SHIFT TO USERROLE;
GRANT EXECUTE ON ATT_IN.FN_TOOTHOURJOBCARD4SHIFT TO USERROLE;
GRANT EXECUTE ON ATT_IN.FN_TOSHORTRESTJOBCARD4SHIFT TO USERROLE;
GRANT EXECUTE ON ATT_IN.FN_TOTALLATEHOUR TO USERROLE;
GRANT EXECUTE ON ATT_IN.FOUR_SHIFT_AUTO_ENTRY TO USERROLE;
GRANT EXECUTE ON ATT_IN.LATE_CALCULATION TO USERROLE;
GRANT EXECUTE ON ATT_IN.MAKEMINUTE_FROMVALUE TO USERROLE;
GRANT EXECUTE ON ATT_IN.MAKETIMEFROMMINUTE TO USERROLE;
GRANT SELECT ON ATT_IN.SEQ_COMPID TO USERROLE;
GRANT SELECT ON ATT_IN.SEQ_DEPTID TO USERROLE;
GRANT SELECT ON ATT_IN.SEQ_EMPID TO USERROLE;
GRANT EXECUTE ON ATT_IN.SP_DATA_PROCESS_BIMAN TO USERROLE;
GRANT EXECUTE ON ATT_IN.SP_DATA_PROCESS_BIMAN_02_02 TO USERROLE;
GRANT EXECUTE ON ATT_IN.SP_DATA_PROCESS_BIMAN_CMIS TO USERROLE;
GRANT EXECUTE ON ATT_IN.SP_EMPLOYEE_ADD_UPDATE_BIMAN TO USERROLE;
GRANT EXECUTE ON ATT_IN.SP_FOURSHIFT_LOG_ADD_UPDATE TO USERROLE;
GRANT EXECUTE ON ATT_IN.SP_FOUR_SHIFT_18_07_13_OFF TO USERROLE;
GRANT EXECUTE ON ATT_IN.SP_FOUR_SHIFT_2014 TO USERROLE;
GRANT EXECUTE ON ATT_IN.SP_FOUR_SHIFT_2014_CMIS TO USERROLE;
GRANT EXECUTE ON ATT_IN.SP_GENERAL_SHIFT TO USERROLE;

```

```

GRANT EXECUTE ON ATT_IN.SP_INSERT_ROWDATA TO USERROLE;
GRANT EXECUTE ON ATT_IN.SP_LEAVE_APPLIED_ADD_UPDATE TO USERROLE;
GRANT EXECUTE ON ATT_IN.SP_MANUAL_INSERT TO USERROLE;
GRANT EXECUTE ON ATT_IN.SP_NIGHT_SHIFT_1 TO USERROLE;
GRANT EXECUTE ON ATT_IN.SP_NIGHT_SHIFT_2 TO USERROLE;
GRANT EXECUTE ON ATT_IN.SP_NIGHT_SHIFT_TANIA TO USERROLE;
GRANT EXECUTE ON ATT_IN.SP_OSD_ADD_UPDATE TO USERROLE;
GRANT EXECUTE ON ATT_IN.SP_OT_CALCULATION TO USERROLE;
GRANT EXECUTE ON ATT_IN.SP_OT_EDIT TO USERROLE;
GRANT EXECUTE ON ATT_IN.SP_PROCESSNORMALNIGHT TO USERROLE;
GRANT EXECUTE ON ATT_IN.SP_PROCESSSHIFTNIGHT TO USERROLE;
GRANT EXECUTE ON ATT_IN.SP_SHIFT_ADD_UPDATE TO USERROLE;
GRANT EXECUTE ON ATT_IN.SP_SPECIAL_OT_CALCULATION TO USERROLE;
GRANT EXECUTE ON ATT_IN.SP_SPECIAL_OT_GENERAL TO USERROLE;
GRANT EXECUTE ON ATT_IN.SP_TWO_SHIFT_OT_TEST_TANIA TO USERROLE;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.TBL_4_SHIFT_ROSTER TO USERROLE;
GRANT INSERT, SELECT, UPDATE ON ATT_IN.TBL_BUSINESSDIVISION TO USERROLE;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.TBL_CHECK TO USERROLE;
GRANT INSERT, SELECT, UPDATE ON ATT_IN.TBL_COMPANY_INFO TO USERROLE;
GRANT INSERT, SELECT, UPDATE ON ATT_IN.TBL_DEPARTMENT TO USERROLE;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.TBL_EMP_OT_APPROVE TO USERROLE;
GRANT INSERT, SELECT, UPDATE ON ATT_IN.TBL_EMP_OT_APPROVE_HISTORY TO USERROLE;
GRANT INSERT, SELECT, UPDATE ON ATT_IN.TBL_EMP_PERSONAL_INFO_HISTORY TO USERROLE;
GRANT INSERT, SELECT, UPDATE ON ATT_IN.TBL_FINANCIAL_YEAR TO USERROLE;
GRANT INSERT, SELECT, UPDATE ON ATT_IN.TBL_FORM TO USERROLE;
GRANT INSERT, SELECT, UPDATE ON ATT_IN.TBL_GOVHOLIDAY_LIST TO USERROLE;
GRANT INSERT, SELECT, UPDATE ON ATT_IN.TBL_LABOUR TO USERROLE;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.TBL_LEAVE_APPLY TO USERROLE;
GRANT INSERT, SELECT, UPDATE ON ATT_IN.TBL_LEAVE_APPLY_HISTORY TO USERROLE;
GRANT INSERT, SELECT, UPDATE ON ATT_IN.TBL_LEAVE_TYPE TO USERROLE;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.TBL_OSD_SETUP TO USERROLE;
GRANT INSERT, SELECT, UPDATE ON ATT_IN.TBL_OSD_SETUP_HISTORY TO USERROLE;
GRANT SELECT ON ATT_IN.TBL_OT_EDIT TO USERROLE;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.TBL_OT_ENTRY TO USERROLE;
GRANT INSERT, SELECT, UPDATE ON ATT_IN.TBL_PLANTFACTORY TO USERROLE;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.TBL_PROCESSED_DATA TO USERROLE;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.TBL_RAW_DATA TO USERROLE;
GRANT DELETE, INSERT, SELECT, UPDATE ON ATT_IN.TBL_RAW_DATA_CMIS TO USERROLE;
GRANT INSERT, SELECT, UPDATE ON ATT_IN.TBL_RAW_DATA_HISTORY TO USERROLE;
GRANT INSERT, SELECT, UPDATE ON ATT_IN.TBL_SEASON_SETUP TO USERROLE;
GRANT INSERT, SELECT, UPDATE ON ATT_IN.TBL_SHIFTNAMESETTINGS TO USERROLE;
GRANT INSERT, SELECT, UPDATE ON ATT_IN.TBL_SHIFT_SETUP TO USERROLE;
GRANT INSERT, SELECT, UPDATE ON ATT_IN.TBL_SHIFT_TYPE TO USERROLE;
GRANT INSERT, SELECT, UPDATE ON ATT_IN.TBL_USER TO USERROLE;
GRANT INSERT, SELECT, UPDATE ON ATT_IN.TBL_USERTYPE TO USERROLE;
GRANT INSERT, SELECT, UPDATE ON ATT_IN.TBL_USER_ROLE_PERMISSION TO USERROLE;
GRANT INSERT, SELECT, UPDATE ON ATT_IN.TBL_WEEKLY_HOLIDAY TO USERROLE;
GRANT SELECT ON ATT_IN.V_ATTENDANCE_REPORT TO USERROLE;
GRANT SELECT ON ATT_IN.V_EMPLOYEE_ALL_INFO TO USERROLE;
GRANT SELECT ON ATT_IN.V_EMP_OTLIST TO USERROLE;
GRANT SELECT ON ATT_IN.V_EMP_REGULAR_STATUS TO USERROLE;
GRANT SELECT ON ATT_IN.V_OT_EDIT TO USERROLE;
GRANT EXECUTE ON ATT_IN.WEBUTIL DB TO USERROLE;

```

## OBJECT WISE DESCRIPTION WHICH ARE USED IN "ATT\_IN" SCHEMA

---

### A) TABLES:

#### 1. TABLE: **TBL\_4\_SHIFT\_ROSTER**

Name	Data type	Length	Null	P Key	Description
DAY_NO	INTEGER				
SHIFT_ID	INTEGER				
SHIFT_NAME	NVARCHAR	30			
REPORTING_SHIFT_NAME	NVARCHAR	30			
SHIFT_IN_TIME	DATE				
SHIFT_OUT_TIME	DATE				
SHIFT_NAME_VIEW	VARCHAR2	30			

#### 2. TABLE: **TBL\_BUSINESSDIVISION**

Name	Data type	Length	Null	P Key	Description
BD_ID	INTEGER			YES	
BD_NAME	VARCHAR2	100			
BD_DESCRIPTION	VARCHAR2	500			
COMPID	INTEGER				
CRREATEBY	VARCHAR2	100			
UPDATEBY	VARCHAR2	100			
CREATEDATE	DATE				
UPDATEDATE	DATE				

#### 3. TABLE: **TBL\_CHECK**

Name	Data type	Length	Null	P Key	Description
DESGINATION	VARCHAR2	50			
EMPCODE	VARCHAR2	50			
EMPID	INTERGER				
DEPTID	INTEGER				
DEPARTMENT_NAME	VARCHAR2	50			
PLANT_ID	INTEGER				
LABOURCODE	INTEGER				
LABOURCODE_NAME	VARCHAR2	50			
ENTRYBY	VARCHAR2	50			
ENTRYDATE	DATE				
UPDATEBY	VARCHAR2	50			
UPDATEDATE	DATE				
DELETEBY	VARCHAR2	50			
DELETEDATE	DATE				

CHECK_ID	INTEGER				
DES_YN	INTEGER				
EMP_YN	INTEGER				
DEP_YN	INTEGER				
PLANT_YN	INTEGER				
LABOUR_YN	INTEGER				
MAX_96_HOURS_OT	VARCHAR2	50			
MAX_64_HOURS_OT	VARCHAR2	50			
OT_HOUR_96_YN	INTEGER				
OT_HOUR_64_YN	INTEGER				

#### 4. TABLE: **TBL\_COMPANY\_INFO**

Name	Data type	Length	Null	P Key	Description
COMPID	INTEGER		NO	YES	
OMPFINYID	INTEGER		NO		
COMPCODE	VATCHAR2	50	NO		
COMPNAMEE	VARCHAR2	200	NO		
COMPNAMEB	VARCHAR2	200			
COMPADDE	VARCHAR2	200			
COMPADDB	VARCHAR2	200			
COMPPHONE	VARCHAR2	100			
COMPFAX	VARCHAR2	100			
COMPEMAIL	VARCHAR2	200			
COMPWEB	VARCHAR2	200			
CREATEBY	VARCHAR2	51			
CREATEDATE	DATE				
UPDATEBY	VARCHAR2	50			
UPDATEDATE	DATE				

#### 5. TABLE: **TBL\_DEPARTMENT**

Name	Data type	Length	Null	P Key	Description
DEPTID	INTEGER		NO	YES	
PLANT_ID	INTEGER				
DEPTCODE	VARCHAR2	50			
DEPTNAMEE	VARCHAR2	50			
DEPTNAMEB	VARCHAR2	50			
DEPTDESCRIPTION	VARCHAR2	100			
CREATEBY	VARCHAR2	50			
CREATEDATE	DATE				
UPDATEBY	VARCHAR2	50			
UPDATEDATE	DATE				
BD_ID	INTEGER				

#### 6. TABLE: **TBL\_DESIGNATION\_SETUP**

Name	Data type	Length	Null	P Key	Description
DESIGNATIONID	I NTEGER		NO	YES	
DESIGNAMEE	VARCHAR2	50			
GRADE	VARCHAR2	50			

PRIORITY	INTEGER				
CREATEBY	VARCHAR2	50			
CREATEDATE	DATE				
UPDATEBY	VARCHAR2	50			
UPDATEDATE	DATE				

#### 7. TABLE: **TBL\_EMP\_OT\_APPROVE**

Name	Data type	Length	Null	P Key	Description
EMPID	VARCHAR2	50		YES	
OTDATE	DATE				
OTAPPROVE	VARCHAR2	3			
CREATEDATE	DATE				
CREATEBY	VARCHAR2	20			
UPDATEBY	VARCHAR2	20			
UPDATEDATE	DATE				
DELETEBY	VARCHAR2	20			
DELETEDATE	DATE				
CHECK_YN	CARCHAR2	2			

#### 8. TABLE: **TBL\_EMP\_OT\_APPROVE\_HISTORY**

Name	Data type	Length	Null	P Key	Description
EMPID	VARCHAR2	50			
OTDATE	DATE				
OTAPPROVE	VARCHAR2	3			
CREATEDATE	DATE				
CREATEBY	VARCHAR2	20			
UPDATEBY	VARCHAR2	20			
UPDATEDATE	DATE				
DELETEBY	VARCHAR2	20			
DELETEDATE	DATE				
CHECK_YN	CARCHAR2	2			

#### 9. TABLE: **TBL\_EMP\_PERSONAL\_INFO**

Name	Data type	Length	Null	P Key	Description
EMPID	INTEGER		NO	YES	
EMPCODE	VARCHAR2	50			
COMPCARDID	VARCHAR2	9			
NATIONALIDCARD	VARCHAR2	80			
EMPNAMEE	VARCHAR2	100			
NICKNAME	VARCHAR2	50			
EMP_PHOTO	VARCHAR2	150			
JOININGDATE	DATE				
CONFIRMDATE	DATE				
HOLIDAY_APPLICABLE	INTEGER				
NIGHTSHIFT_APPLICABLE	INTEGER				
OVERTIME_APPLICABLE	INTEGER				
EMPLOYEE_CATEGORY	VARCHAR2	25			
EMPLOYEMENT_CATEGORY	INTEGER				
EMPLOYEE_TYPE	VARCHAR2	30			

COMPID	NUMBER	8			
BD_ID	INTEGER				
PLANT_ID	INTEGER				
DEPTID	INTEGER				
COST_CENTER_ID	INTEGER				
SECTID	INTEGER				
WORKCENTER_ID	INTEGER				
DESIGNATIONID	VARCHAR2	20			
SHIFTID	INTEGER				
FATHERSNAME	VARCHAR2	50			
MOTHERSNAME	VARCHAR2	50			
SPOUSENAME	VARCHAR2	50			
DATEOFBIRTH	DATE				
BLOODGROUP	INTEGER				
SEX	INTEGER				
RELIGION	INTEGER				
MARITALSTATUS	INTEGER				
NOMINEENAME	VARCHAR2	101			
RELATION_NOMINEE	VARCHAR2	30			
NOMINEE_PHOTO	VARCHAR2	150			
TELEPHONE	VARCHAR2	60			
FAX	VARCHAR2	52			
EMAIL	VARCHAR2	200			
CONTACTNUM1	VARCHAR2	50			
CONTACTNUM2	VARCHAR2	50			
EMERGENCYCONTACTNUM	VARCHAR2	60			
PRESENTADD	VARCHAR2	100			
PERMANENTADD	VARCHAR2	100			
CREATEBY	VARCHAR2	50			
CREATEDATE	DATE				
UPDATEBY	VARCHAR2	50			
UPDATEDATE	DATE				
EX_INT_1	INTEGER				
EX_INT_2	INTEGER				
EX_VARCHAR_1	VARCHAR2	100			
EX_VARCHAR_2	VARCHAR2	100			
EX_DATE_1	DATDE				
EX_DATE_2	DATE				
EMPENDISSTATUS	INTEGER				
QUITDATE	DATE				
GREADID	INTEGER				
LABOURCODE	INTEGER				

10. TABLE: **TBL\_EMP\_PERSONAL\_INFO\_HISTORY**

Name	Data type	Length	Null	P Key	Description
EMPID	INTEGER				
EMPCODE	VARCHAR2	50			
COMPCARDID	VARCHAR2	9			
NATIONALIDCARD	VARCHAR2	80			
EMPNAMEE	VARCHAR2	100			
NICKNAME	VARCHAR2	50			
EMP_PHOTO	VARCHAR2	150			
JOININGDATE	DATE				

CONFIRMDATE	DATE				
HOLIDAY_APPLICABLE	INTEGER				
NIGHTSHIFT_APPLICABLE	INTEGER				
OVERTIME_APPLICABLE	INTEGER				
EMPLOYEE_CATEGORY	VARCHAR2	25			
EMPLOYEMENT_CATEGORY	INTEGER				
EMPLOYEE_TYPE	VARCHAR2	30			
COMPID	NUMBER	8			
BD_ID	INTEGER				
PLANT_ID	INTEGER				
DEPTID	INTEGER				
COST_CENTER_ID	INTEGER				
SECTID	INTEGER				
WORKCENTER_ID	INTEGER				
DESIGNATIONID	VARCHAR2	20			
SHIFTID	INTEGER				
FATHERSNAME	VARCHAR2	50			
MOTHERSNAME	VARCHAR2	50			
SPOUSENAME	VARCHAR2	50			
DATEOFBIRTH	DATE				
BLOODGROUP	INTEGER				
SEX	INTEGER				
RELIGION	INTEGER				
MARITALSTATUS	INTEGER				
NOMINEENAME	VARCHAR2	101			
RELATION_NOMINEE	VARCHAR2	30			
NOMINEE_PHOTO	VARCHAR2	150			
TELEPHONE	VARCHAR2	60			
FAX	VARCHAR2	52			
EMAIL	VARCHAR2	200			
CONTACTNUM1	VARCHAR2	50			
CONTACTNUM2	VARCHAR2	50			
EMERGENCYCONTACTNUM	VARCHAR2	60			
PRESENTADD	VARCHAR2	100			
PERMANENTADD	VARCHAR2	100			
CREATEBY	VARCHAR2	50			
CREATEDATE	DATE				
UPDATEBY	VARCHAR2				
UPDATEDATE	DATE				
EX_INT_1	INTEGER				
EX_INT_2	INTEGER				
EX_VARCHAR_1	VARCHAR2	100			
EX_VARCHAR_2	VARCHAR2	100			
EX_DATE_1	DATE				
EX_DATE_2	DATE				
EMPENDISSTATUS	INTEGER				
QUITDATE	DATE				
GREADID	INTEGER				
LABOURCODE	INTEGER				
NEW_EMPCODE	VARCHAR2				
NEW_EMPID	INTEGER				
NEW_COMPCARDID	VARCHAR2	9			
NEW_NAME	VARCHAR2	100			

NEW_JOININGDATE	DATE				
NEW_DESIGNATIONID	VARCHAR2	20			
NEW_EMPLOYEE_TYPE	VARCHAR2	30			
NEW_EMPLOYEE_CATEGORY	VARCHAR2	25			
NEW_LABOURCODE	INTEGER				
NEW_BD_ID	INTEGER				
NEW_PLANT_ID	INTEGER				
NEW_DEPTID	INTEGER				
NEW_SHIFTID	INTEGER				
DELETE_DATE	DATE				
DELETE_USER	VARCHAR2	50			
UPDATE_DATE	DATE				
UPDATE_USER	VARCHAR2	50			

11.TABLE: **TBL\_FINANCIAL\_YEAR**

Name	Data type	Length	Null	P Key	Description
COMPFINYID	INTEGER		NO	YES	
COMPFINYNAME	VARCHAR2	30			
COMPFINSTARTDATE	DATE				
COMPFINENDDATE	DATE				
CREATEBY	VARCHAR2	50			
CREATEDATE	DATE				
UPDATEBY	VARCHAR2	50			
UPDATEDATE	DATE				

12.TABLE: **TBL\_FORM**

Name	Data type	Length	Null	P Key	Description
FORMID	INTEGER				
FORMCONSTANT	VARCHAR2	50			
FORMDESCRIPTION	VARCHAR2	50			
PARENTMENUITEM	VARCHAR2	50			
CHILDMENUITEM	VARCHAR2	50			
CREATEBY	VARCHAR2	50			
CREATEDATE	DATE				
UPDATEBY	VARCHAR2	50			
UPDATEDATE	DATE				

13.TABLE: **TBL\_GOVHOLIDAY\_LIST**

Name	Data type	Length	Null	P Key	Description
GOVTHOLIID	INTEGER		NO	YES	
COMPID	INTEGER				
COMPFINYID	INTEGER		NO		
GOVTHOLINAME	VARCHAR2	50			
GOVTHOLISTARTDATE	DATE				
GOVTHOLIENDDATE	DATE				
GOVTNUMBERDAYS	FLOAT	126			
GOVTREMARKS	VARCHAR2	250			
CREATEBY	VARCHAR2	50			



CREATEDATE	DATE				
UPDATEBY	VARCHAR2	50			
UPDATEDATE	DATE				
COMPANYHOLIDAY	INTEGER				

14. TABLE: **TBL\_LABOUR**

Name	Data type	Length	Null	P Key	Description
LABOURCODE	NUMBER	3	NO	YES	
NAME	VARCHAR2	20			
OTSTATUS	VARCHAR2	10			
OTTYPE	VARCHAR2	10			

15. TABLE: **TBL\_LEAVE\_APPLY**

Name	Data type	Length	Null	P Key	Description
LEAVEAPPLYID	INTEGER		NO	YES	
EMPID	INTEGER		NO		
LEAVETYPEID	INTEGER		NO		
FROMDATE	DATE		NO		
TODATE	DATE		NO		
CAUSE	VARCHAR2	100			
NUMBEROFDAYS	FLOAT	126			
APPROVEDBY	INTEGER				
APPROVAL	INTEGER				
CREATEBY	VARCHAR2	100			
CREATEDATE	DATE				
UPDATEBY	VARCHAR2	100			
UPDATEDATE	DATE				
COMPFINYID	VARCHAR2	100			
EMPCODE	VARCHAR2	50			
DEPTID	INTEGER				
PLANT_ID	INTEGER				

16. Table **TBL\_LEAVECODE**

Name	Data type	Length	Null	P Key	Description
ID	INTEGER				
EMPID	INTEGER				
FROMDATE	DATE				
TODATE	DATE				
STATUS	VARCHAR2	50			

17. TABLE: **TBL\_LEAVE\_APPLY\_HISTORY**

Name	Data type	Length	Null	P Key	Description
LEAVEAPPLYID	INTEGER		NO	YES	
EMPID	INTEGER		NO		
LEAVETYPEID	INTEGER		NO		
FROMDATE	DATE		NO		
TODATE	DATE		NO		

CAUSE	VARCHAR2	100			
NUMBEROFDAYS	FLOAT	126			
APPROVEDBY	INTEGER				
APPROVAL	INTEGER				
CREATEBY	VARCHAR2	100			
CREATEDATE	DATE				
UPDATEBY	VARCHAR2	100			
UPDATEDATE	DATE				
COMPFINYID	VARCHAR2	100			
EMPCODE	VARCHAR2	50			
DEPTID	INTEGER				
PLANT_ID	INTEGER				
DELETEBY	VARCHAR2	50			
DELETE_DATE	DATE				
LEAVEAPPLYID_N	INTEGER				
EMPID_N	INTEGER				
LEAVETYPEID_N	INTEGER				
FROMDATE_N	DATE				
TODATE_N	DATE				
CAUSE_N	VARCHAR2	100			
NUMBEROFDAYS_N	FLOAT	126			
APPROVEDBY_N	INTEGER				
APPROVAL_N	INTEGER				
CREATEBY_N	VARCHAR2	100			
CREATEDATE_N	DATE				
UPDATEBY_N	VARCHAR2	100			
UPDATEDATE_N	DATE				
COMPFINYID_N	VARCHAR2	100			
EMPCODE_N	VARCHAR2	100			
DEPTID_N	INTEGER				
PLANT_ID_N	INTEGER				
NEW_UPDATEDATE	DATE				
NEW_UPDATEUSER	VARCHAR2	100			

18.TABLE: **TBL\_LEAVE\_TYPE**

Name	Data type	Length	Null	P Key	Description
LEAVETYPEID	INTEGER				
COMPID	INTEGER				
LEAVETYPENAME	VARCHAR2	100			
REPORTINGNAME	VARCHAR2	100			
BALANCE	FLOAT	126			
CARRYFORWARDSTATUS	INTEGER				
FSPECIFIC	INTEGER				
INTERCHANGABLE	INTEGER				
ISPAID	INTEGER				
CREATEBY	VARCHAR2	30			
CREATEDATE	DATE				
UPDATEBY	VARCHAR2	30			
UPDATEDATE	DATE				

19.TABLE: **TBL\_MACHINE\_INFO**

Name	Data type	Length	Null	P Key	Description
ID	NUMBER	6		YES	
MA_GROUP	VARCHAR2	30			
DESCRIPTION	VARCHAR2	100			
MODEL	VARCHAR2	80			
PRODUCTKEY	VARCHAR2	60			
ACTIVATION	VARCHAR2	60			
SERIAL_NO	NUMBER	12			
CONNECTION	VARCHAR2	50			
BAUDRATE	NUMBER	10			
IP_ADDRESS	VARCHAR2	30			
PORT	NUMBER	10			
KEY	VARCHAR2	50			
DISABLED	VARCHAR2	15			
CREATEBY	VARCHAR2	20			
UPDATEDATE	DATE				
CREATEDATE	DATE				
UPDATEBY	VARCHAR2	20			

20.TABLE: **TBL\_OSD\_SETUP**

Name	Data type	Length	Null	P Key	Description
OSDID	INTEGER		NO	YES	
EMPID	INTEGER		NO		
OSDSTARTDATE	DATE				
OSDENDDATE	DATE				
OSDNUMDAYS	FLOAT	126			
OSDLOCATION	VARCHAR2	150			
OSDREASON	VARCHAR2	250			
CREATEBY	VARCHAR2	20			
CREATEDATE	DATE				
UPDATEBY	VARCHAR2	20			
UPDATEDATE	DATE				
DEPTID	INTEGER				
EMPCODE	VARCHAR2	50			
PLANT_ID	INTEGER				

21.TABLE: **TBL\_OSD\_SETUP\_HISTORY**

Name	Data type	Length	Null	P Key	Description
OSDID	INTEGER		NO	YES	
EMPID	INTEGER		NO		
OSDSTARTDATE	DATE				
OSDENDDATE	DATE				
OSDNUMDAYS	FLOAT	126			
OSDLOCATION	VARCHAR2	150			
OSDREASON	VARCHAR2	250			
CREATEBY	VARCHAR2	20			

CREATEDATE	DATE				
UPDATEBY	VARCHAR2	20			
UPDATEDATE	DATE				
DEPTID	INTEGER				
EMPCODE	VARCHAR2	50			
PLANT_ID	INTEGER				
DELETE_BY	VARCHAR2	50			
DELETE_DATE	DATE				

22. TABLE: **TBL\_OT\_EDIT**

Name	Data type	Length	Null	P Key	Description
EMPID	INTEGER				
PUNCHDATE	DATE				
OT_HOUR	VARCHAR2	20			
ACTUAL_OT_HOUR	VARCHAR2	20			
REMARKS	VARCHAR2	100			
ENTRY_BY	VARCHAR2	100			
ENTRYDATE	DATE				
UPDATE_BY	VARCHAR2	100			
UPDATEDATE	DATE				

23. TABLE: **TBL\_OT\_ENTRY**

Name	Data type	Length	Null	P Key	Description
EMPCODE	VARCHAR2	30		YES	
EMPID	INTEGER			YES	
OT_OR_REG_DATE	DATE			YES	
DUTY_PATERN	NVARCHAR2	15			
FIRST_DUTY	INTEGER				
OT_SHIFT	VARCHAR2	30			
OT_SHIFT_ID	INTEGER				
OT_SHIFT_IN_TIME	DATE				
OT_SHIFT_OUT_TIME	DATE				
OT_IN_TIME_PUNCH	DATE				
OT_OUT_TIME_PUNCH	DATE				
R_SHIFT	VARCHAR2	30			
R_SHIFT_ID	INTEGER				
R_SHIFT_IN	DATE				
R_SHIFT_OUT	DATE				
R_IN_TIME_PUNCH	DATE				
R_OUT_TIME_PUNCH	DATE				
OVER_NIGHT_MARK_OT	INTEGER				
OVER_NIGHT_MARK_REGULAR	INTEGER				
DAY_MARK_OT	INTEGER				
DAY_MARK_REGULAR	INTEGER				
OT_HOUR	VARCHAR2	30			
SHORT_REST_HOUR	VARCHAR2	30			
DEPTID	INTEGER				
COMPCARDID	VARCHAR2	9			
ENTRYBY	VARCHAR2	30			
ENTRYDATE	DATE				

UPDATEBY	VARCHAR2	30			
UPDATEDATE	DATE				
LATE_ON_REG_SHIFT	VARCHAR2	30			
LATE_ON_OT_SHIFT	VARCHAR2	30			
REG_STATUS	VARCHAR2	30			
OT_STATUS	VARCHAR2	30			
OT_APPROVE	INTEGER				
OT_APPROVE_DATE	DATE				
OT_APPROVED_BY	VARCHAR2	30			
WEEKLY_HOLIDAY	VARCHAR2	30			
LOGNO	INTEGER				
AFTER_OT_LOG	INTEGER				
OT_LOG	INTEGER				
EXTRA_OT_IN	DATE				
EXTRA_OT_OUT	DATE				
SHORT_REST_LOG_B	INTEGER				
SHORT_REST_LOG_A	INTEGER				
SHORT_REST_IN_MIN	NUMBER				
PLANT_ID	INTEGER				
R_SHIFT_VIEW	VARCHAR2	30			
SPECIAL_OT_HOUR	VARCHAR2	30			

24. TABLE: **TBL\_PLANTFACTORY**

Name	Data type	Length	Null	P Key	Description
PLANT_ID	INTEGER				
PLANT_NAME	VARCHAR2	100			
PLANT_DESCRIPTION	VARCHAR2	500			
BD_ID	INTEGER				
CREATEBY	VARCHAR2	100			
CREATEDATE	DATE				
UPDATEBY	VARCHAR2	100			
UPDATEDATE	DATE				

25. TABLE: **TBL\_PROCESSED\_DATA**

Name	Data type	Length	Null	P Key	Description
EMPID	INTEGER			YES	
PUNCHDATE	DATE			YES	
TIMEIN	DATE				
TIMEOUT	DATE				
SHIFTIN	DATE				
SHIFTOUT	DATE				
BREAKIN	DATE				
BREAKOUT	DATE				
BREAKLATE	DATE				
LATE	VATCHAR2	20			
SHIFTID	INTEGER				
REGHOUR	DATE				
OTHOURL	VARCHAR2	50			
STATUS	VARCHAR2	50			
NUMPUNCH	INTEGER				
NIGHTSHIFTDESC	VARCHAR2	40			

OTSHIFTDESC	VARCHAR2	40			
BUYERSHIFTIN	DATE				
BUYERSHIFTOUT	DATE				
BUYERTIME	DATE				
MEALALLOWANCE	VARCHAR2	2			
URGENTOT	DATE				
REMARKS	VARCHAR2	500			
PID	INTEGER				
EXTRAOT	VARCHAR2	50			

26. TABLE: **TBL\_RAW\_DATA**

Name	Data type	Length	Null	P Key	Description
COMPCARDID	VARCHAR2	50		YES	
PUNCHDATE	DATE			YES	
PUNCHTIME	DATE			YES	
LOC_ID	VARCHAR2	50			
INOUT	VARCHAR2	50			
OVNMARK	INTEGER				
REMARKS	VARCHAR2	100			
MANUALENTRYTAG	INTEGER				
CREATEBY	VARCHAR2	50			
UPDATEDEMPLOYEE	INTEGER				
ROW_ID	NUMBER				
MACHINENAME	VARCHAR2	100			
ENTRYDATE	DATE				

27. TABLE: **TBL\_RAW\_DATA\_CMIS**

Name	Data type	Length	Null	P Key	Description
COMPCARDID	VARCHAR2	50			
PUNCHDATE	DATE				
PUNCHTIME	DATE				
LOC_ID	VARCHAR2	50			
INOUT	VARCHAR2	50			
OVNMARK	INTEGER				
REMARKS	VARCHAR2	100			
MANUALENTRYTAG	INTEGER				
CREATEBY	VARCHAR2	50			
UPDATEDEMPLOYEE	INTEGER				
ROW_ID	NUMBER				
MACHINENAME	VARCHAR2	100			
ENTRYDATE	DATE				

28. TABLE: **TBL\_RAW\_DATA\_ERROR**

Name	Data type	Length	Null	P Key	Description
COMPCARDID	VARCHAR2	50			
PUNCHDATE	DATE				
PUNCHTIME	DATE				
LOC_ID	VARCHAR2	50			
INOUT	VARCHAR2	50			

OVNMARK	INTEGER				
REMARKS	VARCHAR2	100			
MANUAENTRYTAG	INTEGER				
CREATEBY	VARCHAR2	50			
UPDATEDEMPLOYEE	INTEGER				
ROW_ID	NUMBER				
MACHINENAME	VARCHAR2	100			
ENTRYDATE	DATE				

29. TABLE: **TBL\_RAW\_DATA\_HISTORY**

Name	Data type	Length	Null	P Key	Description
COMPCARDID	VARCHAR2	50			
PUNCHDATE	DATE				
PUNCHTIME	DATE				
LOC_ID	VARCHAR2	50			
INOUT	VARCHAR2	50			
OVNMARK	INTEGER				
REMARKS	VARCHAR2	100			
MANUAENTRYTAG	INTEGER				
CREATEBY	VARCHAR2	50			
UPDATEDEMPLOYEE	INTEGER				
ROW_ID	NUMBER				
MACHINENAME	VARCHAR2	100			
ENTRYDATE	DATE				
COMPCARDID_N	VARCHAR2	50			
PUNCHDATE_N	DATE				
PUNCHTIME_N	DATE				
LOC_ID_N	VARCHAR2	50			
OVNMARK_N	INTEGER				
REMARKS_N	VARCHAR2	100			
MANUAENTRYTAG_N	INTEGER				
MACHINENAME_N	VARCHAR2	100			
UPDATE_BY	VARCHAR2	50			
UPDATE_DATE	DATE				
DELETE_BY	VARCHAR2	50			
DELETE_DATE	DATE				

30. TABLE: **TBL\_SEASON\_SETUP**

Name	Data type	Length	Null	P Key	Description
SEASONID	INTEGER			YES	
COMPID	INTEGER				
SEASONTYPENAME	VARCHAR2	50			
SEASONSTARTDATE	DATE				
SEASONENDDATE	DATE				
CREATEBY	VARCHAR2	10			
CREATEDATE	DATE				
UPDATEBY	VARCHAR2	10			

UPDATEDATE	DATE				
------------	------	--	--	--	--

31. TABLE: **TBL\_SHIFT\_SETUP**

Name	Data type	Length	Null	P Key	Description
SHIFTSETUPID	INTEGER			YES	
SEASONID	INTEGER				
SECTID	INTEGER				
SHIFTID	INTEGER				
SHIFTNAME	VARCHAR2	50			
LOGINTIME	DATE				
LOGOUTTIME	DATE				
LATEBY	DATE				
LUNCHLOGIN	DATE				
LUNCHLOGOUT	DATE				
REGULARHOUR	DATE				
CREATEBY	VARCHAR2	50			
CREATEDATE	DATE				
UPDATEBY	VARCHAR2	50			
UPDATEDATE	DATE				
COMPID	INTEGER				
DEPTID	INTEGER				
PLANT_ID	INTEGER				

32. TABLE: **TBL\_SHIFT\_TYPE**

Name	Data type	Length	Null	P Key	Description
SHIFTTYPEID	NUMBER				
SHIFTYPENAME	VARCHAR2	30			

33. TABLE: **TBL\_SHIFTNAMESETTINGS**

Name	Data type	Length	Null	P Key	Description
SHIFTID	INTEGER			YES	
SHIFTNAME	VARCHAR2	100			
ISROLLING	VARCHAR2	50			
CREATEBY	VARCHAR2	50			
UPDATEBY	VARCHAR2	50			
CREATEDATE	DATE				
UPDATEDATE	DATE				
SHIFTTYPE	VATCHAR2	20		YES	
LOGINTIME	DATE				
LOGOUTTIME	DATE				

34. TABLE: **TBL\_SPECIAL\_OT\_DAYS**

Name	Data type	Length	Null	P Key	Description
ID	NUMBER			YES	
YEAR	VARCHAR2	30			
STARTDATE	DATE				
ENDDATE	DATE				
HOLIDAYNAME	VARCHAR2	50			



ENTRYBY	VARCHAR2	50			
ENTRYDATE	DATE				
UPDATEBY	VARCHAR2	50			
UPDATEDATE	DATE				

**35.TABLE: TBL\_USER**

<b>Name</b>	<b>Data type</b>	<b>Length</b>	<b>Null</b>	<b>P Key</b>	<b>Description</b>
USERID	INTEGER		NO	YES	
USERNAME	VAECHAR2	100			
USERTYPENAME	VARCHAR2	100			
USERLOGID	VARCHAR2	100			
USERLOGPASSWORD	VARCHar2	100			
ENTRYBY	VARCHAR2	100			
ENTRYDATE	DATE				
UPDATEBY	VARCHAR2	100			
UPDATEDATE	DATE				
ACTIVESTATUS	INTEGER				
INACTIVEDATE	DATE				
EMPID	INTEGER				
EMPCODE	INTEGER				
ROLEID	INTEGER				
USERCATEGORY	INTEGER				
DEPTID	INTEGER				
AUTH_DEPTID	INTEGER				
PLANT_ID	INTEGER				
ENTRY_AUTHORITY_GIVEN	VARCHAR2	1			
ENTRY_AUTHORITY_DATE	DATE				
WITHDRAWN	VARCHAR2	1			
WITHDRAWAL_DATE	DATE				
ROLE_NAME	VARCHAR2	20			
ROLE_GIVEN_USER	VARCHAR2	100			
ROLE_WITHDRAW_USER	VARCHAR2	100			

**36.TABLE: TBL\_USER\_HISTORY**

<b>Name</b>	<b>Data type</b>	<b>Length</b>	<b>Null</b>	<b>P Key</b>	<b>Description</b>
USERID	INTEGER				
USERNAME	VAECHAR2	100			
USERTYPENAME	VARCHAR2	100			
USERLOGID	VARCHAR2	100			
USERLOGPASSWORD	VARCHar2	100			
ENTRYBY	VARCHAR2	100			
ENTRYDATE	DATE				
UPDATEBY	VARCHAR2	100			
UPDATEDATE	DATE				
ACTIVESTATUS	INTEGER				
INACTIVEDATE	DATE				
EMPID	INTEGER				
EMPCODE	INTEGER				
ROLEID	INTEGER				
USERCATEGORY	INTEGER				
DEPTID	INTEGER				

AUTH_DEPTID	INTEGER				
PLANT_ID	INTEGER				
NEW_EMPID	INTEGER				
NEW_USERNAME	VARCHAR2	100			
NEW_USERLOGID	VARCHAR2	100			
NEW_USERLOGPASSWORD	VARCHAR2	100			
NEW_USERTYPENAME	VARCHAR2	100			
NEW_DEPTID	INTEGER				
NEW_AUTH_DEPTID	INTEGER				
NEW_PLANT_ID	INTEGER				
DELETEBY	VARCHAR2	100			
DELETE_DATE	DATE				
NEW_UPDATEBY	VARCHAR2	100			
NEW_UPDATEDATE	DATE				

37.TABLE: **TBL\_USER\_ROLE\_PERMISSION**

Name	Data type	Length	Null	P Key	Description
USERID	INTEGER				
USERNAME	VARCHAR2	50			
FORMID	INTEGER				
FORMCONSTANT	VARCHAR2	50			
FORMDESCRIPTION	VARCHAR2	50			
PARENTMENUITEM	VARCHAR2	50			
CHILDMENUITEM	VARCHAR2	50			
F	INTEGER				
R	INTEGER				
I	INTEGER				
U	INTEGER				
D	INTEGER				
CREATEBY	VARCHAR2	50			
CREATEDATE	DATE				
UPDATEBY	VARCHAR2	50			
UPDATEDATE	DATE				

38.TABLE: **TBL\_USERTYPE**

Name	Data type	Length	Null	P Key	Description
USERTYPEID	INTEGER				
USERTYPENAME	VARCHAR2	100			

39.TABLE: **TBL\_WEEK\_DAY**

Name	Data type	Length	Null	P Key	Description
DAY_ID	INTEGER				
DAYNAME	VARCHAR2	50			

40.TABLE: **TBL\_WEEKLY\_HOLIDAY**

Name	Data type	Length	Null	P Key	Description
WEEKLYHOLIID	INTEGER				
COMPID	INTEGER				
DEPTID	INTEGER				

SECTID	INTEGER				
SHIFTTYPENAME	VARCHAR2	50			
DAYNAME	VARCHAR2	50			
CREATEBY	VARCHAR2	50			
CREATEDATE	DATE				
UPDATEBY	VARCHAR2	50			
UPDATEDATE	DATE				

## B) VIEWS

-----

### 1. VIEW: V\_ATTANDANCE\_REPORT

Name	Data type	Length	Null	P Key	Description
COMPID	VARCHAR2	30			
COMPNAMEE	NUMBER	8			
COMPADDE	VARCHAR2	200			
BD_ID	VARCHAR2	200			
PLANT_ID	VARCHAR2	200			
DEPTID	VARCHAR2	200			
DEPTCODE	INTEGER				
DEPTNAMEE	VARCHAR2	50			
DESIGNAMEE	VARCHAR2	100			
SECTID	VARCHAR2	50			
SECCODE	INTEGGER				
SECNAMEE	VARCHAR2	50			
EMPID	INTEGER		NO		
EMPNAMEE	VARCHAR2	50			
EMPCODE	VARCHAR2	50	NO		
EMPLOYEE_CATEGORY	VARCHAR2	25			
EMPLOYEE_TYPE	VARCHAR2	30			
FATHERSNAME	VARCHAR2	50			
DATEOFBIRTH	DATE				
JOININGDATE	DATE				
QUITDATE	DATE				
PUNCHDATE	DATE				
TIMEIN	VARCHAR2	5			
TIMEOUT	VARCHAR2	5			
LATE	VARCHAR2	20			
OTHOOR	VARCHAR2	50			
STATUS	VARCHAR2	50			
NUMPUNCH	INTEGER				
REMARKS	VARCHAR2	500			
WORKINGSHIFT	VARCHAR2	100			
SHIFTNAME	VARCHAR2	100			
LABOURCODE	INTEGER				
LABOURNAME	VARCHAR2	20			
BD_NAME	VARCHAR2	100			
PLANT_NAME	VARCHAR2	100			
MEALALLOWANCE	VARCHAR2	2			
SHIFTID	INTEGER				
OTTYPE	VARCHAR2	10			
WORKINGSHIFTID	INTEGER				

## 2. VIEW: V\_EMP\_OTLIST

Name	Data type	Length	Null	P Key	Description
EMPID	INTEGER				
EMPCODE	VARCHAR2	50			
EMPNAMEE	VARCHAR2	100			
LABOURCODE	NUMBER	3			
OTSTATUS	VARCHAR2	10			

## 3. VIEW: V\_EMP\_REGULAR\_STATUS

Name	Data type	Length	Null	P Key	Description
COMPID	VARCHAR2	30			
COMPNAMEE	NUMBER	8			
COMPADDE	VARCHAR2	200			
DEPTID	VARCHAR2	200			
DEPTCODE	INTEGER				
DEPTNAMEE	VARCHAR2	50			
DESIGNAMEE	VARCHAR2	100			
SECTID	VARCHAR2	50			
SECCODE	INTEGGER				
SECNAMEE	VARCHAR2	50			
EMPID	INTEGER		NO		
EMPNAMEE	VARCHAR2	50			
EMPCODE	VARCHAR2	50	NO		
FATHERSNAME	VARCHAR2	50			
DATEOFBIRTH	DATE				
JOININGDATE	DATE				
QUITDATE	DATE				
PUNCHDATE	DATE				
TIMEIN	VARCHAR2	5			
TIMEOUT	VARCHAR2	5			
LATE	VARCHAR2	20			
OTHOURL	VARCHAR2	50			
STATUS	VARCHAR2	50			
NUMPUNCH	INTEGER				
REMARKS	VARCHAR2	500			
SHIFTNAME	VARCHAR2	100			
SHIFTID	INTEGER				
EMPLOYEE_CATEGORY	VARCHAR2	25			
EMPLOYEE_TYPE	VARCHAR2	30			
BD_ID	INTEGER				
BD_NAME	VARCHAR2	100			
PLANT_ID	INTEGER				
PLANT_NAME	VARCHAR2	100			
WORKINGSHIFT	VARCHAR2	100			
SHIFTIN	DATE				
SHIFTOUT	DATE				
WORKINGSHIFTID	INTEGER				
LABOURCODE	INTEGER				
LABOURNAME	VARCHAR2	20			
MEALALLOWANCE	VARCHAR2	2			
OTTYPE	VARCHAR2	10			
EXTRAOT	VARCHAR2	50			

#### 4. VIEW: **V\_EMPLOYEE\_ALL\_INFO**

Name	Data type	Length	Null	P Key	Description
EMPID	INTEGER		NO		
EMPCODE	VARCHAR2	50	NO		
COMPCARDID	VARCHAR2	9			
EMPNAMEE	VARCHAR2	100			
NICKNAME	VARCHAR2	50			
EMP_PHOTO	VARCHAR2	150			
JOININGDATE	DATE				
EMPLOYEE_CATEGORY	VARCHAR2	25			
EMPLOYEE_TYPE	VARCHAR2	30			
COMPID	NUMBER	8			
LABOURCODE	INTEGER				
COMPNAMEE	VARCHAR2	200			
COMPADDE	VARCHAR2	200			
BD_ID	INTEGER				
BD_NAME	VARCHAR2	100			
PLANT_ID	INTEGER				
PLANT_NAME	VARCHAR2	100			
DEPTID	INTEGER				
DEPTCODE	VARCHAR2	50			
DEPTNAMEE	VARCHAR2	50			
SECTID	INTEGER				
SECCODE	VARCHAR2	50			
SECNAMEE	VARCHAR2	100			
DESIGNAMEE	VARCHAR2	20			
SHIFTID	INTEGER				
SHIFTNAME	VARCHAR2	100			
FATHERSNAME	VARCHAR2	50			
MOTHERSNAME	VARCHAR2	50			
SPOUSENAME	VARCHAR2	50			
DATEOFBIRTH	DATE				
NOMINEENAME	VARCHAR2	101			
RELATION_NOMINEE	VARCHAR2	30			
NOMINEE_PHOTO	VARCHAR2	150			
TELEPHONE	VARCHAR2	60			
FAX	VARCHAR2	52			
EMAIL	VARCHAR2	200			
CONTACTNUM1	VARCHAR2	50			
CONTACTNUM2	VARCHAR2	50			
EMERGENCYCONTACTNUM	VARCHAR2	60			
PRESENTADD	VARCHAR2	100			
PERMANENTADD	VARCHAR2	100			
CREATEBY	VARCHAR2	50			
CREATEDATE	DATE				
UPDATEBY	VARCHAR2	50			
UPDATEDATE	DATE				
EMPENDISSTATUS	INTEGER				
QUITDATE	DATE				
AGE	VARCHAR2	4000			
SERVICELENGTH	VARCHAR2	4000			
LABOURNAME	VARCHAR2	20			
OTTYPE	VARCHAR2	10			

## 5. VIEW: **V\_OT\_EDIT**

Name	Data type	Length	Null	P Key	Description
PUNCHDATE	DATE				
OTHOURL	VARCHAR2	20			
EMPCODE	VARCHAR2	50			
ACTUAL_OT_HOUR	VARCHAR2	20			

## C) **FUNCTIONS:**

### 1. **DATEDIFF**

```
CREATE OR REPLACE FUNCTION ATT_IN.datediff( p_what in varchar2,
p_d1 in date,
p_d2 in date ) return number
as
l_result number;
begin
select (p_d2-p_d1) *
decode( upper(p_what),
'SS', 24*60*60, 'MI', 24*60, 'HH', 24, NULL )
into l_result from dual;

return l_result;
end;
```

### 2. **FN\_MINUTE\_TO\_TIME\_SHORTREST\_OT**

```
CREATE OR REPLACE function ATT_IN.fn_minute_to_time_shortrest_ot(p_time number) return varchar2 is
v_time number(9,2):=p_time;
v_text varchar2(50):=0;
v_text1 varchar2(50):=0;
v_frac number(5,2):=0;
v_frac2 varchar2(50);
v_frac1 varchar2(5);

begin
select lpad(trunc((v_time/60),0),2,0) into v_text from dual;
v_frac:=v_time mod 60;

if v_frac<=9 then
v_frac2:=to_char('0'||v_frac);
else
v_frac2:=to_char(v_frac);
end if;
if v_frac2 between '01' and '30' then
v_frac1:=v_frac2;
v_text:= nvl(v_text,'00')||':'||nvl(v_frac1,'00')||':'||'00';
```

```

return v_text;
else
v_text1:= nvl(lpad(round((v_time/60),0),2,0),'00')||':'||'00'||':'||'00';
return v_text1;
end if;
end;

```

### 3. FN\_MINUTE\_TO\_TIME\_SPECIAL\_OT

```

CREATE OR REPLACE function ATT_IN.fn_minute_to_time_special_ot(p_time number) return varchar2 is
v_time number(9,2):=p_time;
v_text varchar2(50):=0;
v_text1 varchar2(50):=0;
v_frac number(5,2):=0;
v_frac2 varchar2(50);
v_frac1 varchar2(5);

begin
select lpad(trunc((v_time/60),0),2,0) into v_text from dual;
v_frac:=v_time mod 60;
if v_frac<=9 then
v_frac2:=to_char('0'||v_frac);
else
v_frac2:=to_char(v_frac);
end if;

```

### 4. FN\_MINUTE\_TO\_TIME\_TEXT

```

CREATE OR REPLACE function ATT_IN.fn_minute_to_time_text (p_time number) return varchar2 is
v_time number:=p_time;
v_text varchar2(5);
v_frac number(5);
v_frac2 varchar2(50);
v_frac1 varchar2(5);
begin
select lpad(trunc((v_time/60),0),3,0) into v_text from dual;
v_frac:=v_time mod 60;
if v_frac<=9 then
v_frac2:=to_char('0'||v_frac);
else
v_frac2:=to_char(v_frac);
end if;
if v_frac2 between '01' and '59' then
v_frac1:=v_frac2;
v_text:= v_text||v_frac1;

else
select rpad(lpad(round((v_time/60),0),3,0),5,0) into v_text
from dual;
end if;
return v_text;
end;

```



## 5. FN\_MINUTE\_TO\_TIME\_TEXT\_REPORT

```
CREATE OR REPLACE function ATT_IN.fn_minute_to_time_text_report (p_time number) return varchar2 is
v_time number(9,2):=p_time;
v_text varchar2(50):=0;
v_text1 varchar2(50):=0;
v_frac number(5,2):=0;
v_frac2 varchar2(50);
v_frac1 varchar2(5);
begin

select lpad(trunc((v_time/60),0),3,0) into v_text from dual;
v_frac:=v_time mod 60;
if v_frac<=9 then
v_frac2:=to_char('0'||v_frac);
else
v_frac2:=to_char(v_frac);
end if;
if v_frac2 between '01' and '30' then
v_frac1:='50';
v_text:= v_text||':'||v_frac1||':'||'00';
return v_text;
else
v_text1:= lpad(round((v_time/60),0),3,0)||':'||'00'||':'||'00';
return v_text1;
end if;
end;
```

## 6. MAKEMINUTE\_FROMVALUE

```
CREATE OR REPLACE FUNCTION ATT_IN.MakeMinute_FromValue
(I_myString varchar2
)
return number
Is
I_deliminator varchar2(1):=':';
V_minute number :=0;
V_minute_temp number :=0;
V_hour number :=0;
target number :=0;
i number :=0;
this_delim number;
last_delim number;
delim varchar2(2);
str varchar2(500);
countDept number;
I_BDId number;
retVal number :=0;
Begin
str:=I_myString;
i := 1;
last_delim := 0;
target := splitter_count(str, I_deliminator);
```

```

while i <= target
loop
this_delim := instr(str, I_deliminator, 1, i);
I_BDId := substr(str, last_delim + 1, this_delim - last_delim -1);--this is filtered key
i := i + 1;
last_delim := this_delim;
if i =2
then
V_hour :=I_BDId;
elsif i =3
then
V_minute_temp := I_BDId;
end if;
end loop;
V_minute :=V_hour*60;
V_minute :=V_minute + V_minute_temp;
RETURN (TO_Number(V_minute)) ;
END;

```

## 7. MakeTimeFromMinute

```

CREATE OR REPLACE FUNCTION ATT_IN.MakeTimeFromMinute
(
I_myString int
)
return varchar2
Is
V_hour int:=0;
V_minute int :=0;
retVal varchar2(50) :=0;
Begin
V_hour :=trunc(I_myString/60);
V_minute :=I_myString mod 60;
retVal :=LPAD(TO_Char(V_hour),3,'0') || ':' || LPAD(V_minute,2,'0') || ':00';
RETURN (retVal);
END;

```

## 8. MAKETIMEFROMMINUTE\_FORLATE

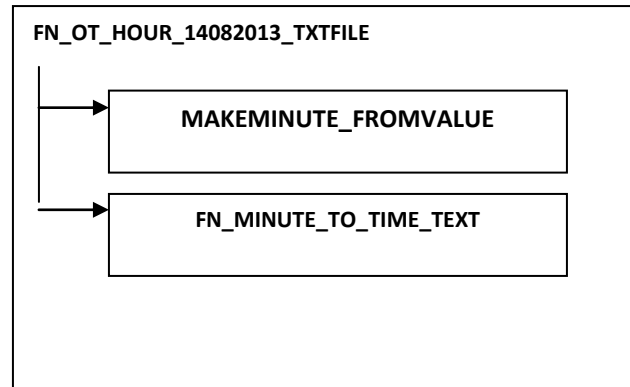
```

CREATE OR REPLACE FUNCTION ATT_IN.MakeTimeFromMinute_Forlate
(I_myString int
)
return varchar2
Is
V_hour int:=0;
V_minute int :=0;

retVal varchar2(50) :=0;
Begin
V_hour :=trunc(I_myString/60);
V_minute :=I_myString mod 60;
retVal :=LPAD(TO_Char(V_hour),2,'0') || ':' || LPAD(V_minute,2,'0') ;
RETURN (retVal);
END;

```

## 9. FUNCTION: FN\_OT\_HOUR\_14082013\_TXTFILE



```
CREATE OR REPLACE function ATT_IN.FN_OT_Hour_14082013_TxtFile
(I_from_Date date,
 I_to_Date date,
 I_Empid int,
 I_shiftid int
)

/* declare
 I fromDate Date:='02-JAN-12';
 I toDate date:='10-JUN-12';
 I_Empid int:=9096;
 I_shiftid int:=441;*/

return varchar2
as

--I_fromDate Date;
I fromDate1 Date;
I fromDate2 Date;
I fromDate3 Date;
I fromDate4 Date;
I toDate date;
workingHour int;
workingHour1 int;
workingHour2 int;
workingHour3 int;
workingHour4 int;

weeklyOTM float;
weeklyOTH float;
weeklyOTH1 float;
weeklyOTH2 float;
weeklyOTH3 float;
weeklyOTH4 float;
weeklyOTH_SHIFT float;
weeklyPaybleOT float:=0;
```

```

weeklyPaybleOT1 float:=0;
weeklyPaybleOT2 float:=0;
weeklyPaybleOT3 float:=0;
weeklyPaybleOT4 float:=0;
weeklyOT float;
weeklyOT1 float;
weeklyOT2 float;
weeklyOT3 float;
weeklyOT4 float;
v_short_rest_mt float;
weeklyOTM_SHIFT float;
weeklyOTH_SHIFT1 float;
weeklyOTH_SHIFT2 float;

weeklyOTdeduct number(9,2);
weeklyOTdeduct1 number(9,2);
weeklyOTdeduct2 number(9,2);
weeklyOTdeduct3 number(9,2);
weeklyOTdeduct4 number(9,2);

v_siftType varchar2(15);
v_employee_type varchar2(15);
v_employee_category varchar2(20);
totalday int;
firstWeeak int:=0;
secondWeeak int:=0;payableOT varchar2(10);
thirdweeak int:=0;forthWeeak int:=0;
fifthWeeak Int:=0;loopStart int:=1;
loopIndex int:=0; WeaklyWorkingDay int:=0;
v_count int;
--I Empid int;
totalOTH float;
totalOTH1 float;
v_shiftid int;
v_desgid varchar2(30);
v_desig_exist_96 int;
v_desig_exist_64 int;
v_labourcode int;
begin
    begin
        select EMPLOYEE_TYPE,EMPLOYEE_CATEGORY,SHIFTID,DESIGNATIONID,LABOURCODE
into v_employee_type,v_employee_category , v_shiftid,v_desgid,v_labourcode
        from att_in.TBL_EMP_PERSONAL_INFO where EMPID=I_Empid;
        EXCEPTION
        when no_data_found then
            null;
        end;

        begin
            select count(*) into v_desig_exist_96 from att_in.TBL_CHECK
            where MAX_96_HOURS_OT=v_labourcode;
            EXCEPTION
            when no_data_found then
                null;
            end;

```

```

begin
select count(*) into v_desig_exist_64 from att_in.TBL_CHECK
where MAX_64_HOURS_OT=v_labourcode;
EXCEPTION
when no_data_found then
null;
end;

select count(*) into v_count from att_in.TBL_OT_ENTRY where empid =I_Empid and
OT_OR_REG_DATE between I_from_Date and I_to_Date ;
Dbms_output.Put_Line('v_desig_exist_96');
Dbms_output.Put_Line(v_desig_exist_96);
Dbms_output.Put_Line(v_desig_exist_64);
Dbms_output.Put_Line(v_count);

if v_desig_exist_96=0 and v_desig_exist_64 =0 then

If v_employee_type in('P','G') and v_employee_category='Permanent' then

Select SUM(MakeMinute_FromValue(OT_HOUR)*2)
into weeklyOTH_SHIFT1
from att_in.TBL_OT_ENTRY where OT_OR_REG_DATE
between TO_Date(I_From_Date,'DD-MM-RR') and TO_Date(I_to_Date,'DD-MM-RR')
and EmpId=I_EmpId;

if weeklyOTH_SHIFT1>15000 then
weeklyOTH_SHIFT1:=15000;
else
weeklyOTH_SHIFT1:=weeklyOTH_SHIFT1;
end if;

Select SUM(MakeMinute_FromValue(SHORT_REST_HOUR)*2)
into weeklyOTH_SHIFT2
from att_in.TBL_OT_ENTRY where OT_OR_REG_DATE
between TO_Date(I_From_Date,'DD-MM-RR') and TO_Date(I_to_Date,'DD-MM-
RR')

and EmpId=I_EmpId;

totalOTH1:= nvl(weeklyOTH_SHIFT1,0)+nvl(weeklyOTH_SHIFT2,0);

/* if totalOTH>19320 then
totalOTH:=19320;
else
totalOTH:= totalOTH;

end if;*/
--return MakeTimeFromMinute(V_ReturnOT);

```

```

        select shifttype into v_siftType from att_in.TBL_SHIFTNAMESETTINGS where
shiftid=I_shiftid;

begin
    select EMPLOYEE_TYPE,EMPLOYEE_CATEGORY into v_employee_type,v_employee_category
    from att_in.TBL_EMP_PERSONAL_INFO where EMPID=I_Empid;
    exception
    when no_data_found then
    null;
end;
--SELECT TO_CHAR(TRUNC(to_date('31-may-2013'), 'MM'), 'DD-MON-RR') into
I_fromDate FROM DUAL;
--SELECT TO_DATE(TRUNC(to_char(I_toDate), 'MM'), 'DD-MON-RRRR') into I_fromDate
FROM DUAL;

--SELECT TRUNC (I_to_Date, 'month') into I_fromDate FROM DUAL;
--totalday:=to_date('31-dec-12')-to_date(I_fromDate,'DD-MON-RR')+1;
    select substr(last_day(to_date(I_from_Date,'DD-MM-RRRR')),1,2) into totalday
from dual;

Select SUM(MakeMinute_FromValue(OT HOUR))
into weeklyOT from V_EMP_REGULAR_STATUS where PunchDate
between TO_Date(I_from_Date,'DD-MM-RRRR') and
TO_Date(I_from_Date + 6,'DD-MM-RRRR')
and EmpId=I_Empid ;
weeklyOTH:=(weeklyOT/60);

-----1st week
Select NVL(SUM(datediff('hh',SHIFTIN,SHIFTOUT)),0) * 60,nvl(count(status),0) into
workingHour ,WeaklyWorkingDay
from V_EMP_REGULAR_STATUS where PunchDate
between TO_Date(I_from_Date,'DD-MM-RRRR') and TO_Date(I_from_Date+6,'DD-MM-RRRR')
and EmpId=I_Empid and status not in ('W') ;

dbms_output.put_line('workingHour');
dbms_output.put_line(workingHour);

--dbms_output.put_line('WeaklyWorkingDay');
--dbms_output.put_line(WeaklyWorkingDay);

--dbms_output.put_line(TO Date(I fromDate,'DD-MM-RRRR'));
--dbms_output.put_line(TO Date(I fromDate+6,'DD-MM-RRRR'));

if workingHour > 2400 then
    workingHour:=2400;
else
    workingHour:=workingHour;
    --dbms_output.put_line('Working Hour Greater then 48');
end if;

If weeklyOT>480 and v_employee_type in('P','G') and
v_employee_category='Permanent' then
    if (v_siftType='General Shift') then
        weeklyOTdeduct:=48*60- workingHour;

```

```

        weeklyPaybleOT:= weeklyOT * 2 - weeklyOTdeduct;

        elsif weeklyPaybleOT<weeklyOT and (v_siftType='General Shift') then
            weeklyPaybleOT:=weeklyOT;
        else -- this condition is only for 4 Shift and 2 Shift
            weeklyPaybleOT := weeklyOT * 2;
        end if;
    else
        weeklyPaybleOT := weeklyOT;
    end if;

    --if loopStart=1 then
        firstWeeak:= firstWeeak+weeklyPaybleOT;
        Dbms_output.Put_Line('firstWeeak');
        Dbms_output.Put_Line(weeklyPaybleOT);
        Dbms_output.Put_Line(weeklyOT);
        Dbms_output.Put_Line(weeklyOTdeduct);

    -----End of 1st week

    -----2nd week-----

    I_fromDate1:=I_from_Date + 7;
    weeklyOT1:=0;
    Select SUM(MakeMinute_FromValue(OTHOURL))
    into weeklyOT1
    from V_EMP_REGULAR_STATUS
    where PunchDate
    between TO_Date(I_fromDate1,'DD-MM-RRRR') and
    TO_Date(I_fromDate1+6,'DD-MM-RRRR')
    and EmpId=I_Empid;
    weeklyOTH1:=(weeklyOT1/60);

    Select NVL(SUM(datediff('hh',SHIFTIN,SHIFTOUT)),0) * 60,nvl(count(status),0) into
workingHour1 ,WeaklyWorkingDay
    from V_EMP_REGULAR_STATUS where PunchDate
    between TO_Date(I_fromDate1,'DD-MM-RRRR') and TO_Date(I_fromDate1+6,'DD-MM-RRRR')
    and EmpId=I_Empid and status not in ('W');

    if workingHour1 > 2400 then
        workingHour1:=2400;
    else
        workingHour1:=workingHour1;
        --dbms_output.put_line('Working Hour Greater then 48');
    end if;

    If v_employee_type in('P','G') and v_employee_category='Permanent' then
        if weeklyOT1>480 and (v_siftType='General Shift' )then
            weeklyOTdeduct1:=48*60- workingHour1;
            weeklyPaybleOT1:= weeklyOT1 * 2 - weeklyOTdeduct1;

```

```

        elsif weeklyPaybleOT1<weeklyOT1 and (v_siftType='General Shift') then
            weeklyPaybleOT1:=weeklyOT1;
        else
            weeklyPaybleOT1:= weeklyOT1 * 2;
        end if;
    else
        weeklyPaybleOT1:= weeklyOT1;
    end if;

```

```

secondWeeak:= secondWeeak+weeklyPaybleOT1;

```

```

        dbms_output.put_line('workingHour1');
        dbms_output.put_line(workingHour1);
        Dbms_output.Put_Line(' secondWeeak');
        Dbms_output.Put_Line(weeklyOT1);
        Dbms_output.Put_Line(weeklyOTdeduct1);

```

```

-----3rd week-----
I_fromDate2:=I_fromDate1+7;

```

```

Select  SUM(MakeMinute_FromValue(OTHOURL))
into weeklyOT2 from V_EMP_REGULAR_STATUS where PunchDate
between TO_Date(I_fromDate2,'DD-MM-RRRR') and
TO_Date(I_fromDate2+6,'DD-MM-RRRR')
and EmpId=I_Empid ;
weeklyOTH2:=(weeklyOT2/60);

```

```

Select  NVL(SUM(datediff('hh',SHIFTIN,SHIFTOUT)),0) * 60,nvl(count(status),0) into
workingHour2 ,WeeklyWorkingDay
from V_EMP_REGULAR_STATUS where PunchDate
between TO_Date(I_fromDate2,'DD-MM-RRRR') and TO_Date(I_fromDate2+6,'DD-MM-RRRR')
and EmpId=I_Empid and status not in ('W') ;

```

```

        --dbms_output.put_line('workingHour2 in Minute on 20/08/2013');
        --dbms_output.put_line(workingHour2);

```

```

if workingHour2 > 2400 then
    workingHour2:=2400;
else
    workingHour2:=workingHour2;
    --dbms_output.put_line('Working Hour Greater then 48');
end if;

```

```

If v_employee_type in('P','G') and v_employee_category='Permanent' then
    if weeklyOT2>480 and (v_siftType='General Shift')then
        weeklyOTdeduct2:=48*60- workingHour2;

```



```

        weeklyPaybleOT2:= weeklyOT2*2 - weeklyOTdeduct2;

        elsif weeklyPaybleOT2<weeklyOT2 and (v_siftType='General Shift') then
            weeklyPaybleOT2:= weeklyOT2;
        else
            weeklyPaybleOT2:= weeklyOT2 * 2;
        end if;
    else
        weeklyPaybleOT2:= weeklyOT2;
    end if;

    thirdweeak:= thirdweeak+weeklyPaybleOT2;

    dbms_output.put_line('workingHour2');
    dbms_output.put_line(workingHour2);
    Dbms_output.Put_Line('thirdweeak');
    Dbms_output.Put_Line(weeklyOT2);
    Dbms_output.Put_Line(weeklyOTdeduct2);

    -----4th week-----
    I_fromDate3:=I_fromDate2+7;

    Select SUM(MakeMinute_FromValue(OTHOURL))
    into weeklyOT3 from V_EMP_REGULAR_STATUS where PunchDate
    between TO_Date(I_fromDate3,'DD-MM-RRRR') and
    TO_Date(I_fromDate3+6,'DD-MM-RRRR')
    and EmpId=I_Empid ;
    weeklyOTH3:=(weeklyOT3/60);

    Select NVL(SUM(datediff('hh',SHIFTIN,SHIFTOUT)),0) * 60,nvl(count(status),0) into
    workingHour3 ,WeeklyWorkingDay
    from V_EMP_REGULAR_STATUS where PunchDate
    between TO_Date(I_fromDate3,'DD-MM-RRRR') and TO_Date(I_fromDate3+6,'DD-MM-RRRR')
    and EmpId=I_Empid and status not in ('W') ;

    if workingHour3 > 2400 then
        workingHour3:=2400;
    else
        workingHour3:=workingHour3;
        --dbms_output.put_line('Working Hour Greater then 48');
    end if;

    If weeklyOT3>480 and v_employee_type in('P','G') and
    v_employee_category='Permanent' then
        if (v_siftType='General Shift')then
            weeklyOTdeduct3:=48*60- workingHour3;
            weeklyPaybleOT3 := weeklyOT3*2 - weeklyOTdeduct3;
        elsif weeklyPaybleOT3 <weeklyOT3 and (v_siftType='General Shift') then
            weeklyPaybleOT3 :=weeklyOT3;
        else
            weeklyPaybleOT3 := weeklyOT3 * 2;
        end if;
    else
        weeklyPaybleOT3 := weeklyOT3;
    end if;

```

```

        forthWeeak:= forthWeeak+weeklyPaybleOT3 ;
        dbms_output.put_line('workingHour3');
        dbms_output.put_line(workingHour3);
        Dbms_output.Put_Line('forthWeeak');
        Dbms_output.Put_Line(weeklyOT3);
        Dbms_output.Put_Line(weeklyOTdeduct3);

-----end of 4th week-----

-----5th week-----

I_fromDate4:=I_fromDate3+7;

Select  SUM(MakeMinute_FromValue(OTHOURL))
into weeklyOT4 from V_EMP_REGULAR_STATUS where PunchDate
between TO_Date(I_fromDate4 , 'DD-MM-RRRR') and
TO_Date(I_to_Date, 'DD-MM-RRRR')
and EmpId=I_Empid ;
weeklyOTH4:=(weeklyOTH4/60);

Select  NVL(SUM(datediff('hh',SHIFTIN,SHIFTOUT)),0) * 60,nvl(count(status),0) into
workingHour4 ,WeaklyWorkingDay
from V_EMP_REGULAR_STATUS where PunchDate
between TO_Date(I_fromDate4 , 'DD-MM-RRRR') and TO_Date(I_to_Date, 'DD-MM-RRRR')
and EmpId=I_Empid and status not in ('W') ;

if workingHour4 > 2400 then
    workingHour4:=2400;
else
    workingHour4:=workingHour4;
    --dbms_output.put_line('Working Hour Greater then 48');
end if;

If v_employee_type in('P','G') and v_employee_category='Permanent' then
    if(weeklyOT4>=240 and weeklyOT4<=720 and (v_siftType='General Shift' ))then
        --weeklyOTdeduct:=48*60- workingHour*60;
        weeklyPaybleOT4:=  weeklyOT4 * 2 - 240;

        elsif weeklyPaybleOT4<weeklyOT4 and (v_siftType='General Shift' ) then
            weeklyPaybleOT4:=weeklyOT4;
        else
            weeklyPaybleOT4:=  weeklyOT4*2;
        end if;
    else
        weeklyPaybleOT4:=  weeklyOT4;
    end if;

    fifthWeeak:= fifthWeeak+weeklyPaybleOT4;
    Dbms_output.Put_Line( firstWeeak);
    Dbms_output.Put_Line( secondWeeak);
    Dbms_output.Put_Line( thirdweeak);
    Dbms_output.Put_Line( forthWeeak);

```

```

Dbms_output.Put_Line( fifthWeeak);

    totalOTH :=nvl(firstWeeak,0) + nvl(secondWeeak,0) + nvl(thirdweeak,0) +
nvl(forthWeeak,0) + nvl(fifthWeeak,0)+nvl(totalOTH1,0);

    if v_shiftid=602 and v_employee_type in('P','G') and
v_employee_category='Permanent' then

        Select nvl( SUM(MakeMinute_FromValue(EXTRAOT)),0)
        into v_short_rest_mt from V_EMP_REGULAR STATUS where PunchDate
        between TO_Date(I_from_Date , 'DD-MM-RRRR') and
        TO_Date(I_to_Date, 'DD-MM-RRRR')
        and EmpId=I_Empid ;

        totalOTH:=totalOTH+v_short_rest_mt;
    else
        totalOTH:=totalOTH;

    end if;

if totalOTH>19320 then
    totalOTH:=19320;
else
    totalOTH:= totalOTH;

end if;

else

        Select SUM(MakeMinute_FromValue(OT_HOUR))
        into totalOTH1
        from att_in.TBL_OT_ENTRY where OT_OR_REG_DATE
        between TO_Date(I_From_Date, 'DD-MM-RR') and
TO_Date(I_to_Date, 'DD-MM-RR')
        and EmpId=I_EmpId;

        select shifttype into v_siftType from att_in.TBL_SHIFTNAMESETTINGS where
shiftid=I_shiftid;
    begin
        select EMPLOYEE_TYPE,EMPLOYEE_CATEGORY into v_employee_type,v_employee_category
        from att_in.TBL_EMP_PERSONAL_INFO where EMPID=I_Empid;
    exception
    when no_data_found then
        null;
    end;
    --SELECT TO_CHAR(TRUNC(to_date('31-may-2013'), 'MM'), 'DD-MON-RR') into
I_fromDate FROM DUAL;
    --SELECT TO_DATE(TRUNC(to_char(I_toDate), 'MM'), 'DD-MON-RRRR') into I_fromDate
FROM DUAL;

    --SELECT TRUNC (I_to_Date, 'month') into I_fromDate FROM DUAL;
    --totalday:=to_date('31-dec-12')-to_date(I_fromDate, 'DD-MON-RR')+1;
    select substr(last_day(to_date(I_from_Date, 'DD-MM-RRRR')),1,2) into totalday
from dual;

```

```

Select  SUM(MakeMinute_FromValue(OTHOURL))
into weeklyOT from V_EMP_REGULAR_STATUS where PunchDate
between TO_Date(I_from_Date,'DD-MM-RRRR') and
TO_Date(I_from_Date + 6,'DD-MM-RRRR')
and EmpId=I_Empid ;
weeklyOTH:=(weeklyOT/60);

-----1st week
Select  NVL(SUM(datediff('hh',SHIFTIN,SHIFTOUT)),0) * 60,nvl(count(status),0) into
workingHour ,WeaklyWorkingDay
from V_EMP_REGULAR_STATUS where PunchDate
between TO_Date(I_from_Date,'DD-MM-RRRR') and TO_Date(I_from_Date+6,'DD-MM-RRRR')
and EmpId=I_Empid and status not in ('W') ;

dbms_output.put_line('workingHour');
dbms_output.put_line(workingHour);

--dbms_output.put_line('WeaklyWorkingDay');
--dbms_output.put_line(WeaklyWorkingDay);

--dbms_output.put_line(TO_Date(I_fromDate,'DD-MM-RRRR'));
--dbms_output.put_line(TO_Date(I_fromDate+6,'DD-MM-RRRR'));

if workingHour > 2400 then
    workingHour:=2400;
else
    workingHour:=workingHour;
    --dbms_output.put_line('Working Hour Greater then 48');
end if;

If weeklyOT>480 and v_employee_type in('P','G') and
v_employee_category='Permanent' then
    if (v_siftType='General Shift') then
        weeklyOTdeduct:=48*60- workingHour;
        weeklyPaybleOT:= weeklyOT * 2 - weeklyOTdeduct;

        elsif weeklyPaybleOT<weeklyOT and (v_siftType='General Shift') then
            weeklyPaybleOT:=weeklyOT;
        else -- this condition is only for 4 Shift and 2 Shift
            weeklyPaybleOT := weeklyOT * 2;
        end if;
    else
        weeklyPaybleOT := weeklyOT;
    end if;

    --if loopStart=1 then
        firstWeeak:= firstWeeak+weeklyPaybleOT;
        Dbms_output.Put_Line('firstWeeak');
        Dbms_output.Put_Line(weeklyPaybleOT);
        Dbms_output.Put_Line(weeklyOT);
        Dbms_output.Put_Line(weeklyOTdeduct);

-----End of 1st week

```

```

-----2nd week-----

I_fromDate1:=I_from_Date + 7;
weeklyOT1:=0;
Select SUM(MakeMinute_FromValue(OTHOURL))
into weeklyOT1
from V_EMP_REGULAR_STATUS
where PunchDate
between TO_Date(I_fromDate1,'DD-MM-RRRR') and
TO_Date(I_fromDate1+6,'DD-MM-RRRR')
and EmpId=I_Empid;
weeklyOTH1:=(weeklyOT1/60);

Select NVL(SUM(datediff('hh',SHIFTIN,SHIFTOUT)),0) * 60,nvl(count(status),0) into
workingHour1 ,WeeklyWorkingDay
from V_EMP_REGULAR_STATUS where PunchDate
between TO_Date(I_fromDate1,'DD-MM-RRRR') and TO_Date(I_fromDate1+6,'DD-MM-RRRR')
and EmpId=I_Empid and status not in ('W');

if workingHour1 > 2400 then
workingHour1:=2400;
else
workingHour1:=workingHour1;
--dbms_output.put_line('Working Hour Greater then 48');
end if;

If v_employee_type in('P','G') and v_employee_category='Permanent' then
if weeklyOT1>480 and (v_siftType='General Shift' )then
weeklyOTdeduct1:=48*60- workingHour1;
weeklyPaybleOT1:= weeklyOT1 * 2 - weeklyOTdeduct1;

elseif weeklyPaybleOT1<weeklyOT1 and (v_siftType='General Shift') then
weeklyPaybleOT1:=weeklyOT1;
else
weeklyPaybleOT1:= weeklyOT1 * 2;
end if;
else
weeklyPaybleOT1:= weeklyOT1;
end if;

secondWeeak:= secondWeeak+weeklyPaybleOT1;

dbms_output.put_line('workingHour1');
dbms_output.put_line(workingHour1);
Dbms_output.Put_Line(' secondWeeak');
Dbms_output.Put_Line(weeklyOT1);
Dbms_output.Put_Line(weeklyOTdeduct1);

-----3rd week-----
I_fromDate2:=I_fromDate1+7;

```

```

Select  SUM(MakeMinute_FromValue(OTHOURL))
into weeklyOT2 from V_EMP_REGULAR_STATUS where PunchDate
between TO_Date(I_fromDate2,'DD-MM-RRRR') and
TO_Date(I_fromDate2+6,'DD-MM-RRRR')
and EmpId=I_Empid ;
weeklyOTH2:=(weeklyOT2/60);

```

```

Select  NVL(SUM(datediff('hh',SHIFTIN,SHIFTOUT)),0) * 60,nvl(count(status),0) into
workingHour2 ,WeaklyWorkingDay
from V_EMP_REGULAR_STATUS where PunchDate
between TO_Date(I_fromDate2,'DD-MM-RRRR') and TO_Date(I_fromDate2+6,'DD-MM-RRRR')
and EmpId=I_Empid and status not in ('W') ;

```

```

--dbms_output.put_line('workingHour2 in Minute on 20/08/2013');
--dbms_output.put_line(workingHour2);

```

```

if workingHour2 > 2400 then
    workingHour2:=2400;
else
    workingHour2:=workingHour2;
--dbms_output.put_line('Working Hour Greater then 48');
end if;

```

```

If v_employee_type in('P','G') and v_employee_category='Permanent' then
    if weeklyOT2>480 and (v_siftType='General Shift')then
        weeklyOTdeduct2:=48*60- workingHour2;
        weeklyPaybleOT2:= weeklyOT2*2 - weeklyOTdeduct2;

        elsif weeklyPaybleOT2<weeklyOT2 and (v_siftType='General Shift') then
            weeklyPaybleOT2:= weeklyOT2;
        else
            weeklyPaybleOT2:= weeklyOT2 * 2;
        end if;
    else
        weeklyPaybleOT2:= weeklyOT2;
    end if;

```

```

thirdweeak:= thirdweeak+weeklyPaybleOT2;

```

```

    dbms_output.put_line('workingHour2');
    dbms_output.put_line(workingHour2);
    Dbms_output.Put_Line('thirdweeak');
    Dbms_output.Put_Line(weeklyOT2);
    Dbms_output.Put_Line(weeklyOTdeduct2);

```

```

-----4th week-----
I_fromDate3:=I_fromDate2+7;

```

```

Select  SUM(MakeMinute_FromValue(OTHOURL))
into weeklyOT3 from V_EMP_REGULAR_STATUS where PunchDate
between TO_Date(I_fromDate3,'DD-MM-RRRR') and
TO_Date(I_fromDate3+6,'DD-MM-RRRR')
and EmpId=I_Empid ;
weeklyOTH3:=(weeklyOT3/60);

Select  NVL(SUM(datediff('hh',SHIFTIN,SHIFTOUT)),0) * 60,nvl(count(status),0) into
workingHour3 ,WeaklyWorkingDay
from V_EMP_REGULAR_STATUS where PunchDate
between TO_Date(I_fromDate3,'DD-MM-RRRR') and TO_Date(I_fromDate3+6,'DD-MM-RRRR')
and EmpId=I_Empid and status not in ('W') ;

if workingHour3 > 2400 then
    workingHour3:=2400;
else
    workingHour3:=workingHour3;
    --dbms_output.put_line('Working Hour Greater then 48');
end if;

If weeklyOT3>480 and v_employee_type in('P','G') and
v_employee_category='Permanent' then
    if (v_siftType='General Shift')then
        weeklyOTdeduct3:=48*60- workingHour3;
        weeklyPaybleOT3 := weeklyOT3*2 - weeklyOTdeduct3;
    elsif weeklyPaybleOT3 <weeklyOT3 and (v_siftType='General Shift') then
        weeklyPaybleOT3 :=weeklyOT3;
    else
        weeklyPaybleOT3 := weeklyOT3 * 2;
    end if;
    else
        weeklyPaybleOT3 := weeklyOT3;
    end if;

    forthWeeak:= forthWeeak+weeklyPaybleOT3 ;
    dbms_output.put_line('workingHour3');
    dbms_output.put_line(workingHour3);
    Dbms_output.Put_Line('forthWeeak');
    Dbms_output.Put_Line(weeklyOT3);
    Dbms_output.Put_Line(weeklyOTdeduct3);

    -----end of 4th week-----

    -----5th week-----

    I_fromDate4:=I_fromDate3+7;

Select  SUM(MakeMinute_FromValue(OTHOURL))
into weeklyOT4 from V_EMP_REGULAR_STATUS where PunchDate
between TO_Date(I_fromDate4 , 'DD-MM-RRRR') and
TO_Date(I_to_Date,'DD-MM-RRRR')
and EmpId=I_Empid ;
weeklyOTH4:=(weeklyOTH4/60);

Select  NVL(SUM(datediff('hh',SHIFTIN,SHIFTOUT)),0) * 60,nvl(count(status),0) into

```

```

workingHour4 ,WeaklyWorkingDay
from V_EMP_REGULAR_STATUS where PunchDate
between TO_Date(I_fromDate4 , 'DD-MM-RRRR') and TO_Date(I_to_Date, 'DD-MM-RRRR')
and EmpId=I_Empid and status not in ('W') ;

if workingHour4 > 2400 then
    workingHour4:=2400;
else
    workingHour4:=workingHour4;
    --dbms_output.put_line('Working Hour Greater then 48');
end if;

If v_employee_type in('P','G') and v_employee_category='Permanent' then
    if(weeklyOT4>=240 and weeklyOT4<=720 and (v_siftType='General Shift' ))then
        --weeklyOTdeduct:=48*60- workingHour*60;
        weeklyPaybleOT4:= weeklyOT4 * 2 - 240;

        elsif weeklyPaybleOT4<weeklyOT4 and (v_siftType='General Shift' ) then
            weeklyPaybleOT4:=weeklyOT4;
        else
            weeklyPaybleOT4:= weeklyOT4*2;
        end if;
    else
        weeklyPaybleOT4:= weeklyOT4;
    end if;

    fifthWeeak:= fifthWeeak+weeklyPaybleOT4;
    Dbms_output.Put_Line( firstWeeak);
    Dbms_output.Put_Line( secondWeeak);
    Dbms_output.Put_Line( thirdweeak);
    Dbms_output.Put_Line( forthWeeak);
    Dbms_output.Put_Line( fifthWeeak);

    totalOTH :=nvl(firstWeeak,0) + nvl(secondWeeak,0) + nvl(thirdweeak,0) +
nvl(forthWeeak,0) + nvl(fifthWeeak,0)+nvl(totalOTH1,0);

    if v_shiftid=602 and v_employee_type in('P','G') and
v_employee_category='Permanent' then

        Select nvl( SUM(MakeMinute_FromValue(EXTRAOT)),0)
        into v_short_rest_mt from V_EMP_REGULAR_STATUS where PunchDate
        between TO_Date(I_from_Date , 'DD-MM-RRRR') and
        TO_Date(I_to_Date, 'DD-MM-RRRR')
        and EmpId=I_Empid ;

        totalOTH:=totalOTH+v_short_rest_mt;
    else
        totalOTH:=totalOTH;
    end if;

    if totalOTH>7500 then
        totalOTH:=7500;
    end if;

```



```

        else
            totalOTH:=totalOTH;
        end if;

end if;

payableOT:=att_in.fn_minute_to_time_text_payable(totalOTH);

--payableOT:=att_in.FN_MINUTE_TO_TIME_TEXT_REPORT(totalOTH1);

        --Dbms_output.Put_Line('v_desig_exist_96');
        Dbms_output.Put_Line( totalOTH1);
        Dbms_output.Put_Line( payableOT);

elsif    v_desig_exist_96=1 then

    --if v_employee_type in('P','G') and v_employee_category='Permanent' then

        Dbms_output.Put_Line('v_desig_exist_96');
        Dbms_output.Put_Line(v_desig_exist_96);
        Dbms_output.Put_Line(v_employee_type);
        Dbms_output.Put_Line(v_employee_category);
        Dbms_output.Put_Line(v_shiftid);

        dbms_output.put_line(v_count);

        if v_count>0 then

            Select    SUM(MakeMinute_FromValue(OT_HOUR))
            into weeklyOTH_SHIFT1
            from att_in.TBL_OT_ENTRY where OT_OR_REG_DATE
            between TO_Date(I_From_Date,'DD-MM-RR') and
TO_Date(I_to_Date,'DD-MM-RR')
            and EmpId=I_EmpId;

        else

            Select    SUM(MakeMinute_FromValue(OTHOURL))
            into weeklyOTH_SHIFT1
            from V_EMP_REGULAR_STATUS where PunchDate
            between TO_Date(I_From_Date,'DD-MM-RR') and
TO_Date(I_to_Date,'DD-MM-RR')
            and EmpId=I_EmpId;
        end if;

        if weeklyOTH_SHIFT1>5760 then
            totalOTH1:=5760;
        else
            totalOTH1:=weeklyOTH_SHIFT1;
        end if;

        payableOT:=att_in.fn_minute_to_time_text_payable(totalOTH1);

```

```

        Dbms_output.Put_Line( totalOTH1);
        Dbms_output.Put_Line( payableOT);

        --end if;

    elsif    v_desig_exist_64=1 then
        Dbms_output.Put_Line('v_desig_exist_64');
        Dbms_output.Put_Line(v_desig_exist_64);

        --if v_employee_type in('P','G') and v_employee_category='Permanent'
then
        if v_count>0 then

            Select    SUM(MakeMinute_FromValue(OT_HOUR))
            into weeklyOTH_SHIFT1
            from att_in.TBL_OT_ENTRY where OT_OR_REG_DATE
            between TO_Date(I_From_Date,'DD-MM-RR') and
TO_Date(I_to_Date,'DD-MM-RR')
            and EmpId=I_EmpId;
            else
            Select    SUM(MakeMinute_FromValue(OTHOURL))
            into weeklyOTH_SHIFT1
            from V_EMP_REGULAR_STATUS where PunchDate
            between TO_Date(I_From_Date,'DD-MM-RR') and
TO_Date(I_to_Date,'DD-MM-RR')
            and EmpId=I_EmpId;

            end if;

            if weeklyOTH_SHIFT1>3840 then
                totalOTH1:=3840;
            else
                totalOTH1:=weeklyOTH_SHIFT1;
            end if;

            payableOT:=att_in.fn_minute_to_time_text_payable(totalOTH1);

            Dbms_output.Put_Line( totalOTH1);
            Dbms_output.Put_Line( payableOT);

        --end if;
        --end if;

        /*elsif v_employee_type in('P','G') and v_employee_category='Permanent'and
v_shiftid =602 then

            Select    SUM(MakeMinute_FromValue(OTHOURL)*2)
            into weeklyOTH_SHIFT1

```

```

from att_in.V_EMP_REGULAR_STATUS where PunchDate
between TO_Date(I_From_Date,'DD-MM-RR') and TO_Date(I_to_Date,'DD-MM-RR')
and EmpId=I_EmpId;

totalOTH1:= nvl(weeklyOTH_SHIFT1,0);

if totalOTH1>19320 then
totalOTH1:=19320;
else
totalOTH1:= totalOTH1;

end if;

payableOT:=att_in.FN_MINUTE_TO_TIME_TEXT_REPORT(totalOTH1);

Dbms_output.Put_Line('a');
Dbms_output.Put_Line(weeklyOTH_SHIFT1);
Dbms_output.Put_Line(totalOTH1);
Dbms_output.Put_Line(payableOT);*/

--return payableOT;

end if;
--end;

return payableOT;
end;
/

```

## 10. FN\_PAYBLE\_OT\_HOUR\_14082013

```

CREATE OR REPLACE function ATT_IN.FN_Payble_OT_Hour_14082013
(I_from_Date date,
I_to_Date date,
I_Empid int,
I_shiftid int
)

/* declare
I_fromDate Date:='02-JAN-12';
I_toDate date:='10-JUN-12';
I_Empid int:=9096;
I_shiftid int:=441;*/

return varchar2
as

```

```

--I_fromDate Date;
I_fromDate1 Date;
I_fromDate2 Date;
I_fromDate3 Date;
I_fromDate4 Date;
I_toDate date;
workingHour int;
workingHour1 int;
workingHour2 int;
workingHour3 int;
workingHour4 int;

weeklyOTM float;
weeklyOTH float;
weeklyOTH1 float;
weeklyOTH2 float;
weeklyOTH3 float;
weeklyOTH4 float;
weeklyOTH_SHIFT float;
weeklyPaybleOT float:=0;
weeklyPaybleOT1 float:=0;
weeklyPaybleOT2 float:=0;
weeklyPaybleOT3 float:=0;
weeklyPaybleOT4 float:=0;
weeklyOT float;
weeklyOT1 float;
weeklyOT2 float;
weeklyOT3 float;
weeklyOT4 float;
v_short_rest_mt float;
weeklyOTM_SHIFT float;
weeklyOTH_SHIFT1 float;
weeklyOTH_SHIFT2 float;

weeklyOTdeduct number(9,2);
weeklyOTdeduct1 number(9,2);
weeklyOTdeduct2 number(9,2);
weeklyOTdeduct3 number(9,2);
weeklyOTdeduct4 number(9,2);

v_siftType varchar2(15);
v_employee_type varchar2(15);
v_employee_category varchar2(20);
totalday int;
firstWeeak int:=0;
secondWeeak int:=0;payableOT varchar2(10);
thirdweeak int:=0;forthWeeak int:=0;
fifthWeeak Int:=0;loopStart int:=1;
loopIndex int:=0; WeaklyWorkingDay int:=0;
v_count int;
--I_Empid int;
totalOTH float;
totalOTH1 float;
v_shiftid int;
v_desgid varchar2(30);
v_desig_exist_96 int;
v_desig_exist_64 int;
v_labourcode int;

```

```

begin

    begin
        select
EMPLOYEE_TYPE,EMPLOYEE_CATEGORY,SHIFTID,DESIGNATIONID,LABOURCODE into
v_employee_type,v_employee_category , v_shiftid,v_desgid,v_labourcode
        from att_in.TBL_EMP_PERSONAL_INFO where EMPID=I_Empid;
        EXCEPTION
        when no_data_found then
        null;
        end;

        begin
        select count(*) into v_desig_exist_96 from att_in.TBL_CHECK
        where MAX_96_HOURS_OT=v_labourcode;
        EXCEPTION
        when no_data_found then
        null;
        end;

        begin
        select count(*) into v_desig_exist_64 from att_in.TBL_CHECK
        where MAX_64_HOURS_OT=v_labourcode;
        EXCEPTION
        when no_data_found then
        null;
        end;

        select  count(*) into v_count from att_in.TBL_OT_ENTRY where empid =I_Empid
and OT_OR_REG_DATE between I_from_Date and I_to_Date ;
        Dbms_output.Put_Line('v_desig_exist_96');
        Dbms_output.Put_Line(v_desig_exist_96);
        Dbms_output.Put_Line(v_desig_exist_64);
        Dbms_output.Put_Line(v_count);

        if  v_desig_exist_96=0 and v_desig_exist_64 =0 then

            If v_employee_type in('P','G') and v_employee_category='Permanent'    then

                Select  SUM(MakeMinute_FromValue(OT_HOUR)*2)
                into weeklyOTH_SHIFT1
                from att_in.TBL_OT_ENTRY where OT_OR_REG_DATE
                between TO_Date(I_From_Date,'DD-MM-RR') and TO_Date(I_to_Date,'DD-
MM-RR')
                and EmpId=I_EmpId;

                if weeklyOTH_SHIFT1>15000 then
                    weeklyOTH_SHIFT1:=15000;
                else
                    weeklyOTH_SHIFT1:=weeklyOTH_SHIFT1;
            end if;
        end if;
    end;
end;

```

```

end if;

Select SUM(MakeMinute_FromValue(SHORT_REST_HOUR)*2)
into weeklyOTH_SHIFT2
from att_in.TBL_OT_ENTRY where OT_OR_REG_DATE
between TO_Date(I_From_Date,'DD-MM-RR') and
TO_Date(I_to_Date,'DD-MM-RR')
and EmpId=I_EmpId;

totalOTH1:= nvl(weeklyOTH_SHIFT1,0)+nvl(weeklyOTH_SHIFT2,0);

/* if totalOTH>19320 then
totalOTH:=19320;
else
totalOTH:= totalOTH;

end if;*/
--return MakeTimeFromMinute(V_ReturnOT);

select shifttype into v_siftType from att_in.TBL_SHIFTNAMESETTINGS where
shiftid=I_shiftid;

begin
select EMPLOYEE_TYPE,EMPLOYEE_CATEGORY into
v_employee_type,v_employee_category
from att_in.TBL_EMP_PERSONAL_INFO where EMPID=I_Empid;
exception
when no_data_found then
null;
end;
--SELECT TO_CHAR(TRUNC(to_date('31-may-2013'),'MM'),'DD-MON-RR') into
I_fromDate FROM DUAL;
--SELECT TO_DATE(TRUNC(to_char(I_toDate),'MM'),'DD-MON-RRRR') into
I_fromDate FROM DUAL;

--SELECT TRUNC (I_to_Date, 'month') into I_fromDate FROM DUAL;
--totalday:=to_date('31-dec-12')-to_date(I_fromDate,'DD-MON-RR')+1;
select substr(last_day(to_date(I_from_Date,'DD-MM-RRRR')),1,2) into
totalday from dual;

Select SUM(MakeMinute_FromValue(OTHOURL))
into weeklyOT from V_EMP_REGULAR_STATUS where PunchDate
between TO_Date(I_from_Date,'DD-MM-RRRR') and
TO_Date(I_from_Date + 6,'DD-MM-RRRR')
and EmpId=I_Empid ;
weeklyOTH:=(weeklyOT/60);

-----1st week
Select NVL(SUM(datediff('hh',SHIFTIN,SHIFTOUT)),0) * 60,nvl(count(status),0)
into workingHour ,WeaklyWorkingDay
from V_EMP_REGULAR_STATUS where PunchDate

```

```

between TO_Date(I_from_Date,'DD-MM-RRRR') and TO_Date(I_from_Date+6,'DD-MM-
RRRR')
and EmpId=I_Empid and status not in ('W') ;

dbms_output.put_line('workingHour');
dbms_output.put_line(workingHour);

--dbms_output.put_line('WeaklyWorkingDay');
--dbms_output.put_line(WeaklyWorkingDay);

--dbms_output.put_line(TO_Date(I_fromDate,'DD-MM-RRRR'));
--dbms_output.put_line(TO_Date(I_fromDate+6,'DD-MM-RRRR'));

if workingHour > 2400 then
    workingHour:=2400;
else
    workingHour:=workingHour;
    --dbms_output.put_line('Working Hour Greater then 48');
end if;

If weeklyOT>480 and v_employee_type in('P','G') and
v_employee_category='Permanent' then
    if (v_siftType='General Shift') then
        weeklyOTdeduct:=48*60- workingHour;
        weeklyPaybleOT:= weeklyOT * 2 - weeklyOTdeduct;

        elsif weeklyPaybleOT<weeklyOT and (v_siftType='General Shift') then
            weeklyPaybleOT:=weeklyOT;
        else -- this condition is only for 4 Shift and 2 Shift
            weeklyPaybleOT := weeklyOT * 2;
        end if;
    else
        weeklyPaybleOT := weeklyOT;
    end if;

    --if loopStart=1 then
        firstWeeak:= firstWeeak+weeklyPaybleOT;
        Dbms_output.Put_Line('firstWeeak');
        Dbms_output.Put_Line(weeklyPaybleOT);
        Dbms_output.Put_Line(weeklyOT);
        Dbms_output.Put_Line(weeklyOTdeduct);

    -----End of 1st week

    -----2nd week-----

    I_fromDate1:=I_from_Date + 7;
    weeklyOT1:=0;
    Select SUM(MakeMinute_FromValue(OT HOUR))
    into weeklyOT1
    from V_EMP_REGULAR_STATUS
    where PunchDate

```

```

between TO_Date(I_fromDate1,'DD-MM-RRRR') and
TO_Date(I_fromDate1+6,'DD-MM-RRRR')
and EmpId=I_Empid;
weeklyOTH1:=(weeklyOT1/60);

Select    NVL(SUM(datediff('hh',SHIFTIN,SHIFTOUT)),0) * 60,nvl(count(status),0)
into    workingHour1 ,WeaklyWorkingDay
from V_EMP_REGULAR_STATUS where PunchDate
between TO_Date(I_fromDate1,'DD-MM-RRRR') and TO_Date(I_fromDate1+6,'DD-MM-
RRRR')
and EmpId=I_Empid and status not in ('W');

if workingHour1 > 2400 then
    workingHour1:=2400;
else
    workingHour1:=workingHour1;
    --dbms_output.put_line('Working Hour Greater then 48');
end if;

If v_employee_type in('P','G') and v_employee_category='Permanent' then
    if weeklyOT1>480 and (v_siftType='General Shift' )then
        weeklyOTdeduct1:=48*60- workingHour1;
        weeklyPaybleOT1:= weeklyOT1 * 2 - weeklyOTdeduct1;

        elsif weeklyPaybleOT1<weeklyOT1 and (v_siftType='General Shift')
then
            weeklyPaybleOT1:=weeklyOT1;
        else
            weeklyPaybleOT1:= weeklyOT1 * 2;
        end if;
    else
        weeklyPaybleOT1:= weeklyOT1;
    end if;

    secondWeeak:= secondWeeak+weeklyPaybleOT1;

    dbms_output.put_line('workingHour1');
    dbms_output.put_line(workingHour1);
    Dbms_output.Put_Line(' secondWeeak');
    Dbms_output.Put_Line(weeklyOT1);
    Dbms_output.Put_Line(weeklyOTdeduct1);

    -----3rd week-----
    I_fromDate2:=I_fromDate1+7;

Select    SUM(MakeMinute_FromValue(OT HOUR))
into weeklyOT2 from V_EMP_REGULAR_STATUS where PunchDate
between TO_Date(I_fromDate2,'DD-MM-RRRR') and
TO_Date(I_fromDate2+6,'DD-MM-RRRR')
and EmpId=I_Empid ;
weeklyOTH2:=(weeklyOT2/60);

```



```

        Select  NVL(SUM(datediff('hh',SHIFTIN,SHIFTOUT)),0) * 60,nvl(count(status),0)
into    workingHour2 ,WeaklyWorkingDay
        from V_EMP_REGULAR_STATUS where PunchDate
        between TO_Date(I_fromDate2,'DD-MM-RRRR') and TO_Date(I_fromDate2+6,'DD-MM-
RRRR')
        and EmpId=I_Empid and status not in ('W') ;

        --dbms_output.put_line('workingHour2 in Minute on 20/08/2013');
        --dbms_output.put_line(workingHour2);


if workingHour2 > 2400 then
    workingHour2:=2400;
else
    workingHour2:=workingHour2;
    --dbms_output.put_line('Working Hour Greater then 48');
end if;

If v_employee_type in('P','G') and v_employee_category='Permanent' then
    if weeklyOT2>480 and (v_siftType='General Shift')then
        weeklyOTdeduct2:=48*60- workingHour2;
        weeklyPaybleOT2:= weeklyOT2*2 - weeklyOTdeduct2;

        elsif weeklyPaybleOT2<weeklyOT2 and (v_siftType='General Shift') then
            weeklyPaybleOT2:= weeklyOT2;
        else
            weeklyPaybleOT2:= weeklyOT2 * 2;
        end if;
    else
        weeklyPaybleOT2:= weeklyOT2;
    end if;

    thirdweeak:= thirdweeak+weeklyPaybleOT2;

        dbms_output.put_line('workingHour2');
        dbms_output.put_line(workingHour2);
        Dbms_output.Put_Line('thirdweeak');
        Dbms_output.Put_Line(weeklyOT2);
        Dbms_output.Put_Line(weeklyOTdeduct2);

    -----4th week-----
    I_fromDate3:=I_fromDate2+7;

    Select  SUM(MakeMinute_FromValue(OTHOURL))
into    weeklyOT3 from V_EMP_REGULAR_STATUS where PunchDate
    between TO_Date(I_fromDate3,'DD-MM-RRRR') and
    TO_Date(I_fromDate3+6,'DD-MM-RRRR')
    and EmpId=I_Empid ;
    weeklyOTH3:=(weeklyOT3/60);

```

```

Select    NVL(SUM(datediff('hh',SHIFTIN,SHIFTOUT)),0) * 60,nvl(count(status),0)
into    workingHour3 ,WeaklyWorkingDay
from V_EMP_REGULAR_STATUS where PunchDate
between TO_Date(I_fromDate3,'DD-MM-RRRR') and TO_Date(I_fromDate3+6,'DD-MM-
RRRR')
and EmpId=I_Empid and status not in ('W') ;

if workingHour3 > 2400 then
    workingHour3:=2400;
else
    workingHour3:=workingHour3;
    --dbms_output.put_line('Working Hour Greater then 48');
end if;
If weeklyOT3>480 and v_employee_type in('P','G') and
v_employee_category='Permanent' then
    if (v_siftType='General Shift')then
        weeklyOTdeduct3:=48*60- workingHour3;
        weeklyPaybleOT3 :=  weeklyOT3*2 - weeklyOTdeduct3;
        elsif weeklyPaybleOT3 <weeklyOT3 and (v_siftType='General Shift')
then
            weeklyPaybleOT3 :=weeklyOT3;
        else
            weeklyPaybleOT3 :=  weeklyOT3 * 2;
        end if;
    else
        weeklyPaybleOT3 :=  weeklyOT3;
    end if;

        forthWeeak:= forthWeeak+weeklyPaybleOT3 ;
        dbms_output.put_line('workingHour3');
        dbms_output.put_line(workingHour3);
        Dbms_output.Put_Line('forthWeeak');
        Dbms_output.Put_Line(weeklyOT3);
        Dbms_output.Put_Line(weeklyOTdeduct3);

        -----end of 4th week-----

        -----5th week-----

        I_fromDate4:=I_fromDate3+7;

        Select    SUM(MakeMinute_FromValue(OTHOURL))
        into weeklyOT4 from V_EMP_REGULAR_STATUS where PunchDate
        between TO_Date(I_fromDate4 , 'DD-MM-RRRR') and
        TO_Date(I_to_Date, 'DD-MM-RRRR')
        and EmpId=I_Empid ;
        weeklyOTH4:=(weeklyOTH4/60);

        Select    NVL(SUM(datediff('hh',SHIFTIN,SHIFTOUT)),0) * 60,nvl(count(status),0)
        into    workingHour4 ,WeaklyWorkingDay
        from V_EMP_REGULAR_STATUS where PunchDate
        between TO_Date(I_fromDate4 , 'DD-MM-RRRR') and TO_Date(I_to_Date, 'DD-MM-RRRR')
        and EmpId=I_Empid and status not in ('W') ;

```

```

if workingHour4 > 2400 then
    workingHour4:=2400;
else
    workingHour4:=workingHour4;
    --dbms_output.put_line('Working Hour Greater then 48');
end if;

If v_employee_type in('P','G') and v_employee_category='Permanent' then
    if(weeklyOT4>=240 and weeklyOT4<=720 and (v_siftType='General Shift'
))then
        --weeklyOTdeduct:=48*60- workingHour*60;
        weeklyPaybleOT4:= weeklyOT4 * 2 - 240;

        elsif weeklyPaybleOT4<weeklyOT4 and (v_siftType='General Shift' ) then
            weeklyPaybleOT4:=weeklyOT4;
        else
            weeklyPaybleOT4:= weeklyOT4*2;
        end if;
    else
        weeklyPaybleOT4:= weeklyOT4;
    end if;

    fifthWeeak:= fifthWeeak+weeklyPaybleOT4;
    Dbms_output.Put_Line( firstWeeak);
    Dbms_output.Put_Line( secondWeeak);
    Dbms_output.Put_Line( thirdweeak);
    Dbms_output.Put_Line( forthWeeak);
    Dbms_output.Put_Line( fifthWeeak);

    totalOTH :=nvl(firstWeeak,0) + nvl(secondWeeak,0) + nvl(thirdweeak,0) +
nvl(forthWeeak,0) + nvl(fifthWeeak,0)+nvl(totalOTH1,0);

    if v_shiftid=602 and v_employee_type in('P','G') and
v_employee_category='Permanent' then

        Select nvl( SUM(MakeMinute_FromValue(EXTRAOT)),0)
        into v_short_rest_mt from V_EMP_REGULAR_STATUS where PunchDate
        between TO_Date(I_from_Date , 'DD-MM-RRRR') and
        TO_Date(I_to_Date, 'DD-MM-RRRR')
        and EmpId=I_Empid ;

        totalOTH:=totalOTH+v_short_rest_mt;
    else
        totalOTH:=totalOTH;
    end if;

    if totalOTH>19320 then
        totalOTH:=19320;
    else
        totalOTH:= totalOTH;
    end if;

```

```

else

        Select      SUM(MakeMinute_FromValue(OT_HOUR))
        into totalOTH1
        from att_in.TBL_OT_ENTRY where OT_OR_REG_DATE
        between TO_Date(I_From_Date,'DD-MM-RR') and
TO_Date(I_to_Date,'DD-MM-RR')
        and EmpId=I_EmpId;

        select shifttype into v_siftType from att_in.TBL_SHIFTNAMESETTINGS
where shiftid=I_shiftid;
        begin
        select EMPLOYEE_TYPE,EMPLOYEE_CATEGORY into
v_employee_type,v_employee_category
        from att_in.TBL_EMP_PERSONAL_INFO where EMPID=I_Empid;
        exception
        when no_data_found then
        null;
        end;
        --SELECT TO_CHAR(TRUNC(to_date('31-may-2013'),'MM'),'DD-MON-RR') into
I_fromDate FROM DUAL;
        --SELECT TO_DATE(TRUNC(to_char(I_toDate),'MM'),'DD-MON-RRRR') into
I_fromDate FROM DUAL;

        --SELECT TRUNC (I_to_Date, 'month') into I_fromDate FROM DUAL;
        --totalday:=to date('31-dec-12')-to date(I_fromDate,'DD-MON-RR')+1;
        select substr(last_day(to_date(I_from_Date,'DD-MM-RRRR')),1,2) into
totalday from dual;

        Select      SUM(MakeMinute_FromValue(OT_HOUR))
        into weeklyOT from V_EMP_REGULAR_STATUS where PunchDate
        between TO_Date(I_from_Date,'DD-MM-RRRR') and
TO_Date(I_from_Date + 6,'DD-MM-RRRR')
        and EmpId=I_Empid ;
        weeklyOTH:=(weeklyOT/60);

        -----1st week
        Select      NVL(SUM(datediff('hh',SHIFTIN,SHIFTOUT)),0) * 60,nvl(count(status),0)
into      workingHour ,WeaklyWorkingDay
        from V_EMP_REGULAR_STATUS where PunchDate
        between TO_Date(I_from_Date,'DD-MM-RRRR') and TO_Date(I_from_Date+6,'DD-MM-
RRRR')
        and EmpId=I_Empid and status not in ('W') ;

        dbms_output.put_line('workingHour');
        dbms_output.put_line(workingHour);

        --dbms_output.put_line('WeaklyWorkingDay');
        --dbms_output.put_line(WeaklyWorkingDay);

        --dbms_output.put_line(TO_Date(I_fromDate,'DD-MM-RRRR'));
        --dbms_output.put_line(TO_Date(I_fromDate+6,'DD-MM-RRRR'));

        if workingHour > 2400 then
        workingHour:=2400;

```

```

        else
            workingHour:=workingHour;
            --dbms_output.put_line('Working Hour Greater then 48');
        end if;

        If weeklyOT>480 and v_employee_type in('P','G') and
v_employee_category='Permanent' then
            if (v_siftType='General Shift') then
                weeklyOTdeduct:=48*60- workingHour;
                weeklyPaybleOT:= weeklyOT * 2 - weeklyOTdeduct;

                elsif weeklyPaybleOT<weeklyOT and (v_siftType='General Shift') then
                    weeklyPaybleOT:=weeklyOT;
                else -- this condition is only for 4 Shift and 2 Shift
                    weeklyPaybleOT := weeklyOT * 2;
                end if;
            else
                weeklyPaybleOT := weeklyOT;
            end if;

            --if loopStart=1 then
                firstWeeak:= firstWeeak+weeklyPaybleOT;
                Dbms_output.Put_Line('firstWeeak');
                Dbms_output.Put_Line(weeklyPaybleOT);
                Dbms_output.Put_Line(weeklyOT);
                Dbms_output.Put_Line(weeklyOTdeduct);

            -----End of 1st week

            -----2nd week-----

            I_fromDate1:=I_from_Date + 7;
            weeklyOT1:=0;
            Select SUM(MakeMinute_FromValue(OTHOOR))
            into weeklyOT1
            from V_EMP_REGULAR_STATUS
            where PunchDate
            between TO_Date(I_fromDate1,'DD-MM-RRRR') and
            TO_Date(I_fromDate1+6,'DD-MM-RRRR')
            and EmpId=I_Empid;
            weeklyOTH1:=(weeklyOT1/60);

            Select NVL(SUM(datediff('hh',SHIFTIN,SHIFTOUT)),0) * 60,nvl(count(status),0)
into workingHour1 ,WeaklyWorkingDay
            from V_EMP_REGULAR_STATUS where PunchDate
            between TO_Date(I_fromDate1,'DD-MM-RRRR') and TO_Date(I_fromDate1+6,'DD-MM-
RRRR')
            and EmpId=I_Empid and status not in ('W');

            if workingHour1 > 2400 then

```

```

        workingHour1:=2400;
        else
        workingHour1:=workingHour1;
        --dbms_output.put_line('Working Hour Greater then 48');
        end if;

If v_employee_type in('P','G') and v_employee_category='Permanent' then
    if weeklyOT1>480 and (v_siftType='General Shift' )then
        weeklyOTdeduct1:=48*60- workingHour1;
        weeklyPaybleOT1:= weeklyOT1 * 2 - weeklyOTdeduct1;

        elsif weeklyPaybleOT1<weeklyOT1 and (v_siftType='General Shift')
then
            weeklyPaybleOT1:=weeklyOT1;
        else
            weeklyPaybleOT1:= weeklyOT1 * 2;
        end if;
    else
        weeklyPaybleOT1:= weeklyOT1;
    end if;

    secondWeeak:= secondWeeak+weeklyPaybleOT1;

        dbms_output.put_line('workingHour1');
        dbms_output.put_line(workingHour1);
        Dbms_output.Put_Line(' secondWeeak');
        Dbms_output.Put_Line(weeklyOT1);
        Dbms_output.Put_Line(weeklyOTdeduct1);

    -----3rd week-----
    I_fromDate2:=I_fromDate1+7;

    Select  SUM(MakeMinute_FromValue(OTHOURL))
    into weeklyOT2 from V_EMP_REGULAR_STATUS where PunchDate
    between TO_Date(I_fromDate2,'DD-MM-RRRR') and
    TO_Date(I_fromDate2+6,'DD-MM-RRRR')
    and EmpId=I_Empid ;
    weeklyOTH2:=(weeklyOT2/60);

    Select  NVL(SUM(datediff('hh',SHIFTIN,SHIFTOUT)),0) * 60,nvl(count(status),0)
into  workingHour2 ,WeaklyWorkingDay
from V_EMP_REGULAR_STATUS where PunchDate
between TO_Date(I_fromDate2,'DD-MM-RRRR') and TO_Date(I_fromDate2+6,'DD-MM-
RRRR')
and EmpId=I_Empid and status not in ('W') ;

        --dbms_output.put_line('workingHour2 in Minute on 20/08/2013');
        --dbms_output.put_line(workingHour2);

```

```

if workingHour2 > 2400 then
    workingHour2:=2400;
else
    workingHour2:=workingHour2;
    --dbms_output.put_line('Working Hour Greater then 48');
end if;

If v_employee_type in('P','G') and v_employee_category='Permanent' then
    if weeklyOT2>480 and (v_siftType='General Shift')then
        weeklyOTdeduct2:=48*60- workingHour2;
        weeklyPaybleOT2:= weeklyOT2*2 - weeklyOTdeduct2;

        elsif weeklyPaybleOT2<weeklyOT2 and (v_siftType='General Shift') then
            weeklyPaybleOT2:= weeklyOT2;
        else
            weeklyPaybleOT2:= weeklyOT2 * 2;
        end if;
    else
        weeklyPaybleOT2:= weeklyOT2;
    end if;

    thirdweeak:= thirdweeak+weeklyPaybleOT2;

    dbms_output.put_line('workingHour2');
    dbms_output.put_line(workingHour2);
    Dbms_output.Put_Line('thirdweeak');
    Dbms_output.Put_Line(weeklyOT2);
    Dbms_output.Put_Line(weeklyOTdeduct2);

    -----4th week-----
    I_fromDate3:=I_fromDate2+7;

    Select SUM(MakeMinute_FromValue(OTHOURL))
    into weeklyOT3 from V_EMP_REGULAR_STATUS where PunchDate
    between TO_Date(I_fromDate3,'DD-MM-RRRR') and
    TO_Date(I_fromDate3+6,'DD-MM-RRRR')
    and EmpId=I_Empid ;
    weeklyOTH3:=(weeklyOT3/60);

    Select NVL(SUM(datediff('hh',SHIFTIN,SHIFTOUT)),0) * 60,nvl(count(status),0)
into workingHour3 ,WeaklyWorkingDay
    from V_EMP_REGULAR_STATUS where PunchDate
    between TO_Date(I_fromDate3,'DD-MM-RRRR') and TO_Date(I_fromDate3+6,'DD-MM-
RRRR')
    and EmpId=I_Empid and status not in ('W') ;

    if workingHour3 > 2400 then
        workingHour3:=2400;
    else
        workingHour3:=workingHour3;
        --dbms_output.put_line('Working Hour Greater then 48');
    end if;

```

```

        If weeklyOT3>480 and v_employee_type in('P','G') and
v_employee_category='Permanent' then
            if (v_siftType='General Shift')then
                weeklyOTdeduct3:=48*60- workingHour3;
                weeklyPaybleOT3 := weeklyOT3*2 - weeklyOTdeduct3;
                elsif weeklyPaybleOT3 <weeklyOT3 and (v_siftType='General Shift')
then
                    weeklyPaybleOT3 :=weeklyOT3;
                else
                    weeklyPaybleOT3 := weeklyOT3 * 2;
                end if;
            else
                weeklyPaybleOT3 := weeklyOT3;
            end if;

            forthWeeak:= forthWeeak+weeklyPaybleOT3 ;
            dbms_output.put_line('workingHour3');
            dbms_output.put_line(workingHour3);
            Dbms_output.Put_Line('forthWeeak');
            Dbms_output.Put_Line(weeklyOT3);
            Dbms_output.Put_Line(weeklyOTdeduct3);

            -----end of 4th week-----

            -----5th week-----

            I_fromDate4:=I_fromDate3+7;

            Select SUM(MakeMinute_FromValue(OTHOURL))
into weeklyOT4 from V_EMP_REGULAR_STATUS where PunchDate
between TO_Date(I_fromDate4 , 'DD-MM-RRRR') and
TO_Date(I_to_Date, 'DD-MM-RRRR')
and EmpId=I_Empid ;
weeklyOTH4:=(weeklyOTH4/60);

            Select NVL(SUM(datediff('hh',SHIFTIN,SHIFTOUT)),0) * 60,nvl(count(status),0)
into workingHour4 ,WeeklyWorkingDay
from V_EMP_REGULAR_STATUS where PunchDate
between TO_Date(I_fromDate4 , 'DD-MM-RRRR') and TO_Date(I_to_Date, 'DD-MM-RRRR')
and EmpId=I_Empid and status not in ('W') ;

            if workingHour4 > 2400 then
                workingHour4:=2400;
            else
                workingHour4:=workingHour4;
                --dbms_output.put_line('Working Hour Greater then 48');
            end if;

            If v_employee_type in('P','G') and v_employee_category='Permanent' then
                if(weeklyOT4>=240 and weeklyOT4<=720 and (v_siftType='General Shift'
))then
                    --weeklyOTdeduct:=48*60- workingHour*60;
                    weeklyPaybleOT4:= weeklyOT4 * 2 - 240;

```



```

        elsif weeklyPaybleOT4<weeklyOT4 and (v_siftType='General Shift' ) then
            weeklyPaybleOT4:=weeklyOT4;
        else
            weeklyPaybleOT4:= weeklyOT4*2;
        end if;
    else
        weeklyPaybleOT4:= weeklyOT4;
    end if;

    fifthWeeak:= fifthWeeak+weeklyPaybleOT4;
    Dbms_output.Put_Line( firstWeeak);
    Dbms_output.Put_Line( secondWeeak);
    Dbms_output.Put_Line( thirdweeak);
    Dbms_output.Put_Line( forthWeeak);
    Dbms_output.Put_Line( fifthWeeak);

    totalOTH :=nvl(firstWeeak,0) + nvl(secondWeeak,0) + nvl(thirdweeak,0) +
    nvl(forthWeeak,0) + nvl(fifthWeeak,0)+nvl(totalOTH1,0);

    if v_shiftid=602 and v_employee_type in('P','G') and
    v_employee_category='Permanent' then

        Select nvl( SUM(MakeMinute_FromValue(EXTRAOT)),0)
        into v_short_rest_mt from V_EMP_REGULAR_STATUS where PunchDate
        between TO_Date(I_from_Date , 'DD-MM-RRRR') and
        TO_Date(I_to_Date, 'DD-MM-RRRR')
        and EmpId=I_Empid ;

        totalOTH:=totalOTH+v_short_rest_mt;
    else
        totalOTH:=totalOTH;
    end if;

    if totalOTH>7500 then
        totalOTH:=7500;
    else
        totalOTH:=totalOTH;
    end if;

end if;

payableOT:=att_in.fn_minute_to_time_text_report(totalOTH);

--payableOT:=att_in.FN_MINUTE_TO_TIME_TEXT_REPORT(totalOTH1);

--Dbms_output.Put_Line('v_desig_exist_96');
Dbms_output.Put_Line( totalOTH1);
Dbms_output.Put_Line( payableOT);

elsif v_desig_exist_96=1 then

    --if v_employee_type in('P','G') and v_employee_category='Permanent'

```

then

```
Dbms_output.Put_Line('v_desig_exist_96');  
Dbms_output.Put_Line(v_desig_exist_96);  
Dbms_output.Put_Line(v_employee_type);  
Dbms_output.Put_Line(v_employee_category);  
Dbms_output.Put_Line(v_shiftid);
```

```
dbms_output.put_line(v_count);
```

```
if v_count>0 then
```

```
        Select    SUM(MakeMinute_FromValue(OT_HOUR))  
        into weeklyOTH_SHIFT1  
        from att_in.TBL_OT_ENTRY where OT_OR_REG_DATE  
        between TO_Date(I_From_Date, 'DD-MM-RR') and  
TO_Date(I_to_Date, 'DD-MM-RR')  
        and EmpId=I_EmpId;
```

```
    else
```

```
        Select    SUM(MakeMinute_FromValue(OT_HOURL))  
        into weeklyOTH_SHIFT1  
        from V_EMP_REGULAR_STATUS where PunchDate  
        between TO_Date(I_From_Date, 'DD-MM-RR') and  
TO_Date(I_to_Date, 'DD-MM-RR')  
        and EmpId=I_EmpId;  
    end if;
```

```
        if weeklyOTH_SHIFT1>5760 then  
            totalOTH1:=5760;  
        else  
            totalOTH1:=weeklyOTH_SHIFT1;  
        end if;
```

```
        payableOT:=att_in.FN_MINUTE_TO_TIME_TEXT_REPORT(totalOTH1);
```

```
Dbms_output.Put_Line( totalOTH1);  
Dbms_output.Put_Line( payableOT);
```

```
--end if;
```

```
elseif v_desig_exist_64=1 then
```

```
Dbms_output.Put_Line('v_desig_exist_64');  
Dbms_output.Put_Line(v_desig_exist_64);
```

```
--if v_employee_type in('P','G') and  
v_employee_category='Permanent' then
```

```
    if v_count>0 then
```

```
        Select    SUM(MakeMinute_FromValue(OT_HOUR))  
        into weeklyOTH_SHIFT1
```

```

from att_in.TBL_OT_ENTRY where OT_OR_REG_DATE
between TO_Date(I_From_Date, 'DD-MM-RR') and
TO_Date(I_to_Date, 'DD-MM-RR')
and EmpId=I_EmpId;
else
Select SUM(MakeMinute_FromValue(OTHOOR))
into weeklyOTH_SHIFT1
from V_EMP_REGULAR_STATUS where PunchDate
between TO_Date(I_From_Date, 'DD-MM-RR') and
TO_Date(I_to_Date, 'DD-MM-RR')
and EmpId=I_EmpId;

end if;

if weeklyOTH_SHIFT1>3840 then
totalOTH1:=3840;
else
totalOTH1:=weeklyOTH_SHIFT1;
end if;

payableOT:=att_in.FN_MINUTE_TO_TIME_TEXT_REPORT(totalOTH1);

Dbms_output.Put_Line( totalOTH1);
Dbms_output.Put_Line( payableOT);

--end if;
--end if;

/*elsif v_employee_type in('P','G') and v_employee_category='Permanent'and
v_shiftid =602 then

Select SUM(MakeMinute_FromValue(OTHOOR)*2)
into weeklyOTH_SHIFT1
from att_in.V_EMP_REGULAR_STATUS where PunchDate
between TO_Date(I_From_Date, 'DD-MM-RR') and TO_Date(I_to_Date, 'DD-
MM-RR')
and EmpId=I_EmpId;

totalOTH1:= nvl(weeklyOTH_SHIFT1,0);

if totalOTH1>19320 then
totalOTH1:=19320;
else
totalOTH1:= totalOTH1;

end if;

payableOT:=att_in.FN_MINUTE_TO_TIME_TEXT_REPORT(totalOTH1);

Dbms_output.Put_Line('a');
Dbms_output.Put_Line(weeklyOTH_SHIFT1);
Dbms_output.Put_Line(totalOTH1);

```

```

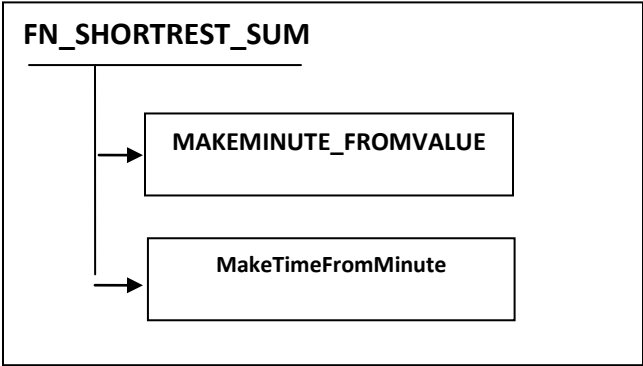
Dbms_output.Put_Line (payableOT);*/

--return payableOT;

end if;
--end;
return payableOT;
end;
/

```

11.FUNCTION: FN\_SHORTREST\_SUM



```

CREATE OR REPLACE FUNCTION ATT_IN.fn_Shortrest_Sum
(
I_FromDate Date,
I_ToDate Date,
I_EmpId int
)
return varchar2
IS
v_count int;
V_ReturnOT varchar(50);
V_ReturnOT1 varchar(50);
V_ReturnOT2 number(9,2);
v_employee_type varchar2(15);
v_employee_category varchar2(20);
v_short_rest_mt number(9,2);
v_shiftid int;
v_desgid varchar2(30);

Begin

begin
select EMPLOYEE_TYPE,EMPLOYEE_CATEGORY,SHIFTID,DESIGNATIONID into v_employee_type,v_employee_category , v_shiftid,v_desgid
from att_in.TBL_EMP_PERSONAL_INFO where EMPID=I_Empid;
EXCEPTION
when no_data_found then
null;
end;

select count(*) into v_count from att_in.TBL_OT_ENTRY

```

```

where empid =I_Empid
and OT_OR_REG_DATE between I_fromDate and I_ToDate ;
if v_count>0 then

If v_employee_type in('P','G') and v_employee_category='Permanent' then
Select  SUM(MakeMinute_FromValue(SHORT_REST_HOUR))
into V_ReturnOT2 from att_in.TBL_OT_ENTRY
where OT_OR_REG_DATE between TO_Date(I_FromDate,'DD-MM-RR') and TO_Date(I_ToDate,'DD-MM-RR')
and EmpId=I_EmpId;

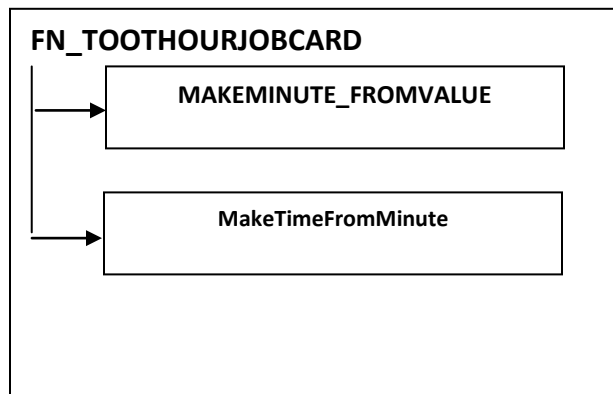
V_ReturnOT:= MakeTimeFromMinute(V_ReturnOT2); else
V_ReturnOT:= '000:00:00';
end if;
elsif v_shiftid=602 and v_employee_type in('P','G') and v_employee_category='Permanent' then

Select nvl( SUM(MakeMinute_FromValue(EXTRAOT)),0)
into v_short_rest_mt from V_EMP_REGULAR_STATUS where PunchDate
between TO_Date(I_FromDate,'DD-MM-RR') and TO_Date(I_ToDate,'DD-MM-RR')
and EmpId=I_Empid ;

V_ReturnOT:= MakeTimeFromMinute(v_short_rest_mt);
else
V_ReturnOT:= '000:00:00';
end if;
return (V_ReturnOT);
exception
when others then
V_ReturnOT:='000:00:00';
return (V_ReturnOT);
END;
/

```

## 12. FUNCTION: FN\_TOOTHOURJOBCARD



```

CREATE OR REPLACE FUNCTION ATT_IN.fn_ToOTHourJobCard
(
I_FromDate Date,
I_ToDate Date,
I_EmpId int
)
return varchar2
IS
v_count int;
V_ReturnOT varchar(50);
V_ReturnOT1 number(9,2);
V_ReturnOT2 number(9,2);
v_employee_type varchar2(15);
v_employee_category varchar2(20);

```

```

v_desig_exist_96 int;
v_desig_exist_64 int;
v_labourcode int;
v_shiftid int;
v_desgid varchar(50);

Begin

begin
select EMPLOYEE_TYPE,EMPLOYEE_CATEGORY,SHIFTID,DESIGNATIONID,LABOURCODE into v_employee_type,v_employee_category ,
v_shiftid,v_desgid,v_labourcode
from att_in.TBL_EMP_PERSONAL_INFO where EMPID=I_Empid;
EXCEPTION
when no_data_found then
null;
end;

begin
select count(*) into v_desig_exist_96 from att_in.TBL_CHECK
where MAX_96_HOURS_OT=v_labourcode;
EXCEPTION
when no_data_found then
null;
end;

begin
select count(*) into v_desig_exist_64 from att_in.TBL_CHECK
where MAX_64_HOURS_OT=v_labourcode;
EXCEPTION
when no_data_found then
null;
end;

select count(*) into v_count from att_in.TBL_OT_ENTRY
where empid =I_Empid
and OT_OR_REG_DATE between I_fromDate and I_ToDate ;

if v_count>0 and v_desig_exist_96=0 and v_desig_exist_64 =0 then

Select  nvl(SUM(MakeMinute_FromValue(OT_HOUR)),0) into V_ReturnOT1 from att_in.TBL_OT_ENTRY where OT_OR_REG_DATE
between TO_Date(I_FromDate,'DD-MM-RR') and TO_Date(I_ToDate,'DD-MM-RR')
and EmpId=I_EmpId;

if V_ReturnOT1>7500 then
V_ReturnOT1:=7500;
else
V_ReturnOT1:=V_ReturnOT1;
end if;
V_ReturnOT:= MakeTimeFromMinute(V_ReturnOT1);
return (V_ReturnOT);
elsif v_desig_exist_96=1 then

if v_count>0 then

Select SUM(MakeMinute_FromValue(OT_HOUR))
into V_ReturnOT1
from att_in.TBL_OT_ENTRY where OT_OR_REG_DATE
between TO_Date(I_FromDate,'DD-MM-RR') and TO_Date(I_ToDate,'DD-MM-RR')
and EmpId=I_EmpId;

else

Select SUM(MakeMinute_FromValue(OT_HOUR))
into V_ReturnOT1
from V_EMP_REGULAR_STATUS where PunchDate

```

```

between TO_Date(I_FromDate,'DD-MM-RR') and TO_Date(I_ToDate,'DD-MM-RR')
and EmpId=I_EmpId;
end if;

if V_ReturnOT1>5760 then
V_ReturnOT1:=5760;
else
V_ReturnOT1:=V_ReturnOT1;
end if;

V_ReturnOT:= MakeTimeFromMinute(V_ReturnOT1);
return (V_ReturnOT);
elsif v_desig_exist_64=1 then

if v_count>0 then

Select SUM(MakeMinute_FromValue(OT_HOUR))
into V_ReturnOT1
from att_in.TBL_OT_ENTRY where OT_OR_REG_DATE
between TO_Date(I_FromDate,'DD-MM-RR') and TO_Date(I_ToDate,'DD-MM-RR')
and EmpId=I_EmpId;

else

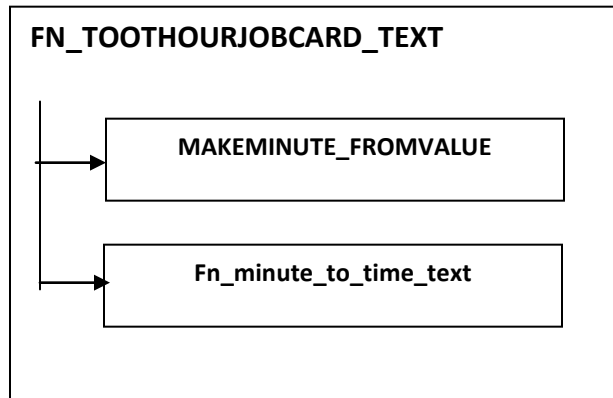
Select SUM(MakeMinute_FromValue(OTHOURL))
into V_ReturnOT1
from V_EMP_REGULAR_STATUS where PunchDate
between TO_Date(I_FromDate,'DD-MM-RR') and TO_Date(I_ToDate,'DD-MM-RR')
and EmpId=I_EmpId;
end if;

if V_ReturnOT1>3840 then
V_ReturnOT1:=3840;
else
V_ReturnOT1:=V_ReturnOT1;
end if;
V_ReturnOT:= MakeTimeFromMinute(V_ReturnOT1);
return (V_ReturnOT);

else
Select nvl( SUM(MakeMinute_FromValue(OTHOURL)),0) into V_ReturnOT2 from V_EMP_REGULAR_STATUS where PunchDate
between TO_Date(I_FromDate,'DD-MM-RR') and TO_Date(I_ToDate,'DD-MM-RR')
and EmpId=I_EmpId;
if V_ReturnOT2>7500 then
V_ReturnOT:=7500;
else
V_ReturnOT:=V_ReturnOT2;
end if;
return MakeTimeFromMinute(V_ReturnOT);
end if;
exception
when others then
V_ReturnOT:='000:00:00';
return (V_ReturnOT);
END;

```

### 13. **FUNCTION:** FN\_TOOTHOURJOB CARD\_TEXT



```
CREATE OR REPLACE FUNCTION ATT_IN.fn_ToOTHourJobCard_Text
(
  I_FromDate Date,
  I_ToDate Date,
  I_EmpId int
)
return varchar2
IS
  v_count int;
  V_ReturnOT varchar(50);
  V_ReturnOT1 varchar(50);
  V_ReturnOT2 varchar(50);
  v_desig_exist_96 int;
  v_desig_exist_64 int;
  v_labourcode int;
  v_shiftid int;
  v_desgid varchar(50);
  v_employee_type varchar2(15);
  v_employee_category varchar2(20);
Begin
begin
select EMPLOYEE_TYPE,EMPLOYEE_CATEGORY,SHIFTID,DESIGNATIONID,LABOURCODE into v_employee_type,v_employee_category ,
v_shiftid,v_desgid,v_labourcode
from att_in.TBL_EMP_PERSONAL_INFO where EMPID=I_Empid;
EXCEPTION
when no_data_found then
null;
end;

begin
select count(*) into v_desig_exist_96 from att_in.TBL_CHECK
where MAX_96_HOURS_OT=v_labourcode;
EXCEPTION
when no_data_found then
null;
end;

begin
select count(*) into v_desig_exist_64 from att_in.TBL_CHECK
where MAX_64_HOURS_OT=v_labourcode;
EXCEPTION
when no_data_found then
null;
end;

select count(*) into v_count from att_in.TBL_OT_ENTRY
```



```

where empid =I_Empid
and OT_OR_REG_DATE between I_fromDate and I_ToDate ;

if v_count>0 and v_desig_exist_96=0 and v_desig_exist_64 =0 then

Select  SUM(MakeMinute_FromValue(OT_HOUR)) into V_ReturnOT1 from att_in.TBL_OT_ENTRY where OT_OR_REG_DATE
between TO_Date(I_FromDate,'DD-MM-RR') and TO_Date(I_ToDate,'DD-MM-RR')
and EmpId=I_EmpId;

if V_ReturnOT1>7500 then
V_ReturnOT1:=7500;
else
V_ReturnOT1:=V_ReturnOT1;
end if;

V_ReturnOT:= V_ReturnOT1;
return Fn_minute_to_time_text(V_ReturnOT);
elsif v_desig_exist_96=1 then
if v_count>0 then
Select  SUM(MakeMinute_FromValue(OT_HOUR))
into V_ReturnOT1
from att_in.TBL_OT_ENTRY where OT_OR_REG_DATE
between TO_Date(I_FromDate,'DD-MM-RR') and TO_Date(I_ToDate,'DD-MM-RR')
and EmpId=I_EmpId;
else
Select  SUM(MakeMinute_FromValue(OT_HOUR))
into V_ReturnOT1
from V_EMP_REGULAR_STATUS where PunchDate
between TO_Date(I_FromDate,'DD-MM-RR') and TO_Date(I_ToDate,'DD-MM-RR')
and EmpId=I_EmpId;
end if;
if V_ReturnOT1>5760 then
V_ReturnOT1:=5760;
else
V_ReturnOT1:=V_ReturnOT1;
end if;
V_ReturnOT:= Fn_minute_to_time_text(V_ReturnOT1);
return (V_ReturnOT);
elsif v_desig_exist_64=1 then

if v_count>0 then

Select SUM(MakeMinute_FromValue(OT_HOUR))
into V_ReturnOT1
from att_in.TBL_OT_ENTRY where OT_OR_REG_DATE
between TO_Date(I_FromDate,'DD-MM-RR') and TO_Date(I_ToDate,'DD-MM-RR')
and EmpId=I_EmpId;
else
Select SUM(MakeMinute_FromValue(OT_HOUR))
into V_ReturnOT1
from V_EMP_REGULAR_STATUS where PunchDate
between TO_Date(I_FromDate,'DD-MM-RR') and TO_Date(I_ToDate,'DD-MM-RR')
and EmpId=I_EmpId;
end if;
if V_ReturnOT1>3840 then
V_ReturnOT1:=3840;
else
V_ReturnOT1:=V_ReturnOT1;
end if;
V_ReturnOT:= Fn_minute_to_time_text(V_ReturnOT1);
return (V_ReturnOT);
else
Select SUM(MakeMinute_FromValue(OT_HOUR))
into V_ReturnOT from V_EMP_REGULAR_STATUS
where PunchDate between TO_Date(I_FromDate,'DD-MM-RR') and TO_Date(I_ToDate,'DD-MM-RR')
and EmpId=I_EmpId;

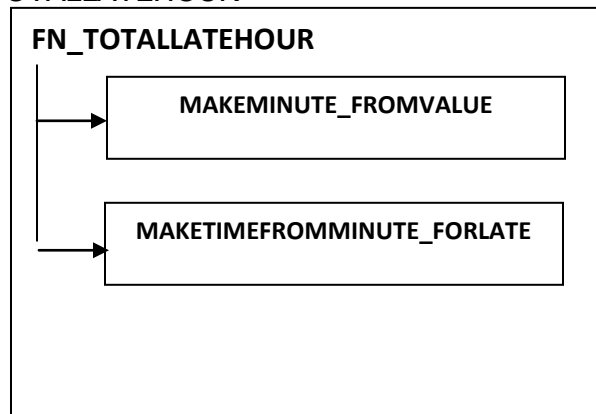
```

```

if V_ReturnOT>7500 then
V_ReturnOT:=7500;
else
V_ReturnOT:=V_ReturnOT;
end if;
return Fn_minute_to_time_text(V_ReturnOT);
end if;
exception
when others then
V_ReturnOT:='000:00:00';
return(V_ReturnOT);
END;

```

#### 14. **FUNCTION:** FN\_TOTALLATEHOUR

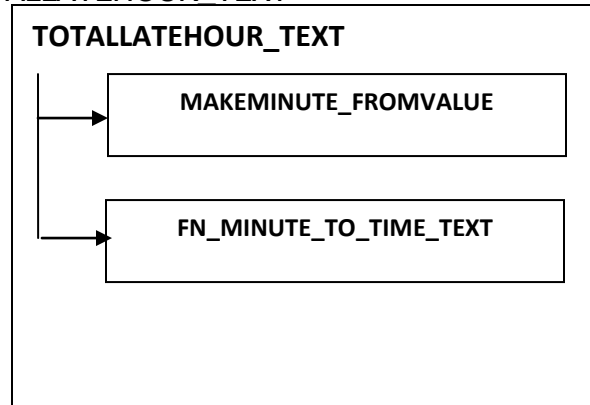


```

CREATE OR REPLACE FUNCTION ATT_IN.Fn_TotallateHour
(
I_FromDate Date,
I_ToDate Date,
I_EmpId int
)
return varchar2
IS
V_ReturnOT varchar(50);
Begin
Select MakeTimeFromMinute_Forlate(SUM(MakeMinute_FromValue(LATE)))
into V_ReturnOT from V_EMP_REGULAR_STATUS
where PunchDate between TO_Date(I_FromDate,'DD-MM-RR') and TO_Date(I_ToDate,'DD-MM-RR')
and EmpId=I_EmpId;
return(V_ReturnOT);
END;

```

## 15. FUNCTION: TOTALLATEHOUR TEXT



```
CREATE OR REPLACE FUNCTION ATT_IN.Fn_TotalLateHour_Text
(
  I_FromDate Date,
  I_ToDate Date,
  I_EmpId int
)
return varchar2
IS
  V_ReturnOT varchar(50);
Begin
  Select  Fn_minute_to_time_text(SUM(MakeMinute_FromValue(LATE)))
into V_ReturnOT from V_EMP_REGULAR_STATUS
where PunchDatebetween TO_Date(I_FromDate,'DD-MM-RR') and TO_Date(I_ToDate,'DD-MM-RR')
and EmpId=I_EmpId;
return(V_ReturnOT);
END;
```

## 16. FN\_LEAVECODE

```
CREATE OR REPLACE FUNCTION ATT_IN.Fn_LeaveCode
(
  V_empid in int,
  V_fromdate in date,
  V_todate in date
)
return Varchar2
AS
BEGIN
  declare
  Ret_Status varchar2(50);
  I_C int;
  begin
  Ret_Status:="" ;
  I_C :=0;

  select count(status) into I_C from att_in.TBL_PROCESSED_DATA where EmpId=V_empid and PunchDate
  between TO_Date(V_fromDate,'DD-MM-RR') and TO_Date(V_todate,'DD-MM-RR') and status ='QL';
  if I_C>0 then
  if Ret_Status is null then
  Ret_Status:='1';
  else
  Ret_Status:=Ret_Status||','||'1';
  end if;
  end if ;

  select count(status) into I_C from att_in.TBL_PROCESSED_DATA where EmpId=V_empid and PunchDate
  between TO_Date(V_fromDate,'DD-MM-RR') and TO_Date(V_todate,'DD-MM-RR') and status ='ML';
```

```

if I_C>0 then
if Ret_Status is null then
Ret_Status:='2';
else
Ret_Status:=Ret_Status||','||'2';
end if;
end if ;

select count(status) into I_C from att_in.TBL_PROCESSED_DATA where EmpId=V_empid and PunchDate between TO_Date(V_fromDate,'DD-MM-RR') and TO_Date(V_todate,'DD-MM-RR') and status ='AL';
if I_C>0 then
if Ret_Status is null then
Ret_Status:='3';
else
Ret_Status:=Ret_Status||','||'3';
end if;
end if ;

select count(status) into I_C from att_in.TBL_PROCESSED_DATA
where EmpId=V_empid
and PunchDate between TO_Date(V_fromDate,'DD-MM-RR') and TO_Date(V_todate,'DD-MM-RR')
and status ='STL';
if I_C>0 then
if Ret_Status is null then
Ret_Status:='4';
else
Ret_Status:=Ret_Status||','||'4';
end if;

end if ;

select count(status) into I_C from att_in.TBL_PROCESSED_DATA where EmpId=V_empid and PunchDate
between TO_Date(V_fromDate,'DD-MM-RR') and TO_Date(V_todate,'DD-MM-RR') and status ='SPL';
if I_C>0 then
if Ret_Status is null then
Ret_Status:='5';
else
Ret_Status:=Ret_Status||','||'5';
end if;

end if ;

select count(status) into I_C from att_in.TBL_PROCESSED_DATA where EmpId=V_empid and PunchDate
between TO_Date(V_fromDate,'DD-MM-RR') and TO_Date(V_todate,'DD-MM-RR') and status ='SWPL';
if I_C>0 then
if Ret_Status is null then
Ret_Status:='6';
else
Ret_Status:=Ret_Status||','||'6';
end if;

end if ;

select count(status) into I_C from att_in.TBL_PROCESSED_DATA where EmpId=V_empid and PunchDate
between TO_Date(V_fromDate,'DD-MM-RR') and TO_Date(V_todate,'DD-MM-RR') and status ='JL';
if I_C>0 then
if Ret_Status is null then
Ret_Status:='7';
else
Ret_Status:=Ret_Status||','||'7';
end if;

end if ;

select count(status) into I_C from att_in.TBL_PROCESSED_DATA where EmpId=V_empid and PunchDate
between TO_Date(V_fromDate,'DD-MM-RR') and TO_Date(V_todate,'DD-MM-RR') and status ='DFL';
if I_C>0 then
if Ret_Status is null then
Ret_Status:='8';

```

```

else
Ret_Status:=Ret_Status||','||'8';
end if;

end if ;
select count(status) into I_C from att_in.TBL_PROCESSED_DATA where EmpId=V_empid and PunchDate
between TO_Date(V_fromDate,'DD-MM-RR') and TO_Date(V_todate,'DD-MM-RR') and status ='SSL';
if I_C>0 then
if Ret_Status is null then
Ret_Status:='9';
else
Ret_Status:=Ret_Status||','||'9';
end if;

end if ;
return Ret_Status;

end ;
END Fn_LeaveCode ;

```

## 17. FUNCTION: FN\_LEAVECODE1

```

CREATE OR REPLACE FUNCTION ATT_IN.Fn_LeaveCode1
(
V_empid in int,
V_fromdate in date,
V_todate in date
)
return Varchar2
AS
BEGIN
declare
Ret_Status varchar2(50):='0';
v_status varchar2(50);
v_rownum int;
I_C int;
begin
Ret_Status:='0' ;
-- I_C :=0;

begin
select distinct status into v_status from TBL_LEAVECODE
where empid=V_empid
and FROMDATE=V_fromdate
and TODATE =V_todate
and id=1
order by id;
exception
when no_data_found then
Ret_Status:='0';
end;

if v_status='QL' THEN
Ret_Status:='1';

ELSIF v_status='ML' THEN
Ret_Status:='2';
ELSIF v_status='AL' THEN
Ret_Status:='3';

ELSIF v_status='STL' THEN
Ret_Status:='4';
ELSIF v_status='SPL/PRL' THEN

```

```

        Ret_Status:='5';

        ELSIF v_status='SWPL' THEN
        Ret_Status:='6';
ELSIF v_status='JL' THEN
        Ret_Status:='7';

        ELSIF v_status='DFL' THEN
        Ret_Status:='8';
        ELSIF v_status='SSL' THEN
        Ret_Status:='9';
        --ELSE
        --Ret_Status:='0';
        END IF;

        DBMS_OUTPUT.PUT_LINE( Ret_Status );

    ----end if;
    return Ret_Status;

end ;
END Fn_LeaveCode1 ;
/

```

## 18. FUNCTION: FN\_LEAVECODE2

```

CREATE OR REPLACE FUNCTION ATT_IN.Fn_LeaveCode2
(
V_empid in int,
V_fromdate in date,
V_todate in date
)
return Varchar2
AS
BEGIN
declare
Ret_Status varchar2(50):='0';
v_status varchar2(50);
v_rownum int;
I_C int;
begin
    Ret_Status:=0 ;
    -- I_C :=0;

    --select distinct(count (rownum) )into v_rownum from TBL_LEAVECODE;

    ---if v_rownum>=2 then
    begin
        select status into v_status from TBL_LEAVECODE
        where empid=V_empid
        and FROMDATE=V_fromdate
        and TODATE =V_todate
        and id=2
        order by id;
        exception
        when no_data_found then
        Ret_Status:='0';
        end;
    end;

```

```

        if      v_status='QL' THEN
            Ret_Status:='1';

            ELSIF v_status='ML' THEN
                Ret_Status:='2';
            ELSIF v_status='AL' THEN
                Ret_Status:='3';

            ELSIF v_status='STL' THEN
                Ret_Status:='4';
            ELSIF v_status='SPL/PRL' THEN
                Ret_Status:='5';

            ELSIF v_status='SWPL' THEN
                Ret_Status:='6';
            ELSIF v_status='JL' THEN
                Ret_Status:='7';

            ELSIF v_status='DFL' THEN
                Ret_Status:='8';
            ELSIF v_status='SSL' THEN
                Ret_Status:='9';
            --ELSE
            --Ret_Status:='0';
            END IF;

            DBMS_OUTPUT.PUT_LINE( Ret_Status );

        ---end if;
        return Ret_Status;
    end ;
END Fn_LeaveCode2 ;
/

```

## 19. FUNCTION: FN\_LEAVEDAYS1

```

CREATE OR REPLACE FUNCTION ATT_IN.Fn_LeaveDays1
(
    V_empid in int,
    V_fromdate in date,
    V_todate in date
)
return Varchar2
AS
BEGIN
    declare
        Ret_Days varchar2(50):='0';
        v_status varchar2(50);
        v_rownum int;
        I_C int;
    begin
        Ret_Days:='0' ;
        -- I_C :=0;

        begin
            select distinct status into v_status from TBL_LEAVECODE

```

```

where empid=V_empid
and FROMDATE=V_fromdate
and TODATE =V_todate
and id=1
order by id;
exception
when no_data_found then
Ret_Days:='0';
end;
begin
select count(status) into Ret_Days  from att_in.V_ATTENDANCE_REPORT
where empid=V_empid
and PUNCHDATE between  V_fromdate and V_todate
and STATUS=v_status;

exception
when no_data_found then
Ret_Days:='0';

end;
if Ret_Days <=9 then
  Ret_Days:='0' || Ret_Days;
else
  Ret_Days:=Ret_Days;
end if;

/* if      v_status='QL' THEN
  Ret_Status:='1';

ELSIF v_status='ML' THEN
  Ret_Status:='2';
ELSIF v_status='AL' THEN
  Ret_Status:='3';

ELSIF v_status='STL' THEN
  Ret_Status:='4';
ELSIF v_status='SPL/PRL' THEN
  Ret_Status:='5';

ELSIF v_status='SWPL' THEN
  Ret_Status:='6';
ELSIF v_status='JL' THEN
  Ret_Status:='7';

ELSIF v_status='DFL' THEN
  Ret_Status:='8';
ELSIF v_status='SSL' THEN
  Ret_Status:='9';
--ELSE
--Ret_Status:='0';
END IF; */

--DBMS_OUTPUT.PUT_LINE( Ret_Status );

return Ret_Days ;

end ;
END Fn_LeaveDays1 ;
/

```



## 20. FUNCTION: FN\_LEAVEDEAYS2

```
CREATE OR REPLACE FUNCTION ATT_IN.Fn_LeaveDays2
(
V_empid in int,
V_fromdate in date,
V_todate in date
)
return Varchar2
AS
BEGIN
declare
Ret_Days varchar2(50):='0';
v_status varchar2(50);
v_rownum int;
I_C int;
begin
    Ret_Days:='0' ;
    -- I_C :=0;

    begin
        select distinct status into v_status    from TBL_LEAVECODE
        where empid=V_empid
        and FROMDATE=V_fromdate
        and TODATE =V_todate
        and id=2
        order by id;
    exception
    when no_data_found then
        Ret_Days:='0';
    end;

    begin
        select count(status) into Ret_Days    from att_in.V_ATTENDANCE_REPORT
        where empid=V_empid
        and PUNCHDATE between  V_fromdate and V_todate
        and STATUS=v_status;

    exception
    when no_data_found then
        Ret_Days:='0';

    end;

    if Ret_Days <=9 then
        Ret_Days:='0'||Ret_Days;
    else
        Ret_Days:=Ret_Days;
    end if;

    /* if      v_status='QL' THEN
        Ret_Status:='1';

        ELSIF v status='ML' THEN
```

```

Ret_Status:='2';
ELSIF v_status='AL' THEN
Ret_Status:='3';

ELSIF v_status='STL' THEN
Ret_Status:='4';
ELSIF v_status='SPL/PRL' THEN
Ret_Status:='5';

ELSIF v_status='SWPL' THEN
Ret_Status:='6';
ELSIF v_status='JL' THEN
Ret_Status:='7';

ELSIF v_status='DFL' THEN
Ret_Status:='8';
ELSIF v_status='SSL' THEN
Ret_Status:='9';
--ELSE
--Ret_Status:='0';
END IF; */

--DBMS_OUTPUT.PUT_LINE( Ret_Status );

----end if;
return Ret_Days ;

end ;
END Fn_LeaveDays2 ;
/

```

## 21. FUNCTION:FN\_LEAVE\_OSD\_COUNT

```

CREATE OR REPLACE FUNCTION ATT_IN.fn_Leave_OSD_Count
(
I_FromDate In Date,
I_ToDate In date,
I_EmpId In Int
)
return int
as
I_result int:=0;
leave_counter int:=0;
osd_counter int:=0;

Begin
select count(EMPID) into leave_counter
from att_in.TBL_LEAVE_APPLY
where EMPID = I_EmpId
and FROMDATE <= TO_DATE(to_char(I_FromDate,'DD/MM/RRRR'),'DD/MM/RRRR')
and TODATE >= TO_DATE(to_char(I_ToDate,'DD/MM/RRRR'),'DD/MM/RRRR');

select count(EMPID) into osd_counter
from att_in.TBL_OSD_SETUP
where EMPID = I_EmpId
and OSDSTARTDATE <= TO_DATE(to_char(I_FromDate,'DD/MM/RRRR'),'DD/MM/RRRR')
and OSDENDDATE >= TO_DATE(to_char(I_ToDate,'DD/MM/RRRR'),'DD/MM/RRRR');

I_result:=leave_counter+osd_counter;
return(I_result);

END;

```

## 22. FUNCTION:FN\_SHIFTDAYS

```
CREATE OR REPLACE FUNCTION ATT_IN.Fn_ShiftDays
(
V_empid in int,
V_fromdate in date,
V_todate in date
)
return int
AS
BEGIN
declare
Ret_Status int;
I_C int;
begin
select count(status) into I_C from att_in.V_ATTENDANCE_REPORT
where EmpId=V_empid
and PunchDate between TO_Date(V_fromDate,'DD-MM-RR') and TO_Date(V_todate,'DD-MM-RR')
and status in('P','L') and WORKINGSHIFT
not in ('G Shift','MT Operator G Shift','G Shift BATC');
if I_C>0 then
Ret_Status:=I_C;
else
Ret_Status:=0;
end if ;
return Ret_Status;
end ;
END Fn_ShiftDays ;
```

## 23. FUNCTION:FN\_MEALALLOWANCE

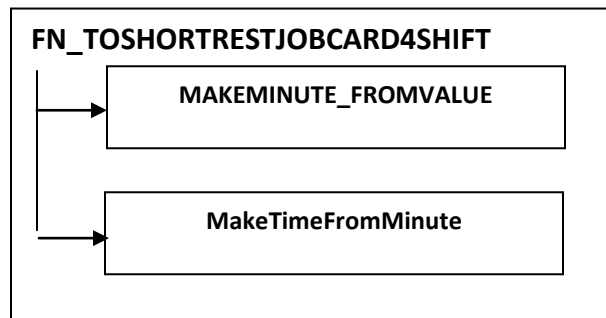
```
CREATE OR REPLACE function ATT_IN.FN_Mealallowance
( I_EmpId int,
I_PunchTime date,
I_shiftid int,
I_OT_Hour varchar2
)
return varchar2
IS
v_employee_type varchar2(20);
v_employee_category varchar2(20);
V_shiftype varchar2(20);
V_mealallowance varchar2(2);
V_shiftExist int;
v_punchtime varchar2(10);
v_OT_Hour int ;
v_OT_Hour1 int ;
Begin
select EMPLOYEE_TYPE,EMPLOYEE_CATEGORY
into v_employee_type ,v_employee_category from TBL_EMP_PERSONAL_INFO
where EMPID=I_Empid;
v_punchtime:=to_char(I_PunchTime,'HH24:MI:SS');
v_OT_Hour:=floor((MakeMinute_FromValue(I_OT_Hour))/60);
v_OT_Hour1:=round((MakeMinute_FromValue(I_OT_Hour))/60);
```

```

If v_employee_type in('P','G') and v_employee_category='Permanent' then
if v_OT_Hour >=2 and v_OT_Hour1 <=3 then
V_mealallowance:='1';
elsif v_OT_Hour1 >3 and v_OT_Hour1<=7 then
V_mealallowance:='2';
elsif v_OT_Hour1 > 7 and v_OT_Hour1<=13 then
V_mealallowance:='3';
elsif v_OT_Hour1 >13 then
V_mealallowance:='4';
else
V_mealallowance:='0';
end if;
else
V_mealallowance:='0';
end if;
return V_mealallowance;
End;

```

24. **FUNCTION:** FN\_TOSHORTRESTJOBCARD4SHIFT

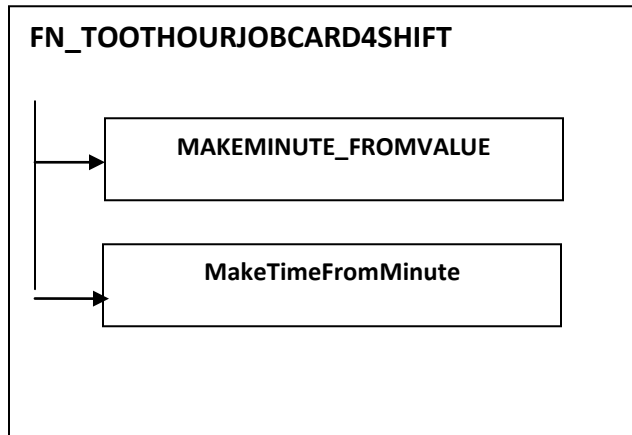


```

CREATE OR REPLACE FUNCTION ATT_IN.fn_ToOTHourJobCard4Shift
(
I_FromDate Date,I_ToDate Date,
I_EmpId int
)
return varchar2
IS
V_ReturnOT1 number(9,2);
Begin
Select SUM(MakeMinute_FromValue(OT_HOUR)) into V_ReturnOT1
from att_in.TBL_OT_ENTRY
where OT_OR_REG_DATE
between TO_Date(I_FromDate,'DD-MM-RR') and TO_Date(I_ToDate,'DD-MM-RR')
and EmpId=I_EmpId;
if V_ReturnOT1>7500 then
V_ReturnOT1:=7500;
else
V_ReturnOT1:=V_ReturnOT1;
end if;
return MakeTimeFromMinute(V_ReturnOT1);
exception
when others then
V_ReturnOT:='000:00:00';
END;

```

25. **FUNCTION:** FN\_TOOTHOURJOBCARD4SHIFT



```

CREATE OR REPLACE FUNCTION ATT_IN.fn_ToOTHourJobCard4Shift
(
  I_FromDate Date,
  I_ToDate Date,
  I_EmpId int
)
return varchar2
IS
  V_ReturnOT varchar(50);
  V_ReturnOT1 number(9,2);
  V_ReturnOT2 number(9,2);
Begin
  Select SUM(MakeMinute_FromValue(OT_HOUR)) into V_ReturnOT1 from att_in.TBL_OT_ENTRY where
OT_OR_REG_DATE
  between TO_Date(I_FromDate,'DD-MM-RR') and TO_Date(I_ToDate,'DD-MM-RR')
  and EmpId=I_EmpId;

  --MakeTimeFromMinute(
  if V_ReturnOT1>7500 then
  V_ReturnOT1:=7500;
  else
  V_ReturnOT1:=V_ReturnOT1;
  end if;
  return MakeTimeFromMinute(V_ReturnOT1);
  exception
  when others then
  V_ReturnOT:='000:00:00';
END;
/
  
```

## D) TRIGGERS

### **TRIGGER:** DELETE\_UPDATE\_LEAVE\_APPLY

Used Table Name: TBL\_FINANCIAL\_YEAR

Description	Trigger Body
<p>CREATE OR REPLACE TRIGGER ATT_IN.DELETE_UPDATE_LEAVE_ APPLY</p> <p>after delete or update ON ATT_IN.TBL_LEAVE_APPLY referencing old as old new as new</p> <p>FOR EACH ROW</p>	<pre> BEGIN if deleting then insert into ATT_IN.TBL_LEAVE_APPLY_HISTORY (LEAVEAPPLYID,EMPID,LEAVETYPEID,FROMDATE,TODATE,CAUSE, NUMBEROFDAYS,APPROVEDBY,APPROVAL, CREATEBY,CREATEDATE,UPDATEBY,UPDATEDATE,COMPFINYID,E MPCODE,DEPTID,PLANT_ID,DELETEBY,DELETE_DATE)  values(:old.LEAVEAPPLYID,:old.EMPID,:old.LEAVETYPEID,:old.FRO MDATE,:old.TODATE,:old.CAUSE,:old.NUMBEROFDAYS,:old.APPRO VEDBY,:old.APPROVAL, :old.CREATEBY,:old.CREATEDATE,:old.UPDATEBY,:old.UPDATEDAT E,:old.COMPFINYID,:old.EMPCODE,:old.DEPTID,:old.PLANT_ID,USE R,SYSDATE);  END IF;  IF UPDATING THEN insert into ATT_IN.TBL_LEAVE_APPLY_HISTORY (LEAVEAPPLYID_N,EMPID_N,LEAVETYPEID_N,FROMDATE_N,TODA TE_N,CAUSE_N,NUMBEROFDAYS_N,APPROVEDBY_N,APPROVAL_N, CREATEBY_N,CREATEDATE_N,UPDATEBY_N,UPDATEDATE_N,COM PFINYID_N,EMPCODE_N,DEPTID_N,PLANT_ID_N,NEW_UPDATEDA TE, NEW_UPDATEUSER) VALUES(:NEW.LEAVEAPPLYID,:NEW.EMPID,:NEW.LEAVETYPEID,:N EW.FROMDATE,:NEW.TODATE,:NEW.CAUSE,:NEW.NUMBEROFDAY S,:NEW.APPROVEDBY,:NEW.APPROVAL, :NEW.CREATEBY,:NEW.CREATEDATE,:NEW.UPDATEBY,:NEW.UPDA TEDATE,:NEW.COMPFINYID,:NEW.EMPCODE,:NEW.DEPTID,:NEW. PLANT_ID,SYSDATE,USER); END IF; END; </pre>

**TRIGGER:** DELETE\_UPDATE\_TBL\_EMP\_INFO

Used Table Name: TBL\_EMP\_PERSONAL\_INFO

Description	Trigger Body
CREATE OR REPLACE TRIGGER ATT_IN.DELETE_UPDATE_TBL_EMP_INFO after delete or update ON ATT_IN.TBL_EMP_PERSONAL_INFO  referencing old as old new as new  FOR EACH ROW	BEGIN if deleting then INSERT INTO ATT_IN.TBL_EMP_PERSONAL_INFO_HISTORY (EMPID,EMPCODE,COMPCARDID,NATIONALIDCARD,EMPNAMEE,NICKNAME,EMP_PHOTO,JOININGDATE,CONFIRMDATE,HOLIDAY_APPLICABLE, NIGHTSHIFT_APPLICABLE,OVERTIME_APPLICABLE,EMPLOYEE_CATEGORY,EMPLOYEMENT_CATEGORY,EMPLOYEE_TYPE,COMPID,BD_ID,PLANT_ID, DEPTID,COST_CENTER_ID,SECTID,WORKCENTER_ID,DESIGNATIONID,SHIFTID,FATHERSNAME,MOTHERSNAME,SPOUSENAME,DATEOFBIRTH,BLOODGROUP, SEX,RELIGION,MARITALSTATUS,NOMINEENAME,RELATION_NOMINEE,NOMINEE_PHOTO,TELEPHONE,FAX,EMAIL,CONTACTNUM1,CONTACTNUM2, EMERGENCYCONTACTNUM,PRESENTADD,PERMANENTADD,CREATEBY,CREATEDATE,UPDATEBY,UPDATEDATE,EX_INT_1,EX_INT_2,EX_VARCHAR_1, EX_VARCHAR_2,EX_DATE_1,EX_DATE_2,EMPENDISSTATUS,QUITDATE,GREADID,LABOURCODE,DELETE_DATE,DELETE_USER) VALUES(:old.EMPID,:old.EMPCODE,:old.COMPCARDID,:old.NATIONALIDCARD,:old.EMPNAMEE,:old.NICKNAME,:old.EMP_PHOTO,:old.JOININGDATE,:old.CONFIRMDATE,:old.HOLIDAY_APPLICABLE, :old.NIGHTSHIFT_APPLICABLE,:old.OVERTIME_APPLICABLE,:old.EMPLOYEE_CATEGORY,:old.EMPLOYEMENT_CATEGORY,:old.EMPLOYEE_TYPE,:old.COMPID,:old.BD_ID,:old.PLANT_ID, :old.DEPTID,:old.COST_CENTER_ID,:old.SECTID,:old.WORKCENTER_ID,:old.DESIGNATIONID,:old.SHIFTID,:old.FATHERSNAME,:old.MOTHERSNAME,:old.SPOUSENAME,:old.DATEOFBIRTH,:old.BLOODGROUP, :old.SEX,:old.RELIGION,:old.MARITALSTATUS,:old.NOMINEENAME,:old.RELATION_NOMINEE,:old.NOMINEE_PHOTO,:old.TELEPHONE, :old.FAX,:old.EMAIL,:old.CONTACTNUM1,:old.CONTACTNUM2, :old.EMERGENCYCONTACTNUM,:old.PRESENTADD,:old.PERMANENTADD,:old.CREATEBY,:old.CREATEDATE,:old.UPDATEBY,:old.UPDATEDATE,:old.EX_INT_1,:old.EX_INT_2,:old.EX_VARCHAR_1, :old.EX_VARCHAR_2,:old.EX_DATE_1,:old.EX_DATE_2,:old.EMPENDISSTATUS,:old.QUITDATE,:old.GREADID,:old.LABOURCODE,SYSDATE,USER); end if; IF UPDATING THEN INSERT INTO ATT_IN.TBL_EMP_PERSONAL_INFO_HISTORY (EMPCODE,NEW_EMPCODE,EMPID,NEW_EMPID,COMPCARDID,NEW_COMPCARDID,EMPNAMEE,NEW_NAME,JOININGDATE,NEW_JOININGDATE, DESIGNATIONID,NEW_DESIGNATIONID,EMPLOYEE_TYPE,NEW_EMPLOYEE_TYPE,EMPLOYEE_CATEGORY,NEW_EMPLOYEE_CATEGORY, RY,

	LABOURCODE,NEW_LABOURCODE,BD_ID,NEW_BD_ID,PLANT_ID, NEW_PLANT_ID,DEPTID,NEW_DEPTID,SHIFTID, NEW_SHIFTID,UPDATE_DATE,UPDATE_USER) VALUES(:old.EMPCODE,:NEW.EMPCODE,:old.EMPID,:old.EMPID,:ol d.COMPCARDID,:NEW.COMPCARDID,:old.EMPNAMEE,:NEW.EMPNA MEE, :old.JOININGDATE,:NEW.JOININGDATE,:old.DESIGNATIONID,:NE W.DESIGNATIONID,:old.EMPLOYEE_TYPE,:NEW.EMPLOYEE_TYPE,: old.EMPLOYEMENT_CATEGORY, :NEW.EMPLOYEMENT_CATEGORY,:old.LABOURCODE,:NEW.LABOU RCODE,:OLD.BD_ID,:NEW.BD_ID,:old.PLANT_ID,:NEW.PLANT_ID,: OLD.DEPTID, :NEW.DEPTID,:old.SHIFTID,:NEW.SHIFTID,SYSDATE,USER); END IF; END;
--	--

### **TRIGGER: T\_COMPFINYID\_INCR**

Used Table Name: TBL\_FINANCIAL\_YEAR

Description	Trigger Body
CREATE OR REPLACE TRIGGER "ATT_IN".t_COMPFINYID_incr before insert ON TBL_FINANCIAL_YEAR for each row	WHEN ( new.COMPFINYID is null ) begin select seq_COMPFINYID.Nextval into :New.COMPFINYID from dual; end;

### **TRIGGER: T\_COMPID\_INCR**

Used Table Name: TBL\_COMPANY\_INFO

Description	Trigger Body
CREATE OR REPLACE TRIGGER "ATT_IN".t_COMPID_incr before insert ON TBL_COMPANY_INFO for each row	WHEN ( new.COMPID is null ) begin select seq_COMPID.Nextval into :New.COMPID from dual; end;

### **TRIGGER: T\_DEPTIDINCREMENT**

Used Table Name: TBL\_DEPARTMENT

Description	Trigger Body
CREATE OR REPLACE TRIGGER "ATT_IN".t_DeptIdIncrement before insert ON tbl_Department for each row	WHEN ( new.DeptId is null ) begin select SEQ_DEPTID.Nextval into :New.DeptId from dual; end;



**TRIGGER: T\_EMP\_OT\_APPROVE\_HISTORY**

Used Table Name: T\_EMP\_OT\_APPROVE\_HISTORY

Description	Trigger Body
CREATE OR REPLACE TRIGGER ATT_IN.T_EMP_OT_APPROVE_HIS TORY AFTER UPDATE OR DELETE ON ATT_IN.TBL_EMP_OT_APPROVE REFERENCING OLD AS OLD NEW AS NEW FOR EACH ROW	BEGIN IF DELETING THEN INSERT INTO ATT_IN.TBL_EMP_OT_APPROVE_HISTORY VALUES(:OLD.EMPID,:OLD.OTDATE,:OLD.OTAPPROVE,:OLD.CREAT EDATE,:OLD.CREATEBY,:OLD.UPDATEBY,:OLD.UPDATEDATE,USER, SYSDATE,:OLD.CHECK_YN);  END IF; END;

**TRIGGER: T\_FORM\_ID\_INCR**

Used Table Name: TBL\_FORE

Description	Trigger Body
CREATE OR REPLACE TRIGGER "ATT_IN".t_FORM_ID_incr before insert ON TBL_FORM for each row	WHEN ( new.FORMID is null ) begin select SEQ_FORM_ID.Nextval into :New.FORMID from dual; end;

**TRIGGER: T\_GOVTHOLIID\_INCR**

Used Table Name: TBL\_GOVT\_HOLIDAY\_LIST

Description	Trigger Body
CREATE OR REPLACE TRIGGER "ATT_IN".t_GOVTHOLIID_incr before insert ON TBL_GOVT_HOLIDAY_LIST for each row	WHEN ( new.GOVTHOLIID is null ) begin select seq_GOVTHOLIID.Nextval into :New.GOVTHOLIID from dual; end;

**TRIGGER: T\_LEAVEAPPLYID\_INCR**

Used Table Name: TBL\_LEAVE\_APPLY

Description	Trigger Body
CREATE OR REPLACE TRIGGER "ATT_IN".t_LEAVEAPPLYID_incr before insert ON TBL_LEAVE_APPLY for each row	WHEN ( new.LEAVEAPPLYID is null ) begin select seq_LEAVEAPPLYID.Nextval into :New.LEAVEAPPLYID from dual; end;

**TRIGGER: T\_LEAVEYPEID\_INCR**

Used Table Name: TBL\_LEAVE\_TYPE

Description	Trigger Body
CREATE OR REPLACE TRIGGER "ATT_IN".t_LEAVEYPEID_incr before insert ON TBL_LEAVE_TYPE for each row	WHEN ( new.LEAVEYPEID is null ) begin select seq_LEAVEYPEID.Nextval into :New.LEAVEYPEID from dual; end;

**TRIGGER: T\_MACHINEID\_INCR**

Used Table Name: TBL\_MACHINE\_INFO

Description	Trigger Body
CREATE OR REPLACE TRIGGER "ATT_IN".t_MACHINEID_incr before insert ON TBL_MACHINENAME for each row	WHEN ( new.MACHINEID is null ) begin select seq_MACHINEID.Nextval into :New.MACHINEID from dual; end;

**TRIGGER: T\_OSDID\_INCR**

Used Table Name: TBL\_OSD\_SETUP

Description	Trigger Body
CREATE OR REPLACE TRIGGER "ATT_IN".t_OSDID_incr before insert ON TBL_OSD_SETUP for each row	WHEN ( new.OSDID is null ) begin select seq_OSDID.Nextval into :New.OSDID from dual; end;

**TRIGGER: T\_SEASONID\_INCR**

Used Table Name: TBL\_SEASON\_SETUP

Description	Trigger Body
CREATE OR REPLACE TRIGGER "ATT_IN".t_SEASONID_incr before insert ON TBL_SEASON_SETUP for each row	WHEN ( new.SEASONID is null ) begin select seq_SEASONID.Nextval into :New.SEASONID from dual; end;

**TRIGGER: T\_SHIFTID\_INCR**

Used Table Name: tbl\_ShiftNameSettings

Description	Trigger Body
CREATE OR REPLACE TRIGGER "ATT_IN".t_ShiftId_incr before insert ON tbl_ShiftNameSettings for each row	WHEN ( new.ShiftId is null ) begin select seq_ShiftId.Nextval into :New.ShiftId from dual; end;

### **TRIGGER: T\_SHIFTSETUPID\_INCR**

Used Table Name: tbl\_shift\_setup

Description	Trigger Body
CREATE OR REPLACE TRIGGER "ATT_IN".t_SHIFTSETUPID_incr before insert ON tbl_shift_setup for each row	WHEN ( new.SHIFTSETUPID is null ) begin select seq_SHIFTSETUPID.Nextval into :New.SHIFTSETUPID from dual; end;

### **TRIGGER: T\_USER\_TYPE**

Used Table Name: TBL\_USER

Description	Trigger
CREATE OR REPLACE TRIGGER ATT_IN.T_USER_HISTORY AFTER UPDATE OR DELETE ON ATT_IN.TBL_USER REFERENCING OLD AS OLD NEW AS NEW FOR EACH ROW	BEGIN IF UPDATING THEN  INSERT INTO ATT_IN.TBL_USER_HISTORY(NEW_EMPID,NEW_USERNAME,NE W_USERLOGID,NEW_USERLOGPASSWORD,NEW_USERTYPENAM E,NEW_DEPTID,NEW_AUTH_DEPTID,NEW_PLANT_ID, NEW_UPDATEBY,NEW_UPDATEDATE) VALUES(:NEW.EMPID,:NEW.USERNAME,:NEW.USERLOGID,:NEW. USERLOGPASSWORD,:NEW.USERTYPENAME,:NEW.DEPTID,:NEW .AUTH_DEPTID,:NEW.PLANT_ID, USER,SYSDATE); END IF ; IF DELETING THEN INSERT INTO ATT_IN.TBL_USER_HISTORY(USERID,USERNAME,USERTYPENAM E,USERLOGID,USERLOGPASSWORD,ENTRYBY,ENTRYDATE,UPDA TEBY,UPDATEDATE,ACTIVESTATUS,INACTIVEDATE,EMPID, EMPCODE,ROLEID,USERCATEGORY,DEPTID,AUTH_DEPTID,PLAN T_ID,DELETEBY,DELETE_DATE) VALUES(:OLD.USERID,:OLD.USERNAME,:OLD.USERTYPENAME,:O LD.USERLOGID,:OLD.USERLOGPASSWORD,:OLD.ENTRYBY,:OLD. ENTRYDATE,:OLD.UPDATEBY,:OLD.UPDATEDATE,:OLD.ACTIVES TATUS,:OLD.INACTIVEDATE,:OLD.EMPID, :OLD.EMPCODE,:OLD.ROLEID,:OLD.USERCATEGORY,:OLD.DEPTI D,:OLD.AUTH_DEPTID,:OLD.PLANT_ID,USER,SYSDATE); END IF;  END;

**TRIGGER: T\_USER\_TYPE**

Used Table Name: TBL\_USERTYPE

Description	Trigger Body
CREATE OR REPLACE TRIGGER "ATT_IN".t_USER_TYPE before insert ON TBL_USERTYPE for each row	WHEN ( new.USERTYPEID is null ) begin select SEQ_USER_TYPE_ID.Nextval into :New.USERTYPEID from dual; end;

**TRIGGER: TBL\_RAW\_DATA\_HISTORY**

Used Table Name: TBL\_RAW\_DATA\_HISTORY

Description	Trigger Body
CREATE OR REPLACE TRIGGER ATT_IN.TBL_RAW_DATA_HISTORY AFTER UPDATE OR DELETE ON ATT_IN.TBL_RAW_DATA REFERENCING OLD AS OLD NEW AS NEW FOR EACH ROW	BEGIN  IF DELETING THEN INSERT INTO ATT_IN.TBL_RAW_DATA_HISTORY(COMPCARDID,PUNCHDATE,PU NCHTIME,LOC_ID,INOUT,OVNMARK,REMARKS,MANUALENTRYTAG, CREATEBY, UPDATEDEMPLOYEE,ROW_ID,MACHINENAME,ENTRYDATE,COMPC ARDID_N,PUNCHDATE_N,PUNCHTIME_N,LOC_ID_N, OVNMARK_N,REMARKS_N,MANUALENTRYTAG_N,MACHINENAME_ N,UPDATE_BY,UPDATE_DATE,DELETE_BY,DELETE_DATE) VALUES(:OLD.COMPCARDID,:OLD.PUNCHDATE,:OLD.PUNCHTIME,: OLD.LOC_ID,:OLD.INOUT,:OLD.OVNMARK,:OLD.REMARKS, :OLD.MANUALENTRYTAG,:OLD.CREATEBY,:OLD.UPDATEDEMPLOY EE,:OLD.ROW_ID,:OLD.MACHINENAME,:OLD.ENTRYDATE,:OLD.CO MPCARDID,:NEW.PUNCHDATE,:NEW.PUNCHTIME,:NEW.LOC_ID, :NEW.OVNMARK,:NEW.REMARKS,:NEW.MANUALENTRYTAG,:NEW. MACHINENAME,"", USER,TO_DATE(TO_CHAR(SYSDATE,'DD/MM/RRRR HH:MI:SS AM'),'DD/MM/RRRR HH:MI:SS AM')); END IF; END;

**TRIGGER: UPDATE\_DELETE\_OCS**

Used Table Name: TBL\_OSD\_SETUP

Description	Trigger Body
CREATE OR REPLACE TRIGGER ATT_IN.UPDATE_DELETE_OCS after delete ON ATT_IN.TBL_OSD_SETUP  referencing old as old new as new  FOR EACH ROW	BEGIN if deleting then INSERT INTO ATT_IN.TBL_OSD_SETUP_HISTORY VALUES(:OLD.OSDID,:OLD.EMPID,:OLD.OSDSTARTDATE,:OLD.OS DENDDATE,:OLD.OSDNUMDAYS,:OLD.OSDLOCATION,:OLD.OSDRE ASON,:OLD.CREATEBY, :OLD.CREATEDATE,:OLD.UPDATEBY,:OLD.UPDATEDATE,:OLD.DEP TID,:OLD.EMPCODE,:OLD.PLANT_ID,USER,SYSDATE); END IF; END;

## E) PROCEDURE

### 1. SP\_COMPANY\_ADD\_UPDATE

```
CREATE OR REPLACE PROCEDURE ATT_IN.sp_Company_Add_Update
(
I_CompId IN int,
I_CompCode IN varchar2,
I_CompNameE IN varchar2,
I_CompNameB IN varchar2,
I_CompAddE IN varchar2,
I_CompAddB IN varchar2,
I_CompPhone IN varchar2,
I_CompFax IN varchar2,
I_CompEmail IN varchar2,
I_CompFinYId IN int,
I_CompWeb IN varchar2,
I_CreateBy IN varchar2,
I_UpdateBy IN varchar2,
I_AddOrUpdate IN varchar2,
I_PreviousCode IN varchar2
)
as
Begin
    declare CompNo int:=0;
    CompCount int :=0;
    CompNameNo int:=0;
    O_retVal int;
Begin
if I_AddOrUpdate='Saved' Then
Select count(*) into CompCount from tbl_Company_Info;
if CompCount <1 Then
Select count(CompId) into CompNo from tbl_Company_Info where CompCode=I_CompCode and
CompFinYId = I_CompFinYId;
Select count(CompId) into CompNameNo from tbl_Company_Info where CompNameE=I_CompNameE and
CompFinYId = I_CompFinYId;
if CompNo=0 and CompNameNo = 0 then
Insert Into
tbl_Company_Info(CompFinYId,CompCode,CompNameE,CompNameB,CompAddE,CompAddB,CompPhone,
CompFax,CompEmail,CompWeb,CreateBy,CreateDate,UpdateBy,UpdateDate)
values(I_CompFinYId,I_CompCode,I_CompNameE,I_CompNameB,I_CompAddE,I_CompAddB,I_CompPhone,
I_CompFax,I_CompEmail,I_CompWeb,I_CreateBy,SYSDATE,I_UpdateBy,SYSDATE);
Commit;
O_retVal := 0;
elsif CompNo=0 and CompNameNo <> 0 Then
O_retVal := 1;
```

```

elseif CompNo <> 0 and CompNameNo = 0 Then
O_retVal := 2;
elseif CompNo<>0 and CompNameNo <> 0 then
O_retVal := 3;
End If;
ELSE
O_retVal := 4;
END IF;
else
update tbl_Company_Info
set CompFinYId=I_CompFinYId,
CompCode=I_CompCode,
CompNameE=I_CompNameE,
CompNameB=I_CompNameB,
CompAddE=I_CompAddE,
CompAddB=I_CompAddB,
CompPhone=I_CompPhone,
CompFax=I_CompFax,
CompEmail=I_CompEmail,
CompWeb=I_CompWeb,
UpdateBy=I_UpdateBy,
UpdateDate= SYSDATE
where CompId=I_CompId;
Commit;
O_retVal := 0;
end if;
End;

```

#### EXCEPTION

```

WHEN OTHERS THEN
RAISE_APPLICATION_ERROR (-20001,
    I_CompFinYId || ':$:' ||
    I_CompCode || ':$:' ||
    I_CompNameE || ':$:' ||
    I_CompNameB || ':$:' ||
    I_CompAddE || ':$:' ||
    I_CompAddB || ':$:' ||
    I_CompPhone || ':$:' ||
    I_CompFax || ':$:' ||
    I_CompEmail || ':$:' ||
    I_CompWeb || ':$:' ||
    I_CreateBy || ':$:' ||
    SQLERRM, TRUE) ;
End sp_Company_Add_Update;

```

## 2. SP\_DATA\_PROCESS\_BIMAN\_CMIS

```
CREATE OR REPLACE PROCEDURE ATT_IN.sp_Data_Process_Biman_cmis
(
  I_processStartingDate In date,
  I_Criteria_Name In int
)
As
Begin
  Declare
    v_weeklyholiday varchar2(100);
    v_countrowdata int;
    V_SQL_DateCompare varchar2(500);
    O_retVal int :=0;
    V_Buyerdate date;
    V_Remarks varchar2(100);
    V_NightShiftDesc varchar2(40);
    V_Status varchar2(100);
    V_NumberOfPunch int;
    V_SQLEmp varchar2(1999);
    V_GiveResult varchar2(1000);
    V_ParameterDefination varchar2(1000);
    V_SelectSeasonId int;
    V_OTHOURD date;
    V_DeptCode varchar2(500);
    V_EmpId varchar2(20);
    V_HoliDayStatus int;
    V_returnSuccessValue int;
    V_CompCardId varchar2(20);
    V_ShiftId int;
    V_SectId int;
    V_JoiningDate date;
    V_QuitDate date;
    V_OTStatus int;
    V_Punchdate date;
    V_PunchTime date;
    V_PunchTime1 date;
    V_PunchTime2 date;
    V_CurDate date:=To_Date('01-Jan-1900 00:00:00', 'DD/MM/YYYY HH24:MI:SS');
    V_OvNMark int;
    V_dateIn date;
    V_shiftLATE date:=To_Date('01-Jan-1900 00:00:00','DD/MM/YYYY HH24:MI:SS');
    V_dateOut date;
    V_Indate date;
    V_Outdate date;
    V_Late date;
    V_OTHour_Val varchar2(50);
    V_LateHr int;
    V_LateVal varchar2(50);
    V_OTStatusMinutes int;
    V_ShiftBeforOTMin int;
    V_NextDate date;
```

```

v_DESIGNATIONID varchar2(50);
V_LeaveStatus varchar2(50);
V_ProcessType varchar2(5);
V_CountRollingStatus int;
V_CountDataNotFound int;
V_unrestCount int;
V_RollingStatus varchar2(50);
V_RegHour date;
V_BuyerShiftIn date;
V_BuyerShiftOut date;
V_BuyerTime date;
V_BreakIn date;
V_BreakOut date;
V_BreakLate date;
V_ShiftOut date;
V_ShiftIn date;
exists_wkly_holiday_compWise int;
exists_wkly_holiday_Shift int;
exists_govt_holiday int;
exists_SeasonId int;
exists_OSD int;
exists_Row_C int;
exists_Row_Leave int;
V_InTime Date;
V_OutTime Date;
V_date_increment date;
V_retPunchTime date;
V_MinValue int;
V_punchTimeMin int;
V_shiftOutMin int;
V_retPTime_min int;
V_OTFraction int;
V_Sex int;
V_Grade varchar2(50);
V_LabourCode int;
V_Labour_OT_Status varchar2(20);
V_OTTYPE varchar2(20);
V_OTAPPROVE varchar2(20):="";
V_SHIFTTYPE varchar2(20);
V_EMPLOYMENT_CATEGORY varchar2(50);
V_EMPLOYEE_TYPE varchar2(50);
V_TDay_OutTime date;
V_CheckFloatShift int;
V_FloatShiftId int;
w_status int;
V_EMPCODE varchar2(20);
v_WORKDAYLOG int;
v_OFFDAYLOG int;
v_shiftTypeCount int:=0;
v_DEPTID int;
v_MEALALLOWANCE varchar2(10);
O_OT_Hour varchar(10);
o_remarks varchar(50);
o_ACTUAL_OT_HOUR varchar2(20);
v_assign_Shift_exist int;

```



```

v_shiftname varchar(50);
v_emp_id_exist int;
v_ot_hour_ot_entry_tbl varchar2(50);
v_r_status_ot_entry_tbl varchar2(50);
v_late_ot_entry_tbl nvarchar2(50);
v_s_rest_ot_entry_tbl nvarchar2(50);
ot_count int;
ot_h_int int;
ot_h_int_spe int;
ot_m_int int;
ot_m_int_spe int;
ot_hour_hh24 date;
s_r_hour_hh24 date;
ot_hour_mm int;
s_r_hour_mm int;
t_ot_min int;
s_rest_h_int int;
s_rest_m_int int;
total_ot_hour nvarchar2(30);
two_shift_ot int;
ocs_check int;
chk_duty_pattern nvarchar2(30);
v_call_sp_short_rest int;
chk_p_exist int;
ecode NUMBER;
emesg VARCHAR2(400);
begin
begin
select distinct e.EmpId,e.EMPCODE,e.CompCardId,e.ShiftId,e.SectId,e.DEPTID,
e.DESIGNATIONID,e.JoiningDate,e.QUITDATE,e.OVERTIME_APPLICABLE OTStatus,
e.EMPLOYEE_CATEGORY,e.EMPLOYEE_TYPE into V_EmpId,v_EmpCode, V_CompCardId, V_ShiftId,
V_SectId,v_DEPTID,v_DESIGNATIONID, V_JoiningDate,V_QuitDate,
V_OTStatus,V_EMPLOYEMENT_CATEGORY,V_EMPLOYEE_TYPE
from tbl_Emp_Personal_Info e
where EmpEnDisStatus = 1
and e.EmpId = I_Criteria_Name ;
exception
when no_data_found then
null;
end;

select count(*) into v_assign_Shift_exist from att_in.TBL_OT_ENTRY where empid =I_Criteria_Name
and OT_OR_REG_DATE =I_processStartingDate;

if v_assign_Shift_exist>0 then
select R_SHIFT_ID into V_ShiftId from att_in.TBL_OT_ENTRY
where empid =I_Criteria_Name and OT_OR_REG_DATE = I_processStartingDate;
else
select SHIFTID into V_ShiftId from att_in.TBL_EMP_PERSONAL_INFO
where empid =I_Criteria_Name ;
end if;
select Count(SHIFTTYPE) into v_shiftTypeCount from att_in.tbl_ShiftNameSettings where
ShiftId=V_ShiftId;
if(v_shiftTypeCount>0) then
begin

```

```

select Upper(nvl(SHIFTTYPE,'')),Upper(SHIFTNAME) into V_SHIFTTYPE, v_shiftname from
att_in.tbl_ShiftNameSettings
where ShiftId=V_ShiftId;
exception
when no_data_found then
null;
end;
else
V_SHIFTTYPE :=0;
end if;
If UPPER(V_SHIFTTYPE)=upper('4-SHIFT') then
ATT_IN.sp_Four_Shift_2014_cmis(I_Criteria_Name , I_processStartingDate);
commit;
ATT_IN.sp_special_ot_calculation(I_Criteria_Name , I_processStartingDate);
commit;
ATT_IN.sp_ot_calculation(I_Criteria_Name , I_processStartingDate);
att_in.SP_SHORT_REST_CALCULATION(I_Criteria_Name , I_processStartingDate);
commit;
INSERT INTO tbl_Processed_Data (
EmpId , PunchDate , TimeIn , TimeOut , ShiftIn , ShiftOut ,
BreakIn , BreakOut , BreakLate , Late , ShiftId , RegHour ,
OTHour , Status , NumPunch , NightShiftDesc ,OTShiftDesc ,
BuyerShiftIn,BuyerShiftOut , BuyerTime, Remarks,MEALALLOWANCE )
VALUES(V_EmpId,I_processStartingDate,V_InTime,V_OutTime,V_ShiftIn,V_ShiftOut,
V_BreakIn,V_BreakOut,V_BreakLate,V_LateVal,V_ShiftId,V_RegHour,
'000:00:00',V_Status,V_NumberOfPunch,V_NightShiftDesc,"
V_BuyerShiftIn,V_BuyerShiftOut,V_BuyerTime,V_Remarks,v_MEALALLOWANCE);
Commit;
ELSIF UPPER(V_SHIFTTYPE)=upper('2-SHIFT') then

ATT_IN.sp_Two_Shift_OT_test_tania(I_Criteria_Name , I_processStartingDate);
ATT_IN.SP_Night_shift_1(I_Criteria_Name , I_processStartingDate);
att_in.sp_Night_shift_2(I_Criteria_Name , I_processStartingDate);
commit;
att_in.SP_SHORT_REST_CALCULATION(I_Criteria_Name , I_processStartingDate);

INSERT INTO tbl_Processed_Data (EmpId , PunchDate , TimeIn , TimeOut , ShiftIn , ShiftOut
,BreakIn , BreakOut , BreakLate , Late , ShiftId , RegHour , OTHour , Status , NumPunch ,
NightShiftDesc ,OTShiftDesc ,
BuyerShiftIn,BuyerShiftOut , BuyerTime, Remarks,MEALALLOWANCE )
VALUES(V_EmpId,I_processStartingDate,V_InTime,V_OutTime,V_ShiftIn,V_ShiftOut,
V_BreakIn,V_BreakOut,V_BreakLate,V_LateVal,V_ShiftId,V_RegHour,
'000:00:00',V_Status,V_NumberOfPunch,V_NightShiftDesc,"
V_BuyerShiftIn,V_BuyerShiftOut,V_BuyerTime,V_Remarks,v_MEALALLOWANCE);
Commit;

elsif UPPER(V_SHIFTTYPE)=upper('General Shift') then

ATT_IN.sp_general_shift(V_EMPID, I_processStartingDate);
If UPPER(V_SHIFTTYPE) not in (upper('4-SHIFT'),upper('2-Shift')) and V_OTStatus=1 AND
V_OTAPPROVE>0 and V_EMPLOYEE_TYPE in('P','G')
and V_EMPLOYEMENT_CATEGORY='Permanent' then
ATT_IN.sp_special_ot_general(V_EMPID, I_processStartingDate);
end if;
end if;

```

```

if upper(V_SHIFTTYPE) in( upper('4-Shift'),upper('2-Shift')) then
select nvl(SHORT_REST_HOUR,'0'),nvl(OT_HOUR,0),LATE_ON_REG_SHIFT,REG_STATUS
into
v_s_rest_ot_entry_tbl,v_ot_hour_ot_entry_tbl,v_late_ot_entry_tbl,
v_r_status_ot_entry_tbl
from att_in.TBL_OT_ENTRY
where empid=V_EMPID and OT_OR_REG_DATE =
to_date(to_char(I_processStartingDate,'DD/MM/YYYY'),'DD/MM/YYYY');
v_MEALALLOWANCE:=FN_Mealallowance(V_EmpId,V_OutTime,V_ShiftId,
v_ot_hour_ot_entry_tbl);

if substr(v_ot_hour_ot_entry_tbl,1,2)>'23' then
ot_h_int_spe:= substr(v_ot_hour_ot_entry_tbl,1,2);
ot_h_int_spe:= to_number(ot_h_int_spe)*60; ot_m_int_spe:=substr(v_ot_hour_ot_entry_tbl,4,2);
ot_m_int_spe:= to_number( ot_m_int_spe);
else
ot_hour_hh24 := to_date(to_char(v_ot_hour_ot_entry_tbl),'HH24:MI:SS');
ot_hour_mm := to_number(to_char(ot_hour_hh24,'MI'));
end if;

s_r_hour_hh24 := to_date(to_char(v_s_rest_ot_entry_tbl),'HH24:MI:SS');
s_r_hour_mm := to_number(to_char(s_r_hour_hh24,'MI'));
ot_h_int := to_number(to_char(ot_hour_hh24,'HH24')) * 60;
s_rest_h_int := to_number(to_char(s_r_hour_hh24,'HH24')) * 60;
if substr(v_ot_hour_ot_entry_tbl,1,2)>'23' then
t_ot_min :=( ot_h_int_spe+ot_m_int_spe+s_r_hour_mm+s_rest_h_int);
else
t_ot_min := (ot_hour_mm + s_r_hour_mm) + (ot_h_int + s_rest_h_int);
end if;
ot_hour_mm := (t_ot_min mod 60);
ot_h_int := (t_ot_min - ot_hour_mm) / 60;
if ot_h_int > 9 then
if ot_hour_mm >= 0 and ot_hour_mm < 10 then
total_ot_hour := to_char(ot_h_int)||':0'||to_char(ot_hour_mm)||':00';
elsif ot_hour_mm >=10 and ot_hour_mm <=59 then
total_ot_hour := to_char(ot_h_int)||':'||to_char(ot_hour_mm)||':00';
end if;
elsif ot_h_int <=9 then
if ot_hour_mm >= 0 and ot_hour_mm < 10 then
total_ot_hour:=to_char('0'||to_char(ot_h_int)||':0'||to_char(ot_hour_mm)
||':00');
elsif ot_hour_mm >=10 and ot_hour_mm <=59 then
total_ot_hour:=to_char('0'||to_char(ot_h_int)||':'||to_char(ot_hour_mm)
||':00');
end if;
end if;

if total_ot_hour is not null then
V_OTHour_Val:=total_ot_hour;
else
V_OTHour_Val:='00:00:00';
end if;

update att_in.TBL_PROCESSED_DATA

```

```

set MEALALLOWANCE=v_MEALALLOWANCE,
OTHOURL =V_OTHour_Val,
STATUS = v_r_status_ot_entry_tbl,
LATE = v_late_ot_entry_tbl
where EMPID = V_EMPID and
PUNCHDATE =to_date(to_char(I_processStartingDate,'DD/MM/YYYY'),'DD/MM/YYYY');
commit
end if;
end;
end;

```

### 3. SP\_DESIGNATION\_ADD\_UPDATE

```

CREATE OR REPLACE PROCEDURE ATT_IN.sp_Designation_Add_Update
(
    I_DesignationId IN Int,
    I_DesignNameE In Varchar2,
    I_Grade In varchar2,
    I_Priority In Int,

    I_CreateBy IN  varchar2,
    I_UpdateBy IN  varchar2,
    I_AddOrUpdate IN  varchar2,
    O_refcursor_retVal OUT SYS_REFCURSOR
)
as
Begin
    --Add
    declare  count_Bd_Name int:=0;
    O_retVal int;
    Begin
        if I_AddOrUpdate='Saved' Then
            --      Select count(BD_NAME) into count_Bd_Name from
TBL_BUSINESSSDIVISION
            --      where BD_NAME=I_BDevisionName;

            if count_Bd_Name=0
            Then
                --Not Duplicate
                Insert Into TBL_DESIGNATION_SETUP
                (
                    DESIGNATIONID,
                    DesigNameE,
                    Grade,
                    Priority,
                    CREATEBY,
                    UPDATEBY,
                    CreateDate,
                    UpdateDate

                )
                values
                (
                    Seq_Designation.nextval,
                    I_DesignNameE,
                    I_Grade,
                    I_Priority,

```

```

        I_CreateBy,
        I_UpdateBy,
        SYSDATE,
        SYSDATE
    );
    Commit;
    O_retVal := 0;
else
    O_retVal := 1;

End If;
else
    update TBL_DESIGNATION_SETUP
    set DesigNameE=I_DesigNameE,
        Grade=I_Grade,
        Priority=I_Priority,
        CREATEBY=I_CreateBy,
        UPDATEBY=I_UpdateBy,
        CreateDate=sysdate,
        UpdateDate=sysdate

        where DESIGNATIONID=I_DesignationId;

    Commit;
    O_retVal := 0;
end if;
open O_refcursor_retVal for select O_retVal from dual;
End;

EXCEPTION
    WHEN OTHERS THEN
        RAISE_APPLICATION_ERROR (-20001,
            I_DesigNameE || ' :$:' ||
            I_Grade || ' :$:' ||
            I_Priority || ' :$:' ||

            I_CreateBy || ' :$:' ||
            I_UpdateBy || ' :$:' ||

            SQLERRM, TRUE) ;

End sp_Designation_Add_Update;
/

```

#### 4. SP\_EMPLOYEE\_ADD\_UPDATE\_BIMAN

```

CREATE OR REPLACE PROCEDURE ATT_IN.sp_Employee_Add_Update_Biman
(
    I_EMPID                In    INTEGER,
    I_EMPCODE              In    VARCHAR2,
    I_COMPCARDID           In    VARCHAR2,
    I_EMPNAMEE             In    VARCHAR2,
    I_JOININGDATE           In    DATE,
    I_COMPID               In    INTEGER,
    I_BD_ID                In    INTEGER,
    I_PLANT_ID             In    INTEGER,
    I_DEPTID               In    INTEGER,
    I_DESIGNATIONID        IN    VARCHAR2,
    I_SHIFTID              IN    INTEGER,

```

```

        I_EmpEndDisStatus    in integer,
        I_LabourCode         in integer,
        I_EmpType            in VARCHAR2,
        I_EmpStatus          in VARCHAR2

)
as
Begin
    --Add
    declare
        EmpCount int:=0;
        empID_For_Update int;
        --CompNameNo int:=0;
        --NationalIDCount int :=0;
        --O_retVal int:=-1;

    Begin
        --if I_AddOrUpdate='Saved' Then

            Select count(*) into EmpCount from att_in.TBL_EMP_PERSONAL_INFO
where EmpCode=I_EmpCode;
            --Select count(*) into CompNameNo from TBL_EMP_PERSONAL_INFO
where COMPCARDID=I_COMPCARDID;
            --select count(*) into NationalIDCount from
TBL_EMP_PERSONAL_INFO where NATIONALIDCARD=I_NATIONALIDCARD;
            if EmpCount=0 then --and CompNameNo = 0 and NationalIDCount=0
then

                --Not Duplicate
                Insert Into att_in.TBL_EMP_PERSONAL_INFO
                (
                    EMPID,
                    EMPCODE,
                    COMPCARDID,
                    EMPNAMEEE,
                    JOININGDATE,
                    COMPID,
                    BD_ID,
                    PLANT_ID,
                    DEPTID,
                    DESIGNATIONID,
                    SHIFTID,
                    EMPENDISSTATUS,
                    LABOURCODE,
                    EMPLOYEE_TYPE,
                    EMPLOYEE_CATEGORY

                )
                values
                (
                    I_EMPID,
                    I_EMPCODE,
                    I_COMPCARDID,
                    I_EMPNAMEEE,
                    I_JOININGDATE,
                    I_COMPID,
                    I_BD_ID,
                    I_PLANT_ID,
                    I_DEPTID,
                    I_DESIGNATIONID,

```

```

        I_SHIFTID,
        I_EmpEnDisStatus,
        I_LabourCode,
        I_EmpType,
        I_EmpStatus
    );
    Commit;
    --      O_retVal := 0;
    --
    -- Employee Id duplicate
    --
    elsif EmpCount>0 Then
        Select empid into empID_For_Update from
att_in.TBL_EMP_PERSONAL_INFO where EmpCode=I_EmpCode;
        update att_in.TBL_EMP_PERSONAL_INFO
        set EmpCode=I_EmpCode,
        COMPCARDID=I_COMPCARDID,
        EMPNAMEE=I_EMPNAMEE,
        JOININGDATE=I_JOININGDATE,
        COMPID=I_COMPID,
        BD_ID=I_BD_ID,
        PLANT_ID=I_PLANT_ID,
        DEPTID=I_DEPTID,
        DESIGNATIONID=I_DESIGNATIONID,
        SHIFTID=I_SHIFTID,
        EmpEnDisStatus=I_EmpEnDisStatus,
        LABOURCODE=I_LabourCode,
        EMPLOYEE_TYPE=I_EmpType,
        EMPLOYEE_CATEGORY=I_EmpStatus
        where EmpId=I_EmpId;
        Commit;

        End If;

    --else
        --O_retVal := 0;

    --end if;
    --open O_refcursor_Return for select O_retVal from dual;
End;

EXCEPTION
    WHEN OTHERS THEN
        RAISE_APPLICATION_ERROR (-20001,
            I_EMPCODE || ' :$:' ||
            I_COMPCARDID || ' :$:' ||
            I_EMPNAMEE || ' :$:' ||
            I_JOININGDATE || ' :$:' ||
            I_COMPID || ' :$:' ||
            I_DEPTID || ' :$:' ||
            I_DESIGNATIONID || ' :$:' ||
            I_SHIFTID || ' :$:' ||
            I_EmpType || ' :$:' ||
            I_EmpStatus || ' :$:' ||
            SQLERRM, TRUE) ;

        --null;
    End sp_Employee_Add_Update_Biman;
/

```

## 5. SP\_FINANTIALYEAR\_ADD\_UPDATE

```
CREATE OR REPLACE PROCEDURE ATT_IN.sp_FinantialYear_Add_Update
(
    I_CompFinYId IN INT,
    I_CompFinYName IN varchar2,
    I_CompFinStartDate IN Date,
    I_CompFinEndDate IN Date,
    I_CreateBy IN varchar2,
    I_UpdateBy IN varchar2,
    I_AddOrUpdate IN varchar2
    --O_refcursor_retVal OUT SYS_REFCURSOR
)
as
Begin
    --Add
    declare count_Bd_Name int:=0;
    O_retVal int;
    Begin
        if I_AddOrUpdate='Saved' Then
            Select count(*) into count_Bd_Name from tbl_Financial_Year where
COMPFINYNNAME=I_CompFinYName;

            if count_Bd_Name=0
            Then
                --Not Duplicate
                Insert Into tbl_Financial_Year
                (
                    COMPFINYNNAME,
                    CompFinStartDate,
                    CompFinEndDate,
                    CreateBy,
                    UpdateBy,
                    createDate,
                    updatedate
                )
                values
                (
                    I_CompFinYName,
                    I_CompFinStartDate,
                    I_CompFinEndDate,
                    I_CreateBy,
                    I_UpdateBy,
                    sysdate,
                    sysdate
                );
                Commit;
                O_retVal := 0;
            else
                O_retVal := 1;
            End If;
        else
            update tbl_Financial_Year
            set CompFinYName=I_CompFinYName,
                CompFinStartDate=I_CompFinStartDate,
                CompFinEndDate= I_CompFinEndDate,
```



```

        CREATEBY=I_CreateBy,
        UPDATEBY=I_UpdateBy
    where CompFinYId=I_CompFinYId;
    Commit;
    O_retVal := 0;
end if;
-- open O_refcursor_retVal for select O_retVal from dual;
End;

EXCEPTION
    WHEN OTHERS THEN
        RAISE_APPLICATION_ERROR (-20001,
            I_CompFinYName || ' :$:' ||
            I_CompFinStartDate || ' :$:' ||
            I_CompFinEndDate || ' :$:' ||

            SQLERRM, TRUE) ;

End sp_FinantialYear_Add_Update;
/

```

## 6. SP\_FORM\_ADD\_UPDATE

```

CREATE OR REPLACE PROCEDURE ATT_IN.sp_Form_Add_Update
(
    I_Form_Id int,
    I_Form_Constant varchar2,
    I_Form_Description varchar2,
    I_Parent_Menu_Item varchar2 ,
    I_Child_Menu_Item varchar2 ,
    I_EntryBy varchar2
    --I_UpdateBy varchar2,
    --I_addOrUpdate varchar2,
    --I_ActiveStatus int,
    --I_InactiveDate date,
    --I_EmpId int,
    --I_EmpCode int
    --O_RetVal OUT SYS_REFCURSOR
)
as
BEGIN
    Declare V_userNameCount int;V_DuplicateFormNameCount int;V_returnVal
    int;V_Form_ID int;
    BEGIN
        V_userNameCount :=0;
        V_returnVal :=0;
        V_DuplicateFormNameCount :=0;
        --if I_addOrUpdate='Saved'
        --Then
            --select count(*) into V_userNameCount from tbl_User;
            --if V_userNameCount<6
            --Then
                select count(FORMCONSTANT) into V_DuplicateFormNameCount
            from tbl_form where FORMCONSTANT=I_Form_Constant;
            if V_DuplicateFormNameCount=0
            Then
                INSERT INTO tbl_form
                (

```

```

        FORMCONSTANT,
        FORMDESCRIPTION,
        PARENTMENUITEM,
        CHILDMENUITEM,
        CREATEBY,
        CREATEDATE
        --ActiveStatus,
        --InactiveDate,
        --EmpId,
        --EMPCODE
        --RoleId
    )
VALUES
(
    I_Form_Constant,
    I_Form_Description,
    I_Parent_Menu_Item,
    I_Child_Menu_Item,
    I_EntryBy,
    sysdate
    --I_ActiveStatus,
    --I_InactiveDate,
    --I_EmpId,
    --I_EmpCode
    --0
);
commit;

Else
    --V_returnVal :=1;
    select FORMID into V_Form_ID from tbl_form where
FORMCONSTANT=I_Form_Constant;
    update tbl_form
    set FORMCONSTANT=I_Form_Constant,
    FORMDESCRIPTION=I_Form_Description,
    PARENTMENUITEM=I_Parent_Menu_Item,
    CHILDMENUITEM=I_Child_Menu_Item,
    --ActiveStatus=I_ActiveStatus,
    UPDATEBY=I_EntryBy,
    UpdateDate=sysdate
    --EmpId=I_EmpId,
    --EMPCODE=I_EmpCode
    where FORMID=V_Form_ID;
    Commit;

    END IF;
--ELSE
--    V_returnVal :=2;
--END IF;

END;
END;
/

```

## 7. SP\_FOUR\_SHIFT\_2014\_CMIS

```
CREATE OR REPLACE PROCEDURE ATT_IN.sp_Four_Shift_2014_cmis
(
  I_EmpId In int,
  I_OT_Date In date
)
As
Begin
  declare
    v_shift_in_time_except date;
    v_dreg_hour date;
    V_REG_STATUS varchar2(30);
    diff_time_between_two_shift int;
    previous_shift varchar2(30);
    current_shift varchar2(30);
    v_check_extra_ot_log int;
    I_Sts varchar2(15);
    v_diff_between_extra_ot int;
    v_ot_in date;
    v_ot_out date;
    v_extra_in date;
    v_extra_out date;
    v_reg_ot_in date;
    v_reg_ot_out date;
    v_shift_extra_ot_log int:=0;
    v_shift_extra_ot_in date;
    v_shift_extra_ot_out date;
    v_shift_out_extra_OT date;
    v_pre_day_after_ot_log int;
    v_for_pick_after_ot date;
    v_shift_after_ot_in date;
    v_shift_after_ot_out date;
    v_shift_before_ot_in date;
    v_shift_before_ot_out date;
    v_regDutyStatusInA int:=0;
    v_regDutyStatusOutA int:=0;
    v_regDutyStatusInC int:=0;
    v_regDutyStatusOutC int:=0;
    v_shift_in_s_range date;
    v_shift_in_e_range date;
    v_shift_in_s_range_BOT date;
    v_shift_in_e_range_BOT date;
    v_shift_out_s_range date;
    v_shift_out_e_range date;
    v_shift_out_s_range_AOT date;
    v_shift_out_e_range_AOT date;
    v_shift_in_time date;
    v_shift_out_time date;
    diff_in_out_time int;
    var_ot_calculation int;
    var_late_calculation int;
    var_late_calculation_final int;
    ot_min_late int;
    ot_hour_late int;
```

```

total_ot_hour_late nvarchar2(20);
v_shift_in_time_c date;
v_shift_out_time_c date;
v_emp_exist_in_tbl_ot_entry int;
v_emp_code nvarchar2(20);
v_comp_cardid nvarchar2(20);
v_dept_id int;
v_shiftname nvarchar2(20);
v_shiftin date;
v_shiftout date;
v_shiftin_s_range date;
I_Prev_Shift_Before_D nvarchar2(20);
leave_osd_counter int;
leave_counter int;
leave_id int;
ITypeName nvarchar2(100);
osd_counter int;
b_ot int :=0;
a_ot int :=0;
I_counder_D int;
in_time_C date;
out_time_C date;
in_punch_C date;
ot_inpunch_C date;
ot_outpunch_C date;
set_reg_out_time date;
in_time_D date;
out_time_D date;
in_time_D_2nd_min date;
diff_mind_maxd int;
diff_in_out_C int;
diff_2min_1min_D int;
chk_ot_log int;
var_diff_dnight int;
d1_outtime nvarchar2(50);
chk_C2nd_ot_in date;
chk_C2nd_ot_out date;
I_CounterW int;
I_Sts_W nvarchar2(5);
I_WRINTIME nvarchar2(50);
I_WROUTTIME nvarchar2(50);
I_WOINTIME nvarchar2(50);
I_WOOUTTIME nvarchar2(50);
I_WIN date;
I_WOUT date;
var_diff_wnight int;
d2_outtime date;
var_reg_out_punch date;
var_reg_punch date;
v_status nvarchar2(5);
v_rintime date;
v_routtime date;
v_oointime date;
v_oouttime date;
v_check_in_punch int;

```

*v\_check\_out\_punch int;*

*begin*

*SELECT count(\*) into v\_emp\_exist\_in\_tbl\_ot\_entry FROM att\_in.tbl\_ot\_entry*

*WHERE EMPID = I\_EmpId and OT\_OR\_REG\_DATE =*

*to\_date(to\_char(I\_OT\_Date,'DD/MM/RRRR'),'DD/MM/RRRR');*

*if v\_emp\_exist\_in\_tbl\_ot\_entry = 0 then null;*

*else*

*select EMPCODE,to\_char(COMPCARDID),DEPTID into v\_emp\_code,v\_comp\_cardid,v\_dept\_id*

*from att\_in.TBL\_EMP\_PERSONAL\_IN where EMPID = I\_EmpId end if*

*begin*

*select R\_SHIFT into v\_shiftname from att\_in.TBL\_OT\_ENTRY where EMPID = I\_EmpId and*

*OT\_OR\_REG\_DATE=to\_date(to\_char(I\_OT\_Date,'DD/MM/RRRR'),'DD/MM/RRRR');*

*EXCEPTION when too\_many\_rows then null; when no\_data\_found then null;*

*when others then null; end; if v\_shiftname <> '4 Shift-D' then*

*begin select nvl(AFTER\_OT\_LOG,0),R\_SHIFT into*

*v\_pre\_day\_after\_ot\_log,previous\_shift*

*from att\_in.TBL\_OT\_ENTRY where EMPID = I\_EmpId*

*and OT\_OR\_REG\_DATE = to\_date(to\_char(I\_OT\_Date -1,'DD/MM/RRRR'),'DD/MM/RRRR'); EXCEPTION*

*when too\_many\_rows then null; when no\_data\_found then null;*

*when others then null;*

*begin select to\_date(to\_char(R\_SHIFT\_IN,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'),*

*to\_date(to\_char(R\_SHIFT\_OUT,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')*

*into v\_shiftin, v\_shiftout from att\_in.TBL\_OT\_ENTRY where EMPID = I\_EmpId*

*and OT\_OR\_REG\_DATE = to\_date(to\_char(I\_OT\_Date,'DD/MM/RRRR'),'DD/MM/RRRR')*

*and R\_SHIFT = v\_shiftname; EXCEPTION when too\_many\_rows then null;*

*when no\_data\_found then null; when others then*

*null; end;*

*begin*

*select to\_date(to\_char(v\_shiftin - interval '120' MINUTE,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR*

*HH24:MI:SS') into v\_shift\_in\_s\_range*

*from dual; end;*

*begin select to\_date(to\_char(v\_shiftin + interval '119' MINUTE,'DD/MM/RRRR*

*HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')*

*into v\_shift\_in\_e\_range from dual; end;*

*begin select to\_date(to\_char(v\_shiftout - interval '120' MINUTE,'DD/MM/RRRR*

*HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')*

*into v\_shift\_out\_s\_range from dual; end;*

*begin select to\_date(to\_char(v\_shiftout + interval '119' MINUTE,'DD/MM/RRRR*

*HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')*

*into v\_shift\_out\_e\_range from dual;*

*end;*

*begin*

*select to\_date(to\_char(v\_shiftin - interval '601' MINUTE,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR*

*HH24:MI:SS')*

*into v\_shift\_in\_s\_range\_BOT from dual; end;*

*begin select to\_date(to\_char(v\_shiftin - interval '361' MINUTE,'DD/MM/RRRR*

*HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')*

*into v\_shift\_in\_e\_range\_BOT from dual; end;*

*begin*

*select to\_date(to\_char(v\_shiftout + interval '359' MINUTE,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR*

*HH24:MI:SS')*

*into v\_shift\_out\_s\_range\_AOT from dual; end;*

```

begin select to_date(to_char(v_shiftout + interval '720' MINUTE,'DD/MM/RRRR
HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
into v_shift_out_e_range_AOT from dual; end;
begin select to_date(to_char( v_shift_in_s_range - interval '960' MINUTE,'DD/MM/RRRR
HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
into v_shift_out_extra_OT from dual; end;
begin select to_date(to_char(min(PunchTime),'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
into v_shift_in_time
from att_in.TBL_RAW_DATA_CMIS where COMPCARDID = v_comp_cardid
and to_date(to_char(punchtime,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS') between
to_date(to_char(v_shift_in_s_range,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS') and
to_date(to_char(v_shift_in_e_range,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS');
begin select to_date(to_char(R_IN_TIME_PUNCH,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS'), to_date(to_char(R_OUT_TIME_PUNCH,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS'),
to_date(to_char(OT_IN_TIME_PUNCH,'DD/MM/RRRR
HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'), to_date(to_char(OT_OUT_TIME_PUNCH,'DD/MM/RRRR
HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS') into v_rintime,v_routtime,v_overtime,v_overtime from
att_in.TBL_OT_ENTRY
where EMPID = I_EmpId and OT_OR_REG_DATE = to_date(to_char(I_OT_Date -
1,'DD/MM/RRRR'),'DD/MM/RRRR');
EXCEPTION
when too_many_rows then null;
when no_data_found then null;
when others then null; end;
if v_routtime is not null or v_overtime is not null or v_overtime is not null then
if v_shift_in_time is not null and v_overtime = v_shift_in_time or v_routtime =v_shift_in_time then
v_shift_in_time:="";
b_ot := 0; else v_shift_in_time:=v_shift_in_time; end if;end if;
EXCEPTION when too_many_rows then null;
when no_data_found then null; when others then null;end
if v_shift_in_time is null then begin select to_date(to_char(min(PunchTime),'DD/MM/RRRR
HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS') into v_shift_in_time from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = v_comp_cardid and to_date(to_char(punchtime,'DD/MM/RRRR
HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS') between to_date(to_char(v_shift_in_s_range,'DD/MM/RRRR
HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
and to_date(to_char(v_shift_out_s_range,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS');
begin select to_date(to_char(R_IN_TIME_PUNCH,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS'),
to_date(to_char(R_OUT_TIME_PUNCH,'DD/MM/RRRR
HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'),
to_date(to_char(OT_IN_TIME_PUNCH,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'),
to_date(to_char(OT_OUT_TIME_PUNCH,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS') into
v_rintime,v_routtime,v_overtime,v_overtime
from att_in.TBL_OT_ENTRY where EMPID = I_EmpId and OT_OR_REG_DATE =
to_date(to_char(I_OT_Date - 1,'DD/MM/RRRR'),'DD/MM/RRRR');
EXCEPTION when too_many_rows then null; when no_data_found then null;
when others then null; end;
if v_routtime is not null or v_overtime is not null or v_overtime is not null then
if v_shift_in_time is not null and v_overtime = v_shift_in_time or v_routtime =v_shift_in_time then
v_shift_in_time:="";
b_ot := 0;
else
v_shift_in_time:=v_shift_in_time;
--b_ot := 1;
end if;
end if;

```

*EXCEPTION*

*when too\_many\_rows then  
null;  
when no\_data\_found then  
null;  
when others then  
null;  
end;  
end if;*

*if v\_shift\_in\_time is null then  
v\_check\_in\_punch := 0;  
else  
v\_check\_in\_punch := 1;  
end if;*

*if v\_shift\_in\_time is not null then  
v\_regDutyStatusInA :=1;  
else  
v\_regDutyStatusInA :=0;  
end if;*

*if v\_shift\_in\_time is null then*

*v\_rintime:= '';  
v\_routtime:='';  
v\_overtime:='';  
v\_overtime:='';*

*b\_ot := 0;*

*begin*

*select to\_date(to\_char(R\_IN\_TIME\_PUNCH,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'),  
to\_date(to\_char(R\_OUT\_TIME\_PUNCH,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'),  
to\_date(to\_char(OT\_IN\_TIME\_PUNCH,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'),  
to\_date(to\_char(OT\_OUT\_TIME\_PUNCH,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')*

*into  
v\_rintime,v\_routtime,v\_overtime,v\_overtime*

*from att\_in.TBL\_OT\_ENTRY*

*where EMPID = I\_EmpId*

*and OT\_OR\_REG\_DATE = to\_date(to\_char(I\_OT\_Date - 1,'DD/MM/RRRR'),'DD/MM/RRRR') ;*

*EXCEPTION*

*when too\_many\_rows then  
null;  
when no\_data\_found then  
null;  
when others then  
null;  
end;*

*begin*

*select to\_date(to\_char(min(PunchTime),'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')*

*into v\_shift\_in\_time*

*from att\_in.TBL\_RAW\_DATA\_CMIS*

*where COMPCARDID = v\_comp\_cardid*

```

and to_date(to_char(punchtime,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
between to_date (to_char(v_shift_in_s_range_BOT,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS')
and to_date(to_char(v_shift_in_e_range_BOT,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS');
EXCEPTION
when too_many_rows then
null;
when no_data_found then
null;
when others then
null;
end;
if v_shift_in_time is null then
v_check_in_punch := 0;
else
v_check_in_punch := 1;
end if;

if v_routtime is not null or v_overtime is not null or v_overtime is not null then

if v_shift_in_time is not null and v_overtime = v_shift_in_time or v_routtime =v_shift_in_time or
v_overtime =v_shift_in_time then
v_shift_in_time:="";
b_ot := 0;
else
v_shift_in_time:=v_shift_in_time;
b_ot := 1;
end if;
end if;

if v_shift_in_time is not null then
b_ot := 1;
end if;

if v_shift_in_time is null then
b_ot := 0;
select nvl(to_date(to_char(min(PunchTime),'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'), '')
into v_shift_in_time
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = v_comp_cardid
and to_date(to_char(punchtime,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')between
to_date (to_char(v_shift_in_s_range_BOT,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')and
to_date(to_char(v_shift_out_s_range,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS');

if v_shift_in_time is null then
v_check_in_punch := 0;
else
v_check_in_punch := 1;
end if;

if v_routtime is not null and v_overtime is not null and v_overtime is not null then

if v_shift_in_time is not null and v_overtime= v_shift_in_time or v_routtime=v_shift_in_time or v_overtime
=v_shift_in_time then
v_shift_in_time:="";

```



```

b_ot := 0;
else
v_shift_in_time:=v_shift_in_time;
b_ot := 1;
end if;
end if;

if v_shift_in_time is not null then
b_ot := 1;
end if;

end if;
end if;

begin

select to_date(to_char(max(PunchTime),'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
into v_shift_out_time
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = v_comp_cardid
and to_date(to_char(punchtime,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
between to_date(to_char(v_shift_out_s_range,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
and to_date(to_char(v_shift_out_e_range,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS');

exception
when too_many_rows then
null;
when no_data_found then
null;
when others then
null;
end;

if v_shift_out_time is not null then
v_regDutyStatusOutA :=1;
else
v_regDutyStatusOutA :=0;
end if;

if v_shift_out_time is null and b_ot = 0 then

begin
select to_date(to_char(max(PunchTime),'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
into v_shift_out_time
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = v_comp_cardid
and to_date(to_char(punchtime,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
between to_date(to_char(v_shift_out_s_range_AOT,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS')
and to_date(to_char(v_shift_out_e_range_AOT,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS');

exception
when too_many_rows then
null;

```

```

when no_data_found then
null;
when others then
null;

end;

if v_shift_out_time is not null then
a_ot := 1;
end if;

if v_shift_out_time is null then
select to_date(to_char(max(PunchTime), 'DD/MM/RRRR HH24:MI:SS'), 'DD/MM/RRRR HH24:MI:SS')
into v_shift_out_time
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = v_comp_cardid
and to_date(to_char(punchtime, 'DD/MM/RRRR HH24:MI:SS'), 'DD/MM/RRRR HH24:MI:SS')
between to_date(to_char(v_shift_out_s_range, 'DD/MM/RRRR HH24:MI:SS'), 'DD/MM/RRRR HH24:MI:SS')
and to_date(to_char(v_shift_out_e_range_AOT, 'DD/MM/RRRR HH24:MI:SS'), 'DD/MM/RRRR
HH24:MI:SS');

if v_shift_out_time is not null then
a_ot := 1;
end if;
end if;
else
if v_shift_out_time is null then
select to_date(to_char(max(PunchTime), 'DD/MM/RRRR HH24:MI:SS'), 'DD/MM/RRRR HH24:MI:SS')
into v_shift_out_time
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = v_comp_cardid
and to_date(to_char(punchtime, 'DD/MM/RRRR HH24:MI:SS'), 'DD/MM/RRRR HH24:MI:SS')
between to_date(to_char(v_shiftout, 'DD/MM/RRRR HH24:MI:SS'), 'DD/MM/RRRR HH24:MI:SS')
and to_date(to_char(v_shift_out_s_range_AOT, 'DD/MM/RRRR HH24:MI:SS'), 'DD/MM/RRRR
HH24:MI:SS');
end if;
end if;

if v_regDutyStatusInA = 1 and v_regDutyStatusOutA = 1 then

begin

select to_date(to_char(max(PunchTime), 'DD/MM/RRRR HH24:MI:SS'), 'DD/MM/RRRR HH24:MI:SS')
into v_shift_after_ot_in
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = v_comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date, 'DD/MM/RRRR'), 'DD/MM/RRRR')
and to_date(to_char(punchtime, 'DD/MM/RRRR HH24:MI:SS'), 'DD/MM/RRRR HH24:MI:SS') >
to_date(to_char(v_shift_out_time, 'DD/MM/RRRR HH24:MI:SS'), 'DD/MM/RRRR HH24:MI:SS');
exception
when too_many_rows then
v_shift_out_time:=null;
when no_data_found then
v_shift_out_time:=null;
when others then

```

```

v_shift_out_time:=null;
end;
select to_date(to_char(v_shift_after_ot_in + interval '600' MINUTE,'DD/MM/RRRR
HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
into v_for_pick_after_ot
from dual;
begin
select to_date(to_char(max(PunchTime),'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
into v_shift_after_ot_out
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = v_comp_cardid
and to_date(to_char(punchtime,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')>
to_date(to_char(v_shift_after_ot_in,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
and to_date(to_char(punchtime,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')<
to_date(to_char(v_for_pick_after_ot,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS');

exception
when too_many_rows then
null;
when no_data_found then
null;
when others then
null;
end;

begin

select to_date(to_char(min(PunchTime),'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
into v_shift_before_ot_in
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = v_comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR')
and to_date(to_char(punchtime,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS') <
to_date(to_char(v_shift_in_time,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS');

begin
select to_date(to_char(R_IN_TIME_PUNCH,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS'),to_date(to_char(R_OUT_TIME_PUNCH,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS'),to_date(to_char(OT_IN_TIME_PUNCH,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS'),
to_date(to_char(OT_OUT_TIME_PUNCH,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
into
v_rintime,v_routtime,v_oointime,v_oouttime
from att_in.TBL_OT_ENTRY
where EMPID = I_EmpId
and OT_OR_REG_DATE = to_date(to_char(I_OT_Date - 1,'DD/MM/RRRR'),'DD/MM/RRRR') ;

EXCEPTION
when too_many_rows then
null;
when no_data_found then
null;
when others then
null;
end;

```

```

if v_routtime is not null or v_ointime is not null or v_oouttime is not null then

if v_shift_before_ot_in is not null and v_oouttime= v_shift_before_ot_in or v_ointime
=v_shift_before_ot_in then
v_shift_before_ot_in:="";
else
v_shift_before_ot_in:=v_shift_before_ot_in;
end if;
end if;
exception
when too_many_rows then
null;
when no_data_found then
null;
when others then
null;
end;

```

```

if v_shift_before_ot_in is not null then

```

```

begin

```

```

select to_date(to_char(max(PunchTime), 'DD/MM/RRRR HH24:MI:SS'), 'DD/MM/RRRR HH24:MI:SS')
into v_shift_before_ot_out
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = v_comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date, 'DD/MM/RRRR'), 'DD/MM/RRRR')
and to_date(to_char(punchtime, 'DD/MM/RRRR HH24:MI:SS'), 'DD/MM/RRRR HH24:MI:SS') >
to_date(to_char(v_shift_before_ot_in, 'DD/MM/RRRR HH24:MI:SS'), 'DD/MM/RRRR HH24:MI:SS')
and to_date(to_char(punchtime, 'DD/MM/RRRR HH24:MI:SS'), 'DD/MM/RRRR HH24:MI:SS') <
to_date(to_char(v_shift_in_time, 'DD/MM/RRRR HH24:MI:SS'), 'DD/MM/RRRR HH24:MI:SS');

```

```

begin

```

```

select to_date(to_char(R_IN_TIME_PUNCH, 'DD/MM/RRRR HH24:MI:SS'), 'DD/MM/RRRR HH24:MI:SS'),
to_date(to_char(R_OUT_TIME_PUNCH, 'DD/MM/RRRR HH24:MI:SS'), 'DD/MM/RRRR HH24:MI:SS'),
to_date(to_char(OT_IN_TIME_PUNCH, 'DD/MM/RRRR HH24:MI:SS'), 'DD/MM/RRRR HH24:MI:SS'),
to_date(to_char(OT_OUT_TIME_PUNCH, 'DD/MM/RRRR HH24:MI:SS'), 'DD/MM/RRRR HH24:MI:SS')
into
v_rintime, v_routtime, v_ointime, v_oouttime
from att_in.TBL_OT_ENTRY
where EMPID = I_EmpId
and OT_OR_REG_DATE = to_date(to_char(I_OT_Date - 1, 'DD/MM/RRRR'), 'DD/MM/RRRR') ;

```

```

EXCEPTION

```

```

when too_many_rows then
v_shift_in_time:=null;
when no_data_found then
v_shift_in_time:=null;
when others then
v_shift_in_time:=null;
end;

```

```

if v_routtime is not null or v_overtime is not null or v_overtime is not null then
if v_shift_before_ot_out is not null and v_overtime= v_shift_before_ot_out or v_overtime
=v_shift_before_ot_out then
v_shift_before_ot_out:="";
else
v_shift_before_ot_out:=v_shift_before_ot_out;
end if;
end if;
exception
when too_many_rows then
null;
when no_data_found then
null;
when others then
null;
end;
end if;

if v_shift_in_time is not null and v_shift_out_time is not null
and v_shift_before_ot_in is not null and v_shift_before_ot_out is not null then
b_ot := 1;
end if;

if v_shift_after_ot_out is null then
select to_date(to_char(v_for_pick_after_ot + interval '600' MINUTE,'DD/MM/RRRR
HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
into v_for_pick_after_ot
from dual;

begin
select to_date(to_char(max(PunchTime),'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
into v_shift_after_ot_out
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = v_comp_cardid
and to_date(to_char(punchtime,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')>
to_date(to_char(v_shift_after_ot_in,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
and to_date(to_char(punchtime,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')<
to_date(to_char(v_for_pick_after_ot,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS');

exception
when too_many_rows then
null;
when no_data_found then
null;
when others then
null;
end;

if v_shift_after_ot_out is not null and to_char(v_shift_after_ot_in,'HH24:MI:SS') > '20:00:00' then
v_shift_after_ot_out := to_date(to_char(I_OT_Date + 1,'DD/MM/RRRR') || '06:00:00','DD/MM/RRRR
HH24:MI:SS');
end if;
end if;
elsif v_pre_day_after_ot_log = 1 then

```

```

if previous_shift= v_shiftname and b_ot = 1 then
b_ot:=0;
else
b_ot:=b_ot;
end if;

begin

select to_date(to_char(max(PunchTime), 'DD/MM/RRRR HH24:MI:SS'), 'DD/MM/RRRR HH24:MI:SS')
into v_shift_after_ot_in
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = v_comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date, 'DD/MM/RRRR'), 'DD/MM/RRRR')
and to_date(to_char(punchtime, 'DD/MM/RRRR HH24:MI:SS'), 'DD/MM/RRRR HH24:MI:SS') >
to_date(to_char(v_shift_out_time, 'DD/MM/RRRR HH24:MI:SS'), 'DD/MM/RRRR HH24:MI:SS');
exception
when too_many_rows then
null;
when no_data_found then
null;
when others then
null;
end;

select to_date(to_char(v_shift_after_ot_in + interval '600' MINUTE, 'DD/MM/RRRR
HH24:MI:SS'), 'DD/MM/RRRR HH24:MI:SS')
into v_for_pick_after_ot
from dual;
begin
select to_date(to_char(max(PunchTime), 'DD/MM/RRRR HH24:MI:SS'), 'DD/MM/RRRR HH24:MI:SS')
into v_shift_after_ot_out
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = v_comp_cardid
and to_date(to_char(punchtime, 'DD/MM/RRRR HH24:MI:SS'), 'DD/MM/RRRR HH24:MI:SS') >
to_date(to_char(v_shift_after_ot_in, 'DD/MM/RRRR HH24:MI:SS'), 'DD/MM/RRRR HH24:MI:SS')
and to_date(to_char(punchtime, 'DD/MM/RRRR HH24:MI:SS'), 'DD/MM/RRRR HH24:MI:SS') <
to_date(to_char(v_for_pick_after_ot, 'DD/MM/RRRR HH24:MI:SS'), 'DD/MM/RRRR HH24:MI:SS');
exception
when too_many_rows then
null;
when no_data_found then
null;
when others then
null;
end;

if v_shift_after_ot_out is null then
select to_date(to_char(v_for_pick_after_ot + interval '600' MINUTE, 'DD/MM/RRRR
HH24:MI:SS'), 'DD/MM/RRRR HH24:MI:SS')
into v_for_pick_after_ot
from dual;
begin
select to_date(to_char(max(PunchTime), 'DD/MM/RRRR HH24:MI:SS'), 'DD/MM/RRRR HH24:MI:SS')
into v_shift_after_ot_out

```

```

from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = v_comp_cardid
and to_date(to_char(punchtime,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')>
to_date (to_char(v_shift_after_ot_in,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
and to_date(to_char(punchtime,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')<
to_date (to_char(v_for_pick_after_ot,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS');
exception
when too_many_rows then
null;
when no_data_found then
null;
when others then
null;
end;
end if;
end if;

Select abs(NVL(SUM(datediff('hh',v_shift_in_time,v_shift_out_time)),0) * 60)
into diff_in_out_time
from dual;

```

```

if v_shift_in_time is not null and v_shift_out_time is not null and diff_in_out_time > 780
and v_pre_day_after_ot_log=1 then
if v_overtime = v_shift_in_time and v_shift_out_time is not null and diff_in_out_time > 780
and v_pre_day_after_ot_log=1 then
v_shift_in_time := to_date(to_char(I_OT_Date,'DD/MM/RRRR') || '06:00:00','DD/MM/RRRR
HH24:MI:SS');
end if;
else

```

```

if diff_in_out_time > 500 then

```

```

select to_date(to_char(min(PunchTime),'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
into v_shift_extra_ot_in
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = v_comp_cardid
and to_date(to_char(punchtime,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')>
to_date (to_char(v_shift_out_extra_OT,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
and to_date(to_char(punchtime,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')<
to_date (to_char(v_shift_in_time,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS');

```

```

select to_date(to_char(max(PunchTime),'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
into v_shift_extra_ot_out
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = v_comp_cardid
and to_date(to_char(punchtime,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')>
to_date (to_char(v_shift_out_extra_OT,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
and to_date(to_char(punchtime,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')<
to_date (to_char(v_shift_in_time,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS');

```

```

begin
select to_date(to_char(R_IN_TIME_PUNCH,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'),
to_date(to_char(R_OUT_TIME_PUNCH,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'),
to_date(to_char(OT_IN_TIME_PUNCH,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'),
to_date(to_char(OT_OUT_TIME_PUNCH,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')

```

```

into
v_rintime,v_routtime,v_overtime,v_overtime
from att_in.TBL_OT_ENTRY
where EMPID = I_EmpId
and OT_OR_REG_DATE = to_date(to_char(I_OT_Date - 1,'DD/MM/RRRR'),'DD/MM/RRRR') ;

EXCEPTION
when too_many_rows then
null;
when no_data_found then
null;
when others then
null;
end;
if (v_shift_extra_ot_in = v_rintime or v_shift_extra_ot_in = v_overtime) and (v_shift_extra_ot_out =
v_routtime or v_shift_extra_ot_out = v_overtime) then
v_shift_extra_ot_in := '';
v_shift_extra_ot_out := '';
else
v_shift_extra_ot_in := v_shift_extra_ot_in;
v_shift_extra_ot_out := v_shift_extra_ot_out;

end if;

Select abs(NVL(SUM(datediff('hh',v_shift_extra_ot_in,v_shift_extra_ot_out)),0) * 60)
into v_diff_between_extra_ot
from dual;
if v_shift_extra_ot_in is not null and v_shift_extra_ot_out is not null and v_diff_between_extra_ot > 120
then
v_shift_extra_ot_log:=1;
else
v_shift_extra_ot_log:=0;
end if;

if v_shift_extra_ot_log=1 then

update att_in.TBL_OT_ENTRY
set REG_STATUS = upper('P'),
EXTRA_OT_IN= to_date(to_char(v_shift_extra_ot_in,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS')--I_TempOTIn
EXTRA_OT_OUT = to_date(to_char(v_shift_extra_ot_out,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS')--I_TempOTIn
where EMPID = I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR');
commit;
end if;
end if;
end if;
begin
select to_date(to_char(EXTRA_OT_IN,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'),
to_date(to_char(EXTRA_OT_OUT,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'),
to_date(to_char(OT_IN_TIME_PUNCH,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'),

```



```

to_date(to_char(OT_OUT_TIME_PUNCH,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
into
v_extra_in,v_extra_out,v_reg_ot_in,v_reg_ot_out
from att_in.TBL_OT_ENTRY
where EMPID = I_EmpId
and OT_OR_REG_DATE = to_date(to_char(I_OT_Date - 1,'DD/MM/RRRR'),'DD/MM/RRRR');
EXCEPTION
when too_many_rows then
null;
when no_data_found then
null;
when others then
null;
end;

begin
select to_date(to_char(OT_IN_TIME_PUNCH,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'),
to_date(to_char(OT_OUT_TIME_PUNCH,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
into
v_ot_in,v_ot_out
from att_in.TBL_OT_ENTRY
where EMPID = I_EmpId
and OT_OR_REG_DATE = to_date(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR');
EXCEPTION
when too_many_rows then
null;
when no_data_found then
null;
when others then
null;
end;

if v_shift_in_time is not null and v_shift_out_time is not null then

if b_ot = 0 and a_ot = 0 then
if v_regDutyStatusInA = 1 and v_regDutyStatusOutA = 1 then
update att_in.TBL_OT_ENTRY
set REG_STATUS = upper('P'),
R_IN_TIME_PUNCH = to_date(to_char(v_shift_in_time,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS'),
R_OUT_TIME_PUNCH = to_date(to_char(v_shift_out_time,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS'),
OT_LOG = 0,
AFTER_OT_LOG = 0
where EMPID = I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR');
commit;
end if;

if v_regDutyStatusInA = 1 and v_regDutyStatusOutA = 1
and v_shift_after_ot_in is not null and v_shift_after_ot_out is not null Then

update att_in.TBL_OT_ENTRY
set REG_STATUS = upper('P'),

```

```

OT_IN_TIME_PUNCH = to_date(to_char(v_shift_after_ot_in,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS'),--I_TempOTIn
OT_OUT_TIME_PUNCH = to_date(to_char(v_shift_after_ot_out,'DD/MM/RRRR
HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'),--I_TempOTIn
OT_LOG = 0,
AFTER_OT_LOG = 1
where EMPID = I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR');
commit;
end if;

```

```

if v_regDutyStatusInA = 1 and v_regDutyStatusOutA = 1
and v_shift_before_ot_in is not null and v_shift_before_ot_out is not null then
update att_in.TBL_OT_ENTRY
set REG_STATUS = upper('P'),
OT_IN_TIME_PUNCH = to_date(to_char(v_shift_before_ot_in,'DD/MM/RRRR
HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'),--I_TempOTIn
OT_OUT_TIME_PUNCH = to_date(to_char(v_shift_before_ot_out,'DD/MM/RRRR
HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'),--I_TempOTIn
OT_LOG = 1,
AFTER_OT_LOG = 0
where EMPID = I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR');
commit;
end if;

```

```

if v_regDutyStatusInA = 1 and v_regDutyStatusOutA = 1
and v_shift_after_ot_in is not null and v_shift_after_ot_out is not null
and v_pre_day_after_ot_log = 1 then

```

```

update att_in.TBL_OT_ENTRY
set REG_STATUS = upper('P'),
OT_IN_TIME_PUNCH = to_date(to_char(v_shift_after_ot_in,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS'),--I_TempOTIn
OT_OUT_TIME_PUNCH = to_date(to_char(v_shift_after_ot_out,'DD/MM/RRRR
HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'),--I_TempOTIn
OT_LOG = 0,
AFTER_OT_LOG = 1
where EMPID = I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR');
commit;
end if;

```

```

if v_regDutyStatusInA = 0 and v_regDutyStatusOutA = 1 and v_shift_in_time is not null and
v_shift_out_time is not null
and v_pre_day_after_ot_log=1 and diff_in_out_time > 780 then

```

```

update att_in.TBL_OT_ENTRY
set REG_STATUS = upper('P'),
R_IN_TIME_PUNCH = to_date(to_char(v_shift_in_time,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS'),
R_OUT_TIME_PUNCH = to_date(to_char(v_shift_out_time,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS'),
OT_IN_TIME_PUNCH = to_date(to_char(v_shift_after_ot_in,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS'),--I_TempOTIn

```

```

OT_OUT_TIME_PUNCH = to_date(to_char(v_shift_after_ot_out,'DD/MM/RRRR
HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'),--I_TempOTIn
OT_LOG = 0,
AFTER_OT_LOG = 1
where EMPID = I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR');
commit;
end if;

elsif b_ot = 1 and a_ot = 0 then
if v_routtime is not null or v_overtime is not null or v_overtime is not null then

Select abs(NVL(SUM(datediff('hh',v_shift_in_time,v_overtime)),0) * 60)
into diff_time_between_two_shift
from dual;
end if;

if v_pre_day_after_ot_log = 0 then
if v_shift_before_ot_in is not null and v_shift_before_ot_out is not null then
update att_in.TBL_OT_ENTRY
set REG_STATUS = upper('P'),
R_IN_TIME_PUNCH = to_date(to_char(v_shift_in_time,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS'),
R_OUT_TIME_PUNCH = to_date(to_char(v_shift_out_time,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS'),
OT_IN_TIME_PUNCH = to_date(to_char(v_shift_before_ot_in,'DD/MM/RRRR
HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'),--I_TempOTIn
OT_OUT_TIME_PUNCH = to_date(to_char(v_shift_before_ot_out,'DD/MM/RRRR
HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'),--I_TempOTIn
OT_LOG = 1,
AFTER_OT_LOG = 0
where EMPID = I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR');
commit;

else

update att_in.TBL_OT_ENTRY
set REG_STATUS = upper('P'),
R_IN_TIME_PUNCH = to_date(to_char(v_shiftin,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS'),
R_OUT_TIME_PUNCH = to_date(to_char(v_shift_out_time,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS'),
OT_IN_TIME_PUNCH = to_date(to_char(v_shift_in_time,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS'),--I_TempOTIn
OT_OUT_TIME_PUNCH = to_date(to_char(v_shiftin,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS'),--I_TempOTIn
OT_LOG = 1,
AFTER_OT_LOG = 0
where EMPID = I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR');
commit;

```

```

end if;
elsif v_pre_day_after_ot_log = 1 and diff_time_between_two_shift>240 then
update att_in.TBL_OT_ENTRY
set REG_STATUS = upper('P'),
R_IN_TIME_PUNCH = to_date(to_char(v_shiftin,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS'),
R_OUT_TIME_PUNCH = to_date(to_char(v_shift_out_time,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS'),
OT_IN_TIME_PUNCH = to_date(to_char(v_shift_in_time,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS'),--I_TempOTIn
OT_OUT_TIME_PUNCH = to_date(to_char(v_shiftin,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS'),--I_TempOTIn
OT_LOG = 1,
AFTER_OT_LOG = 0
where EMPID = I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR');
commit;
else
update att_in.TBL_OT_ENTRY
set REG_STATUS = upper('P'),
R_IN_TIME_PUNCH = to_date(to_char(v_shiftin,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS'),
R_OUT_TIME_PUNCH = to_date(to_char(v_shift_out_time,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS'),
OT_LOG = 0,
AFTER_OT_LOG = 0
where EMPID = I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR');
commit;
end if;
elsif b_ot = 0 and a_ot = 1 then
update att_in.TBL_OT_ENTRY
set REG_STATUS = upper('P'),
R_IN_TIME_PUNCH = to_date(to_char(v_shift_in_time,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS'),
R_OUT_TIME_PUNCH = to_date(to_char(v_shiftout,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS'),
OT_IN_TIME_PUNCH = to_date(to_char(v_shiftout,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS'),--I_TempOTIn
OT_OUT_TIME_PUNCH = to_date(to_char(v_shift_out_time,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS'),--I_TempOTIn
OT_LOG = 0,
AFTER_OT_LOG = 1
where EMPID = I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR');
commit;
elsif b_ot = 1 and a_ot = 1 then

select to_date(to_char(v_shift_in_time + interval '480' MINUTE,'DD/MM/RRRR
HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
into v_shift_in_time_exceot
from dual;

update att_in.TBL_OT_ENTRY
set REG_STATUS = upper('P'),

```

```

R_IN_TIME_PUNCH = to_date(to_char(v_shift_in_time,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS'),
R_OUT_TIME_PUNCH = to_date(to_char(v_shift_in_time_exceot,'DD/MM/RRRR
HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'),
OT_IN_TIME_PUNCH = to_date(to_char(v_shift_in_time_exceot,'DD/MM/RRRR
HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'),--I_TempOTIn
OT_OUT_TIME_PUNCH = to_date(to_char(v_shift_out_time,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS'),--I_TempOTIn
OT_LOG = 0,
AFTER_OT_LOG = 1
where EMPID = I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR');
commit;

end if;
elsif v_shift_in_time is null and v_shift_out_time is not null then

if v_shift_after_ot_in is not null and v_shift_after_ot_out is not null then
update att_in.TBL_OT_ENTRY
set REG_STATUS = upper('P'),
R_IN_TIME_PUNCH = to_date(to_char(v_shiftin,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS'),
R_OUT_TIME_PUNCH = to_date(to_char(v_shift_out_time,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS'),
OT_IN_TIME_PUNCH = to_date(to_char(v_shift_after_ot_in,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS'),--I_TempOTIn
OT_OUT_TIME_PUNCH = to_date(to_char(v_shift_after_ot_out,'DD/MM/RRRR
HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'),--I_TempOTIn
OT_LOG = 1,
AFTER_OT_LOG = 0
where EMPID = I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR');
commit;
elsif v_shift_after_ot_in is null and v_shift_after_ot_out is null then

if v_shift_in_time is not null and v_shift_out_time is not null then
update att_in.TBL_OT_ENTRY
set REG_STATUS = upper('P'),
R_IN_TIME_PUNCH = to_date(to_char(v_shiftin,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS'),
R_OUT_TIME_PUNCH = to_date(to_char(v_shift_out_time,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS'),
OT_LOG = 1,
AFTER_OT_LOG = 0
where EMPID = I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR');
commit;

elsif v_shift_in_time is null and v_shift_out_time is not null then
if v_check_in_punch = 1 then
update att_in.TBL_OT_ENTRY
set REG_STATUS = upper('P'),
R_IN_TIME_PUNCH = to_date(to_char(v_shiftin,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS'),
R_OUT_TIME_PUNCH = to_date(to_char(v_shift_out_time,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR

```

```

HH24:MI:SS'),
OT_LOG = 1,
AFTER_OT_LOG = 0
where EMPID = I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR');
commit;
elsif v_check_in_punch = 0 then
update att_in.TBL_OT_ENTRY
set REG_STATUS ='A'
where EMPID = I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR');
commit;
end if;
end if;
else
update att_in.TBL_OT_ENTRY
set REG_STATUS ='A',
R_OUT_TIME_PUNCH = to_date(to_char(v_shiftout,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS')
where EMPID = I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR');
commit;
end if;

elsif v_shift_in_time is not null and v_shift_out_time is null then
update att_in.TBL_OT_ENTRY
set REG_STATUS ='A',
R_IN_TIME_PUNCH = to_date(to_char(v_shift_in_time,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS')
where EMPID = I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR');
commit;
elsif v_shift_in_time is null and v_shift_out_time is null then
update ATT_IN.TBL_OT_ENTRY
set REG_STATUS ='A'
where EMPID = I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR');
commit;
end if;

if v_shift_in_time is null and v_shift_out_time is not null then
v_shift_out_time := '';
elsif v_shift_in_time is not null and v_shift_out_time is null then
v_shift_in_time := '';
end if;

if v_shift_in_time is null and v_shift_out_time is null then

select ATT_IN.fn_Leave_OSD_Count(I_OT_Date,I_OT_Date,I_EmpId) into leave_osd_counter from dual;

if leave_osd_counter > 0 then
begin
select count(EMPID) into leave_counter
from att_in.TBL_LEAVE_APPLY
where EMPID = I_EmpId

```

```

and FROMDATE = TO_DATE(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR') ;
EXCEPTION
when too_many_rows then
null;
when no_data_found then
null;
when others then
null;
end;

if leave_counter > 0 then

begin
select LEAVETYPEID into leave_id
from att_in.TBL_LEAVE_APPLY
where EMPID = I_EmpId
and FROMDATE = TO_DATE(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR') ;
EXCEPTION
when too_many_rows then
null;
when no_data_found then
null;
when others then
null;
end;

begin
select REPORTINGNAME into ITypeName
from att_in.TBL_LEAVE_TYPE
where LEAVETYPEID = leave_id;
EXCEPTION
when too_many_rows then
null;
when no_data_found then
null;
when others then
null;
end;
update att_in.tbl_ot_entry
set REG_STATUS = upper(ITypeName),
R_IN_TIME_PUNCH="",
R_OUT_TIME_PUNCH=""
where EMPID = I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR');
commit;
else
begin
select count(EMPID) into osd_counter
from att_in.TBL_OSD_SETUP
where EMPID = I_EmpId
and OSDSTARTDATE = TO_DATE(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR');
EXCEPTION
when too_many_rows then
null;
when no_data_found then

```

```

null;
when others then
null;
end;

if osd_counter > 0 then
update att_in.tbl_ot_entry
set REG_STATUS = upper('OCS'),
R_IN_TIME_PUNCH="",
R_OUT_TIME_PUNCH=""
where EMPID = I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR');
commit;
else
update att_in.TBL_OT_ENTRY
set REG_STATUS = 'A'
where EMPID = I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR');
commit;
end if;
end if;
end if;
end if;

begin
select REG_STATUS into I_Sts
from att_in.TBL_OT_ENTRY
where EMPID = I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR');
EXCEPTION when no_data_found then
null;
end;

if v_extra_in is not null and v_extra_out is not null and v_reg_ot_in is not null and v_reg_ot_out is not
null
and v_ot_in is null and v_ot_out is null and I_Sts not in('CL','SL','PL','D/O','OCS','GHD') then

update att_in.TBL_OT_ENTRY
set
OT_IN_TIME_PUNCH = to_date(to_char(v_reg_ot_in,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS'),--I_TempOTIn
OT_OUT_TIME_PUNCH = to_date(to_char(v_reg_ot_out,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS'),--I_TempOTIn
OT_LOG = 1,
AFTER_OT_LOG = 0
where EMPID = I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR');
commit;

update att_in.TBL_OT_ENTRY
set
OT_IN_TIME_PUNCH = to_date(to_char(v_extra_in,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS'),--I_TempOTIn
OT_OUT_TIME_PUNCH = to_date(to_char(v_extra_out,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR

```



```

HH24:MI:SS'),--I_TempOTIn
OT_LOG = 1,
AFTER_OT_LOG = 0
where EMPID = I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date - 1,'DD/MM/RRRR'),'DD/MM/RRRR');
commit;
end if;
elsif v_shiftname = '4 Shift-D' then

begin
select nvl(R_SHIFT,")
into I_Prev_Shift_Before_D
from att_in.TBL_OT_ENTRY
where EMPID = I_EmpId
and OT_OR_REG_DATE = (select to_date(to_char(I_OT_Date - interval '1'
DAY,'DD/MM/RRRR'),'DD/MM/RRRR')
from dual);
EXCEPTION when no_data_found then
I_Prev_Shift_Before_D:= null;
end;

if I_Prev_Shift_Before_D = '4 Shift-C' then
select count(punchtime)
into I_counder_D
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = v_comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR');

begin
select to_date(to_char(R_SHIFT_IN,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'),
to_date(to_char(R_IN_TIME_PUNCH,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'),
to_date(to_char(R_OUT_TIME_PUNCH,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'),
to_date(to_char(OT_IN_TIME_PUNCH,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'),
to_date(to_char(OT_OUT_TIME_PUNCH,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'),
to_date(to_char(R_SHIFT_OUT,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
into
in_time_C,
in_punch_C,out_time_C,ot_inpunch_C,ot_outpunch_C,
set_reg_out_time
from att_in.TBL_OT_ENTRY
where EMPID = I_EmpId
and OT_OR_REG_DATE = (select to_date(to_char(I_OT_Date - interval '1'
DAY,'DD/MM/RRRR'),'DD/MM/RRRR')
from dual);
EXCEPTION when no_data_found then
null;
end;

if I_counder_D > 0 then

begin
select nvl(to_date(to_char(min(PunchTime),'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'),")
into in_time_D
from att_in.TBL_RAW_DATA_CMIS

```

```

where COMPCARDID = v_comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR');

if in_time_C is not null or in_punch_C is not null or v_oouttime is not null then

if in_punch_C is not null and in_time_D=in_punch_C or in_time_D=out_time_C
or in_time_D=ot_inpunch_C or in_time_D=ot_outpunch_C then
in_time_D:="";
else
in_time_D:= in_time_D;
end if;

end if;

EXCEPTION when no_data_found then
null;
end;

begin
select nvl(to_date(to_char(max(PunchTime),'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'), '')
into out_time_D
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = v_comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR');

if v_routtime is not null or v_overtime is not null or v_oouttime is not null then

if out_time_D is not null and v_rintime= out_time_D or v_overtime =out_time_D or v_routtime =
out_time_D then
out_time_D:="";
else
out_time_D:= out_time_D;
end if;

end if;

EXCEPTION when no_data_found then
null;
end;

begin
select nvl(to_date(to_char(min(PunchTime),'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'), '')
into in_time_D_2nd_min
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = v_comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR')
and PunchTime > in_time_D and PunchTime < out_time_D;
EXCEPTION when no_data_found then
null;
end;

if in_time_D_2nd_min is null then
in_time_D_2nd_min := in_time_D;
end if;

```

```

Select abs(NVL(SUM(datediff('hh',in_time_D_2nd_min,out_time_D)),0) * 60)
into diff_mind_maxd
from dual;

if diff_mind_maxd < 120 then
out_time_D := null;
else
out_time_D := out_time_D;

end if;

if in_time_D_2nd_min is null then

begin
select nvl(to_date(to_char(max(PunchTime),'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'),")
into in_time_D_2nd_min
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = v_comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR');
EXCEPTION when no_data_found then
null;
end;
out_time_D := null;
end if;

if out_time_D is null then
begin
select nvl(to_date(to_char(min(PunchTime),'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'),")
into out_time_D
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = v_comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date + 1,'DD/MM/RRRR'),'DD/MM/RRRR');
EXCEPTION when no_data_found then
null;
end;

if v_routtime is not null or v_overtime is not null or v_overtime is not null then

if out_time_D is not null and v_rintime= out_time_D or v_overtime =out_time_D or v_routtime =
out_time_D then
out_time_D:="";
else
out_time_D:= out_time_D;
end if;

end if;

end if;
end if;

Select abs(NVL(SUM(datediff('hh',in_time_C,in_time_D)),0) * 60)
into diff_in_out_C
from dual;

Select abs(NVL(SUM(datediff('hh',in_time_D,in_time_D_2nd_min)),0) * 60)

```

```

into diff_2min_1min_D
from dual;

select nvl(OT_LOG,0),REG_STATUS
into chk_ot_log,V_REG_STATUS
from att_in.TBL_OT_ENTRY
where EMPID = I_EmpId
and OT_OR_REG_DATE = (select to_date(to_char(I_OT_Date - interval '1'
DAY,'DD/MM/RRRR'),'DD/MM/RRRR')
from dual);

if chk_ot_log = 0 then

if diff_in_out_C >= 960 and diff_in_out_C < 1080 then
diff_in_out_C :=960;
end if;

if diff_in_out_C > 600 and V_REG_STATUS in ('P','L') then

update att_in.TBL_OT_ENTRY
set R_OUT_TIME_PUNCH = set_reg_out_time,
OT_IN_TIME_PUNCH = set_reg_out_time,
OT_OUT_TIME_PUNCH = in_time_D,
AFTER_OT_LOG = 1,
OT_LOG =0
where EMPID = I_EmpId
and OT_OR_REG_DATE = (select to_date(to_char(I_OT_Date - interval '1'
DAY,'DD/MM/RRRR'),'DD/MM/RRRR')
from dual);
commit;

update att_in.TBL_OT_ENTRY
set REG_STATUS = 'W'
where EMPID = I_EmpId
and OT_OR_REG_DATE = to_date(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR');
commit;

elsif I_Prev_Shift_Before_D = '4 Shift-C' and V_REG_STATUS = 'A' then

select to_date(to_char(in_time_D_2nd_min + interval '480' MINUTE,'DD/MM/RRRR
HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
into v_dreg_hour
from dual;

if diff_mind_maxd > 480 then

update att_in.TBL_OT_ENTRY
set REG_STATUS = 'P',
R_IN_TIME_PUNCH = in_time_D_2nd_min,
R_OUT_TIME_PUNCH = v_dreg_hour,
OT_IN_TIME_PUNCH = v_dreg_hour,
OT_OUT_TIME_PUNCH = out_time_D,
AFTER_OT_LOG = 1,
OT_LOG =0
where EMPID = I_EmpId

```

```

and OT_OR_REG_DATE = (select to_date(to_char(I_OT_Date , 'DD/MM/RRRR'), 'DD/MM/RRRR')
from dual);
commit;
else
update att_in.TBL_OT_ENTRY
set REG_STATUS = 'W'
where EMPID = I_EmpId
and OT_OR_REG_DATE = to_date(to_char(I_OT_Date, 'DD/MM/RRRR'), 'DD/MM/RRRR');
commit;
end if;

elsif diff_in_out_C = 0 then

update att_in.TBL_OT_ENTRY
set REG_STATUS = 'W'
where EMPID = I_EmpId
and OT_OR_REG_DATE = to_date(to_char(I_OT_Date, 'DD/MM/RRRR'), 'DD/MM/RRRR');
commit;
else
update att_in.TBL_OT_ENTRY
set REG_STATUS = 'W'
where EMPID = I_EmpId
and OT_OR_REG_DATE = to_date(to_char(I_OT_Date, 'DD/MM/RRRR'), 'DD/MM/RRRR');
commit;

end if;

if diff_2min_1min_D > 120 then

Select abs(NVL(SUM(datediff('hh', in_time_D_2nd_min, out_time_D)), 0) * 60)
into var_diff_dnight
from dual;

if var_diff_dnight >= 509 and var_diff_dnight < 1080 then

if to_char(in_time_D_2nd_min, 'HH24:MI:SS') > '10:00:00' and
to_char(in_time_D_2nd_min, 'HH24:MI:SS') < '16:00:00' then
d1_outtime := '22:00:00';
elsif to_char(in_time_D_2nd_min, 'HH24:MI:SS') > '18:00:00' and
to_char(in_time_D_2nd_min, 'HH24:MI:SS') < '23:59:00' then
d1_outtime := '06:00:00';
end if;

update att_in.TBL_OT_ENTRY
set REG_STATUS = 'P',
R_IN_TIME_PUNCH = in_time_D_2nd_min,
R_OUT_TIME_PUNCH = to_date(to_char(I_OT_Date + 1, 'DD/MM/RRRR') || ' ' ||
d1_outtime, 'DD/MM/RRRR HH24:MI:SS'),
OT_IN_TIME_PUNCH = to_date(to_char(I_OT_Date + 1, 'DD/MM/RRRR') || ' ' || d1_outtime, 'DD/MM/RRRR
HH24:MI:SS'),
OT_OUT_TIME_PUNCH = out_time_D,
AFTER_OT_LOG = 1
where EMPID = I_EmpId
and OT_OR_REG_DATE = to_date(to_char(I_OT_Date , 'DD/MM/RRRR'), 'DD/MM/RRRR');
commit;

```

```

else
update att_in.TBL_OT_ENTRY
set REG_STATUS = 'P',
R_IN_TIME_PUNCH = in_time_D_2nd_min,
R_OUT_TIME_PUNCH = out_time_D,
AFTER_OT_LOG = 1
--OT_LOG =1
where EMPID = I_EmpId
and OT_OR_REG_DATE = to_date(to_char(I_OT_Date , 'DD/MM/RRRR'), 'DD/MM/RRRR');
commit;
end if;
begin
select to_date(to_char(OT_SHIFT_IN_TIME, 'DD/MM/RRRR HH24:MI:SS'), 'DD/MM/RRRR HH24:MI:SS'),
to_date(to_char(OT_SHIFT_OUT_TIME, 'DD/MM/RRRR HH24:MI:SS'), 'DD/MM/RRRR HH24:MI:SS')
into
chk_C2nd_ot_in,
chk_C2nd_ot_out
from att_in.TBL_OT_ENTRY
where EMPID = I_EmpId
and OT_OR_REG_DATE = (select to_date(to_char(I_OT_Date - interval '1'
DAY, 'DD/MM/RRRR'), 'DD/MM/RRRR')
from dual);
EXCEPTION when no_data_found then
chk_C2nd_ot_in:= null;
chk_C2nd_ot_out:= null;
end;

if chk_C2nd_ot_in is not null and chk_C2nd_ot_out is not null then
update att_in.TBL_OT_ENTRY
set OT_SHIFT_IN_TIME = null,
OT_SHIFT_OUT_TIME = null
where EMPID = I_EmpId
and OT_OR_REG_DATE = (select to_date(to_char(I_OT_Date - interval '1'
DAY, 'DD/MM/RRRR'), 'DD/MM/RRRR')
from dual);
commit;
end if;

elsif in_time_D is not null then

if to_char(in_time_D_2nd_min, 'HH24:MI:SS') >
'10:00:00' and to_char(in_time_D_2nd_min, 'HH24:MI:SS') < '16:00:00' then
d1_outtime := '22:00:00';
elsif to_char(in_time_D_2nd_min, 'HH24:MI:SS')
> '18:00:00' and to_char(in_time_D_2nd_min, 'HH24:MI:SS') < '23:59:00' then
d1_outtime := '06:00:00';
end if;

update att_in.TBL_OT_ENTRY
set REG_STATUS = 'P',
R_IN_TIME_PUNCH = in_time_D_2nd_min,
R_OUT_TIME_PUNCH = to_date(to_char(I_OT_Date + 1, 'DD/MM/RRRR') || ' ' ||
d1_outtime, 'DD/MM/RRRR HH24:MI:SS'),
OT_IN_TIME_PUNCH = to_date(to_char(I_OT_Date + 1, 'DD/MM/RRRR') || ' ' || d1_outtime, 'DD/MM/RRRR

```

```

HH24:MI:SS'),
OT_OUT_TIME_PUNCH = out_time_D,
AFTER_OT_LOG = 1
where EMPID = I_EmpId
and OT_OR_REG_DATE = to_date(to_char(I_OT_Date, 'DD/MM/RRRR'), 'DD/MM/RRRR');
commit;
end if;
elsif chk_ot_log = 1 then
if diff_in_out_C >= 960 and diff_in_out_C < 1080 then
diff_in_out_C := 960;
select to_date(to_char(OT_IN_TIME_PUNCH, 'DD/MM/RRRR HH24:MI:SS'), 'DD/MM/RRRR HH24:MI:SS'),
to_date(to_char(OT_OUT_TIME_PUNCH, 'DD/MM/RRRR HH24:MI:SS'), 'DD/MM/RRRR HH24:MI:SS')
into
chk_C2nd_ot_in,
chk_C2nd_ot_out
from att_in.TBL_OT_ENTRY
where EMPID = I_EmpId
and OT_OR_REG_DATE = (select to_date(to_char(I_OT_Date - interval '1'
DAY, 'DD/MM/RRRR'), 'DD/MM/RRRR')
from dual);
if chk_C2nd_ot_in is not null and chk_C2nd_ot_out is not null then
update att_in.TBL_OT_ENTRY
set OT_IN_TIME_PUNCH = set_reg_out_time,
OT_OUT_TIME_PUNCH = in_time_D,
AFTER_OT_LOG = 1,
OT_LOG = 1
where EMPID = I_EmpId
and OT_OR_REG_DATE = (select to_date(to_char(I_OT_Date - interval '1'
DAY, 'DD/MM/RRRR'), 'DD/MM/RRRR')
from dual);
commit;

update att_in.TBL_OT_ENTRY
set REG_STATUS = 'W'
where EMPID = I_EmpId
and OT_OR_REG_DATE = to_date(to_char(I_OT_Date, 'DD/MM/RRRR'), 'DD/MM/RRRR');
commit;
else
null;
end if;

elsif diff_in_out_C > 540 then

select to_date(to_char(OT_IN_TIME_PUNCH, 'DD/MM/RRRR HH24:MI:SS'), 'DD/MM/RRRR HH24:MI:SS'),
to_date(to_char(OT_OUT_TIME_PUNCH, 'DD/MM/RRRR HH24:MI:SS'), 'DD/MM/RRRR HH24:MI:SS')
into
chk_C2nd_ot_in,
chk_C2nd_ot_out
from att_in.TBL_OT_ENTRY
where EMPID = I_EmpId
and OT_OR_REG_DATE = (select to_date(to_char(I_OT_Date - interval '1'
DAY, 'DD/MM/RRRR'), 'DD/MM/RRRR')
from dual);

if chk_C2nd_ot_in is not null and chk_C2nd_ot_out is not null then

```

```

update att_in.TBL_OT_ENTRY
set OT_IN_TIME_PUNCH = set_reg_out_time,
OT_OUT_TIME_PUNCH = in_time_D,
AFTER_OT_LOG = 1,
OT_LOG =1
where EMPID = I_EmpId
and OT_OR_REG_DATE = (select to_date(to_char(I_OT_Date - interval '1'
DAY,'DD/MM/RRRR'),'DD/MM/RRRR')
from dual);
commit;

update att_in.TBL_OT_ENTRY
set REG_STATUS = 'W'
where EMPID = I_EmpId
and OT_OR_REG_DATE = to_date(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR');
commit;
else
null;
end if;
else
update att_in.TBL_OT_ENTRY
set REG_STATUS = 'W'
where EMPID = I_EmpId
and OT_OR_REG_DATE = to_date(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR');
commit;
end if;

if diff_2min_1min_D > 120 then
Select abs(NVL(SUM(datediff('hh',in_time_D_2nd_min,out_time_D)),0) * 60)
into var_diff_dnight
from dual;
if var_diff_dnight >= 509 and var_diff_dnight < 1080 then
if to_char(in_time_D_2nd_min,'HH24:MI:SS') > '10:00:00' and
to_char(in_time_D_2nd_min,'HH24:MI:SS') < '16:00:00' then
d1_outtime := '22:00:00';
elsif to_char(in_time_D_2nd_min,'HH24:MI:SS') > '18:00:00' and
to_char(in_time_D_2nd_min,'HH24:MI:SS') < '23:59:00' then
d1_outtime := '06:00:00';
end if;

update att_in.TBL_OT_ENTRY
set REG_STATUS = 'P',
R_IN_TIME_PUNCH = in_time_D_2nd_min,
R_OUT_TIME_PUNCH = to_date(to_char(I_OT_Date + 1,'DD/MM/RRRR') ||' '||
d1_outtime,'DD/MM/RRRR HH24:MI:SS'),
OT_IN_TIME_PUNCH = to_date(to_char(I_OT_Date + 1,'DD/MM/RRRR') ||' '|| d1_outtime,'DD/MM/RRRR
HH24:MI:SS'),
OT_OUT_TIME_PUNCH = out_time_D,
AFTER_OT_LOG = 1
where EMPID = I_EmpId
and OT_OR_REG_DATE = to_date(to_char(I_OT_Date , 'DD/MM/RRRR'),'DD/MM/RRRR');
commit;

else
update att_in.TBL_OT_ENTRY

```



```

set REG_STATUS = 'P',
R_IN_TIME_PUNCH = in_time_D_2nd_min,
R_OUT_TIME_PUNCH = out_time_D,
AFTER_OT_LOG = 1
where EMPID = I_EmpId
and OT_OR_REG_DATE = to_date(to_char(I_OT_Date , 'DD/MM/RRRR'), 'DD/MM/RRRR');
commit;
end if;
begin
select to_date(to_char(OT_SHIFT_IN_TIME, 'DD/MM/RRRR HH24:MI:SS'), 'DD/MM/RRRR HH24:MI:SS'),
to_date(to_char(OT_SHIFT_OUT_TIME, 'DD/MM/RRRR HH24:MI:SS'), 'DD/MM/RRRR HH24:MI:SS')
into
chk_C2nd_ot_in,
chk_C2nd_ot_out
from att_in.TBL_OT_ENTRY
where EMPID = I_EmpId
and OT_OR_REG_DATE = (select to_date(to_char(I_OT_Date - interval '1'
DAY, 'DD/MM/RRRR'), 'DD/MM/RRRR')
from dual);
EXCEPTION when no_data_found then
chk_C2nd_ot_in:= null;
chk_C2nd_ot_out:= null;
end;

if chk_C2nd_ot_in is not null and chk_C2nd_ot_out is not null then
update att_in.TBL_OT_ENTRY
set OT_SHIFT_IN_TIME = null,
OT_SHIFT_OUT_TIME = null
where EMPID = I_EmpId
and OT_OR_REG_DATE = (select to_date(to_char(I_OT_Date - interval '1'
DAY, 'DD/MM/RRRR'), 'DD/MM/RRRR')
from dual);
commit;
end if;
end if;
end if;
elsif I_Prev_Shift_Before_D = '4 Shift-D' then
select count(punchtime)
into I_CounterW
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = v_comp_cardid
and PUNCHDATE= to_date(to_char(I_OT_Date , 'DD/MM/RRRR'), 'DD/MM/RRRR');

select REG_STATUS into I_Sts_W
from att_in.TBL_OT_ENTRY
where EMPID = I_EmpId
and OT_OR_REG_DATE = to_date(to_char(I_OT_Date - 1, 'DD/MM/RRRR'), 'DD/MM/RRRR');

if I_CounterW>0 then

if I_Sts_W in('P','L') then

select to_char(R_IN_TIME_PUNCH, 'DD/MM/RRRR HH24:MI:SS'),
to_char(R_OUT_TIME_PUNCH, 'DD/MM/RRRR HH24:MI:SS'),
to_char(OT_IN_TIME_PUNCH, 'DD/MM/RRRR HH24:MI:SS'),

```

```

to_char(OT_OUT_TIME_PUNCH,'DD/MM/RRRR HH24:MI:SS')
into
I_Wrintime,I_Wrouvertime,I_Wointime,I_Woouttime
from att_in.TBL_OT_ENTRY
where EMPID = I_EmpId
and OT_OR_REG_DATE = to_date(to_char(I_OT_Date - 1,'DD/MM/RRRR'),'DD/MM/RRRR');
end if;

if I_Sts_W in ('L','P') and I_Wointime is not null and I_Woouttime is not null then --or I_Wointime
= '0' and or I_Woouttime = '0'

select to_date(to_char(min(PunchTime),'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
into I_WIN
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = v_comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR')
and to_char(punchtime,'DD/MM/RRRR HH24:MI:SS')> I_Woouttime;

select to_date(to_char(max(PunchTime),'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
into I_WOUT
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = v_comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR')
and to_char(punchtime,'DD/MM/RRRR HH24:MI:SS')> I_Woouttime;

elsif I_Sts_W in ('L','P') and I_Wrouvertime is not null or I_Wrouvertime = '0' and I_Wrouvertime is not
null or I_Wrouvertime= '0' then

select to_date(to_char(min(PunchTime),'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
into I_WIN
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = v_comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR')
and to_char(punchtime,'DD/MM/RRRR HH24:MI:SS')> I_Wrouvertime;

select to_date(to_char(max(PunchTime),'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
into I_WOUT
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = v_comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR')
and to_char(punchtime,'DD/MM/RRRR HH24:MI:SS')> I_Wrouvertime;

elsif I_Sts_W in ('W') and I_Wointime is null or I_Wointime = '0' and I_Woouttime is null or
I_Woouttime = '0' then
select to_date(to_char(min(PunchTime),'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
into I_WIN
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = v_comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR');

select to_date(to_char(max(PunchTime),'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
into I_WOUT
from att_in.TBL_RAW_DATA_CMIS

```

```

where COMPCARDID = v_comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR');
end if;

Select abs(NVL(SUM(datediff('hh',I_WIN,I_WOUT)),0) * 60)
into var_diff_wnight
from dual;

if var_diff_wnight=0 then

select to_date(to_char(min(PunchTime),'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
into I_WOUT
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = v_comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date+1,'DD/MM/RRRR'),'DD/MM/RRRR');
end if;

Select abs(NVL(SUM(datediff('hh',I_WIN,I_WOUT)),0) * 60)
into var_diff_wnight
from dual;

if var_diff_wnight>360 and var_diff_wnight<=510 then

update att_in.TBL_OT_ENTRY
set REG_STATUS = 'P',
R_IN_TIME_PUNCH = I_WIN,
R_OUT_TIME_PUNCH = I_WOUT,
AFTER_OT_LOG = 1
where EMPID = I_EmpId
and OT_OR_REG_DATE = to_date(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR');
commit;

elsif var_diff_wnight>510 and var_diff_wnight<1080 then
select to_date(to_char(I_WIN + interval '480' MINUTE,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS')
into d2_outtime
from dual;

update att_in.TBL_OT_ENTRY
set REG_STATUS = 'P',
R_IN_TIME_PUNCH = I_WIN,
R_OUT_TIME_PUNCH = d2_outtime,
OT_IN_TIME_PUNCH = d2_outtime,
OT_OUT_TIME_PUNCH = I_WOUT,
AFTER_OT_LOG = 1
where EMPID = I_EmpId
and OT_OR_REG_DATE = to_date(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR');
commit;

elsif var_diff_wnight=0 then
update att_in.TBL_OT_ENTRY
set REG_STATUS = 'W'
where EMPID = I_EmpId
and OT_OR_REG_DATE = to_date(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR');

commit;

```

```

end if;
else
update att_in.TBL_OT_ENTRY
set REG_STATUS = 'W'
where EMPID = I_EmpId
and OT_OR_REG_DATE = to_date(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR');
commit;

end if;

else
update att_in.TBL_OT_ENTRY
set REG_STATUS = 'W'
where EMPID = I_EmpId
and OT_OR_REG_DATE = to_date(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR');
commit;
end if;
end if;

begin
select to_date(to_char(R_OUT_TIME_PUNCH,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'),
to_date(to_char(R_IN_TIME_PUNCH,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'),
REG_STATUS
into var_reg_out_punch,var_reg_punch,v_status
from att_in.tbl_ot_entry
where EMPID = I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR')
and OT_IN_TIME_PUNCH is null
and OT_OUT_TIME_PUNCH is null;

if var_reg_punch >=v_shiftin and v_status not in ('A','W') and var_reg_out_punch is not null then
if v_shiftname= '4 Shift-D' and var_reg_punch is not null then

if to_char(var_reg_punch,'HH24:MI:SS') > '02:00:00' and to_char(var_reg_punch,'HH24:MI:SS') <
'10:00:00' then
v_shiftout := to_date(to_char(I_OT_Date,'DD/MM/RRRR')||' 14:00:00','DD/MM/RRRR HH24:MI:SS');

elsif to_char(var_reg_punch,'HH24:MI:SS') > '10:00:00' and to_char(var_reg_punch,'HH24:MI:SS') <
'18:00:00' then
v_shiftout := to_date(to_char(I_OT_Date,'DD/MM/RRRR')||' 22:00:00','DD/MM/RRRR HH24:MI:SS');

elsif to_char(var_reg_punch,'HH24:MI:SS') > '18:00:00' and to_char(var_reg_punch,'HH24:MI:SS') <
'23:59:00' then
v_shiftout := to_date(to_char(I_OT_Date+1,'DD/MM/RRRR')||' 06:00:00','DD/MM/RRRR HH24:MI:SS');
end if;
end if;
end if;

Select NVL(SUM(datediff('hh',v_shiftout,var_reg_out_punch)),0) * 60
into var_ot_calculation
from dual;
if var_ot_calculation > 30 and v_status not in ('A','W') then
update att_in.tbl_ot_entry
set R_OUT_TIME_PUNCH = to_date(to_char(v_shiftout,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS'),

```

```

OT_IN_TIME_PUNCH = to_date(to_char(v_shiftout,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS'),
AFTER_OT_LOG=1,
OT_OUT_TIME_PUNCH = var_reg_out_punch
where EMPID = I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR')
and OT_IN_TIME_PUNCH is null
and OT_OUT_TIME_PUNCH is null;
commit;
end if;
EXCEPTION
when too_many_rows then
null;
when no_data_found then
null;
when others then
null;
end;

begin
select to_date(to_char(R_IN_TIME_PUNCH,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS'),REG_STATUS
into var_reg_punch,v_status
from att_in.tbl_ot_entry
where EMPID = I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR');

if v_shiftname= '4 Shift-D' and var_reg_punch is not null then

if to_char(var_reg_punch,'HH24:MI:SS') > '02:00:00' and to_char(var_reg_punch,'HH24:MI:SS') <
'10:00:00' then
v_shiftin := to_date(to_char(I_OT_Date,'DD/MM/RRRR')||' 06:00:00','DD/MM/RRRR HH24:MI:SS');

elsif to_char(var_reg_punch,'HH24:MI:SS') > '10:00:00' and to_char(var_reg_punch,'HH24:MI:SS') <
'18:00:00' then
v_shiftin := to_date(to_char(I_OT_Date,'DD/MM/RRRR')||' 14:00:00','DD/MM/RRRR HH24:MI:SS');

elsif to_char(var_reg_punch,'HH24:MI:SS') > '18:00:00' and to_char(var_reg_punch,'HH24:MI:SS') <
'23:59:00' then
v_shiftin := to_date(to_char(I_OT_Date,'DD/MM/RRRR')||' 22:00:00','DD/MM/RRRR HH24:MI:SS');
end if;
end if;

Select abs(NVL(SUM(datediff('hh',v_shiftin,var_reg_punch)),0) * 60)
into var_late_calculation
from dual;

if var_reg_punch > v_shiftin and v_status <> 'A' then

if var_late_calculation > 30 then
var_late_calculation_final:=var_late_calculation-30;
ot_min_late := floor(( var_late_calculation_final mod 60));
ot_hour_late := ( var_late_calculation_final - ot_min_late) / 60;
if ot_min_late >= 0 and ot_min_late < 10 then
total_ot_hour_late := to_char('0')||to_char(ot_hour_late)||':0'||to_char(ot_min_late)||':00';

```

```

elseif ot_min_late >=10 and ot_min_late <=59 then
total_ot_hour_late := to_char('0')||to_char(ot_hour_late)||':'||to_char(ot_min_late)||':00';
end if;
update att_in.tbl_ot_entry
set REG_STATUS = 'L',
LATE_ON_REG_SHIFT = total_ot_hour_late
where EMPID = I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR');
commit;
end if;
end if;
EXCEPTION
when too_many_rows then
null;
when no_data_found then
null;
when others then
null;
end;
end;
End;

```

## 8. SP\_GENERAL\_SHIFT

```

CREATE OR REPLACE PROCEDURE ATT_IN.sp_general_shift
(
    I_EmpId In int,
    I_OT_Date In date
)
As
Begin
    declare
        v_count int;
        I_Sts varchar2(10);
        v_short_rest_hour_mt varchar2(40);
        exists_govt_holiday int;
        v_present_diff int;
        current_d_name varchar2(40);
        weekly_holiday_counter int;
        var_ot_calculation_before_ot int:=0;
        V_OTHour_Val varchar2(40);
        V_SRHour_Val varchar2(40):='00:00:00';
        V_BreakIn date;
        V_BreakOut date;
        V_BreakLate date;
        V_RegHour date;
        V_NightShiftDesc varchar2(40);
        V_NumberOfPunch int;
        V_BuyerShiftIn date;
        V_BuyerShiftOut date;
        V_BuyerTime date;
        V_Remarks varchar2(100);
        v_MEALALLOWANCE varchar2(10);
        leave_osd_counter int;
        leave_counter int;

```

```

leave_id int;
osd_counter int;
var_late_calculation_final int;
lTypeName nvarchar2(30);
v_in_time date;
v_out_time date;
v_out_ot_time date;
v_out_ot_time_for_mt date;
v_late int;
v_comp_cardid nvarchar2(30);
v_emp_code nvarchar2(30);
v_designation nvarchar2(30);
v_ot_min int;
v_shiftid int;
v_shift_in_time date;
v_shift_out_time date;
V_OTAPPROVE int;
v_deptid int;
var_ot_calculation int:=0;
V_shiftLATE nvarchar2(30);
V_Status nvarchar2(30);
V_LateVal varchar2(50);
ot_min_late int;
ot_hour_late int;
total_ot_hour nvarchar2(20);
total_ot_hour_late nvarchar2(20);
O_OT_Hour varchar(10);
O_SR_Hour varchar(10):='00:00:00';
--O_OT_Hour int;
o_remarks varchar(50);
V_OTStatus int;
v_otstatus_name varchar(50);
v_mipunchtime_mt date;
v_short_rest_mt int;
v_employee_type nvarchar2(30);
v_employee_category nvarchar2(30);
short_rest_hour_mt int;
v_short_rest_in_out int;

begin

begin
select
EMPCODE,to_char(COMPCARDID),DESIGNATIONID,SHIFTID,DEPTID,EMPLOYEE_TYPE,EMPLOYEE_CATEGORY
into
v_emp_code,v_comp_cardid,v_designation,v_shiftid,v_deptid
,v_employee_type,v_employee_category
from att_in.TBL_EMP_PERSONAL_INFO
where EMPID = I_EmpId;
EXCEPTION
when too_many_rows then
null;
when no_data_found then
null;
when others then
null;
end;
end;

```

```

begin
    select OTSTATUS into v_otstatus_name from
att_in.TBL_LABOUR TL,att_in.TBL_EMP_PERSONAL_INFO TE
    where TL.LABOURCODE=TE.LABOURCODE
    and Te.EMPID=I_EmpId;
    if upper(v_otstatus_name)='YES'
    then
        V_OTStatus :=1;
    else
        V_OTStatus :=0;
    END IF;
EXCEPTION
    when too_many_rows then
        null;
    when no_data_found then
        null;
    when others then
        null;

end;

begin
    select distinct to_date(to_char(LOGINTIME,'
HH24:MI:SS'),' HH24:MI:SS'),
        to_date(to_char(LOGOUTTIME,'HH24:MI:SS'),'HH24:MI:SS')
    into v_shift_in_time,v_shift_out_time
    from att_in.TBL_SHIFT_SETUP TS
    where TS.SHIFTID = v_shiftid
    and ts.deptid=v_deptid;

EXCEPTION

    when no_data_found then
        null;
    when others then
        null;
end;

v_shift_in_time := to_date(to_char(I_OT_Date,'DD/MM/RRRR')
||to_char( v_shift_in_time,'HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS');
v_shift_out_time :=
to_date(to_char(I_OT_Date,'DD/MM/RRRR') ||to_char(
v_shift_out_time,'HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS');

begin
    select count(DAYNAME) into weekly_holiday_counter
    from att_in.TBL_WEEKLY_HOLIDAY
    where upper(substr(DAYNAME,1,3)) =
upper(substr(current_d_name,1,3)) ;

EXCEPTION
    when no_data_found then
        null;
    when others then

```



```

        null;
    end;

    begin
        select to_date(to_char(min(PunchTime), 'DD/MM/RRRR
HH24:MI:SS'), 'DD/MM/RRRR HH24:MI:SS')
        into v_in_time
        from att_in.TBL_RAW_DATA_CMIS
        where COMPCARDID = v_comp_cardid
        and PUNCHDATE =
to_date(to_char(I_OT_Date, 'DD/MM/RRRR'), 'DD/MM/RRRR');

        EXCEPTION
        when too_many_rows then
            null;
        when no_data_found then
            null;
        when others then
            null;
    end;

    begin
        select to_date(to_char(max(PunchTime), 'DD/MM/RRRR
HH24:MI:SS'), 'DD/MM/RRRR HH24:MI:SS')
        into v_out_time
        from att_in.TBL_RAW_DATA_CMIS
        where COMPCARDID = v_comp_cardid
        and PUNCHDATE =
to_date(to_char(I_OT_Date, 'DD/MM/RRRR'), 'DD/MM/RRRR');

        EXCEPTION
        when too_many_rows then
            null;
        when no_data_found then
            null;
        when others then
            null;
    end;

    begin
        select count(punchtime)
        into v_count
        from att_in.TBL_RAW_DATA_CMIS
        where COMPCARDID = v_comp_cardid
        and PUNCHDATE =
TO_DATE(to_char(I_OT_Date, 'DD/MM/YYYY'), 'DD/MM/YYYY');
        EXCEPTION
        when too_many_rows then
            null;
        when no_data_found then
            null;
        when others then

```

```

        null;
    end;

    if v_in_time = v_out_time then
        v_out_time := '';
    end if;

    60) Select (NVL(SUM(datediff('hh',v_in_time,v_out_time)),0) *
        into v_present_diff
        from dual;

    if v_in_time is not null and v_out_time is not null and
v_count>1 then
        v_status:='P';
    else
        v_status:='A';
    end if;

    -----Late Calculation Start-----
    -----

    if v_in_time > v_shift_in_time and v_count>1 then
        Select
        (NVL(SUM(datediff('hh',v_shift_in_time,v_in_time)),0) * 60)
        into v_late
        from dual;

        dbms_OUTPUT.PUT_LINE(' v_late');
        dbms_OUTPUT.PUT_LINE( v_late);
        --dbms_OUTPUT.PUT_LINE('O_OT_Hour');
        --dbms_OUTPUT.PUT_LINE( O_OT_Hour);

        if v_late > 30 then

            var_late_calculation_final:=v_late-30;
            ot_min_late := floor((
var_late_calculation_final mod 60));

            ot_hour_late := ( var_late_calculation_final -
ot_min_late) / 60;

            if ot_hour_late>9 then
                if ot_min_late >= 0 and ot_min_late < 10 then
                    total_ot_hour_late :=
to_char(ot_hour_late)||':0'||to_char(ot_min_late)||':00';
                elsif ot_min_late >=10 and ot_min_late <=59
then
                    total_ot_hour_late :=
to_char(ot_hour_late)||':'||to_char(ot_min_late)||':00';
                end if;
            elsif ot_hour_late<=9 then

```

```

                                if ot_min_late >= 0 and ot_min_late < 10
then
                                total_ot_hour_late :=
to_char('0' || to_char(ot_hour_late) || ':' || to_char(ot_min_late) || ':00');
                                elsif ot_min_late >= 10 and ot_min_late <= 59
then
                                total_ot_hour_late :=
to_char('0' || to_char(ot_hour_late) || ':' || to_char(ot_min_late) || ':00');
                                end if;
                                end if;
                                v_status:='L';

                                end if;
                                end if;

                                -----Late Calculation End-----
-----

                                if v_in_time is null and v_out_time is null then
Start-----
                                -----Leave and OSD

                                select
ATT_IN.fn_Leave_OSD_Count(I_OT_Date,I_OT_Date,I_EmpId) into leave_osd_counter from dual;

                                if leave_osd_counter > 0 then
                                begin
                                select
count(EMPID) into leave_counter
                                from
att_in.TBL_LEAVE_APPLY
                                where EMPID =
I_EmpId
                                and FROMDATE <=
TO_DATE(to_char(I_OT_Date, 'DD/MM/RRRR'), 'DD/MM/RRRR')
                                and TODATE >=
TO_DATE(to_char(I_OT_Date, 'DD/MM/RRRR'), 'DD/MM/RRRR');
                                EXCEPTION
                                when too_many_rows
then
                                null;
                                when no_data_found
then
                                null;
                                when others then
                                null;
                                end;

                                if leave_counter
> 0 then
                                begin
                                select LEAVETYPEID into leave_id

```

```

from att_in.TBL_LEAVE_APPLY

where EMPID = I_EmpId

and FROMDATE <= TO_DATE(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR')

and TODATE >= TO_DATE(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR');

EXCEPTION

when too_many_rows then

null;

when no_data_found then

null;

when others then

null;

end;

begin

select REPORTINGNAME into lTypeName

from att_in.TBL_LEAVE_TYPE

where LEAVETYPEID = leave_id;

EXCEPTION

when too_many_rows then

null;

when no_data_found then

null;

when others then

null;

end;

v_status:=upper(lTypeName);

else

begin

select count(EMPID) into osd_counter

from att_in.TBL_OSD_SETUP

where EMPID = I_EmpId

```

```

and OSDSTARTDATE <= TO_DATE(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR')
and OSDENDDATE >= TO_DATE(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR');

EXCEPTION

when too_many_rows then

null;

when no_data_found then

null;

when others then

null;

end;

if

osd_counter > 0 then

v_status:='OCS';

end

if;

end if;

-----

Leave and OSD End -----

else

-----Gogt

Holiday start -----

select
count(NVL(GOVTHOLIID,'0')) into exists_govt_holiday from att_in.tbl_Govt_Holiday_List
where
TO_DATE(GovtHoliStartDate,'DD/MM/YYYY') <= TO_DATE(I_OT_Date,'DD/MM/YYYY')
and
TO_DATE(GovtHoliEndDate,'DD/MM/YYYY') >= TO_DATE(I_OT_Date,'DD/MM/YYYY');

if

exists_govt_holiday=1 Then

v_status:=
'GHD';

END IF;

-----Gogt

Holiday End -----

-----

weekly Off start -----

select
upper(to_char(I_OT_Date,'DAY')) into current_d_name from dual;

```

```

weekly_holiday_counter      select count(DAYNAME) into
                             from
att_in.TBL_WEEKLY_HOLIDAY    where
upper(substr(DAYNAME,1,3)) = upper(substr(current_d_name,1,3)) ;

                             if weekly_holiday_counter >0
then
                             v_status:='W';
                             end if;

                             -----weekly
off end-----

                             end if;

                             end if;

begin
select  count(OTAPPROVE) into V_OTAPPROVE
from att_in.tbl_emp_ot_approve
where EMPID=I_EmpId and otdat=I_OT_Date;

EXCEPTION
  when too_many_rows then
    null;
  when no_data_found then
    null;
  when others then
    null;
end;

-----before ot start-----
-----

if v_in_time < v_shift_in_time then

  Select
(NVL(SUM(datediff('hh',v_in_time,v_shift_in_time)),0) * 60)
  into var_ot_calculation_before_ot
  from dual;

  if var_ot_calculation_before_ot>30 then
    var_ot_calculation_before_ot:=30;
  else
    var_ot_calculation_before_ot:=0;

  end if;
end if;

-----before ot end-----
-----after ot start-----
-----

if V_OTAPPROVE > 0 and V_OTStatus=1 and
weekly_holiday_counter =0 then

```

```

                                if V_ShiftId <> 602 then

                                    if to_char(v_out_time, 'HH24:MI') >'21:00'
then
                                        v_out_ot_time:= to_date(to_char(I_OT_Date
, 'DD/MM/RRRR') || '21:00:00', 'DD/MM/RRRR HH24:MI:SS');
                                        else
                                        v_out_ot_time:=v_out_time;
                                        end if;

                                        if v_out_ot_time>v_shift_out_time then
Select
(NVL(SUM(datediff('hh',v_shift_out_time,v_out_ot_time)),0) * 60)
into var_ot_calculation
from dual;
                                        end if;

                                        if var_ot_calculation>30 then
                                        var_ot_calculation:=var_ot_calculation;
                                        else
                                        var_ot_calculation:=0;
                                        end if;

                                elsif V_ShiftId = 602 then

                                    if to_char(v_out_time, 'HH24:MI') >'22:00' then
                                        v_out_ot_time_for_mt:=
to_date(to_char(I_OT_Date , 'DD/MM/RRRR') || '22:00:00', 'DD/MM/RRRR HH24:MI:SS');
                                        else
                                        v_out_ot_time_for_mt:= v_out_time;
                                        end if;

                                    if v_out_ot_time_for_mt>v_shift_out_time then

                                        Select
(NVL(SUM(datediff('hh',v_shift_out_time,v_out_ot_time_for_mt)),0) * 60)
                                        into var_ot_calculation
                                        from dual;

                                        end if;

                                    if var_ot_calculation > 480 then
                                        var_ot_calculation:= 480;
                                    else
                                        var_ot_calculation := var_ot_calculation;
                                    end if;
                                end if;
                                end if;

                                -----after ot end -----
                                -----for Employee -----

```

```

if V_OTAPPROVE>0 and V_OTStatus=1 and v_designation
<>'MLSS' then

    V_OTHour_Val
:=MakeTimeFromMinute(var_ot_calculation);

    end if;
    if v_designation ='MLSS' then

        if V_OTAPPROVE>0 and V_OTStatus=1 and
v_status<>'A'then
            V_OTHour_Val
:=MakeTimeFromMinute(var_ot_calculation_before_ot+var_ot_calculation);
            end if;

        end if;
        -----For Mlss -----

sp_OT_Edit(I_EmpId,I_OT_Date,V_OTHour_Val,V_SRHour_Val,O_OT_Hour,O_SR_Hour,o_remarks);
v_MEALALLOWANCE:=FN_Mealallowance(I_EmpId,v_out_time,V_ShiftId,O_OT_Hour);

        dbms_OUTPUT.PUT_LINE('V_OTHour_Val');
        dbms_OUTPUT.PUT_LINE( V_OTHour_Val);
        dbms_OUTPUT.PUT_LINE('O_OT_Hour');
        dbms_OUTPUT.PUT_LINE( O_OT_Hour);

INSERT INTO att_in.tbl_Processed_Data
(
    EmpId ,    PunchDate , TimeIn ,    TimeOut , ShiftIn
, ShiftOut ,
    BreakIn , BreakOut , BreakLate , Late ,    ShiftId
, RegHour ,
    OTHour , Status ,    NumPunch , NightShiftDesc
,OTShiftDesc ,
    BuyerShiftIn,BuyerShiftOut , BuyerTime,
Remarks,MEALALLOWANCE
)
VALUES
(
    I_EmpId,
I_OT_Date,v_in_time,v_out_time,v_shift_in_time,v_shift_out_time,
V_BreakIn,V_BreakOut,V_BreakLate,total_ot_hour_late,v_shiftid,V_RegHour,
O_OT_Hour,v_status,V_NumberOfPunch,V_NightShiftDesc,'',
V_BuyerShiftIn,V_BuyerShiftOut,V_BuyerTime,V_Remarks,v_MEALALLOWANCE
);
Commit;

    if v_shiftid=602 and v_deptid =24 and v_employee_type
in('P','G') and v_employee_category='Permanent' then
        v_short_rest_mt:=0;
        short_rest_hour_mt:=0;

```



```

abs((NVL(SUM(datediff('hh',v_out_time,v_in_time)),0) * 60))
into v_short_rest_in_out
from dual;

dbms_OUTPUT.PUT_LINE('v_short_rest_in_out');

dbms_OUTPUT.PUT_LINE(v_short_rest_in_out);

if v_short_rest_in_out>510 then

BEGIN

select
to_date(to_char(min(PUNCHTIME),'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
into v_mipunchtime_mt
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = v_comp_cardid
and PUNCHDATE =
to_date(to_char(I_OT_Date+1,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS') ;

EXCEPTION
when no_data_found then
null;
End;

Select
abs((NVL(SUM(datediff('hh',v_out_time,v_mipunchtime_mt)),0) * 60))
into v_short_rest_mt
from dual;

dbms_OUTPUT.PUT_LINE('maxpunchtime');
dbms_OUTPUT.PUT_LINE(to_char(
v_mipunchtime_mt,'DD/MM/RRRR HH24:MI:SS'));
dbms_OUTPUT.PUT_LINE('v_in_time');
dbms_OUTPUT.PUT_LINE(to_char(v_in_time,'DD/MM/RRRR HH24:MI:SS'));

end if;

DBMS_OUTPUT.PUT_LINE ( v_short_rest_hour_mt);
if v_short_rest_mt>=30 and v_short_rest_mt
< 720 and v_short_rest_in_out>510 then

short_rest_hour_mt := 720 -
v_short_rest_mt;

if short_rest_hour_mt >= 240 then
short_rest_hour_mt := 240;
elsif short_rest_hour_mt < 30 then
short_rest_hour_mt := 0;
else
short_rest_hour_mt :=
short_rest_hour_mt;

end if;

--v_short_rest_hour_mt
:=ATT IN.fn minute to time shortrest ot(short rest hour mt);

```

```

v_short_rest_mt);
ATT_IN.fn_minute_to_time_shortrest_ot(short_rest_hour_mt)
to_date(to_char(I_OT_Date, 'DD/MM/RRRR'), 'DD/MM/RRRR');

        DBMS_OUTPUT.PUT_LINE ( 'SHORT REST');
        DBMS_OUTPUT.PUT_LINE (
                update att_in.TBL_PROCESSED_DATA
                set     EXTRAOT =
                where empid= I_EmpId
                and PUNCHDATE =

                commit;
        DBMS_OUTPUT.PUT_LINE ( 'SHORT REST');
        --DBMS_OUTPUT.PUT_LINE

fn_minute_to_time_shortrest_ot( v_short_rest_mt);

        end if;
        --DBMS_OUTPUT.PUT_LINE ( 'SHORT REST
CALCULATION STARTED');
        --DBMS_OUTPUT.PUT_LINE (
ATT_IN.fn_minute_to_time_shortrest_ot(short_rest_hour_mt));

        begin
                select STATUS      --25-09-0213
                into I_Sts
                from att_in.TBL_PROCESSED_DATA
                where EMPID = I_EmpId
                and PUNCHDATE =

to_date(to_char(I_OT_Date, 'DD/MM/RRRR'), 'DD/MM/RRRR');
                ----DBMS_OUTPUT.PUT_LINE('shift');
                ----DBMS_OUTPUT.PUT_LINE(I_Sts);
                ----DBMS_OUTPUT.PUT_LINE(I_OTLog);

                if I_Sts in ( 'A'
, 'W', 'CL', 'SL', 'PL', 'D/O', 'OCS', 'GHD', 'ML', 'AL', 'QL', 'STL', 'SPL', 'SWPL', 'JL', 'DFL', 'SSL')
then
                UPDATE att_in.TBL_PROCESSED_DATA
                SET EXTRAOT = ''
                where EMPID=I_EmpId
                and PUNCHDATE =

to_date(to_char(I_OT_Date, 'DD/MM/RRRR'), 'DD/MM/RRRR');

                commit;
                END IF;

                EXCEPTION when no_data_found then
                        null;
                end;

        end if;

        -----Enf of Short Rest for MT

end;

end sp_general_shift;
/

```

## 9. SP\_GOVTHOLIDAY\_ADD\_UPDATE

```
CREATE OR REPLACE PROCEDURE ATT_IN.sp_GovtHoliday_Add_Update
(
    I_GovtHoliId In Int,
    I_CompId In Int,
    I_CompFinYId In Int,
    I_GovtHoliName In varchar2,
    I_GovtHoliStartDate In date,
    I_GovtHoliEndDate In date,
    I_GovtNumberDays In float,
    I_GovtRemarks In varchar2,
    I_CreateBy In varchar2,
    I_UpdateBy In varchar2,
    I_AddOrUpdate In varchar2,
    I_CompanyHoliday In Int

    --O_refcursor_out OUT SYS_REFCURSOR
)
is
    BEGIN

        declare Count_No int:=0; O_retVal int :=0;
        BEGIN

            IF I_AddOrUpdate='Saved'
            then
                Select count(GovtHoliId) into Count_No from tbl_Govt_Holiday_List
where GovtHoliName= I_GovtHoliName;

                if Count_No = 0
                then
                    Insert Into tbl_Govt_Holiday_List
                    (
                        CompId,CompFinYId,GovtHoliName,GovtHoliStartDate,GovtHoliEndDate,GovtNumberDays,GovtRemarks,CreateBy, CreateDate, UpdateBy, UpdateDate,CompanyHoliday
                    )
                    values
                    (
                        I_CompId,I_CompFinYId,I_GovtHoliName,I_GovtHoliStartDate,I_GovtHoliEndDate,I_GovtNumberDays,I_GovtRemarks,I_CreateBy, sysdate, I_UpdateBy, sysdate,I_CompanyHoliday
                    );
                    commit;
                    O_retVal := 0;
                else
                    O_retVal := 1;
                END IF;
            Else
                -----Update
                update tbl_Govt_Holiday_List
                set CompId = I_CompId,CompFinYId = I_CompFinYId ,GovtHoliName =
```

```

I_GovtHoliName ,
                GovtHoliStartDate = I_GovtHoliStartDate ,GovtHoliEndDate =
I_GovtHoliEndDate ,
                GovtNumberDays = I_GovtNumberDays,GovtRemarks = I_GovtRemarks,
UpdateBy=I_UpdateBy,UpdateDate=sysdate,CompanyHoliday=I_CompanyHoliday
                where GovtHoliId=I_GovtHoliId;

                Commit;

        END IF;
        --open O_refcursor_out for select O_retVal from dual;
END;

EXCEPTION
WHEN OTHERS THEN
RAISE_APPLICATION_ERROR (-20001,
I_GovtHoliId || ':$:' ||
I_CompId || ':$:' ||
I_CompFinYId || ':$:' ||
I_GovtHoliName || ':$:' ||
I_GovtHoliStartDate || ':$:' ||
I_GovtHoliEndDate || ':$:' ||
I_GovtNumberDays || ':$:' ||
I_GovtRemarks || ':$:' ||
I_CreateBy || ':$:' ||
I_UpdateBy || ':$:' ||
I_AddOrUpdate || ':$:' ||
I_CompanyHoliday || ':$:' ||

SQLERRM, TRUE) ;

end sp_GovtHoliday_Add_Update;
/

```

## 10. SP\_INSERT\_ROWDATA

```

CREATE OR REPLACE PROCEDURE ATT_IN.sp_insert_rowdata
(I_comcard varchar2,I_fromdate date,I_todate date)
as
v_COMPCARDID varchar2(20);
v_PUNCHDATE date;
v_PUNCHTIME date;
v_LOC_ID varchar2(20);
v_INOUT varchar2(20);
v_OVNMARK int;
v_REMARKS varchar2(100);
v_MANUALENTRYTAG int;
v_CREATEBY varchar2(100);
v_UPDATEDEMPLOYEE int;
v_ROW_ID number(20);
v_MACHINENAME varchar2(100);
v_ENTRYDATE date;

begin
declare

cursor new_rowdata is select COMPCARDID,PUNCHDATE,PUNCHTIME,LOC_ID,INOUT,OVNMARK,REMARKS,
MANUALENTRYTAG,CREATEBY,UPDATEDEMPLOYEE,ROW_ID,MACHINENAME,ENTRYDATE from TBL_RAW_DATA

```

```

where COMPCARDID=I_comcard and
PUNCHDATE BETWEEN TO_DATE(to_char(I_fromdate, 'MM/DD/YYYY HH24:MI:SS'), 'MM/DD/YYYY
HH24:MI:SS')
AND TO_DATE(to_char(I_todate+1, 'MM/DD/YYYY HH24:MI:SS'), 'MM/DD/YYYY HH24:MI:SS')
order by PUNCHTIME ;

begin
open new_rowdata;
loop
    FETCH new_rowdata INTO
v_COMPCARDID,v_PUNCHDATE,v_PUNCHTIME,v_LOC_ID,v_INOUT,v_OVNMARK,v_REMARKS,v_MANUALENTRYTA
G,v_CREATEBY,v_UPDATEDEMPLOYEE,v_ROW_ID,v_MACHINENAME,v_ENTRYDATE;
    EXIT WHEN new_rowdata%NOTFOUND;
    --if v_COMPCARDID='000001158' then
        insert into
TBL_RAW_DATA_CMIS (COMPCARDID,PUNCHDATE,PUNCHTIME,LOC_ID,INOUT,OVNMARK,REMARKS,
MANUALENTRYTAG,CREATEBY,UPDATEDEMPLOYEE,ROW_ID,MACHINENAME,ENTRYDATE)
values(v_COMPCARDID,v_PUNCHDATE,v_PUNCHTIME,v_LOC_ID,v_INOUT,v_OVNMARK,v_REMARKS,v_MANUAL
ENTRYTAG,v_CREATEBY,v_UPDATEDEMPLOYEE,v_ROW_ID,v_MACHINENAME,v_ENTRYDATE);
        commit;
    --end if;
end loop;
close new_rowdata;
end;

--EXECUTE att_in.sp_insert_rowdata('000037123','01-nov-2013','30-nov-2013');
end;
/

```

## 11.SP\_LEAVE\_APPLIED\_ADD\_UPDATE

```

CREATE OR REPLACE PROCEDURE ATT_IN.sp_Leave_Applied_Add_Update
(
    I_ApplyNumber In int,
    I_EmpId In int,
    I_FromDate In Date,
    I_ToDate In date,
    I_Avail In number,
    I_LeaveTypeId In int,
    I_Remarks In varchar2,
    I_EntryBy In varchar2,
    I_UpdateBy In varchar2,
    I_AddOrUpdate In varchar2,
    I_CompId In int,
    I_previousLvTypeId In int,
    I_FinYName In varchar2,
    I_EmpCode in varchar2,
    I_DeptId In int,
    I_PlantId In int
    --O_refcursor_retVal OUT SYS_REFCURSOR
)
as
BEGIN

    --Set serveroutput on;

```

```

/* Declare
   I_ApplyNumber int :=42;
   I_EmpId int :=1;
   I_FromDate Date :=TO_Date('05/11/2012','DD-MM-RR');
   I_ToDate date :=TO_Date('07/11/2012','DD-MM-RR');

   I_LeaveTypeId int :=1;
   I_Approval int;
   I_Remarks varchar2(500);
   I_ApproveBy int;
   I_EntryBy varchar2(50);
   I_UpdateBy varchar2(50);
   --I_AddOrUpdate varchar2(50) :='Saved';
   I_AddOrUpdate varchar2(50) :='Update';
   I_CompId int :=3;
   I_previousLvTypeId int :=1;
   I_FinYName varchar2(50) :='2011-2011';*/

```

```

--
--Variable Declaration
--

```

Declare

```

I_Approval int;
I_ApproveBy int;
I_Avail number(5,2);
V_Used_Balance number(5,2);
V_Max_Balance number(5,2);

V_TB_Num_days number(5,2);
V_TB_useBalance number(5,2);
V_TB_totlUsedLeave number(5,2);
V_applyDuration number(5,2);
V_remainBalance number(5,2);
V_existRemainBalnce int;
V_TB_RemainingTotalDays number(5,2);
V_TB_UpdatedBalance number(5,2);
O_retVal int;

V_RollBackLeavdeBalance number(5,2);
V_FinalLeaveBalance number(5,2);
v_days int :=0;
V_DateIncrement date:=I_FromDate ;

BEGIN
   I_Approval :=1;
   I_ApproveBy :=1;
   O_retVal:=0;

   -- I_Avail := I_ToDate - I_FromDate; I_Avail,
   select (I_ToDate - I_FromDate)+1 into v_days from dual;
   IF I_AddOrUpdate='Saved'
   Then
      for i in 1..v_days

```

```

        loop
        INSERT INTO tbl_Leave_Apply
        (
            EmpId,LeaveTypeId,FromDate,ToDate,NumberOfDays,Approval,
Cause,CreateBy,CreateDate,UpdateBy,UpdateDate,COMPFINYID,EMPCODE,DEPTID,PLANT_ID
        )
        VALUES
        (
            I_EmpId, I_LeaveTypeId,V_DateIncrement,V_DateIncrement,
I_Avail,I_Approval,I_Remarks,I_EntryBy,sysdate,I_UpdateBy,sysdate,I_FinYName,I_EmpCode,I_
DeptId,I_PlantId
        );
        commit;
            V_DateIncrement:=V_DateIncrement+1;
        end loop;
    else

        UPDATE tbl_Leave_Apply
        SET
            FromDate = I_FromDate,
            ToDate = I_ToDate,
            NumberOfDays =I_Avail,
            LeaveTypeId =I_LeaveTypeId,
            --CreateBy = I_EntryBy,
            UpdateBy =I_UpdateBy
        WHERE EmpId = I_EmpId
        and LeaveApplyId =I_ApplyNumber;
        Commit;

    end if;
end;

End sp_Leave_Applied_Add_Update;
/

```

## 12.SP\_LEAVE\_TYPE\_ADD\_UPDATE

```

CREATE OR REPLACE PROCEDURE ATT_IN.sp_Leave_Type_Add_Update
(
    I_LeaveTypeId In Int,
    I_CompId In Int,
    I_LeaveTypeName In varchar2,
    I_ReportingName In varchar2,
    --I Balance In float,
    --I_CarryForwardStatus In Int,
    --I_FSpecific In Int,
    --I Interchangeable In Int,
    --I IsPaid In Int,
    I_CreateBy In varchar2,
    I_UpdateBy In varchar2,
    I_AddOrUpdate In varchar2
    --O_refcursor_retVal OUT SYS_REFCURSOR
)
is
BEGIN

```

```

declare Count_No int:=0; O_retVal int :=0; LeaveTypeNo int :=0;
BEGIN

    IF I_AddOrUpdate='Saved'
    then
        Select count (LeaveTypeId) into Count_No from tbl_Leave_Type
where LeaveTypeName=I_LeaveTypeName and CompId= I_CompId;

        if Count_No = 0
        then

            --Not Duplicate
            Insert Into tbl_Leave_Type
            (
                --
CompId,LeaveTypeName,ReportingName,Balance,CarryForwardStatus,
                --
FSpecific,Interchangable,IsPaid,CreateBy,CreateDate,UpdateBy,UpdateDate
CompId,LeaveTypeName,ReportingName,CreateBy,CreateDate,UpdateBy,UpdateDate

            )

            values
            (
                I_CompId,
                I_LeaveTypeName,
                I_ReportingName,
                --I_Balance,
                --I_CarryForwardStatus,
                --I_FSpecific,
                --I_Interchangable,
                --I_IsPaid,
                I_CreateBy,
                sysdate,
                I_UpdateBy,
                sysdate
            );

            commit;
            O_retVal := 0;
        else
            O_retVal :=1;

        END IF;
    Else
        -----Update
        Update tbl_Leave_Type
        SET CompId =I_CompId,
        LeaveTypeName = I_LeaveTypeName,
        ReportingName = I_ReportingName,
        --Balance =I_Balance,
        --CarryForwardStatus =I_CarryForwardStatus,
        --FSpecific = I_FSpecific,
        --Interchangable =I_Interchangable,
        --IsPaid =I_IsPaid,
        UpdateBy = I_UpdateBy,

```



```

        UpdateDate=sysdate
        where LeaveTypeId = I_LeaveTypeId;

        Commit;

    END IF;
    --open O_refcursor_retVal for select O_retVal from dual;
END;

EXCEPTION
WHEN OTHERS THEN
RAISE_APPLICATION_ERROR (-20001,

    I_LeaveTypeId || ':$:' ||
    I_CompId || ':$:' ||
    I_LeaveTypeName || ':$:' ||
    I_ReportingName || ':$:' ||
    --I_Balance || ':$:' ||
    --I_CarryForwardStatus || ':$:' ||
    --I_FSpecific || ':$:' ||
    --I_Interchangable || ':$:' ||
    --I_IsPaid || ':$:' ||
    I_CreateBy || ':$:' ||
    I_UpdateBy || ':$:' ||
    I_AddOrUpdate || ':$:' ||

    SQLERRM, TRUE) ;

end sp_Leave_Type_Add_Update;
/

```

### 13.SP\_MANUAL\_INSERT

```

CREATE OR REPLACE PROCEDURE ATT_IN.sp_Manual_Insert
(
    I_IdentifyName In Int,
    I_IdentifyString In varchar2,
    I_PunchFromDate In Date,
    I_PunchToDate In date,
    I_PunchTime In date,
    I_PunchTimeOut In date,
    I_CreateBy In varchar2,
    I_Remarks In varchar2,
    I_AddOrUpdate In varchar2,
    I_rowId In int,
    I_MachineName In varchar2,
    I_CompcardId In varchar2
    --O_refcursor_out OUT SYS_REFCURSOR
)
is
BEGIN
    declare V_SQLEmp varchar2(5000);
            CompCardId varchar2(50);
            EmpCode varchar2(50); O_retVal int :=0;
            V_Pre_Time date; V_CompCardId varchar2(100);
            v_days int:=0;V_dayStart int:=0;
            -- I_SearchString varchar2(500);
            ---
            --- for testing

```

```

--
/*I_IdentifyName  Int:=0;
I_IdentifyString  varchar2(20):='where 1=1';
I_PunchFromDate  Date:='24-May-12';
I_PunchToDate    date:='24-May-12';
I_PunchTime      date:= to_date('26-MAY-12 06:00:00','DD-MON-YYYY
HH24:MI:SS');
I_PunchTimeOut   date:=to_date('26-MAY-12 14:00:00','DD-MON-YYYY
HH24:MI:SS');
I_CreateBy      varchar2(20):='Sam';
I_Remarks       varchar2(20):='test';
I_AddOrUpdate   varchar2(20):='Saved';
I_rowId         int:=0;
I_MachineName   varchar2(10):='soft4';*/
V_DateIncrement date:=I_PunchFromDate ;
v_punchtime     date:=I_PunchTime ;
v_punchtimeout  date:=I_PunchTimeOut ;
-- select to_date(I_PunchToDate,'DD-MM-RR')- to_date(I_PunchFromDate,'DD-
MM-RR') into v_days from dual;
-- v_days :=to_date(I_PunchToDate,'DD-MM-RR')-
to_date(I_PunchFromDate,'DD-MM-RR') ;
--
--cursor declare 1
--
--TYPE EmpCurTyp IS REF CURSOR;
--curEmp1 EmpCurTyp;
BEGIN

-- I_SearchString:=' where 1=1 and DeptId=2';
if I_AddOrUpdate='Saved' Then
  if I_PunchTime is not null and I_PunchTimeOut is not null then
    select (I_PunchToDate - I_PunchFromDate)+1 into v_days from dual;
    --V_SQLEmp := 'select CompCardId,EmpCode from tbl_Emp_Personal_Info
' || I_IdentifyString;
    --cursor
    --DBMS_OUTPUT.PUT_LINE(V_SQLEmp);
    --OPEN curEmp1 FOR V_SQLEmp;
    -- LOOP
      --FETCH curEmp1 INTO CompCardId,EmpCode;
      --EXIT WHEN curEmp1%NOTFOUND;
      V_dayStart:=0;
      V_DateIncrement:=I_PunchFromDate;
      v_punchtime:=I_PunchTime ;
      v_punchtimeout:=I_PunchTimeOut ;
      while V_dayStart< v_days
      Loop
        insert into tbl_Raw_Data
        (
          CompCardId,
          PunchDate,
          PunchTime,
          Loc_Id,
          InOut,
          REMARKS,
          ManualEntryTag,
          CreateBy,
          MachineName,
          EntryDate)
          values

```

```

        (
            I_CompcardId,
            V_DateIncrement,
            v_punchtime,
            '0',
            '0',
            I_Remarks,
            1,
            I_CreateBy,
            I_MachineName,
            sysdate()
        );
        commit;
        insert into tbl_Raw_Data
        (
            CompCardId,
            PunchDate,
            PunchTime,
            Loc_Id,
            InOut,
            REMARKS,
            ManualEntryTag,
            CreateBy,
            MachineName,
            EntryDate
        )
        values
        (
            I_CompcardId,
            V_DateIncrement,
            v_punchtimeout,
            '0',
            '0',
            I_Remarks,
            1,
            I_CreateBy,
            I_MachineName,
            sysdate()
        );
        commit;
        V_DateIncrement:=V_DateIncrement+1;
        --I_PunchFromDate:=V_DateIncrement;
        v_punchtime:=v_punchtime+1;
        v_punchtimeout:=v_punchtimeout+1;
        V_dayStart:=V_dayStart+1;
        end loop;
    elsif I_PunchTime is not null then
        select (I_PunchToDate - I_PunchFromDate)+1 into v_dayses from
dual;
        --V_SQLEmp := 'select CompCardId,EmpCode from
tbl_Emp_Personal_Info ' || I_IdentifyString;
        --cursor
        --DBMS_OUTPUT.PUT_LINE(V_SQLEmp);
        --OPEN curEmp1 FOR V_SQLEmp;
        -- LOOP
        --FETCH curEmp1 INTO CompCardId,EmpCode;
        --EXIT WHEN curEmp1%NOTFOUND;
        V_dayStart:=0;
        V_DateIncrement:=I_PunchFromDate;

```

```

        v_punchtime:=I_PunchTime ;
        v_punchtimeout:=I_PunchTimeOut ;
        while V_dayStart< v_dayes

Loop
    insert into tbl_Raw_Data
    (
        CompCardId,
        PunchDate,
        PunchTime,
        Loc_Id,
        InOut,
        REMARKS,
        ManualEntryTag,
        CreateBy,
        MachineName,
        EntryDate)
    values
    (
        I_CompcardId,
        V_DateIncrement,
        v_punchtime,
        '0',
        '0',
        I_Remarks,
        1,
        I_CreateBy,
        I_MachineName,
        sysdate()
    );
    commit;
    V_DateIncrement:=V_DateIncrement+1;
    --I_PunchFromDate:=V_DateIncrement;
    v_punchtime:=v_punchtime+1;
    v_punchtimeout:=v_punchtimeout+1;
    V_dayStart:=V_dayStart+1;
    end loop;
elseif I_PunchTimeOut is not null then
    select (I_PunchToDate - I_PunchFromDate)+1 into v_dayes from
dual;

        --V_SQLEmp := 'select CompCardId,EmpCode from
tbl_Emp_Personal_Info ' || I_IdentifyString;
        --cursor
        --DBMS_OUTPUT.PUT_LINE(V_SQLEmp);
        --OPEN curEmp1 FOR V_SQLEmp;
        -- LOOP
        --FETCH curEmp1 INTO CompCardId,EmpCode;
        --EXIT WHEN curEmp1%NOTFOUND;
        V_dayStart:=0;
        V_DateIncrement:=I_PunchFromDate;
        v_punchtime:=I_PunchTime ;
        v_punchtimeout:=I_PunchTimeOut ;
        while V_dayStart< v_dayes

Loop

    insert into tbl_Raw_Data
    (
        CompCardId,

```

```

        PunchDate,
        PunchTime,
        Loc_Id,
        InOut,
        REMARKS,
        ManualEntryTag,
        CreateBy,
        MachineName,
        EntryDate
    )
    values
    (
        I_CompcardId,
        V_DateIncrement,
        v_punchtimeout,
        '0',
        '0',
        I_Remarks,
        1,
        I_CreateBy,
        I_MachineName,
        sysdate()
    );
    commit;
    V_DateIncrement:=V_DateIncrement+1;
    --I_PunchFromDate:=V_DateIncrement;
    v_punchtime:=v_punchtime+1;
    v_punchtimeout:=v_punchtimeout+1;
    V_dayStart:=V_dayStart+1;
    end loop;

    --O_retVal := 0;

    end if;

    end if;

    --open O_refcursor_out for select O_retVal from dual;
END;
EXCEPTION
WHEN OTHERS THEN
    RAISE_APPLICATION_ERROR (-20001,
        I_IdentifyName || ':$:' ||
        I_IdentifyString || ':$:' ||
        I_PunchFromDate || ':$:' ||
        I_PunchToDate || ':$:' ||
        -- I_PunchTime || ':$:' ||
        --- I_PunchTimeOut || ':$:' ||
        I_CreateBy || ':$:' ||
        I_Remarks || ':$:' ||
        I_AddOrUpdate || ':$:' ||
        I_rowId || ':$:' ||
        I_MachineName || ':$:' ||
        SQLERRM, TRUE) ;

end sp_Manual_Insert;
/

```

## 14.SP\_NIGHT\_SHIFT\_1

```
CREATE OR REPLACE PROCEDURE ATT_IN.SP_Night_shift_1
(
    I_EmpId In int,
    I_OT_Date In date
)
As
Begin
    declare
        chk_duty_pattern nvarchar2(30);
        check_ot_shift_name nvarchar2(30);
        check_reg_shift_name nvarchar2(30);
        comp_cardid nvarchar2(9);
        otDate nvarchar2(30);
        r_shift_in_check1 int;
        r_shift_out_check1 int;
        v_otapprove int;
        I_ShiftIn Date;
        I_ShiftOut Date;
        I_ShiftName nvarchar2(30);
        I_ShiftIn1 date;
        I_ShiftIn2 date;
        I_ShiftOut1 date;
        I_ShiftOut2 date;
        I_DutyIn Date;
        I_DutyOut Date;
        I_NextDayPunch Date;
        I_TempDayIn nvarchar2(30);--Date;
        I_TempDayOut nvarchar2(30);
        I_C_ShiftRegOutHH nvarchar2(30);
        nextday date;
        I_C_OUT_PUNCH date;
        v_I_ShiftIn2 int;
        v_I_ShiftIn1 int;

    begin
        select
            R_SHIFT,to_char(COMPCARDID),to_char(OT_OR_REG_DATE,'DD/MM/YYYY'),to_number(to_char(R_SHIF
            T_IN,'HH24')) ,
            to_number(to_char(R_SHIFT_OUT,'HH24'))
            into check_reg_shift_name,comp_cardid,otDate,
            r_shift_in_check1,r_shift_out_check1
            from att_in.TBL_OT_ENTRY
            where empid=I_EmpId and OT_OR_REG_DATE =
            TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');
        EXCEPTION
            when no_data_found then
                null;
            end;

        if check_reg_shift_name = 'Night Shift-1' then

            begin
                select R_SHIFT,
```

```

to_date(to_char(R_SHIFT_IN,'DD/MM/RRRR
HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'),
to_date(to_char(R_SHIFT_OUT,'DD/MM/RRRR
HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')

into I_ShiftName,

I_ShiftIn,
I_ShiftOut
from att_in.TBL_OT_ENTRY
where EMPID = I_EmpId
and OT_OR_REG_DATE =
to_date(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR');
select to_date(to_char(I_OT_Date + interval '1'
DAY,'DD/MM/RRRR'),'DD/MM/RRRR')
into nextday--nextday --r_shift_c_out_date
from dual;

select to_char(I_ShiftOut,'HH24:MI:SS')
into I_C_ShiftRegOutHH
from dual;

update att_in.TBL_OT_ENTRY
set R_SHIFT_OUT = to_date(to_char(nextday,'DD/MM/RRRR')||'
'||I_C_ShiftRegOutHH,'DD/MM/RRRR HH24:MI:SS') --'0'||
where empid= I_EmpId
and OT_OR_REG_DATE =
TO_DATE(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR'); --to_date(I_OT_Date,'DD/MM/YYYY')
--T

commit;
select
to_date(to_char(R_SHIFT_OUT,'DD/MM/RRRR
HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
into
I_ShiftOut
from att_in.TBL_OT_ENTRY
where EMPID = I_EmpId
and OT_OR_REG_DATE =
to_date(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR');

select to_date(to_char(I_ShiftOut - interval '120'
MINUTE,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/YYYY HH24:MI:SS')
into I_ShiftOut1
from dual;

select to_date(to_char(I_ShiftOut + interval '180'
MINUTE,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/YYYY HH24:MI:SS')
into I_ShiftOut2
from dual;

select to_date(to_char(max(PunchTime),'DD/MM/RRRR
HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
into I_C_OUT_PUNCH
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = comp_cardid
and to_char(punchtime,'DD/MM/RRRR HH24:MI:SS')
between to_char(I_ShiftOut1,'DD/MM/RRRR HH24:MI:SS')
and to_char(I_ShiftOut2,'DD/MM/RRRR HH24:MI:SS');

```

```

select to_date(to_char(I_ShiftIn - interval '120'
MINUTE, 'DD/MM/RRRR HH24:MI:SS'), 'DD/MM/RRRR HH24:MI:SS')
into I_ShiftIn1
from dual;

select to_date(to_char(I_ShiftIn + interval '115'
MINUTE, 'DD/MM/RRRR HH24:MI:SS'), 'DD/MM/RRRR HH24:MI:SS')
into I_ShiftIn2
from dual;

select to_date(to_char(min(PunchTime), 'DD/MM/RRRR
HH24:MI:SS'), 'DD/MM/RRRR HH24:MI:SS')
into I_DutyIn
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = comp_cardid
and PUNCHDATE =
to_date(to_char(I_OT_Date, 'DD/MM/RRRR'), 'DD/MM/RRRR')
and to_char(punchtime, 'DD/MM/RRRR HH24:MI:SS')
between to_char(I_ShiftIn1, 'DD/MM/RRRR HH24:MI:SS')
and to_char(I_ShiftIn2, 'DD/MM/RRRR HH24:MI:SS');
EXCEPTION
when no_data_found then
null;
end;

select count(I_ShiftIn2) into v_I_ShiftIn2 from
att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = comp_cardid
and PUNCHDATE =
to_date(to_char(I_OT_Date, 'DD/MM/RRRR'), 'DD/MM/RRRR');

select count(I_ShiftIn1) into v_I_ShiftIn1 from
att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = comp_cardid
and PUNCHDATE =
to_date(to_char(I_OT_Date, 'DD/MM/RRRR'), 'DD/MM/RRRR');
if v_I_ShiftIn2>0 or v_I_ShiftIn2>0 then

update att_in.tbl_ot_entry
set R_IN_TIME_PUNCH = to_date(to_char(I_DutyIn, 'DD/MM/RRRR
HH24:MI:SS'), 'DD/MM/RRRR HH24:MI:SS'),
R_OUT_TIME_PUNCH =
to_date(to_char(I_C_OUT_PUNCH, 'DD/MM/RRRR HH24:MI:SS'), 'DD/MM/RRRR HH24:MI:SS'),
REG_STATUS = upper('P')
where EMPID = I_EmpId and OT_OR_REG_DATE =
TO_DATE(to_char(I_OT_Date, 'DD/MM/RRRR'), 'DD/MM/RRRR');
commit;
end if;
end if;end;

end;
/

```



## 15.SP\_NIGHT\_SHIFT\_2

```
CREATE OR REPLACE PROCEDURE ATT_IN.sp_Night_shift_2
(
    I_EmpId In int,
    I_OT_Date In date
)
As
Begin
    declare
        chk_duty_pattern nvarchar2(30);
        check_ot_shift_name nvarchar2(30);
        check_reg_shift_name nvarchar2(30);
        chk_p_exist_in int;
        chk_p_exist_out int;
        chk_p_exist int;
        -----
        v_compcardid varchar2(30);
        comp_cardid nvarchar2(9);
        otDate nvarchar2(30);
        r_shift_in_check1 int;
        r_shift_in_check2 int;
        r_shift_out_check1 int;
        r_shift_out_check2 int;
        v_otapprove int;
        I_ShiftIn Date;
        I_ShiftOut Date;
        I_ShiftName nvarchar2(30);
        I_ShiftIn1 date;
        I_ShiftIn2 date;
        I_ShiftOut1 date;
        I_ShiftOut2 date;
        I_DutyIn Date;
        I_DutyOut Date;
        I_NextDayPunch Date;
        I_TempDayIn nvarchar2(30);--Date;
        I_TempDayOut nvarchar2(30);
        I_C_ShiftRegOutHH nvarchar2(30);
        nextday date;
        I_C_OUT_PUNCH date;

begin
            begin
                select
                    R_SHIFT,to_char(COMPCARDID),to_char(OT_OR_REG_DATE,'DD/MM/YYYY'),
                    to_number(to_char(R_SHIFT_IN,'HH24')) ,
                    to_number(to_char(R_SHIFT_OUT,'HH24'))
                    into check_reg_shift_name,comp_cardid,otDate,
                    r_shift_in_check1,r_shift_out_check1
                    from att_in.TBL_OT_ENTRY
                    where empid=I_EmpId
```

```

        and OT_OR_REG_DATE =
TO_DATE(to_char(I_OT_Date, 'DD/MM/YYYY'), 'DD/MM/YYYY');
        EXCEPTION
        when no_data_found then
        null;

        end;

        if check_reg_shift_name = 'Night Shift-2' then
        begin
        select R_SHIFT,
        to_date(to_char(R_SHIFT_IN, 'DD/MM/RRRR
HH24:MI:SS'), 'DD/MM/RRRR HH24:MI:SS'),
        to_date(to_char(R_SHIFT_OUT, 'DD/MM/RRRR
HH24:MI:SS'), 'DD/MM/RRRR HH24:MI:SS')

        into I_ShiftName,
        I_ShiftIn,
        I_ShiftOut
        from att_in.TBL_OT_ENTRY
        where EMPID = I_EmpId
        and OT_OR_REG_DATE =
to_date(to_char(I_OT_Date, 'DD/MM/RRRR'), 'DD/MM/RRRR');

        select to_date(to_char(I_OT_Date + interval '1'
DAY, 'DD/MM/RRRR'), 'DD/MM/RRRR')
        into nextday--nextday --r_shift_c_out_date
        from dual;

        select to_char(I_ShiftOut, 'HH24:MI:SS')
        into I_C_ShiftRegOutHH
        from dual;

        update att_in.TBL_OT_ENTRY
        set R_SHIFT_OUT =
to_date(to_char(nextday, 'DD/MM/RRRR') || ' ' || I_C_ShiftRegOutHH, 'DD/MM/RRRR
HH24:MI:SS') --'0'||
        where empid= I_EmpId
        and OT_OR_REG_DATE =
TO_DATE(to_char(I_OT_Date, 'DD/MM/RRRR'), 'DD/MM/RRRR');
        commit;

        select
        to_date(to_char(R_SHIFT_OUT, 'DD/MM/RRRR
HH24:MI:SS'), 'DD/MM/RRRR HH24:MI:SS')
        into
        I_ShiftOut
        from att_in.TBL_OT_ENTRY
        where EMPID = I_EmpId
        and OT_OR_REG_DATE =
to_date(to_char(I_OT_Date, 'DD/MM/RRRR'), 'DD/MM/RRRR');

        select to_date(to_char(I_ShiftOut - interval '120'
MINUTE, 'DD/MM/RRRR HH24:MI:SS'), 'DD/MM/YYYY HH24:MI:SS')
        into I_ShiftOut1
        from dual;

```

```

        select to_date(to_char(I_ShiftOut + interval '180'
MINUTE, 'DD/MM/RRRR HH24:MI:SS'), 'DD/MM/YYYY HH24:MI:SS')
        into I_ShiftOut2
        from dual;

        select to_date(to_char(max(PunchTime), 'DD/MM/RRRR
HH24:MI:SS'), 'DD/MM/RRRR HH24:MI:SS')
        into I_C_OUT_PUNCH
        from att_in.TBL_RAW_DATA_CMIS
        where COMPCARDID = comp_cardid
        and to_char(punchtime, 'DD/MM/RRRR HH24:MI:SS')
        between to_char(I_ShiftOut1, 'DD/MM/RRRR HH24:MI:SS')
        and to_char(I_ShiftOut2, 'DD/MM/RRRR HH24:MI:SS');

        select to_date(to_char(I_ShiftIn - interval '120'
MINUTE, 'DD/MM/RRRR HH24:MI:SS'), 'DD/MM/RRRR HH24:MI:SS')
        into I_ShiftIn1
        from dual;

        select to_date(to_char(I_ShiftIn + interval '115'
MINUTE, 'DD/MM/RRRR HH24:MI:SS'), 'DD/MM/RRRR HH24:MI:SS')
        into I_ShiftIn2
        from dual;

        select to_date(to_char(min(PunchTime), 'DD/MM/RRRR
HH24:MI:SS'), 'DD/MM/RRRR HH24:MI:SS')
        into I_DutyIn
        from att_in.TBL_RAW_DATA_CMIS
        where COMPCARDID = comp_cardid
        and PUNCHDATE =
to_date(to_char(I_OT_Date, 'DD/MM/RRRR'), 'DD/MM/RRRR')
        and to_char(punchtime, 'DD/MM/RRRR HH24:MI:SS')
        between to_char(I_ShiftIn1, 'DD/MM/RRRR HH24:MI:SS')
        and to_char(I_ShiftIn2, 'DD/MM/RRRR HH24:MI:SS');

EXCEPTION
when no_data_found then
null;

end;

update att_in.tbl_ot_entry
set R_IN_TIME_PUNCH = to_date(to_char(I_DutyIn, 'DD/MM/RRRR
HH24:MI:SS'), 'DD/MM/RRRR HH24:MI:SS'),
    R_OUT_TIME_PUNCH =
to_date(to_char(I_C_OUT_PUNCH, 'DD/MM/RRRR HH24:MI:SS'), 'DD/MM/RRRR HH24:MI:SS'),
    REG_STATUS = upper('P')
where EMPID = I_EmpId
and OT_OR_REG_DATE =
TO_DATE(to_char(I_OT_Date, 'DD/MM/RRRR'), 'DD/MM/RRRR');
commit;

```

```

        select count(punchtime)
        into chk_p_exist
        from att_in.TBL_RAW_DATA_CMIS
        where  COMPCARDID =comp_cardid
        and PUNCHDATE = I_OT_Date;

        select count(punchtime)
        into chk_p_exist_in
        from att_in.TBL_RAW_DATA_CMIS
        where COMPCARDID = comp_cardid
        and PUNCHDATE =
to_date(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR')
        and to_char(punchtime,'DD/MM/RRRR HH24:MI:SS')
        between to_char(I_ShiftIn1,'DD/MM/RRRR HH24:MI:SS')
        and to_char(I_ShiftIn2,'DD/MM/RRRR HH24:MI:SS');

        if chk_p_exist_in=0 then
            update att_in.tbl_ot_entry
            set REG_STATUS = upper('A')
            where EMPID = I_EmpId
            and OT_OR_REG_DATE =
TO_DATE(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR');
            commit;
        end if;

        if chk_p_exist=1 then
            update att_in.tbl_ot_entry
            set REG_STATUS = upper('P')
            where EMPID = I_EmpId
            and OT_OR_REG_DATE =
TO_DATE(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR');
            commit;
        end if;

        select count(punchtime)
        into chk_p_exist_out
        from att_in.TBL_RAW_DATA_CMIS
        where COMPCARDID = comp_cardid
        and to_char(punchtime,'DD/MM/RRRR HH24:MI:SS')
        between to_char(I_ShiftOut1,'DD/MM/RRRR HH24:MI:SS')
        and to_char(I_ShiftOut2,'DD/MM/RRRR HH24:MI:SS');

        if chk_p_exist_out=0 then
            update att_in.tbl_ot_entry
            set REG_STATUS = upper('A')
            where EMPID = I_EmpId
            and OT_OR_REG_DATE =
TO_DATE(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR');
            commit;

        end if;

        end if;

    end;

/

```

## 16.SP\_OSD\_ADD\_UPDATE

```
CREATE OR REPLACE PROCEDURE ATT_IN.sp_OSD_Add_Update
(
    I_OsdId In Int,
    I_EmpId In Int,
    I_OsdStartDate In date,
    I_OsdEndDate In date,
    I_OsdNumDays In float,
    I_OsdLocation In varchar2,
    I_OsdReason In varchar2,
    I_CreateBy In varchar2,
    I_UpdateBY In varchar2,
    I_AddOrUpdate In varchar2,
    I_DeptId in Int,
    I_EmpCode in  varchar2,
    I_PlantId Int

    --O_refcursor_out OUT SYS_REFCURSOR
)
is
BEGIN

    declare
        Count_No int:=0;
        O_retVal int :=0;
        v_days int :=0;
        V_DateIncrement date:=I_OsdStartDate ;
    BEGIN
        select (I_OsdEndDate - I_OsdStartDate)+1 into v_days from dual;
        IF I_AddOrUpdate='Saved'
        then
            --Select count(*) into Count_No from Table_Name
            -- where condition of duplicate;

            if Count_No = 0
            then

                --Not Duplicate
                for i in 1..v_days
                loop
                    Insert Into att_in.tbl_OSD_Setup
                    (
                        EmpId,OsdStartDate,OsdEndDate,OsdNumDays,OsdLocation,OsdReason,
                        CreateBy,CreateDate,UpdateBy,UpdateDate,DEPTID,EMPCODE,PLANT_ID
                    )
                    values
                    (
                        I_EmpId,
                        V_DateIncrement,
                        V_DateIncrement,
                        I_OsdNumDays,
                        I_OsdLocation,
                        I_OsdReason,
                        I_CreateBy,
```

```

                                sysdate,
                                I_UpdateBY,
                                sysdate,
                                I_DeptId,
                                I_EmpCode,
                                I_PlantId

);

        commit;
        V_DateIncrement:=V_DateIncrement+1;
    end loop;
    O_retVal := 0;
    else
        O_retVal := 1;

END IF;
Else
-----
-----Update
update      att_in.tbl_OSD_Setup
set  OsdStartDate = I_OsdStartDate,
    OsdEndDate =I_OsdEndDate,
    OsdNumDays =I_OsdNumDays,
    OsdLocation =I_OsdLocation,
    OsdReason = I_OsdReason,
    UpdateBy = I_UpdateBY,
    UpdateDate=sysdate
where OsdId =I_OsdId;

    Commit;
END IF;
--open O_refcursor_out for select O_retVal from dual;
END;

EXCEPTION
WHEN OTHERS THEN
RAISE_APPLICATION_ERROR (-20001,

    I_OsdId || ' :$:' ||
    I_EmpId || ' :$:' ||
    I_OsdStartDate || ' :$:' ||
    I_OsdEndDate || ' :$:' ||
    I_OsdNumDays || ' :$:' ||
    I_OsdLocation || ' :$:' ||
    I_OsdReason || ' :$:' ||
    I_CreateBy || ' :$:' ||
    I_UpdateBY || ' :$:' ||
    I_AddOrUpdate || ' :$:' ||
SQLERRM, TRUE) ;

end sp_OSD_Add_Update;
/

```

## 17.SP\_OT\_APPROVE\_ADD\_UPDATE

```
CREATE OR REPLACE PROCEDURE ATT_IN.sp_OT_Approve_Add_Update
(
    I_EMPID IN INT,
    I_OTDATE IN Date,
    I_OTAPPROVE IN INT,
    I_CreateBy IN varchar2,
    I_UpdateBy IN varchar2,
    I_AddOrUpdate IN varchar2
    --O_refcursor_retVal OUT SYS_REFCURSOR
)
as
Begin
    --Add
    declare count_Emp_OT_Date int:=0;
    O_retVal int;
    Begin
        if I_AddOrUpdate='Saved' Then
            Select count(*) into count_Emp_OT_Date from att_in.TBL_EMP_OT_APPROVE
            where EMPID=I_EMPID and OTDATE=I_OTDATE;

            if count_Emp_OT_Date=0
            Then
                --Not Duplicate
                Insert Into att_in.TBL_EMP_OT_APPROVE
                (
                    EMPID,
                    OTDATE,
                    OTAPPROVE,
                    CREATEDATE,
                    CREATEBY,
                    UPDATEBY,
                    updatedate
                )
                values
                (
                    I_EMPID,
                    I_OTDATE,
                    I_OTAPPROVE,
                    sysdate,
                    I_CreateBy,
                    I_UpdateBy,
                    sysdate
                );
                Commit;
                O_retVal := 0;
            else
                O_retVal := 1;
            End If;
        else
            update att_in.TBL_EMP_OT_APPROVE
            set EMPID=I_EMPID,
                OTDATE=I_OTDATE,
                OTAPPROVE= I_OTAPPROVE,
                CREATEBY=I_CreateBy,
                UPDATEBY=I_UpdateBy
```

```

        where EMPID=I_EMPID;
        Commit;
        O_retVal := 0;
    end if;
    -- open O_refcursor_retVal for select O_retVal from dual;
End;

EXCEPTION
    WHEN OTHERS THEN
        RAISE_APPLICATION_ERROR (-20001,
            I_EMPID || ' :$: ' ||
            I_OTDATE || ' :$: ' ||
            I_OTAPPROVE || ' :$: ' ||

            SQLERRM, TRUE) ;

End sp_OT_Approve_Add_Update;
/

```

## 18.SP\_OT\_CALCULATION

```

CREATE OR REPLACE PROCEDURE ATT_IN.sp_ot_calculation
(
    I_EmpId In int,
    I_OT_Date In date
)
As
Begin
    declare

        -- 01/01/2014 -- for stop double calculation in OT HOUR

        v_rintime date;
        v_routtime date;
        v_overtime date;
        v_overtime date;
        previous_ot_in date;
        previous_ot_out date;
        special_ot_start_date_pre date;
        special_ot_end_date_pre date;
        special_ot_start_date date;
        special_ot_end_date date;
        special_day_row_count int:=0;
        special_day_row_count_pre int:=0;
        -----

        ot_counter int;
        ot_entry_counter int;

        chk_p_exist int;
        --chk duty pattern nvarchar2(30);
        -----

        ot_in time date;

```



```

ot_punch_out date;

-----

ot_hour int; --19_06_2013
ot_min int; --19_06_2013
v_ot_intimehh number;
v_ot_intimemi number;
v_ot_outtimehh number;
v_ot_outtimemi number;
v_ot_in_time number;
v_ot_punch_out number;

ot_p_out_time_hour int; --19_06_2013
total_ot_hour nvarchar2(30); --19_06_2013
ot_count int; --19_06_2013
double_ot_hour int;
double_ot_min int;
special_ot_count int;
difrence_In_Hours number;
difrence_In_minutes number;
difrence_In_seconds number;
ot_punch_count int;
spe_ot_count int;
I_Sts nvarchar2(30);
srt_count int;
emp_shift nvarchar2(30);
-----

--otDate varchar2(30);

v_otapprove int;
v_special_ot_entry_tbl nvarchar2(50);
s_ot_hour_hh24 date;
s_ot_hour_mm int;
s_ot_h_int int;
total_spe_hour_in_minute int;
grand_total int;
v_shiftid int;
v_desgid varchar2(30);
v_labourcode int;
v_employee_type varchar2(20);
v_employee_category varchar2(20);
v_desig_exist_96 int;
v_desig_exist_64 int;
-----

begin

begin

begin
select EMPLOYEE_TYPE,EMPLOYEE_CATEGORY,SHIFTID,DESIGNATIONID,LABOURCODE
into v_employee_type,v_employee_category , v_shiftid,v_desgid,v_labourcode
from att_in.TBL_EMP_PERSONAL_INFO where EMPID=I_Empid;
EXCEPTION

```

```

        when no_data_found then
            null;
        end;

        begin
            select count(*) into v_desig_exist_96 from att_in.TBL_CHECK
            where MAX_96_HOURS_OT=v_labourcode;
            EXCEPTION
            when no_data_found then
                null;
            end;

            begin
                select count(*) into v_desig_exist_64 from att_in.TBL_CHECK
                where MAX_64_HOURS_OT=v_labourcode;
                EXCEPTION
                when no_data_found then
                    null;
                end;

            begin
                select R_SHIFT
                into emp_shift
                from att_in.TBL_OT_ENTRY
                where empid= I_EmpId
                and
                OT_OR_REG_DATE = to_date(to_char(I_OT_Date, 'DD/MM/RRRR'), 'DD/MM/RRRR');
                EXCEPTION when no_data_found then
                    null;
                end;

                DBMS_OUTPUT.PUT_LINE('R_SHIFT');
                DBMS_OUTPUT.PUT_LINE( emp_shift);

            if emp_shift like '4 Shift%' then

                begin
                    select to_date(to_char(R_IN_TIME_PUNCH, 'DD/MM/RRRR
HH24:MI:SS'), 'DD/MM/RRRR HH24:MI:SS'),
                    to_date(to_char(R_OUT_TIME_PUNCH, 'DD/MM/RRRR HH24:MI:SS'), 'DD/MM/RRRR
HH24:MI:SS'),
                    to_date(to_char(OT_IN_TIME_PUNCH, 'DD/MM/RRRR HH24:MI:SS'), 'DD/MM/RRRR
HH24:MI:SS'),
                    to_date(to_char(OT_OUT_TIME_PUNCH, 'DD/MM/RRRR
HH24:MI:SS'), 'DD/MM/RRRR HH24:MI:SS')
                    into
                    v_rintime,v_routtime,v_overtime,v_overtime
                    from att_in.TBL_OT_ENTRY
                    where EMPID = I_EmpId
                    and OT_OR_REG_DATE = to_date(to_char(I_OT_Date-
1, 'DD/MM/RRRR'), 'DD/MM/RRRR') ;

                    EXCEPTION
                    when too_many_rows then
                        null;
                    when no_data_found then
                        null;
                    when others then
                        null;

```

```

end;

select nvl(OT_APPROVE,0) into v_otapprove
from att_in.TBL_OT_ENTRY
where empid=I_EmpId
and OT_OR_REG_DATE =
to_date(to_char(I_OT_Date, 'DD/MM/YYYY'), 'DD/MM/YYYY');

select nvl(count(OT_IN_TIME_PUNCH),0) into ot_count
from att_in.TBL_OT_ENTRY
where empid= I_EmpId and OT_OR_REG_DATE = I_OT_Date ;--
to_date(I_OT_Date, 'DD/MM/YYYY');

select nvl(count(SHORT_REST_HOUR),0) into srt_count
from att_in.TBL_OT_ENTRY
where empid= I_EmpId and OT_OR_REG_DATE = I_OT_Date ;--
to_date(I_OT_Date, 'DD/MM/YYYY');

select nvl(count(SPECIAL_OT_HOUR),0) into spe_ot_count
from att_in.TBL_OT_ENTRY
where empid= I_EmpId and OT_OR_REG_DATE = I_OT_Date ;

DBMS_OUTPUT.PUT_LINE('ot_count');
DBMS_OUTPUT.PUT_LINE(ot_count);
DBMS_OUTPUT.PUT_LINE(spe_ot_count );
--DBMS_OUTPUT.PUT_LINE('Tanvir 01/01/2014' );

begin

select
to_date(to_char(STARTDATE, 'DD/MM/RRRR'), 'DD/MM/RRRR'), to_date(to_char(ENDDATE, 'DD/MM/RRRR'), 'DD/MM/RRRR')
into special_ot_start_date, special_ot_end_date
from att_in.TBL_SPECIAL_OT_DAYS
where STARTDATE =
TO_DATE(to_char(I_OT_Date, 'DD/MM/RRRR'), 'DD/MM/RRRR');

select nvl(count(SPECIAL_OT_HOUR),0) into special_day_row_count
from att_in.TBL_OT_ENTRY
where (to_date(OT_IN_TIME_PUNCH, 'DD/MM/RRRR') between
to_date(special_ot_start_date, 'DD/MM/RRRR') and
to_date(special_ot_end_date, 'DD/MM/RRRR'))
and (to_date(OT_OUT_TIME_PUNCH, 'DD/MM/RRRR') between
to_date(special_ot_start_date, 'DD/MM/RRRR') and
to_date(special_ot_end_date, 'DD/MM/RRRR'))
and EMPID = I_EmpId;

EXCEPTION
when too_many_rows then
null;
when no_data_found then
null;
when others then
null;
end;

```

```

      ----here is
      ----before days otintime special ot day er mache ki na

      begin
      select count(to_char(v_overtime,'DD/MM/RRRR')) into
special_day_row_count_pre
      from att_in.TBL_OT_ENTRY
      where (to_date(v_overtime,'DD/MM/RRRR') between
to_date(special_ot_start_date,'DD/MM/RRRR') and
to_date(special_ot_end_date,'DD/MM/RRRR'))
and (to_date(v_overtime,'DD/MM/RRRR') between
to_date(special_ot_start_date,'DD/MM/RRRR') and
to_date(special_ot_end_date,'DD/MM/RRRR'))
and EMPID = I_EmpId
and OT_OR_REG_DATE = to_date(to_char(I_OT_Date-
1,'DD/MM/RRRR'),'DD/MM/RRRR') ;
      EXCEPTION
      when too_many_rows then
      null;
      when no_data_found then
      null;
      when others then
      null;
      end;

      dbms_OUTPUT.PUT_LINE(to_char(v_overtime,'DD/MM/RRRR HH24:MI:SS'));
      dbms_OUTPUT.PUT_LINE(to_char(v_overtime,'DD/MM/RRRR HH24:MI:SS'));
      dbms_OUTPUT.PUT_LINE(to_char(special_ot_start_date,'DD/MM/RRRR
HH24:MI:SS'));
      dbms_OUTPUT.PUT_LINE(to_char(special_ot_end_date,'DD/MM/RRRR
HH24:MI:SS'));
      DBMS_OUTPUT.PUT_LINE (special_day_row_count_pre ) ;
      DBMS_OUTPUT.PUT_LINE ('special_ot_start_date_pre');

      DBMS_OUTPUT.PUT_LINE ('Before Calculation');
      DBMS_OUTPUT.PUT_LINE (special_day_row_count ) ;

      if v_otapprove = 1 then

      Select
abs(NVL(SUM(datediff('hh',OT_IN_TIME_PUNCH,OT_OUT_TIME_PUNCH)),0) * 60)
      into difrence_In_Hours from att_in.TBL_OT_ENTRY
      where empid= I_EmpId and OT_OR_REG_DATE = I_OT_Date;
      If v_employee_type in('P','G') and v_employee_category='Permanent'
and v_desig_exist_96=0 and v_desig_exist_64 =0 then

      if special_day_row_count > 0 then
      difrence_In_Hours := 0;
      else
      difrence_In_Hours := difrence_In_Hours;
      end if;
      else
      difrence_In_Hours := difrence_In_Hours;
      end if;

      if difrence_In_Hours >= 480 then
      difrence_In_Hours := 480;

```

```

else
    difrence_In_Hours := difrence_In_Hours ;

end if;

select nvl(SPECIAL_OT_HOUR,'0')
into
v_special_ot_entry_tbl
from att_in.TBL_OT_ENTRY
where empid= I_EmpId and OT_OR_REG_DATE = I_OT_Date;

s_ot_hour_hh24 :=
to_date(to_char(v_special_ot_entry_tbl),'HH24:MI:SS');
s_ot_hour_mm := to_number(to_char(s_ot_hour_hh24,'MI'));
s_ot_h_int := to_number(to_char(s_ot_hour_hh24,'HH24')) * 60;
total_spe_hour_in_minute:=s_ot_h_int+s_ot_hour_mm ;
--DBMS_OUTPUT.PUT_LINE( 'total_spe_hour_in_minute');
--DBMS_OUTPUT.PUT_LINE( total_spe_hour_in_minute);
if spe_ot_count >0 then

    grand_total:=difrence_In_Hours+total_spe_hour_in_minute;

    If grand_total> 960 then
        grand_total:=960;
    else
        grand_total:=grand_total;
    end if;

    ot_min := floor((grand_total mod 60));
    ot_hour := (grand_total - ot_min) / 60;
    if ot_hour > 9 then

        if ot_min >= 0 and ot_min < 10 then
            total_ot_hour :=
to_char(ot_hour)||':0'||to_char(ot_min)||':00';
            elsif ot_min >=10 and ot_min <=59 then
                total_ot_hour :=
to_char(ot_hour)||':'||to_char(ot_min)||':00';
            end if;
            elsif ot_hour <=9 then
                if ot_min >= 0 and ot_min < 10 then
                    total_ot_hour :=
to_char('0'||to_char(ot_hour)||':0'||to_char(ot_min)||':00');
                elsif ot_min >=10 and ot_min <=59 then
                    total_ot_hour :=
to_char('0'||to_char(ot_hour)||':'||to_char(ot_min)||':00');
                end if;

            end if;
            DBMS_OUTPUT.PUT_LINE(' grand_total');
            DBMS_OUTPUT.PUT_LINE( grand_total);

        else
            grand_total:=difrence_In_Hours;

            ot_min := floor((grand_total mod 60));
            ot_hour := (grand total - ot_min) / 60;

```

```

        if ot_hour > 9 then

            if ot_min >= 0 and ot_min < 10 then
                total_ot_hour :=
to_char(ot_hour)||':0'||to_char(ot_min)||':00';
                elsif ot_min >=10 and ot_min <=59 then
                total_ot_hour :=
to_char(ot_hour)||': '||to_char(ot_min)||':00';
                end if;
            elsif ot_hour <=9 then
                if ot_min >= 0 and ot_min < 10 then
                    total_ot_hour :=
to_char('0'||to_char(ot_hour)||':0'||to_char(ot_min)||':00');
                    elsif ot_min >=10 and ot_min <=59 then
                        total_ot_hour :=
to_char('0'||to_char(ot_hour)||': '||to_char(ot_min)||':00');
                    end if;
                end if;
                DBMS_OUTPUT.PUT_LINE(' grand_total');
                DBMS_OUTPUT.PUT_LINE( grand_total);

            end if;

            DBMS_OUTPUT.PUT_LINE('total_ot_hour');
            DBMS_OUTPUT.PUT_LINE(total_ot_hour);

            update att_in.TBL_OT_ENTRY
            set OT_HOUR = to_char(total_ot_hour)
            where empid= I_EmpId and OT_OR_REG_DATE = I_OT_Date ;
            commit;

        end if;

        If v_employee_type in('P','G') and v_employee_category='Permanent' and
v_desig_exist_96=0 and v_desig_exist_64 =0 then
            if special_day_row_count_pre>0 then

                UPDATE att_in.TBL_OT_ENTRY
                SET OT_HOUR=''
                where EMPID=I_EmpId
                and OT_OR_REG_DATE = to_date(to_char(I_OT_Date-
1, 'DD/MM/RRRR'), 'DD/MM/RRRR');
                commit;

            end if;
        end if;
    end if;

    if ot_count=0 and srt_count=0 then

        update att_in.TBL_PROCESSED_DATA
        set MEALALLOWANCE=0
        where EMPID = I_EmpId and PUNCHDATE = I_OT_Date ;
        commit;
    end if;

```

```

        end if;

        select REG_STATUS    --25-09-0213
        into I_Sts
        from att_in.TBL_OT_ENTRY
        where EMPID = I_EmpId
        and OT_OR_REG_DATE = to_date(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR');

        if I_Sts = 'A' then
            UPDATE att_in.TBL_OT_ENTRY
            SET OT_HOUR=' '
            where EMPID=I_EmpId
            and OT_OR_REG_DATE = to_date(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR');
            commit;
            END IF;
    EXCEPTION when no_data_found then
        null;
        end;
    end;
    --EXECUTE ATT_IN.sp_Four_Shift_18_07_13_OFF(35287,'15-MAY-2013');
    --EXECUTE ATT_IN.Four_Shift_Auto_Entry(37124,'01-May-2013','31-May-2013','4 Shift-C
2nd Day','REG');
    --EXECUTE ATT_IN.sp_Data_Process_Biman('01-May-2013','31-May-
2013',5,141,0,1226,1,5227);
    -- EXECUTE ATT_IN.SP_INSERT_ROWDATA('000050388','01-mar-2014','31-mar-2014');
    --EXECUTE ATT_IN.sp_ot_calculation(50388,'02-Mar-2014');
    --EXECUTE att_in.sp_Four_Shift_11_07_13(37123,'01-May-2013');
    --EXECUTE att_in.sp_Four_Shift(37123,'10-May-2013');
    --EXECUTE att_in.sp_Data_Process_Biman('01-may-2013','31-may-
2013',5,141,0,1226,1,5227);

End;
/

```

## 19.SP\_ROLE\_ADD\_UPDATE

```

CREATE OR REPLACE PROCEDURE ATT_IN.sp_Role_Add_Update
(
    I_RoleId int,
    I_RoleName varchar2,
    I_CreateBy varchar2,
    I_UpdateBy varchar2,
    I_AddOrUpdate varchar2,
    O_IdDuplicate OUT SYS_REFCURSOR
)
as
BEGIN
    --Add
    Declare I_IdDuplicate int;
    BEGIN
        I_IdDuplicate :=0;
        if I_AddOrUpdate='Saved' then
            select count(*) into I_IdDuplicate from tbl_Role_Creation where
RoleName=I_RoleName;
            if I_IdDuplicate=0 then
                Insert Into tbl_Role_Creation
                (

```

```

        RoleName,CreateBy,CreateDate,UpdateBy,UpdateDate
    )
    values
    (
        I_RoleName,I_CreateBy,sysdate,I_UpdateBy,sysdate
    );
    commit;

    Declare V_RoleId int;
    Begin
        select RoleId into V_RoleId from  tbl_Role_Creation where
RoleName=I_RoleName;
        sp_InitialRoleAssign(V_RoleId,I_CreateBy);
    END;
    else
        update  tbl_Role_Creation
        set RoleName=I_RoleName,
        UpdateDate=sysdate
        where RoleId=I_RoleId;
        commit;
    End IF;

    END IF;
    Open O_IdDuplicate for select I_IdDuplicate from dual;
END;

End  sp_Role_Add_Update;
/

```

## 20.SP\_SEASON\_ADD\_UPDATE

```

CREATE OR REPLACE PROCEDURE ATT_IN.sp_Season_Add_Update
(
    I_SeasonId In Int,
    I_CompId In Int,
    I_SeasonTypeName In varchar2,
    I_SeasonStartDate In Date,
    I_SeasonEndDate In Date,
    I_CreateBy In varchar2,
    I_UpdateBy In varchar2,
    I_AddOrUpdate In varchar2
    --O_refcursor_out OUT SYS_REFCURSOR
)
is
    BEGIN

        declare Count_No int:=0; O_retVal int :=0;
        BEGIN

            IF I_AddOrUpdate='Saved'
            then
                Select count(*) into  Count_No from TBL_SEASON_SETUP
                where  SEASONTYPENAME=I_SeasonTypeName;

                if Count_No = 0
                then

```



```

--Not Duplicate
Insert into TBL_SEASON_SETUP
(
    COMPID,
    SEASONTYPENAME,
    SEASONSTARTDATE,
    SEASONENDDATE,
    CREATEBY,
    CREATEDATE,
    UPDATEBY,
    UPDATEDATE
)
Values
(
    I_CompId,
    I_SeasonTypeName,
    I_SeasonStartDate,
    I_SeasonEndDate,
    I_CreateBy,
    sysdate,
    I_UpdateBy ,
    sysdate
);
commit;
O_retVal := 0;
else
    O_retVal := 1;
END IF;
Else
Update TBL_SEASON_SETUP
SET COMPID=I_CompId,
SEASONTYPENAME=I_SeasonTypeName,
SEASONSTARTDATE=I_SeasonStartDate,
SEASONENDDATE=I_SeasonEndDate,
CREATEBY=I_CreateBy,
CREATEDATE=sysdate,
UPDATEBY=I_UpdateBy,
UPDATEDATE=sysdate
Where SEASONID=I_SeasonId;
Commit;
END IF;
END;
EXCEPTION
WHEN OTHERS THEN
RAISE_APPLICATION_ERROR (-20001,
    I_SeasonId || ':$:' ||
    I_CompId || ':$:' ||
    I_SeasonTypeName || ':$:' ||
    I_SeasonStartDate || ':$:' ||
    I_SeasonEndDate || ':$:' ||
    I_CreateBy || ':$:' ||
    I_UpdateBy || ':$:' ||
    I_AddOrUpdate || ':$:' ||

    SQLERRM, TRUE) ;

end sp_Season_Add_Update;
/

```

## 21.SP\_SHIFTNAME\_ADD\_UPDATE

```
CREATE OR REPLACE PROCEDURE ATT_IN.sp_ShiftName_Add_Update
(
    I_ShiftId In Int,
    I_ShiftName In Varchar2,
    I_IsRolling In varchar2,
    I_CreateBy IN  varchar2,
    I_UpdateBy IN  varchar2,
    I_AddOrUpdate IN  varchar2,
    I_ShiftType IN varchar2,
    I_LOGINTIME    in    Date,
    I_LOGOUTTIME   in    Date
    --O_refcursor_retVal OUT SYS_REFCURSOR

    -- execute sp_ShiftName_Add_Update(0,'4
Shift','Rolling','Sam','sam','Saved','4-Shift');
)
as
Begin
    --Add
    declare
    count_shift_Name int:=0;

    --entryBy nvarchar(50);

    O_retVal int;
    Begin
        if I_AddOrUpdate='Saved' Then
            Select count(ShiftName) into    count_shift_Name from
TBL_SHIFTNAMESETTINGS
            where ShiftName=I_ShiftName;

            if count_shift_Name=0
            Then
                --Not Duplicate
                --entryBy := (SELECT sys_context('USERENV', 'SESSION_USER')
into CreateBy FROM dual);
                Insert Into TBL_SHIFTNAMESETTINGS
                (
                    ShiftName,
                    IsRolling,
                    CREATEBY,
                    UPDATEBY,
                    CreateDate,
                    UpdateDate,
                    SHIFTTYPE,
                    LOGINTIME,
                    LOGOUTTIME

                )
                values
                (
                    I_ShiftName,
                    I_IsRolling,
                    --entryBy
                    I_CreateBy,
```

```

        I_UpdateBy,
        SYSDATE,
        SYSDATE,
        I_ShiftType,
        I_LOGINTIME,
        I_LOGOUTTIME

    );
    Commit;
    O_retVal := 0;
else
    O_retVal := 1;

End If;
else
    update TBL_SHIFTNAMESETTINGS
    set ShiftName=I_ShiftName,
        IsRolling=I_IsRolling,
        --CREATEBY=entryBy
        CREATEBY=I_CreateBy,

        UPDATEBY=I_UpdateBy,
        CreateDate=sysdate,
        UpdateDate=sysdate,
        SHIFTTYPE=I_ShiftType

    where ShiftId=I_ShiftId;

    Commit;
    O_retVal := 0;
end if;
--open O_refcursor_retVal for select O_retVal from dual;
End;

EXCEPTION
    WHEN OTHERS THEN
        RAISE_APPLICATION_ERROR (-20001,
            I_ShiftName || ' :$:' ||
            I_IsRolling || ' :$:' ||
            I_CreateBy || ' :$:' ||
            I_UpdateBy || ' :$:' ||

            SQLERRM, TRUE) ;

End sp_ShiftName_Add_Update;
/

```

## 22.SP\_SHIFT\_ADD\_UPDATE

```

CREATE OR REPLACE PROCEDURE ATT_IN.sp_Shift_Add_Update
(
    I_SHIFTSETUPID In int,
    I_SEASONID In int,
    I_COMPID In int,
    I_SECTID In int,
    I_SHIFTID In int,
    I_SHIFTNAME In Varchar2,
    I_LOGINTIME In Date,

```

```

I_LOGOUTTIME In Date,
I_LATEBY In Date,
I_LUNCHLOGIN In Date,
I_LUNCHLOGOUT In Date,
I_REGULARHOUR In Date,
I_CreateBy IN varchar2,
I_UpdateBy IN varchar2,
I_AddOrUpdate IN varchar2
--O_refcursor_animation OUT SYS_REFCURSOR
)
as

/* Declare
I_SHIFTSETUPID int;
I_SEASONID int;
I_COMPID int;
I_SECTID int;
I_SHIFTID int;
I_SHIFTNAME Varchar2(50);
I_LOGINTIME Date;
I_LOGOUTTIME Date;
I_LATEBY Date;
I_LUNCHLOGIN Date;
I_LUNCHLOGOUT Date;
I_REGULARHOUR Date;
I_CreateBy varchar2(50);
I_UpdateBy varchar2(50);
I_AddOrUpdate varchar2(50);
ShiftNo int;
O_retVal int;
Begin
I_SHIFTSETUPID:=0;
I_SEASONID:=89;
I_COMPID :=1;
I_SECTID:=5;
I_SHIFTID:=321;
I_SHIFTNAME:='General Shift';
I_LOGINTIME:=(to_date('01-Jan-1900 08:00:00', 'DD-MM-YYYY
hh24:mi:ss')));
I_LOGOUTTIME:=(to_date('01-Jan-1900 17:00:00', 'DD-MM-YYYY
hh24:mi:ss')));
I_LATEBY:=(to_date('01-Jan-1900 08:05:00', 'DD-MM-YYYY
hh24:mi:ss')));
I_LUNCHLOGIN:=(to_date('01-Jan-1900 13:00:00', 'DD-MM-YYYY
hh24:mi:ss')));
I_LUNCHLOGOUT:=(to_date('01-Jan-1900 14:00:00', 'DD-MM-YYYY
hh24:mi:ss')));
I_REGULARHOUR:=(to_date('01-Jan-1900 08:00:00', 'DD-MM-YYYY
hh24:mi:ss')));
I_CreateBy:='superadmin';
I_UpdateBy:='superadmin';
I_AddOrUpdate:='Saved';
ShiftNo:=0;
O_retVal :=0;
*/

Begin
--Add

```

```

declare ShiftNo  int:=0;
O_retVal int;
Begin
    if I_AddOrUpdate='Saved' Then
        Select count(CompId) into ShiftNo from  tbl_Shift_Setup where
SectId=I_SectID  and SHIFTID = I_SHIFTID and SEASONID=I_SEASONID;
        if ShiftNo =0  then
            --DBMS_Output.put_Line('Test');

            INSERT INTO tbl_Shift_Setup
            (
                SEASONID,
                SECTID,
                SHIFTID,
                SHIFTNAME,
                LOGINTIME,
                LOGOUTTIME,
                LATEBY,
                LUNCHLOGIN,
                LUNCHLOGOUT,
                REGULARHOUR,
                CREATEBY,
                CreateDate,
                UPDATEBY,
                Updatedate,
                COMPID,
                DEPTID
            )
            values
            (
                I_SEASONID,
                I_SECTID,
                I_SHIFTID,
                I_SHIFTNAME,
                I_LOGINTIME,
                I_LOGOUTTIME,
                I_LATEBY,
                I_LUNCHLOGIN,
                I_LUNCHLOGOUT,
                I_REGULARHOUR,
                I_CREATEBY,
                SYSDATE,
                I_UpdateBy,
                SYSDATE,
                I_COMPID,
                I_SECTID
            );
            Commit;
            --DBMS_OUTPUT.PUT_LINE('Success');
            O_retVal := 0;
        else
            O_retVal := 1;
        End If;
    else
        update tbl_Shift_Setup
        set  SEASONID=I_SEASONID,
            COMPID=I_COMPID,
            SECTID=I_SECTID,
            SHIFTID=I_SHIFTID,

```

```

        SHIFTNAME=I_SHIFTNAME,
        LOGINTIME=I_LOGINTIME,
        LOGOUTTIME=I_LOGOUTTIME,
        LATEBY=I_LATEBY,
        LUNCHLOGIN=I_LUNCHLOGIN,
        LUNCHLOGOUT=I_LUNCHLOGOUT,
        REGULARHOUR=I_REGULARHOUR,
        CREATEBY=I_CREATEBY,
        CREATEDATE=sysdate,
        UPDATEBY=I_UPDATEBY,
        UPDATEDATE=sysdate
    where SHIFTSETUPID=I_SHIFTSETUPID;
    Commit;
    O_retVal := 0;
end if;
--open O_refcursor_animation for select O_retVal from dual;
--End;
EXCEPTION
    WHEN OTHERS THEN
        RAISE_APPLICATION_ERROR (-20001,
            I_SHIFTSETUPID || ' :$:' ||
            I_SEASONID || ' :$:' ||
            I_SECTID || ' :$:' ||
            I_SHIFTID || ' :$:' ||
            I_SHIFTNAME || ' :$:' ||
            I_LOGINTIME || ' :$:' ||
            I_LOGOUTTIME || ' :$:' ||
            I_LATEBY || ' :$:' ||
            I_LUNCHLOGIN || ' :$:' ||
            I_LUNCHLOGOUT || ' :$:' ||
            I_REGULARHOUR || ' :$:' ||
            I_CREATEBY || ' :$:' ||
            I_UPDATEBY || ' :$:' ||
            I_COMPID || ' :$:' ||
            I_SECTID || ' :$:' ||

            SQLERRM, TRUE) ;

End sp_Shift_Add_Update;

end;
/

```

## 23.SP\_SHORT\_REST\_CALCULATION

```

CREATE OR REPLACE PROCEDURE ATT_IN.sp_short_rest_calculation
(
    I_EmpId In int,
    I_OT_Date In date

)
As
Begin

    declare
        ----- HERE IS THE PROCEDURE

```

```

-- we have already picked the REG_IN,REG_OUT, OT_IN & OT_OUT of
previous day right ?
-- PICK KORA ACHE
-- SOB VALUE PICK KORA ACHE

-- now please do it again from the beginnig

--I Sts nvarchar2(30);
V_SHIFTLOG int;
cur_day_short_rest int;
diff_curr_in_pre_out int;
v_curr_day_sec_min_punch date;
v_previous_day_short_rest nvarchar2(30);
v_curr_day_min_punch date;
V_CUR_DAY_MAX_PUNCH DATE;
v_ot_count int;
diff_gen_punch int;
v_max_punch_time date;
diff_cur_ot_in_cur_reg_out int;
diff_pre_ot_in_cur_reg_out int;
diff_ot_out_reg_in int;
v_sec_min_punch date;
v_min_punch_time date;
V_COMPCARDID nvarchar2(30);
diff_min_punch_reg_out int;
diff_curr_in_cur_out int;
--pre day ot in date;
--pre day ot out date;
diff_min_punch_ot_out int;
diff_curr_out_ot_in int;
diff_curr_in_ot_out int;
diff_pre_out_in int;
emp_shift_previous nvarchar2(30);
diff_pre_out_in_srt int;
pre_day_reg_in_srt date;
pre_day_reg_out_srt date;
pre_day_ot_in_srt date;
pre_day_ot_out_srt date;
c_shift_in1 nvarchar2(30);
c_shift_in2 nvarchar2(30);
c_shift_out1 nvarchar2(30);
c_shift_out2 nvarchar2(30);

a_shift_out1 nvarchar2(30);
a_shift_out2 nvarchar2(30);

previous_day_counter int;

previous_day_in_time1 date;
previous_day_in_time2 date;

curr_day_c_in_time1 date;
curr_day_c_in_time2 date;

next_day_c_out_time1 date;
next_day_c_out_time2 date;

curr_day_in_time1 date;

```

```

curr_day_in_time2 date;

curr_day_out_time1 date;
curr_day_out_time2 date;


previous_day_out_time date;
special_ot_count int;


sp_ot_for_pre_c_shift date;

total_minute int;
total_minute_ot_sft int;

total_minut1 int;

total_minute_ot_sft1 int;

total_minute2 int;
total_minute_ot_sft2 int;

total_minute3 int;
total_minute_ot_sft3 int;

total_minute4 int;
total_minute_ot_sft4 int;

total_minute_reg int;

special_ot_actual nvarchar2(30);
special_ot_actual_ot_sft nvarchar2(30);
total_ot_reg_minute int;
special_ot_reg nvarchar2(30);

--26/08/2013
short_rest_default_val_in_min int;
short_rest_min_pre_day int;
short_rest_min_curr_day int;
previous_day date;
nextday date;

pre_day_ot_in date;
pre_day_ot_out date;
pre_day_reg_in date;
pre_day_reg_out date;

curr_day_ot_in date;
curr_day_ot_out date;
curr_day_reg_in date;
curr_day_reg_out date;

next_day_ot_in date;
next_day_ot_out date;
next_day_reg_in date;
next_day_reg_out date;

max_out_punch_pre_day date;
max_out_punch_curr_day date;
max_out_punch_next_day date;

```



```

pre_day_ot_o_g_reg_in int; -- pre_day_ot_o_g_reg_in(PREVIOUS DAT OT IS
GREATER THAN PREVIOUS DAY REGULAR IN)
pre_day_ot_o_s_reg_in int; -- pre_day_ot_o_g_reg_in(PREVIOUS DAT OT IS
GREATER THAN PREVIOUS DAY REGULAR IN)

curr_day_ot_o_g_reg_in int;
curr_day_ot_o_s_reg_in int;

curr_s_r_diff_pre_max_curr_min int;

previous_day_short_rest_srt int;

previous_day_short_rest int;
current_day_short_rest int;

pre_max_curr_min int;

pre_day_max date;
pre_day_min date;
curr_day_min date;
curr_day_max date;

--26/08/2013
--08/09/2013

emp_shift nvarchar2(30);
diff_pre_out_curr_in int;
short_rest_hour_2_shift int;
v_employee_type varchar2(15);
v_employee_category varchar2(20);
I_Sts varchar2(20);
--v_desgid varchar2(60);
v_desig_exist int;

v_shiftid int;
v_desgid varchar2(30);
v_desig_exist_96 int;
v_desig_exist_64 int;
v_labourcode int;
v_empcount int;
v_otapprove int;
emp_shift_next varchar2(20);

diff_next_in_next_out int;
v_shiftin date;
v_shiftout date;
v_after_ot int;
v_before_ot int;
diff_cot_out_nminpunch int;
v_shift_nminpunch date;
diff_cot_in_pmaxpunch int;
diff_pre_ot_in_pre_reg_out int;
V_SHIFT_ID int;

--08/09/2013

```

```

begin

    begin
        select
EMPLOYEE_TYPE,EMPLOYEE_CATEGORY,SHIFTID,DESIGNATIONID,LABOURCODE ,COMPCARDID into
v_employee_type,v_employee_category , v_shiftid,v_desgid,v_labourcode,
V_COMPCARDID from att_in.TBL_EMP_PERSONAL_INFO where EMPID=I_Empid;
    EXCEPTION
        when no_data_found then
            null;
        end;

    begin
        select count(*) into v_desig_exist from att_in.TBL_CHECK
        where DESGINATION=v_desgid;
    EXCEPTION
        when no_data_found then
            null;
        end;

    begin
        select count(*) into v_desig_exist_96 from att_in.TBL_CHECK
        where MAX_96_HOURS_OT=v_labourcode;
    EXCEPTION
        when no_data_found then
            null;
        end;

    begin
        select count(*) into v_desig_exist_64 from att_in.TBL_CHECK
        where MAX_64_HOURS_OT=v_labourcode;
    EXCEPTION
        when no_data_found then
            null;
        end;

    begin
        select count(*) into v_empcount from att_in.TBL_OT_ENTRY
        where EMPID=I_Empid
        and
        OT_OR_REG_DATE =
to_date(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR');
    EXCEPTION
        when no_data_found then
            null;
        end;

    begin
        select nvl(OT_APPROVE,0) into v_otapprove
        from att_in.TBL_OT_ENTRY
        where empid=I_EmpId
        and OT_OR_REG_DATE =
to_date(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');
    EXCEPTION
        when no_data_found then
            null;
        end;

```

```

        If v_employee_type in('P','G') and v_employee_category='Permanent' and
v_desig_exist=0 and v_desig_exist_96=0 and v_desig_exist_64 =0 and v_otapprove=1
then

```

```

        DBMS_OUTPUT.PUT_LINE ( 'v_empcount ');
        DBMS_OUTPUT.PUT_LINE ( v_empcount);

```

```

short_rest_default_val_in_min := 720;

```

```

begin

```

```

    select R_SHIFT
    into emp_shift_previous
    from att_in.TBL_OT_ENTRY
    where empid= I_EmpId
    and

```

```

OT_OR_REG_DATE = to_date(to_char(I_OT_Date-
1, 'DD/MM/RRRR'), 'DD/MM/RRRR');

```

```

EXCEPTION when no_data_found then
null;
end;

```

```

begin

```

```

    select R_SHIFT
    into emp_shift_next
    from att_in.TBL_OT_ENTRY
    where empid= I_EmpId
    and

```

```

OT_OR_REG_DATE =
to_date(to_char(I_OT_Date+1, 'DD/MM/RRRR'), 'DD/MM/RRRR');
EXCEPTION when no_data_found then
null;
end;

```

```

begin

```

```

    select R_SHIFT,R_SHIFT_ID
    into emp_shift,V_SHIFT_ID
    from att_in.TBL_OT_ENTRY
    where empid= I_EmpId
    and

```

```

OT_OR_REG_DATE =
to_date(to_char(I_OT_Date, 'DD/MM/RRRR'), 'DD/MM/RRRR');
EXCEPTION when no_data_found then
null;
end;

```

```

        If emp_shift like '4 Shift%' and v_empcount=1 then

```

```

begin

```

```

    select
    nvl(SHORT_REST_LOG_B,0)
    into
    previous_day_short_rest --

```

```

previous ot in, previous ot out,

```

```

from att_in.TBL_OT_ENTRY
where empid= I_EmpId
and
OT_OR_REG_DATE =
to_date(to_char(I_OT_Date-1,'DD/MM/RRRR'),'DD/MM/RRRR');
EXCEPTION
when too_many_rows then
null;
when no_data_found then
null;
when others then
null;

end;

begin
select
nvl(to_date(to_char(R_IN_TIME_PUNCH,'DD/MM/RRRR
HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'),''),
nvl(to_date(to_char(R_OUT_TIME_PUNCH,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS'),''),
to_date(to_char(OT_IN_TIME_PUNCH,'DD/MM/RRRR
HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'),
to_date(to_char(OT_OUT_TIME_PUNCH,'DD/MM/RRRR
HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'),
to_date(to_char(R_SHIFT_IN,'DD/MM/RRRR
HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'),
to_date(to_char(R_SHIFT_OUT,'DD/MM/RRRR
HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
into curr_day_reg_in,
curr_day_reg_out,curr_day_ot_in,curr_day_ot_out, v_shiftin, v_shiftout
--previous_ot_in, previous_ot_out,
from att_in.TBL_OT_ENTRY
where empid= I_EmpId
and
OT_OR_REG_DATE =
to_date(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR');
EXCEPTION
when too_many_rows then
null;
when no_data_found then
null;
when others then
null;
end;

select count (OT_HOUR) into v_ot_count
from att_in.TBL_OT_ENTRY
where empid= I_EmpId
and
OT_OR_REG_DATE =
to_date(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR');

If curr_day_reg_in is not null and
curr day reg out is not null and curr day ot in is not null

```

```

                                and curr_day_ot_out is not null then
                                if curr_day_reg_in<curr_day_ot_in then
                                v_after_ot:=1;
                                if curr_day_ot_in= curr_day_reg_out or
curr_day_ot_out=curr_day_reg_in then
                                dbms_OUTPUT.PUT_LINE(' A ot condition
true');

dbms_OUTPUT.PUT_LINE(to_char(curr_day_reg_in,'DD/MM/RRRR HH24:MI:SS'));
dbms_OUTPUT.PUT_LINE(to_char(curr_day_reg_out,'DD/MM/RRRR HH24:MI:SS'));
dbms_OUTPUT.PUT_LINE(to_char(curr_day_ot_in,'DD/MM/RRRR HH24:MI:SS'));
dbms_OUTPUT.PUT_LINE(to_char(curr_day_ot_out,'DD/MM/RRRR HH24:MI:SS'));

                                select
to_date(to_char(min(PunchTime),'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
                                into v_min_punch_time
                                from att_in.TBL_RAW_DATA_CMIS
                                where COMPCARDID = V_COMPCARDID
                                and PUNCHDATE =
to_date(to_char(I_OT_Date+1,'DD/MM/RRRR'),'DD/MM/RRRR');

dbms_OUTPUT.PUT_LINE('diff_cot_out_nminpunch1111111111111111');
dbms_OUTPUT.PUT_LINE(to_char(v_min_punch_time,'DD/MM/RRRR HH24:MI:SS'));

                                if v_min_punch_time<=curr_day_ot_out then

                                begin
                                select
to_date(to_char(v_min_punch_time + interval '60' MINUTE,'DD/MM/RRRR
HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')

                                into v_shift_nminpunch
                                from dual;
                                end;

                                select
to_date(to_char(min(PunchTime),'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
                                into v_min_punch_time
                                from att_in.TBL_RAW_DATA_CMIS
                                where COMPCARDID = V_COMPCARDID
                                and PUNCHDATE =
to_date(to_char(I_OT_Date+1,'DD/MM/RRRR'),'DD/MM/RRRR')
                                and PunchTime>v_shift_nminpunch;

                                end if;

dbms_OUTPUT.PUT_LINE('diff_cot_out_nminpunch1706');
dbms_OUTPUT.PUT_LINE(to_char(v_min_punch_time,'DD/MM/RRRR HH24:MI:SS'));

                                Select

```

```

abs(NVL(SUM(datediff('hh',v_min_punch_time,curr_day_ot_out)),0) * 60)
        into diff_cot_out_nminpunch
        from dual;

        if diff_cot_out_nminpunch >30 and
diff_cot_out_nminpunch < 720 then

        short_rest_hour_2_shift :=
short_rest_default_val_in_min - diff_cot_out_nminpunch;

        if short_rest_hour_2_shift >= 240 then
            short_rest_hour_2_shift := 240;
        elsif short_rest_hour_2_shift < 30 then
            short_rest_hour_2_shift:=0;
        else
            short_rest_hour_2_shift :=
short_rest_hour_2_shift;
        end if;

dbms_OUTPUT.PUT_LINE('diff_cot_out_nminpunch');
        ----
dbms_OUTPUT.PUT_LINE(to_char(curr_day_reg_in,'DD/MM/RRRR HH24:MI:SS'));
dbms_OUTPUT.PUT_LINE(short_rest_hour_2_shift );

        --if previous_day_short_rest<>2
then

        update att_in.TBL_OT_ENTRY
        set SHORT_REST_LOG_B = 1,

        SHORT_REST_IN_MIN =
short_rest_hour_2_shift,
        SHORT_REST_HOUR =
ATT_IN.fn_minute_to_time_shortrest_ot(short_rest_hour_2_shift)
        where empid= I_EmpId
        and
        OT_OR_REG_DATE =
to_date(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR');
        commit;
        dbms_OUTPUT.PUT_LINE(' Update
complete');

        --end if;
        end if;
    end if;

    elsif curr_day_reg_in>curr_day_ot_in then

dbms_OUTPUT.PUT_LINE(to_char(curr_day_reg_in,'DD/MM/RRRR HH24:MI:SS'));
dbms_OUTPUT.PUT_LINE(to_char(curr_day_ot_in,'DD/MM/RRRR HH24:MI:SS'));

        v_before_ot:=1;

```

```

true');
                                dbms_OUTPUT.PUT_LINE(' B ot condition

                                if curr_day_ot_in= curr_day_reg_out or
curr_day_ot_out=curr_day_reg_in then

                                begin
                                select

nvl(to_date(to_char(R_IN_TIME_PUNCH, 'DD/MM/RRRR HH24:MI:SS'), 'DD/MM/RRRR
HH24:MI:SS'), ''),

nvl(to_date(to_char(R_OUT_TIME_PUNCH, 'DD/MM/RRRR HH24:MI:SS'), 'DD/MM/RRRR
HH24:MI:SS'), ''),

to_date(to_char(OT_IN_TIME_PUNCH, 'DD/MM/RRRR HH24:MI:SS'), 'DD/MM/RRRR HH24:MI:SS'),

to_date(to_char(OT_OUT_TIME_PUNCH, 'DD/MM/RRRR HH24:MI:SS'), 'DD/MM/RRRR
HH24:MI:SS'),

                                nvl(SHORT_REST_LOG_B,0)
                                into pre_day_reg_in,

pre_day_reg_out,pre_day_ot_in,pre_day_ot_out,
                                previous_day_short_rest --
previous_ot_in, previous_ot_out,

                                from att_in.TBL_OT_ENTRY
                                where empid= I_EmpId
                                and
                                OT_OR_REG_DATE =
to_date(to_char(I_OT_Date-1, 'DD/MM/RRRR'), 'DD/MM/RRRR');
                                EXCEPTION
                                when too_many_rows then
                                null;
                                when no_data_found then
                                null;
                                when others then
                                null;
                                end;

                                dbms_OUTPUT.PUT_LINE('ppunch');

dbms_OUTPUT.PUT_LINE(to_char(pre_day_reg_in, 'DD/MM/RRRR HH24:MI:SS'));
dbms_OUTPUT.PUT_LINE(to_char(pre_day_reg_out, 'DD/MM/RRRR HH24:MI:SS'));
dbms_OUTPUT.PUT_LINE(to_char(pre_day_ot_in, 'DD/MM/RRRR HH24:MI:SS'));
dbms_OUTPUT.PUT_LINE(to_char(pre_day_ot_out, 'DD/MM/RRRR HH24:MI:SS'));

                                dbms_OUTPUT.PUT_LINE('cpunch');

dbms_OUTPUT.PUT_LINE(to_char(curr_day_reg_in, 'DD/MM/RRRR HH24:MI:SS'));
dbms_OUTPUT.PUT_LINE(to_char(curr_day_reg_out, 'DD/MM/RRRR HH24:MI:SS'));
dbms_OUTPUT.PUT_LINE(to_char(curr_day_ot_in, 'DD/MM/RRRR HH24:MI:SS'));

```

```

dbms_OUTPUT.PUT_LINE(to_char(curr_day_ot_out,'DD/MM/RRRR HH24:MI:SS'));
                                if previous_day_short_rest<>1 or
previous_day_short_rest is null then
                                if pre_day_ot_out is null
then
                                v_max_punch_time:=pre_day_reg_out;
                                elsif pre_day_reg_out is not
null and pre_day_ot_out is not null then
                                if
pre_day_reg_out>pre_day_ot_out then
                                v_max_punch_time:=pre_day_reg_out;
                                elsif
pre_day_ot_out>pre_day_reg_out then
                                v_max_punch_time:=pre_day_ot_out;

                                end if;

                                end if;

dbms_OUTPUT.PUT_LINE(to_char(v_max_punch_time,'DD/MM/RRRR HH24:MI:SS'));
                                --
dbms_OUTPUT.PUT_LINE(to_char(curr_day_reg_in,'DD/MM/RRRR HH24:MI:SS'));

                                Select
abs(NVL(SUM(datediff('hh',v_max_punch_time,curr_day_ot_in)),0) * 60)
                                into diff_cot_in_pmaxpunch
                                from dual;

dbms_OUTPUT.PUT_LINE(to_char(v_max_punch_time,'DD/MM/RRRR HH24:MI:SS'));
dbms_OUTPUT.PUT_LINE(to_char(curr_day_reg_in,'DD/MM/RRRR HH24:MI:SS'));

                                if diff_cot_in_pmaxpunch >30
and diff_cot_in_pmaxpunch < 720 then

                                short_rest_hour_2_shift :=
short_rest_default_val_in_min - diff_cot_in_pmaxpunch;

                                if short_rest_hour_2_shift >=
240 then
                                short_rest_hour_2_shift :=
240;
                                elsif short_rest_hour_2_shift <
30 then
                                short_rest_hour_2_shift:=0;
                                else
                                short_rest_hour_2_shift :=
short_rest_hour_2_shift;

                                end if;

```



```

dbms_OUTPUT.PUT_LINE('diff_cot_in_pmaxpunch');

dbms_OUTPUT.PUT_LINE(diff_cot_in_pmaxpunch);

dbms_OUTPUT.PUT_LINE(short_rest_hour_2_shift );

short_rest_hour_2_shift,
ATT_IN.fn_minute_to_time_shortrest_ot(short_rest_hour_2_shift)
to_date(to_char(I_OT_Date, 'DD/MM/RRRR'), 'DD/MM/RRRR');
complete');

update att_in.TBL_OT_ENTRY
set SHORT_REST_LOG_B = 2,
SHORT_REST_IN_MIN =
SHORT_REST_HOUR =
where empid= I_EmpId
and
OT_OR_REG_DATE =
commit;
dbms_OUTPUT.PUT_LINE(' Update
end if;

end if;
end if;

end if;

--dbms_OUTPUT.PUT_LINE('v_after_ot');
--dbms_OUTPUT.PUT_LINE(v_after_ot);
--dbms_OUTPUT.PUT_LINE('v before ot');
--dbms_OUTPUT.PUT_LINE(v_before_ot);

IF curr_day_reg_in= curr_day_ot_out THEN
--dbms_OUTPUT.PUT_LINE('4 shift Condition
1');

IF curr_day_reg_out> curr_day_ot_in then
V_CUR_DAY_MAX_PUNCH:=curr_day_reg_out;
else
V_CUR_DAY_MAX_PUNCH:=curr_day_ot_in;

END IF;

elsif curr_day_reg_out= curr_day_ot_in then
--dbms_OUTPUT.PUT_LINE('4 shift Condition
2');

IF curr_day_reg_in> curr_day_ot_out then
V_CUR_DAY_MAX_PUNCH:=curr_day_reg_in;
else
V_CUR_DAY_MAX_PUNCH:=curr_day_ot_out;

```

```

END IF;

END IF;
--
dbms_OUTPUT.PUT_LINE(to_char(V_CUR_DAY_MAX_PUNCH,'DD/MM/RRRR HH24:MI:SS'));

/* if curr_day_reg_in is not null and
curr day reg out is not null and curr_day_ot_in is not null and curr_day_ot_out is
not null then

select
to_date(to_char(min(PunchTime),'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
into v_min_punch_time
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = V_COMPCARDID
and PUNCHDATE =
to_date(to_char(I_OT_Date+1,'DD/MM/RRRR'),'DD/MM/RRRR');

--dbms_OUTPUT.PUT_LINE('1st Min Punch of Next
Day');
--
dbms_OUTPUT.PUT_LINE(to_char(v_min_punch_time,'DD/MM/RRRR HH24:MI:SS'));

Select
abs(NVL(SUM(datediff('hh',v_min_punch_time, V_CUR_DAY_MAX_PUNCH)),0) * 60)
into diff_min_punch_ot_out
from dual;

if diff_min_punch_ot_out<60 then
select to_date(to_char(v_min_punch_time +
interval '60' MINUTE,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
into v_sec_min_punch
from dual;

select
to_date(to_char(min(PunchTime),'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
into v_min_punch_time
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = V_COMPCARDID
and PUNCHDATE =
to_date(to_char(I_OT_Date+1,'DD/MM/RRRR'),'DD/MM/RRRR')
and to_date(to_char(punchtime,'DD/MM/RRRR
HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')>
to_date
(to_char(v_sec_min_punch,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS');

--dbms_OUTPUT.PUT_LINE('2nd Min Punch of
Next Day');
--
dbms_OUTPUT.PUT_LINE(to_char(v_min_punch_time,'DD/MM/RRRR HH24:MI:SS'));

Select
abs(NVL(SUM(datediff('hh',v_min_punch_time, V_CUR_DAY_MAX_PUNCH)),0) * 60)
into diff_min_punch_ot_out
from dual;

```

```

end if;

--dbms_OUTPUT.PUT_LINE(diff_min_punch_ot_out
);

--end if;
-- type what is the previous value
-- now check it out
if diff_min_punch_ot_out >30 and
diff_min_punch_ot_out < 720 and v_ot_count =1 then

--dbms_OUTPUT.PUT_LINE('05/04/2014 ');
short_rest_hour_2_shift :=
short_rest_default_val_in_min - diff_min_punch_ot_out;

if short_rest_hour_2_shift >= 240 then
short_rest_hour_2_shift := 240;
elsif short_rest_hour_2_shift < 30 then
short_rest_hour_2_shift:=0;
else
short_rest_hour_2_shift :=
short_rest_hour_2_shift;

end if;

update att_in.TBL_OT_ENTRY
set SHORT_REST_LOG_B = 1,
SHORT_REST_IN_MIN =
short rest min curr day,
SHORT_REST_HOUR =
ATT_IN.fn_minute_to_time_shortrest_ot(short_rest_hour_2_shift)
where empid= I_EmpId
and
OT_OR_REG_DATE =
to_date(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR');
commit;
--dbms_OUTPUT.PUT_LINE(' Update
complete');

end if;

--paste the update here

end if;*/
-- upto here result is fine
-- 05/05/2014 upto here result is OK
-- after here where is the UPDATE part ?
-- r u there ?
-- yes

-- update ta upore dite hobe mone hoi
-- from here BREAK OT is STARTED right ?
-- yes

begin
select
nvl(to date(to char(R IN TIME PUNCH,'DD/MM/RRRR

```

```

HH24:MI:SS'), 'DD/MM/RRRR HH24:MI:SS'), ''),

nvl(to_date(to_char(R_OUT_TIME_PUNCH, 'DD/MM/RRRR HH24:MI:SS'), 'DD/MM/RRRR
HH24:MI:SS'), ''),

to_date(to_char(OT_IN_TIME_PUNCH, 'DD/MM/RRRR
HH24:MI:SS'), 'DD/MM/RRRR HH24:MI:SS'),
to_date(to_char(OT_OUT_TIME_PUNCH, 'DD/MM/RRRR
HH24:MI:SS'), 'DD/MM/RRRR HH24:MI:SS'),
nvl(SHORT_REST_LOG_B, 0)
into pre_day_reg_in,
pre_day_reg_out, pre_day_ot_in, pre_day_ot_out,
previous_day_short_rest --previous_ot_in,

previous_ot_out,

from att_in.TBL_OT_ENTRY
where empid= I_EmpId
and
OT_OR_REG_DATE = to_date(to_char(I_OT_Date-
1, 'DD/MM/RRRR'), 'DD/MM/RRRR');

EXCEPTION
when too_many_rows then
null;
when no_data_found then
null;
when others then
null;
end;

dbms_OUTPUT.PUT_LINE('19062014' );

dbms_OUTPUT.PUT_LINE(to_char(pre_day_ot_in, 'DD/MM/RRRR HH24:MI:SS'));

dbms_OUTPUT.PUT_LINE(to_char(pre_day_ot_out, 'DD/MM/RRRR HH24:MI:SS'));

dbms_OUTPUT.PUT_LINE(to_char(curr_day_reg_in, 'DD/MM/RRRR HH24:MI:SS'));

if pre_day_ot_in is not null
and pre_day_ot_out is not null and curr_day_reg_in is not null then

dbms_OUTPUT.PUT_LINE('short_rest_hour_2_shift2703' );

dbms_OUTPUT.PUT_LINE(short_rest_hour_2_shift );

Select
abs(NVL(SUM(datediff('hh', pre_day_ot_out, curr_day_reg_in)), 0) * 60)
into diff_ot_out_reg_in
from dual;

Select
abs(NVL(SUM(datediff('hh', pre_day_ot_in, curr_day_reg_out)), 0) * 60)
into
diff_pre_ot_in_cur_reg_out
from dual;

Select
abs(NVL(SUM(datediff('hh', pre_day_ot_in, pre_day_reg_out)), 0) * 60)
into
diff_pre_ot_in_pre_reg_out
from dual;

```

```

-- reg_out & ot_in er diff
is not less than 10 or 0 so
--
if (diff_ot_out_reg_in<10
or diff_ot_out_reg_in=0 ) and diff_pre_ot_in_cur_reg_out>510 and v_ot_count =1 then

--if curr_day_ot_in is
null and curr_day_ot_out is null then

if
diff_pre_ot_in_pre_reg_out>30 and diff_pre_ot_in_pre_reg_out<720 then
short_rest_hour_2_shift
:= short_rest_default_val_in_min - diff_pre_ot_in_pre_reg_out;

if
short_rest_hour_2_shift >= 240 then

elseif
short_rest_hour_2_shift < 30 then

else
short_rest_hour_2_shift:=0;

end if;

short_rest_hour_2_shift := short_rest_hour_2_shift ;

--
--
dbms_OUTPUT.PUT_LINE('short_rest_hour_2_shift2703' );
dbms_OUTPUT.PUT_LINE(short_rest_hour_2_shift );

update
att_in.TBL_OT_ENTRY
set
SHORT_REST_LOG_B = 1,
SHORT_REST_IN_MIN = short_rest_min_curr_day,
SHORT_REST_HOUR = ATT_IN.fn_minute_to_time_shortrest_ot(short_rest_hour_2_shift)
where
empid= I_EmpId
and
OT_OR_REG_DATE = to_date(to_char(I_OT_Date-1,'DD/MM/RRRR'),'DD/MM/RRRR');
end if;
commit;

--end if;

/* if
curr_day_ot_in is not null and curr_day_ot_out is not null then
Select
abs(NVL(SUM(datediff('hh',curr_day_ot_in,curr_day_reg_out)),0) * 60)
into
diff_cur_ot_in_cur_reg_out
from
dual;
```

```

diff_cur_ot_in_cur_reg_out >30 and diff_cur_ot_in_cur_reg_out < 720
v ot count =1 then
where we r geting the value of short_rest_default_val_in_min
check
short_rest_hour_2_shift := short_rest_default_val_in_min -
diff_cur_ot_in_cur_reg_out;

short_rest_hour_2_shift >= 240 then
short_rest_hour_2_shift := 240;
short_rest_hour_2_shift < 30 then
short_rest_hour_2_shift:=0;

short rest hour 2 shift := short rest hour 2 shift ;

dbms_OUTPUT.PUT_LINE('short_rest_hour_2_shift2703' );
dbms_OUTPUT.PUT_LINE(short_rest_hour_2_shift );

att in.TBL OT ENTRY
SHORT_REST_LOG_B = 1,
SHORT_REST_IN_MIN = short_rest_min_curr_day,
SHORT_REST_HOUR = ATT_IN.fn_minute_to_time_shortrest_ot(short_rest_hour_2_shift)
empid= I_EmpId

OT_OR_REG_DATE = to_date(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR');

end if;
end if;*/
end if;
--
dbms_OUTPUT.PUT_LINE(diff_ot_out_reg_in );
--
dbms_OUTPUT.PUT_LINE(diff_pre_ot_in_cur_reg_out );

--here
end if;

```

```

        elsif emp_shift like '2 Shift%' and v_empcount=1 then
            BEGIN
                SELECT SHIFT_LOG INTO V_SHIFTLOG FROM
ATT_IN.TBL_SHIFTNAMESETTINGS
                WHERE SHIFTID=V_SHIFT_ID;

                EXCEPTION when no_data_found then
                    null;

                END;

            begin
                select
                    nvl(to_date(to_char(R_IN_TIME_PUNCH,'DD/MM/RRRR
HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'),''),
nvl(to_date(to_char(R_OUT_TIME_PUNCH,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS'),''),
                    to_date(to_char(OT_IN_TIME_PUNCH,'DD/MM/RRRR
HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'),
                    to_date(to_char(OT_OUT_TIME_PUNCH,'DD/MM/RRRR
HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'),
                    SHORT_REST_LOG_B
                    into pre_day_reg_in,
                    pre_day_reg_out,pre_day_ot_in,pre_day_ot_out,
                    previous_day_short_rest --previous_ot_in,
previous ot out,

                    from att_in.TBL_OT_ENTRY
                    where empid= I_EmpId
                    and
                    OT_OR_REG_DATE = to_date(to_char(I_OT_Date-
1,'DD/MM/RRRR'),'DD/MM/RRRR');

                EXCEPTION
                    when too_many_rows then
                        null;
                    when no_data_found then
                        null;
                    when others then
                        null;
                    end;

            begin
                select
                    nvl(to_date(to_char(R_IN_TIME_PUNCH,'DD/MM/RRRR
HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'),''),
nvl(to_date(to_char(R_OUT_TIME_PUNCH,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS'),''),
                    to_date(to_char(OT_IN_TIME_PUNCH,'DD/MM/RRRR
HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'),
                    to_date(to_char(OT_OUT_TIME_PUNCH,'DD/MM/RRRR
HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
                    into curr_day_reg_in,
                    curr_day_reg_out,curr_day_ot_in,curr_day_ot_out
                    from att_in.TBL_OT_ENTRY
                    where empid= I_EmpId
                    and
                    OT_OR_REG_DATE =
to date(to char(I OT Date,'DD/MM/RRRR'),'DD/MM/RRRR');

```

```

EXCEPTION
    when too_many_rows then
        null;
    when no_data_found then
        null;
    when others then
        null;
end;

begin
select
    nvl(to_date(to_char(R_IN_TIME_PUNCH,'DD/MM/RRRR
HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'),''),
    nvl(to_date(to_char(R_OUT_TIME_PUNCH,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS'),''),
    to_date(to_char(OT_IN_TIME_PUNCH,'DD/MM/RRRR
HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'),
    to_date(to_char(OT_OUT_TIME_PUNCH,'DD/MM/RRRR
HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
    into next_day_reg_in,
    next_day_reg_out,next_day_ot_in,next_day_ot_out
from att_in.TBL_OT_ENTRY
where empid= I_EmpId
and
    OT_OR_REG_DATE =
to_date(to_char(I_OT_Date+1,'DD/MM/RRRR'),'DD/MM/RRRR');
EXCEPTION
    when too_many_rows then
        null;
    when no_data_found then
        null;
    when others then
        null;
end;

select count (OT_HOUR) into v_ot_count
from att_in.TBL_OT_ENTRY
where empid= I_EmpId
and
    OT_OR_REG_DATE =
to_date(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR');

    dbms_OUTPUT.PUT_LINE('v_ot_count' );
    dbms_OUTPUT.PUT_LINE(v_ot_count );
    --
dbms_OUTPUT.PUT_LINE(diff_min_punch_ot_out );

dbms_OUTPUT.PUT_LINE(to_char(curr_day_reg_in,'DD/MM/RRRR HH24:MI:SS'));
dbms_OUTPUT.PUT_LINE(to_char(curr_day_reg_out,'DD/MM/RRRR HH24:MI:SS'));
    --
dbms_OUTPUT.PUT_LINE(to_char(pre_day_ot_in,'DD/MM/RRRR HH24:MI:SS'));
    --
dbms_OUTPUT.PUT_LINE(to_char(pre_day_ot_out,'DD/MM/RRRR HH24:MI:SS'));
    Select
abs(NVL(SUM(datediff('hh',curr day reg in,curr day reg out)),0) * 60)

```



```

into diff_curr_in_cur_out
from dual;

Select
abs(NVL(SUM(datediff('hh',next_day_reg_in,next_day_reg_out)),0) * 60)
into diff_next_in_next_out
from dual;

Select
abs(NVL(SUM(datediff('hh',pre_day_reg_out,curr_day_reg_in)),0) * 60)
into diff_curr_in_pre_out
from dual;

--if diff_curr_in_cur_out>510 then
dbms_OUTPUT.PUT_LINE(diff_curr_in_cur_out );
select
to_date(to_char(min(PunchTime), 'DD/MM/RRRR HH24:MI:SS'), 'DD/MM/RRRR HH24:MI:SS')
into v_min_punch_time
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = V_COMPCARDID
and PUNCHDATE =
to_date(to_char(I_OT_Date+1, 'DD/MM/RRRR'), 'DD/MM/RRRR');

dbms_OUTPUT.PUT_LINE(to_char(v_min_punch_time, 'DD/MM/RRRR HH24:MI:SS'));

Select
abs(NVL(SUM(datediff('hh',curr_day_reg_out,v_min_punch_time)),0) * 60)
into diff_min_punch_reg_out
from dual;
dbms_OUTPUT.PUT_LINE('diff_curr_in_pre_out' );
dbms_OUTPUT.PUT_LINE(diff_curr_in_pre_out );
dbms_OUTPUT.PUT_LINE('V_SHIFTLOG' );
dbms_OUTPUT.PUT_LINE(V_SHIFTLOG );

--DBMS_OUTPUT.PUT_LINE (
'previous_day_short_rest');
--DBMS_OUTPUT.PUT_LINE (
previous_day_short_rest);

--end if;
-----
18062014stsrt-----
if diff_curr_in_cur_out>510 and V_SHIFTLOG
=1 then

DBMS_OUTPUT.PUT_LINE ( 'AOT' );

Select
abs(NVL(SUM(datediff('hh',next_day_reg_in,curr_day_reg_out)),0) * 60)
into diff_next_in_next_out
from dual;

if diff_next_in_next_out>30 and
diff_next_in_next_out<720 and v_ot_count =1 then

short_rest_hour_2_shift :=

```

```

short_rest_default_val_in_min - diff_next_in_next_out;

                                if short_rest_hour_2_shift >= 240 then
                                    short_rest_hour_2_shift := 240;
                                elsif short_rest_hour_2_shift < 30 then
                                    short_rest_hour_2_shift:=0;
                                else
                                    short_rest_hour_2_shift :=

short_rest_hour_2_shift;

                                end if;

                                DBMS_OUTPUT.PUT_LINE (

'short_rest_hour_2_shift ');

                                DBMS_OUTPUT.PUT_LINE (

short_rest_hour_2_shift);

                                update att_in.TBL_OT_ENTRY
                                set SHORT_REST_LOG_B = 1,
                                SHORT_REST_IN_MIN =

short_rest_min_curr_day,

                                SHORT_REST_HOUR =
ATT_IN.fn_minute_to_time_shortrest_ot(short_rest_hour_2_shift)
                                where empid= I_EmpId
                                and
                                OT_OR_REG_DATE =
to_date(to_char(I_OT_Date, 'DD/MM/RRRR'), 'DD/MM/RRRR');
                                commit;
                                end if;

                                elsif diff_curr_in_cur_out>510 and
V_SHIFTLOG =2 then

                                DBMS_OUTPUT.PUT_LINE ( 'BOT' );

                                Select
                                abs(NVL(SUM(datediff('hh',pre_day_reg_out,curr_day_reg_in)),0) * 60)
                                into diff_curr_in_pre_out
                                from dual;

                                if diff_curr_in_pre_out>30 and
diff_curr_in_pre_out<720 and v_ot_count =1 then

                                short_rest_hour_2_shift :=
short_rest_default_val_in_min - diff_curr_in_pre_out;

                                if short_rest_hour_2_shift >= 240
then
                                    short_rest_hour_2_shift := 240;
                                elsif short_rest_hour_2_shift < 30
then
                                    short_rest_hour_2_shift:=0;
                                else
                                    short_rest_hour_2_shift :=

short_rest_hour_2_shift;

                                end if;

                                DBMS_OUTPUT.PUT_LINE (

```

```

'short_rest_hour_2_shift ');

short_rest_hour_2_shift);

                                DBMS_OUTPUT.PUT_LINE (

                                -- if previous_day_short_rest<>1 or
previous_day_short_rest is null then
                                update att_in.TBL_OT_ENTRY
                                set SHORT_REST_LOG_B = 2,
                                SHORT_REST_IN_MIN =
short_rest_min_curr_day,
                                SHORT_REST_HOUR =
ATT_IN.fn_minute_to_time_shortrest_ot(short_rest_hour_2_shift)
                                where empid= I_EmpId
                                and
                                OT_OR_REG_DATE =
to_date(to_char(I_OT_Date, 'DD/MM/RRRR'), 'DD/MM/RRRR');
                                commit;

                                --end if;
                                end if;

                                end if;

                                dbms_OUTPUT.PUT_LINE('diff_curr_in_cur_out' );
                                dbms_OUTPUT.PUT_LINE(diff_curr_in_cur_out );
                                /*
                                if previous day short rest is null and
diff_curr_in_cur_out>510 and next_day_reg_in is not null then
                                dbms_OUTPUT.PUT_LINE('A' );

dbms OUTPUT.PUT LINE(cur day short rest );

                                Select
abs(NVL(SUM(datediff('hh',next_day_reg_in,curr_day_reg_out)),0) * 60)
                                into diff_next_in_next_out
                                from dual;

                                if diff_next_in_next_out>30 and
diff_next_in_next_out<720 and v_ot_count =1 then

                                short_rest_hour_2_shift :=
short_rest_default_val_in_min - diff_next_in_next_out;

                                if short_rest_hour_2_shift >= 240 then
                                short_rest_hour_2_shift := 240;
                                elsif short_rest_hour_2_shift < 30 then
                                short_rest_hour_2_shift:=0;
                                else
                                short_rest_hour_2_shift :=
short_rest_hour_2_shift;
                                end if;

                                DBMS_OUTPUT.PUT_LINE (

                                DBMS_OUTPUT.PUT_LINE (
short_rest_hour_2_shift);

```

```

cur day short rest is null then
    if cur_day_short_rest<>1 or
    update att_in.TBL_OT_ENTRY
    set SHORT_REST_LOG_B = 1,
    SHORT_REST_IN_MIN =
short_rest_min_curr_day,
    SHORT REST HOUR =
ATT_IN.fn_minute_to_time_shortrest_ot(short_rest_hour_2_shift)
    where empid= I_EmpId
    and
    OT_OR_REG_DATE =
to_date(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR');
    commit;
    end if;
end if;*/

-----18062014
end-----
-----close18062014-----

/*if diff_min_punch_reg_out >30 and
diff_min_punch_reg_out < 720 and diff_curr_in_cur_out>510 and
emp_shift_next=emp_shift then --and diff_curr_in_cur_out>510 and v_ot_count =1
then

    short_rest_hour_2_shift :=
short_rest_default_val_in_min - diff_min_punch_reg_out;

    if short_rest_hour_2_shift >= 240 then
        short_rest_hour_2_shift := 240;
    elsif short_rest_hour_2_shift < 30 then
        short_rest_hour_2_shift:=0;
    else
        short_rest_hour_2_shift :=
short_rest_hour_2_shift;
    end if;

    DBMS_OUTPUT.PUT_LINE (
'short_rest_hour_2_shift ');
    DBMS_OUTPUT.PUT_LINE (
short_rest_hour_2_shift);

    update att_in.TBL_OT_ENTRY
    set SHORT_REST_LOG_B = 1,
    SHORT_REST_IN_MIN =
short rest min curr day,
    SHORT REST HOUR =
ATT_IN.fn_minute_to_time_shortrest_ot(short_rest_hour_2_shift)
    where empid= I_EmpId
    and
    OT_OR_REG_DATE =
to_date(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR');
    commit;

    elsif diff_min_punch_reg_out >30 and

```

```

diff_min_punch_reg_out < 720 and diff_next_in_next_out>510 then --and
emp_shift_next= emp_shift then
short_rest_hour_2_shift :=
short_rest_default_val_in_min - diff_min_punch_reg_out;

if short_rest_hour_2_shift >=
240 then
short_rest_hour_2_shift :=
240;
elseif short_rest_hour_2_shift <
30 then
short_rest_hour_2_shift:=0;
else
short_rest_hour_2_shift :=
short_rest_hour_2_shift;
end if;

DBMS_OUTPUT.PUT_LINE (
'short_rest_hour_2_shift ');
DBMS_OUTPUT.PUT_LINE (
short_rest_hour_2_shift);

update att_in.TBL_OT_ENTRY
set SHORT_REST_LOG_B = 1,
SHORT_REST_IN_MIN =
SHORT_REST_HOUR =
ATT_IN.fn_minute_to_time_shortrest_ot(short_rest_hour_2_shift)
where empid= I_EmpId
and
OT_OR_REG_DATE =
to_date(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR');
commit;

end if;*/
-----close18062014-----
-----

elseif v_empcount=0 then -----for general shift shortrest

begin
select to_date(to_char(TIMEOUT,'DD/MM/RRRR
HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
into v_max_punch_time
from att_in.TBL_PROCESSED_DATA
where empid= I_EmpId
and PUNCHDATE =
to_date(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR');
EXCEPTION
when too_many_rows then
null;
when no_data_found then
null;
when others then
null;

```

```

end;

begin
select
to_date(to_char(min(PunchTime), 'DD/MM/RRRR HH24:MI:SS'), 'DD/MM/RRRR HH24:MI:SS')
into v_min_punch_time
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = V_COMPCARDID
and PUNCHDATE =
to_date(to_char(I_OT_Date+1, 'DD/MM/RRRR'), 'DD/MM/RRRR');
EXCEPTION
when too_many_rows then
null;
when no_data_found then
null;
when others then
null;
end;

Select
abs(NVL(SUM(datediff('hh',v_max_punch_time,v_min_punch_time)),0) * 60)
into diff_gen_punch
from dual;

dbms_OUTPUT.PUT_LINE(diff_gen_punch );
if diff_gen_punch >30 and diff_gen_punch <
720 then

short_rest_hour_2_shift :=
short_rest_default_val_in_min - diff_gen_punch;

if short_rest_hour_2_shift >= 240 then
short_rest_hour_2_shift := 240;
elsif short_rest_hour_2_shift < 30 then
short_rest_hour_2_shift:=0;
else
short_rest_hour_2_shift :=
short_rest_hour_2_shift;
end if;

DBMS_OUTPUT.PUT_LINE (
'short_rest_hour_2_shift ');
DBMS_OUTPUT.PUT_LINE (
short_rest_hour_2_shift);

/*update att_in.TBL_PROCESSED_DATA
set
EXTRAOT =
ATT_IN.fn_minute_to_time_shortrest_ot(short_rest_hour_2_shift)
where empid= I_EmpId
and
PUNCHDATE =
to_date(to_char(I_OT_Date, 'DD/MM/RRRR'), 'DD/MM/RRRR');
commit;*/

```

```

end if;

-- go to 4 shift procedure
-- current day STATUS pick koren

end if;
else

update att_in.TBL_OT_ENTRY
set  SHORT_REST_IN_MIN = '',
SHORT_REST_HOUR = ''
where empid= I_EmpId
and
OT_OR_REG_DATE =
to_date(to_char(I_OT_Date, 'DD/MM/RRRR'), 'DD/MM/RRRR');
commit;

end if; -- ei end if er if kotha thike suru hoise ?

end;
-- EXECUTE ATT_IN.SP_INSERT_ROWDATA('000035288', '01-may-2014', '30-may-2014');
-- EXECUTE ATT_IN.sp_short_rest_calculation1506(35288, '05-may-2014');
-- now check in OT ENTRY TABLE
end;
/

```

## 24.SP\_SPECIAL\_OT\_CALCULATION

```

CREATE OR REPLACE PROCEDURE ATT_IN.sp_special_ot_calculation
(
I_EmpId In int,
I_OT_Date In date
)
As
Begin
declare
-----
c_shift_in1 nvarchar2(30);
c_shift_in2 nvarchar2(30);
c_shift_out1 nvarchar2(30);
c_shift_out2 nvarchar2(30);
a_shift_out1 nvarchar2(30);
a_shift_out2 nvarchar2(30);
previous_day_counter int;
previous_day_in_time1 date;
previous_day_in_time2 date;
curr_day_c_in_time1 date;
curr_day_c_in_time2 date;
next_day_c_out_time1 date;
next_day_c_out_time2 date;

```

```

curr_day_in_time1 date;
curr_day_in_time2 date;
curr_day_out_time1 date;
curr_day_out_time2 date;
previous_day_out_time date;
special_ot_count int;
previous_day date;
nextday date;
previous_ot_in date;
previous_ot_out date;
previous_reg_in date;
previous_reg_out date;
sp_ot_for_pre_c_shift date;
total_minute int;
total_minute_ot_sft int;
total_minute1 int;
total_minute_ot_sft1 int;
total_minute2 int;
total_minute_ot_sft2 int;
total_minute3 int;
total_minute_ot_sft3 int;
total_minute4 int;
total_minute_ot_sft4 int;
total_minute_reg int;
special_ot_actual nvarchar2(30);
special_ot_actual_ot_sft nvarchar2(30);
total_ot_reg_minute int;
special_ot_reg nvarchar2(30);
emp_shift nvarchar2(30);
comp_card_id nvarchar2(30);
max_p_time date;

begin
select to_char(COMPCARDID)
into comp_card_id
from att_in.TBL_EMP_PERSONAL_INFO
where EMPID = I_EmpId;

begin
select R_SHIFT
into emp_shift
from att_in.TBL_OT_ENTRY
where empid= I_EmpId
and
OT_OR_REG_DATE = to_date(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR');
EXCEPTION
when no_data_found then
null;
end;

if emp_shift like '4 Shift%' then

previous_ot_in := null;

```



```

previous_ot_out := null;
previous_reg_in := null;
previous_reg_out := null;
total_minute := 0;
total_minute_ot_sft:=0;
special_ot_count := 0;

select count(ID) into special_ot_count from att_in.TBL_SPECIAL_OT_DAYS
where STARTDATE <= TO_DATE(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR') and
ENDDATE >= TO_DATE(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR');

```

```

select to_date(to_char(I_OT_Date - interval '1' DAY,'DD/MM/RRRR'),'DD/MM/RRRR')
into previous_day
from dual;

```

```

select to_date(to_char(I_OT_Date + interval '1' DAY,'DD/MM/RRRR'),'DD/MM/RRRR')
into nextday
from dual;

```

```

c_shift_in1 := '20:00:00';
c_shift_in2 := '23:59:00';

```

```

c_shift_out1 := '04:00:00';
c_shift_out2 := '08:00:00';

```

```

a_shift_out1 := '12:00:00';
a_shift_out2 := '16:00:00';

```

*--previous c shift in*

```

previous_day_in_time1 := to_date(to_char(previous_day,'DD/MM/RRRR') || ' ' || c_shift_in1,'DD/MM/RRRR
HH24:MI:SS');
previous_day_in_time2 := to_date(to_char(previous_day,'DD/MM/RRRR') || ' ' || c_shift_in2,'DD/MM/RRRR
HH24:MI:SS');

```

*-- previous c shift out and curr a shift in*

```

curr_day_out_time1 := to_date(to_char(I_OT_Date,'DD/MM/RRRR') || ' ' || c_shift_out1,'DD/MM/RRRR
HH24:MI:SS');
curr_day_out_time2 := to_date(to_char(I_OT_Date,'DD/MM/RRRR') || ' ' || c_shift_out2,'DD/MM/RRRR
HH24:MI:SS');

```

*--curr a shift out and b shift in*

```

curr_day_in_time1 := to_date(to_char(I_OT_Date,'DD/MM/RRRR') || ' ' || a_shift_out1,'DD/MM/RRRR
HH24:MI:SS');
curr_day_in_time2 := to_date(to_char(I_OT_Date,'DD/MM/RRRR') || ' ' || a_shift_out2,'DD/MM/RRRR
HH24:MI:SS');

```

*--curr c shift in and b shift out*

```

curr_day_c_in_time1 := to_date(to_char(I_OT_Date,'DD/MM/RRRR') || ' ' || c_shift_in1,'DD/MM/RRRR

```

```

HH24:MI:SS');
curr_day_c_in_time2 := to_date(to_char(I_OT_Date,'DD/MM/RRRR') || ' ' || c_shift_in2,'DD/MM/RRRR
HH24:MI:SS');

--curr c shift out
next_day_c_out_time1 := to_date(to_char(nextday,'DD/MM/RRRR') || ' ' || c_shift_out1,'DD/MM/RRRR
HH24:MI:SS');
next_day_c_out_time2 := to_date(to_char(nextday,'DD/MM/RRRR') || ' ' || c_shift_out2,'DD/MM/RRRR
HH24:MI:SS');

if special_ot_count > 0 then
previous_ot_in := null;
previous_ot_out := null;
previous_reg_in := null;
previous_reg_out := null;

begin
select
nvl(to_date(to_char(R_IN_TIME_PUNCH,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'),"),
nvl(to_date(to_char(R_OUT_TIME_PUNCH,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'),")
into previous_reg_in,previous_reg_out --previous_ot_in, previous_ot_out,
from att_in.TBL_OT_ENTRY
where empid= I_EmpId
and
R_IN_TIME_PUNCH
between to_date(to_char(previous_day_in_time1,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS')
and to_date(to_char(previous_day_in_time2,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
and R_OUT_TIME_PUNCH
between to_date(to_char(curr_day_out_time1,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
and to_date(to_char(curr_day_out_time2,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS');

EXCEPTION when no_data_found then
null;
end;

if previous_ot_in is not null and previous_ot_out is not null then
Select NVL(SUM(datediff('hh',previous_ot_in,previous_ot_out)),0) * 60
into total_minute_ot_sft
from dual;
if total_minute_ot_sft >= 480 then
total_minute_ot_sft := 480 - 120;
else
total_minute_ot_sft := total_minute_ot_sft - 120;
end if;
end if;

if previous_reg_in is not null and previous_reg_out is not null then
Select NVL(SUM(datediff('hh',previous_reg_in,previous_reg_out)),0) * 60
into total_minute
from dual;

if total_minute >= 480 then
total_minute := 480-120;

```

```

else
total_minute := total_minute - 120;
end if;
end if;
end if;
-----21/08/2013----- AAA

if special_ot_count > 0 then

previous_ot_in := null;
previous_ot_out := null;
previous_reg_in := null;
previous_reg_out := null;

begin
select
nvl(to_date(to_char(R_IN_TIME_PUNCH,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'),"),
nvl(to_date(to_char(R_OUT_TIME_PUNCH,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'),")
into previous_reg_in,previous_reg_out --previous_ot_in, previous_ot_out,
from att_in.TBL_OT_ENTRY
where empid= I_EmpId
and
R_IN_TIME_PUNCH between
to_date(to_char(curr_day_out_time1,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
and to_date(to_char(curr_day_out_time2,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
and R_OUT_TIME_PUNCH between
to_date(to_char(curr_day_in_time1,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
and to_date(to_char(curr_day_in_time2,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS');

EXCEPTION when no_data_found then

begin

select to_date(to_char(min(PunchTime),'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')--
max(punchtime)
into previous_reg_in
from att_in.tbl_raw_data_CMIS
where COMPCARDID = comp_card_id
and punchdate = TO_DATE(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR');

select
nvl(to_date(to_char(R_OUT_TIME_PUNCH,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'),")
into previous_reg_out --previous_reg_in,previous_reg_out
from att_in.TBL_OT_ENTRY
where empid= I_EmpId
and
(R_OUT_TIME_PUNCH
between to_date(to_char(curr_day_in_time1,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
and to_date(to_char(curr_day_in_time2,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'));

EXCEPTION when no_data_found then
null;

```

```

when too_many_rows then
null;
when others then
null;
end;

when too_many_rows then
null;
when others then
null;
end;

begin
select
nvl(to_date(to_char(OT_IN_TIME_PUNCH,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'),"),
nvl(to_date(to_char(OT_OUT_TIME_PUNCH,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'),")
into previous_ot_in, previous_ot_out -- previous_reg_in,previous_reg_out
from att_in.TBL_OT_ENTRY
where empid= I_EmpId
and
OT_IN_TIME_PUNCH
between to_date(to_char(previous_day_in_time1,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS')
and to_date(to_char(previous_day_in_time2,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
and OT_OUT_TIME_PUNCH
between to_date(to_char(curr_day_out_time1,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
and to_date(to_char(curr_day_out_time2,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS');

EXCEPTION when no_data_found then
null;
when too_many_rows then
null;
when others then
null;
end;

begin

select
nvl(to_date(to_char(OT_IN_TIME_PUNCH,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'),"),
nvl(to_date(to_char(OT_OUT_TIME_PUNCH,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'),")
into previous_ot_in, previous_ot_out --previous_reg_in,previous_reg_out
from att_in.TBL_OT_ENTRY
where empid= I_EmpId
and
(OT_IN_TIME_PUNCH between
to_date(to_char(curr_day_out_time1,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS') and
to_date(to_char(curr_day_out_time2,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS') and
OT_OUT_TIME_PUNCH between
to_date(to_char(curr_day_in_time1,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS') and
to_date(to_char(curr_day_in_time2,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'));

EXCEPTION
when no_data_found then

```

```

begin

select to_date(to_char(min(PunchTime),'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')--
max(punchtime)
into previous_ot_in
from att_in.tbl_raw_data_CMIS
where COMPCARDID = comp_card_id
and punchdate = TO_DATE(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR');

select
nvl(to_date(to_char(OT_OUT_TIME_PUNCH,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'),")
into previous_ot_out --previous_reg_in,previous_reg_out
from att_in.TBL_OT_ENTRY
where empid= I_EmpId
and (OT_OUT_TIME_PUNCH
between to_date(to_char(curr_day_in_time1,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
and to_date(to_char(curr_day_in_time2,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'));

EXCEPTION
when no_data_found then
null;
when too_many_rows then
null;
when others then
null;
end;

when too_many_rows then
null;
when others then
null;
end;

if previous_ot_in is not null and previous_ot_out is not null then
Select NVL(SUM(datediff('hh',previous_ot_in,previous_ot_out)),0) * 60
into total_minute_ot_sft1
from dual;
if total_minute_ot_sft1 >= 480 then
total_minute_ot_sft1 := 480;
else
total_minute_ot_sft1 :=total_minute_ot_sft1;
end if;
end if;

if previous_reg_in is not null and previous_reg_out is not null then
Select NVL(SUM(datediff('hh',previous_reg_in,previous_reg_out)),0) * 60
into total_minute1
from dual;
if total_minute1 >= 480 then
total_minute1 := 480;
else
total_minute1 := total_minute1;
end if;
end if;

end if;

```

-----BBB

if special\_ot\_count > 0 then

previous\_ot\_in := null;

previous\_ot\_out := null;

previous\_reg\_in := null;

previous\_reg\_out := null;

begin

select

nvl(to\_date(to\_char(R\_IN\_TIME\_PUNCH,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'),"),

nvl(to\_date(to\_char(R\_OUT\_TIME\_PUNCH,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'),")

into previous\_reg\_in,previous\_reg\_out --previous\_ot\_in, previous\_ot\_out,

from att\_in.TBL\_OT\_ENTRY

where empid= I\_EmpId

and

R\_IN\_TIME\_PUNCH between to\_date(to\_char(curr\_day\_in\_time1,'DD/MM/RRRR

HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS') and

to\_date(to\_char(curr\_day\_in\_time2,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS') and

R\_OUT\_TIME\_PUNCH between to\_date(to\_char(curr\_day\_c\_in\_time1,'DD/MM/RRRR

HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS') and

to\_date(to\_char(curr\_day\_c\_in\_time2,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS');

EXCEPTION

when no\_data\_found then

null;

when too\_many\_rows then

null;

when others then

null;

end;

begin

select

nvl(to\_date(to\_char(OT\_IN\_TIME\_PUNCH,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'),"),

nvl(to\_date(to\_char(OT\_OUT\_TIME\_PUNCH,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'),")

into previous\_ot\_in, previous\_ot\_out --previous\_reg\_in,previous\_reg\_out

from att\_in.TBL\_OT\_ENTRY

where empid= I\_EmpId

and

(OT\_IN\_TIME\_PUNCH between to\_date(to\_char(curr\_day\_in\_time1,'DD/MM/RRRR

HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS') and

to\_date(to\_char(curr\_day\_in\_time2,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS') and

OT\_OUT\_TIME\_PUNCH between to\_date(to\_char(curr\_day\_c\_in\_time1,'DD/MM/RRRR

HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS') and

to\_date(to\_char(curr\_day\_c\_in\_time2,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'));

EXCEPTION when no\_data\_found then

begin

select

```

nvl(to_date(to_char(OT_IN_TIME_PUNCH,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'),")
into previous_ot_in --previous_reg_in,previous_reg_out
from att_in.TBL_OT_ENTRY
where empid= I_EmpId
and
(OT_IN_TIME_PUNCH between to_date(to_char(curr_day_in_time1,'DD/MM/RRRR
HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS') and
to_date(to_char(curr_day_in_time2,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'));

select to_date(to_char(max(PunchTime),'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS') --
max(punchtime)
into previous_ot_out
from att_in.tbl_raw_data_CMIS
where COMPCARDID = comp_card_id
and punchdate = TO_DATE(to_char(I_OT_Date,'DD/MM/RRRR'),'DD/MM/RRRR');
EXCEPTION
when no_data_found then
null;
when too_many_rows then
null;
when others then
null;
end;

when too_many_rows then
null;
when others then
null;
end;

if previous_ot_in is not null and previous_ot_out is not null then
Select NVL(SUM(datediff('hh',previous_ot_in,previous_ot_out)),0) * 60
into total_minute_ot_sft2
from dual;
if total_minute_ot_sft2 >= 480 then
total_minute_ot_sft2 := 480;
else
total_minute_ot_sft2 := total_minute_ot_sft2;
end if;

end if;

if previous_reg_in is not null and previous_reg_out is not null then
Select NVL(SUM(datediff('hh',previous_reg_in,previous_reg_out)),0) * 60
into total_minute2
from dual;
if total_minute2 >= 480 then
total_minute2 := 480;
else
total_minute2 := total_minute2;
end if;
end if;
end if;

-----BBB
-----CCC

```

```

if special_ot_count > 0 then
previous_ot_in := null;
previous_ot_out := null;
previous_reg_in := null;
previous_reg_out := null;

begin
select
nvl(to_date(to_char(R_IN_TIME_PUNCH,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'),"),
nvl(to_date(to_char(R_OUT_TIME_PUNCH,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'),")
into previous_reg_in,previous_reg_out --previous_ot_in, previous_ot_out,
from att_in.TBL_OT_ENTRY
where empid= I_EmpId
and
R_IN_TIME_PUNCH between to_date(to_char(curr_day_c_in_time1,'DD/MM/RRRR
HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS') and
to_date(to_char(curr_day_c_in_time2,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS') and
R_OUT_TIME_PUNCH between to_date(to_char(next_day_c_out_time1,'DD/MM/RRRR
HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS') and
to_date(to_char(next_day_c_out_time2,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS');

EXCEPTION
when no_data_found then
null;
when too_many_rows then
null;
when others then
null;
end;

begin
select
nvl(to_date(to_char(OT_IN_TIME_PUNCH,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'),"),
nvl(to_date(to_char(OT_OUT_TIME_PUNCH,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS'),")
into previous_ot_in, previous_ot_out --previous_reg_in,previous_reg_out
from att_in.TBL_OT_ENTRY
where empid= I_EmpId
and
OT_IN_TIME_PUNCH between to_date(to_char(curr_day_c_in_time1,'DD/MM/RRRR
HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS') and
to_date(to_char(curr_day_c_in_time2,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS') and
OT_OUT_TIME_PUNCH between to_date(to_char(next_day_c_out_time1,'DD/MM/RRRR
HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS') and
to_date(to_char(next_day_c_out_time2,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS');
EXCEPTION when no_data_found then
null;
end;

if previous_ot_in is not null and previous_ot_out is not null then
Select NVL(SUM(datediff('hh',previous_ot_in,previous_ot_out)),0) * 60
into total_minute_ot_sft3
from dual;
if total_minute_ot_sft3 >= 480 then
total_minute_ot_sft3 := 480-360;
else
total_minute_ot_sft3 := total_minute_ot_sft3-360;

```



```

end if;
end if;

if previous_reg_in is not null and previous_reg_out is not null then
Select abs( NVL(SUM(datediff('hh', previous_reg_in,previous_reg_out)),0)) * 60
into total_minute3
from dual;

if total_minute3 >= 480 then
total_minute3 := 480-360;
else
total_minute3 := total_minute3 - 360;
end if;
end if;
end if;

special_ot_actual := att_in.fn_minute_to_time_special_ot( (nvl(total_minute,0)
+ nvl(total_minute1,0)
+ nvl(total_minute2,0)
+ nvl(total_minute3,0)
+ nvl(total_minute_ot_sft,0)
+ nvl(total_minute_ot_sft1,0)
+ nvl(total_minute_ot_sft2,0)
+ nvl(total_minute_ot_sft3,0)));

update att_in.TBL_OT_ENTRY
set SPECIAL_OT_HOUR = to_char(special_ot_actual)
where empid= I_EmpId
and OT_OR_REG_DATE = I_OT_Date;
commit;
-----
end if;
end;
End;

```

## 25.SP\_TWO\_SHIFT\_OT\_TEST\_TANIA

```
CREATE OR REPLACE PROCEDURE ATT_IN.sp_Two_Shift_OT_test_tania
(
I_EmpId In int,
I_OT_Date In date
)
As
Begin
declare
chk_duty_pattern nvarchar2(30);
check_ot_shift_name nvarchar2(30);
check_reg_shift_name nvarchar2(30);
w_off_in_out nvarchar2(30);
emp_code nvarchar2(30);
dept_id int;
counter int;
ot_counter int;
ot_entry_counter int;
leave_counter int;
osd_counter int;
punch_counter int;
lTypeName nvarchar2(30);
unknown_punch1 nvarchar2(30);
unknown_punch2 nvarchar2(30);
chk_prev_duty date;
leave_id int;
leave_name nvarchar2(30);
chk_p_exist int;
ot_s_in_min int;
r_s_in_min int;
-----
comp_cardid nvarchar2(9);
otDate nvarchar2(30);
--ot_in_time date;
ot_in_time number;
ot_punch_out number;
--ot_punch_out date;
test_chk nvarchar2(30);
prev_day_last_punch_time nvarchar2(30);
status nvarchar2(30);
-----
ot_shift nvarchar2(30);
ot_shift_in1 nvarchar2(30);
ot_shift_in2 nvarchar2(30);
ot_shift_out1 nvarchar2(30);
ot_shift_out2 nvarchar2(30);
ot_shift_in_check1 int;
ot_shift_in_check2 int;
ot_shift_out_check1 int;
ot_shift_out_check2 int;
p_time nvarchar2(30);
p_time_h int;--17_06_2013
p_time_m int;--17_06_2013
```

```

p_time_count int;--17_06_2013
p_out_count int; --19_06_2013
p_out_time nvarchar2(30);
ot_shift_c_out_date nvarchar2(30);
ot_shift_late_by int;--17_06_2013
ot_shift_cilling_time int;--17_06_2013
ot_p_time_hour int;--17_06_2013
ot_late_hour int;--17_06_2013
ot_late_min int;--17_06_2013
ot_hour int; --19_06_2013
ot_min int; --19_06_2013
p_out_time_h int; --19_06_2013
p_out_time_m int; --19_06_2013
ot_p_out_time_hour int; --19_06_2013
total_ot_hour nvarchar2(30); --19_06_2013
ot_count int; --19_06_2013
double_ot_hour int;
double_ot_min int;
special_ot_count int;
difrence_In_Hours number;
difrence_In_minutes number;
difrence_In_seconds number;
ot_shift_name nvarchar2(30);
-----
r_shift nvarchar2(30);
r_shift_in1 nvarchar2(30);
r_shift_in2 nvarchar2(30);
r_shift_out1 nvarchar2(30);
r_shift_out2 nvarchar2(30);
r_shift_in_check1 int;
r_shift_in_check2 int;
r_shift_out_check1 int;
r_shift_out_check2 int;
--otDate varchar2(30);
r_punch_in_time nvarchar2(30);
r_punch_out_time nvarchar2(30);
r_shift_c_out_date nvarchar2(30);
r_shift_late_by int;--17_06_2013
r_shift_cilling_time int;--17_06_2013
r_time_h int;--18_06_2013
r_time_m int;--18_06_2013
r_time_count int;--18_06_2013
r_out_count int; --19_06_2013
r_p_time_hour int;--18_06_2013
r_late_hour int;--18_06_2013
r_late_min int;--18_06_2013
reg_shift_name nvarchar2(30);
reg_punch_count int;
current_d_name nvarchar2(15);
weekly_holiday_name nvarchar2(15);
-----
v_minpunchtime varchar2(30);
v_minpunchtimehh number;
v_minpunchtimemi number;
v_maxpunchtimehh number;

```

```

v_maxpunchtimemi number;
v_max_total_time number;
v_r_shift_in number;
v_r_shift_in_min number;
v_r_shift_out_min number;
v_r_shift_out number;
v_r_shift varchar2(30);
v_compcardid varchar2(30);
v_ot_time varchar2(30);
v_otdate varchar2(30);
v_maxpunchtime varchar2(30);
v_otapprove int;
exists_govt_holiday int;

n_shift_t int;

diff_in_shift_t int;
v_maxpunchtime_t date;
v_minpunchtime_t date;
diff_in_Ashift_t int;
v_r_shift_in_t date;
v_r_shift_out_t date;
v_employee_type varchar2(30);
v_employee_category varchar2(30);
V_PUNCHTIME int;
V_REG_STATUS varchar2(30);
diff_in_Ashift_tt int;
total_ot_w number;
V_LOGINTIME date;

-----
begin

SELECT count(*) into counter
FROM att_in.tbl_ot_entry
WHERE EMPID = I_EmpId
and OT_OR_REG_DATE = to_date(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');

if counter = 0 then
select EMPCODE,to_char(COMPCARDID),DEPTID
into emp_code,comp_cardid,dept_id
from att_in.TBL_EMP_PERSONAL_INFO
where EMPID = I_EmpId;

insert into att_in.tbl_ot_entry(EMPCODE,EMPID,OT_OR_REG_DATE,COMPCARDID,DEPTID)
values(emp_code,I_EmpId,I_OT_Date,to_char(comp_cardid),dept_id);
commit;

else
select EMPCODE,to_char(COMPCARDID),DEPTID
into emp_code,comp_cardid,dept_id
from att_in.TBL_EMP_PERSONAL_INFO
where EMPID = I_EmpId;

```

```

end if;

SELECT count(*) into ot_counter
FROM att_in.tbl_ot_entry
WHERE EMPID = I_EmpId
and OT_OR_REG_DATE = to_date(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');

if ot_counter >0 then

select nvl(count(DUTY_PATERN),0) into ot_entry_counter
FROM att_in.tbl_ot_entry
WHERE EMPID = I_EmpId
and OT_OR_REG_DATE = to_date(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');

if ot_entry_counter = 0 then

select count(EMPID) into leave_counter
from att_in.TBL_LEAVE_APPLY
where EMPID = I_EmpId
and FROMDATE = TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');

if leave_counter > 0 then
select LEAVETYPEID into leave_id
from att_in.TBL_LEAVE_APPLY
where EMPID = I_EmpId
and FROMDATE = TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY') ;

select REPORTINGNAME into ITypeName
from att_in.TBL_LEAVE_TYPE
where LEAVETYPEID = leave_id;

update att_in.tbl_ot_entry
set REG_STATUS = upper(ITypeName)
where EMPID = I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');
commit;

else

select count(EMPID) into osd_counter
from att_in.TBL_OSD_SETUP
where EMPID = I_EmpId
and OSDSTARTDATE = TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY') ;

if osd_counter > 0 then

update att_in.tbl_ot_entry
set REG_STATUS = upper('OCS')
where EMPID = I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');
commit;

```

```

elsif osd_counter = 0 then

select count(NVL(GOVTHOLIID,'0'))
into exists_govt_holiday
from att_in.tbl_Govt_Holiday_List
where GOVTHOLISTARTDATE <= TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY')
and GOVTHOLIENDDATE >= TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');

if exists_govt_holiday > 0 then
update att_in.tbl_ot_entry
set REG_STATUS = upper('GHD')
where EMPID = I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');

commit;

else
select WEEKLY_HOLIDAY into weekly_holiday_name
from att_in.tbl_ot_entry
where EMPID = I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');

select to_char(I_OT_Date,'DAY') into current_d_name from dual;

if upper(substr(weekly_holiday_name,1,3)) = upper(substr(current_d_name,1,3)) then

update att_in.tbl_ot_entry
set DUTY_PATERN = upper('WEEKLY OFF'),
REG_STATUS = upper('W')
where EMPID = I_EmpId
and OT_OR_REG_DATE = TO_DATE(I_OT_Date,'DD/MM/YYYY');
commit;

else
update att_in.tbl_ot_entry
set REG_STATUS = upper('A')
where EMPID = I_EmpId and OT_OR_REG_DATE =
TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');
commit;
end if;
end if;
end if;
end if;

else

-----
select count(EMPID) into leave_counter
from att_in.TBL_LEAVE_APPLY
where EMPID = I_EmpId

```

```

and FROMDATE= TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY') ;

if leave_counter > 0 then
select LEAVETYPEID into leave_id
from att_in.TBL_LEAVE_APPLY
where EMPID = I_EmpId
and FROMDATE = TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY') ;

select REPORTINGNAME into ITypeName
from att_in.TBL_LEAVE_TYPE
where LEAVETYPEID = leave_id;

update att_in.tbl_ot_entry
set REG_STATUS = upper(ITypeName)
where EMPID = I_EmpId and OT_OR_REG_DATE =
TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');
commit;
else

select count(EMPID) into osd_counter
from att_in.TBL_OSD_SETUP
where EMPID = I_EmpId
and OSDSTARTDATE = TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY') ;

if osd_counter > 0 then

update att_in.tbl_ot_entry
set REG_STATUS = upper('OCS')
where EMPID = I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');
commit;

elsif osd_counter = 0 then

select count(NVL(GOVTHOLIID,'0'))
into exists_govt_holiday
from att_in.tbl_Govt_Holiday_List
where GOVTHOLISTARTDATE <= TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY')
and GOVTHOLIENDDATE>= TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');

if exists_govt_holiday>0 then
update att_in.tbl_ot_entry
set REG_STATUS = upper('GHD')
where EMPID = I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');

commit;

else

```

```

select upper(WEEKLY_HOLIDAY) into weekly_holiday_name
from att_in.tbl_ot_entry
where EMPID = I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');

select upper(to_char(I_OT_Date,'DAY')) into current_d_name from dual;

if upper(substr(weekly_holiday_name,1,3)) = upper(substr(current_d_name,1,3)) then

update att_in.tbl_ot_entry
set REG_STATUS = 'W'
where EMPID = I_EmpId
and OT_OR_REG_DATE = TO_DATE(otDate,'DD/MM/YYYY');
commit;

end if;
end if;

end if;

end if;
-----
select count(punchtime)
into chk_p_exist
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = comp_cardid
and PUNCHDATE = TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');

select DUTY_PATERN into chk_duty_pattern
from att_in.tbl_ot_entry
where empid=I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');

if chk_p_exist =0 then

if chk_duty_pattern ='REG' then

select count(EMPID) into leave_counter
from att_in.TBL_LEAVE_APPLY
where EMPID = I_EmpId
and FROMDATE = TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY') ;

if leave_counter > 0 then
select LEAVETYPEID into leave_id
from att_in.TBL_LEAVE_APPLY
where EMPID = I_EmpId
and FROMDATE = TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');

select REPORTINGNAME into ITypeName
from att_in.TBL_LEAVE_TYPE
where LEAVETYPEID = leave_id;

```



```

update att_in.tbl_ot_entry
set REG_STATUS = upper(ITypeName)
where EMPID = I_EmpId and OT_OR_REG_DATE =
TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');
commit;
else
select count(EMPID) into osd_counter
from att_in.TBL_OSD_SETUP
where EMPID = I_EmpId
and OSDSTARTDATE = TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');

if osd_counter > 0 then
update att_in.tbl_ot_entry
set REG_STATUS = upper('OCS')
where EMPID = I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');
commit;
elsif osd_counter = 0 then

select count(NVL(GOVTHOLIID,'0'))
into exists_govt_holiday
from att_in.tbl_Govt_Holiday_List
where GOVTHOLISTARTDATE <= TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY')
and GOVTHOLIENDDATE >= TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');

if exists_govt_holiday > 0 then
update att_in.tbl_ot_entry
set REG_STATUS = upper('GHD')
where EMPID = I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');

commit;

else

select WEEKLY_HOLIDAY
into weekly_holiday_name
from att_in.tbl_ot_entry
where EMPID = I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');

select to_char(I_OT_Date,'DAY') into current_d_name from dual;

if upper(substr(weekly_holiday_name,1,3)) = upper(substr(current_d_name,1,3)) then

update att_in.tbl_ot_entry
set REG_STATUS = upper('W')
where EMPID = I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');
commit;
else
update att_in.tbl_ot_entry

```

```

set REG_STATUS = upper('A'),
OT_IN_TIME_PUNCH=null,
OT_OUT_TIME_PUNCH=null,
R_IN_TIME_PUNCH=null,
R_OUT_TIME_PUNCH=null
where EMPID = I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');
commit;

end if;

end if;

end if;
end if;

end if;

elsif chk_p_exist >0 and chk_p_exist<2 then

select nvl(to_char(min(punchtime),'HH24:MI:SS'),'0')
into v_minpunchtime
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY') ;

update att_in.TBL_OT_ENTRY
set REG_STATUS = upper('A'),
R_IN_TIME_PUNCH=to_date(I_OT_Date||' '||v_minpunchtime,'DD/MM/RRRR HH24:MI:SS'),
R_OUT_TIME_PUNCH="
where empid= I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');
commit;

elsif chk_p_exist >1 then
select DUTY_PATERN into chk_duty_pattern
from att_in.TBL_OT_ENTRY
where empid=I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');

if chk_duty_pattern = 'REG' then

select R_SHIFT,to_char(COMPCARDID),to_char(OT_OR_REG_DATE,'DD/MM/YYYY'),
to_number(to_char(R_SHIFT_IN,'HH24')) , to_number(to_char(R_SHIFT_OUT,'HH24'))
into check_reg_shift_name,comp_cardid,otDate, r_shift_in_check1,r_shift_out_check1
from att_in.TBL_OT_ENTRY
where empid=I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY')
and DUTY_PATERN = 'REG';

select nvl(OT_APPROVE,0) into v_otapprove

```

```

from att_in.TBL_OT_ENTRY
where empid=I_EmpId
and OT_OR_REG_DATE = to_date(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');

if check_reg_shift_name = '2 Shift-A' then

begin

select to_date(to_char(min(PUNCHTIME),'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
into v_minpunchtime_t
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS') ;

select to_date(to_char(max(PUNCHTIME),'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
into v_maxpunchtime_t
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS') ;

select to_date(to_char(R_SHIFT_out,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
into v_r_shift_out_t
from att_in.TBL_OT_ENTRY
where empid=I_EmpId
and OT_OR_REG_DATE = to_date(to_char(I_OT_Date,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS');

select count(punchtime)into V_PUNCHTIME from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY') ;

select nvl(to_char(min(punchtime),'HH24:MI:SS'),'0')
into v_minpunchtime
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY') ;

select nvl(to_char(max(punchtime),'HH24:MI:SS'),'0')
into v_maxpunchtime
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY') ;

select nvl(to_number(to_char(min(punchtime),'HH24'),'0'),
nvl(to_number(to_char(min(punchtime),'MI'),'0')
into v_minpunchtimehh,v_minpunchtimemi
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY') ;

```

```

select nvl(to_number(to_char(max(punchtime),'HH24')),0) ,
nvl(to_number(to_char(max(punchtime),'MI')),0)
into v_maxpunchtimehh,v_maxpunchtimemi
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY') ;

```

```

select R_SHIFT,to_char(COMPCARDID),to_char(OT_OR_REG_DATE,'DD/MM/YYYY'),
to_number(to_char(R_SHIFT_IN,'HH24')) , to_number(to_char(R_SHIFT_OUT,'HH24'))
into v_r_shift,v_compcardid,v_otdate,v_r_shift_in,v_r_shift_out
from att_in.TBL_OT_ENTRY
where empid=I_EmpId
and OT_OR_REG_DATE = to_date(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');

```

```

select REG_STATUS into V_REG_STATUS
from att_in.TBL_OT_ENTRY
where empid=I_EmpId
and OT_OR_REG_DATE = to_date(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');

```

```

Select NVL(SUM(datediff('hh',v_minpunchtime_t,v_r_shift_out_t)),0) * 60 --for normal day
into diff_in_Ashift_t
from dual;

```

```

Select NVL(SUM(datediff('hh',v_minpunchtime_t,v_maxpunchtime_t)),0) * 60 --for weekly off day
into diff_in_Ashift_tt
from dual;

```

```

if V_REG_STATUS='W' and V_PUNCHTIME>=2 then

```

```

if diff_in_Ashift_tt >360 and diff_in_Ashift_tt <510 and diff_in_Ashift_t<360 then

```

```

update att_in.TBL_OT_ENTRY
set
REG_STATUS = 'P',
R_IN_TIME_PUNCH=to_date(I_OT_Date||' '||v_minpunchtime,'DD/MM/RRRR HH24:MI:SS'),
R_OUT_TIME_PUNCH=to_date(I_OT_Date||' '|| v_maxpunchtime,'DD/MM/RRRR HH24:MI:SS')
where empid= I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');
commit;

```

```

elsif
diff_in_Ashift_tt >510 and diff_in_Ashift_t<360 then

```

```

update att_in.TBL_OT_ENTRY
set
REG_STATUS = 'P',
R_SHIFT_IN=to_date(I_OT_Date||' '||'14:00:00','DD/MM/RRRR HH24:MI:SS'),
R_SHIFT_OUT=to_date(I_OT_Date||' '|| '20:00:00','DD/MM/RRRR HH24:MI:SS'),
R_IN_TIME_PUNCH=to_date(I_OT_Date||' '||v_minpunchtime,'DD/MM/RRRR HH24:MI:SS'),

```

```

R_OUT_TIME_PUNCH=to_date(I_OT_Date||' '|| v_maxpunchtime,'DD/MM/RRRR HH24:MI:SS')
where empid= I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');
commit;

if v_otapprove=1 then
select nvl(to_number(to_char(min(punchtime),'HH24')),'0'),
nvl(to_number(to_char(min(punchtime),'MI')),'0')
into v_minpunchtimehh,v_minpunchtimemi
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY') ;

select nvl(to_number(to_char(min(R_SHIFT_IN),'HH24')),'0'),
nvl(to_number(to_char(min(R_SHIFT_IN),'MI')),'0')
into v_r_shift_in,v_r_shift_in_min
from att_in.TBL_OT_ENTRY
where empid= I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');
v_r_shift_in:=v_r_shift_in*60;
v_minpunchtimehh:=(v_minpunchtimehh)*60+v_minpunchtimemi;

if v_minpunchtimehh<v_r_shift_in then
v_max_total_time:=(v_r_shift_in -v_minpunchtimehh);
end if;

if v_max_total_time>480 then
v_max_total_time:=480;
else
v_max_total_time:=v_max_total_time;
end if;
ot_min := v_max_total_time mod 60;
ot_hour := floor((v_max_total_time) / 60);
if v_max_total_time >29 then
if ot_min<=9 and ot_hour<=9 then
v_ot_time:= to_char('0'||ot_hour)||':'||to_char('0'|(ot_min))||':00';
else
v_ot_time:= to_char('0'||ot_hour)||':'||to_char(ot_min)||':00';
end if;
else
v_ot_time:= '00:00:00';
end if;

else
v_ot_time:= '00:00:00';
end if;

update att_in.TBL_OT_ENTRY
set OT_HOUR = v_ot_time,
R_IN_TIME_PUNCH=to_date(I_OT_Date||' '||v_minpunchtime,'DD/MM/RRRR HH24:MI:SS'),
R_OUT_TIME_PUNCH=to_date(I_OT_Date||' '|| v_maxpunchtime,'DD/MM/RRRR HH24:MI:SS')
where empid= I_EmpId

```

```

and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');
commit;

else
update att_in.TBL_OT_ENTRY
set OT_HOUR = "",
REG_STATUS = 'A'
where empid= I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');
commit;
end if;

elsif diff_in_Ashift_t <360 or diff_in_Ashift_tt <360 then

update att_in.TBL_OT_ENTRY
set OT_HOUR = "",
REG_STATUS = 'A', LATE_ON_REG_SHIFT ='00:00:00',
R_IN_TIME_PUNCH=to_date(I_OT_Date||' '||v_minpunchtime,'DD/MM/RRRR HH24:MI:SS'),
R_OUT_TIME_PUNCH=to_date(I_OT_Date||' '|| v_maxpunchtime,'DD/MM/RRRR HH24:MI:SS')
where empid= I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');
commit;

else
r_shift_cilling_time:=(v_r_shift_in*60)+30;
r_p_time_hour:=(v_minpunchtimehh*60)+v_minpunchtimemi;

if r_shift_cilling_time >= r_p_time_hour then
update att_in.TBL_OT_ENTRY
set REG_STATUS = 'P', LATE_ON_REG_SHIFT ='00:00:00'
where empid= I_EmpId
and OT_OR_REG_DATE = to_date(otDate,'DD/MM/YYYY');
commit;
elsif r_p_time_hour > r_shift_cilling_time then
r_shift_late_by := r_p_time_hour - r_shift_cilling_time;

if r_shift_late_by >60 then
r_late_hour := (r_shift_late_by / 60);
r_late_min := (r_shift_late_by mod 60);
if r_late_min<=9 then
update att_in.TBL_OT_ENTRY
set REG_STATUS = 'L', LATE_ON_REG_SHIFT = to_char('0'||to_char(r_late_hour)||':'||'0'||
r_late_min||':00')
where empid= I_EmpId
and OT_OR_REG_DATE = to_date(otDate,'DD/MM/YYYY');
commit;
else
update att_in.TBL_OT_ENTRY
set REG_STATUS = 'L', LATE_ON_REG_SHIFT = to_char('0'||to_char(r_late_hour)||':'|| r_late_min||':00')
where empid= I_EmpId
and OT_OR_REG_DATE = to_date(otDate,'DD/MM/YYYY');
commit;
end if;
elsif r_shift_late_by>9 and r_shift_late_by < 60 then

```

```

update att_in.TBL_OT_ENTRY
set REG_STATUS = 'L', LATE_ON_REG_SHIFT = to_char('00:'||r_shift_late_by||':00')
where empid= I_EmpId
and OT_OR_REG_DATE = to_date(otDate,'DD/MM/YYYY');
commit;
elsif r_shift_late_by<=9 then
update att_in.TBL_OT_ENTRY
set REG_STATUS = 'L', LATE_ON_REG_SHIFT = to_char('00:'||'0'||r_shift_late_by||':00')
where empid= I_EmpId
and OT_OR_REG_DATE = to_date(otDate,'DD/MM/YYYY');
commit;
end if;
end if;

IF v_otapprove=1 then
v_r_shift_out:=v_r_shift_out*60;
v_maxpunchtimehh:=(v_maxpunchtimehh)*60+v_maxpunchtimemi;

if v_maxpunchtimehh>v_r_shift_out then
v_max_total_time:=(v_maxpunchtimehh -v_r_shift_out);
end if;

if v_max_total_time>480 then
v_max_total_time:=480;
else
v_max_total_time:=v_max_total_time;
end if;
ot_min := v_max_total_time mod 60;
ot_hour := floor((v_max_total_time) / 60);
if v_max_total_time >29 then
if ot_min<=9 and ot_hour<=9 then
v_ot_time:= to_char('0'||ot_hour)||':'||to_char('0'|(ot_min))||':00';
else
v_ot_time:= to_char('0'||ot_hour)||':'||to_char(ot_min)||':00';
end if;
else
v_ot_time:= '00:00:00';
end if;
else
v_ot_time:= '00:00:00';
end if;

update att_in.TBL_OT_ENTRY
set OT_HOUR = v_ot_time,
R_IN_TIME_PUNCH=to_date(I_OT_Date||' '||v_minpunchtime,'DD/MM/RRRR HH24:MI:SS'),
R_OUT_TIME_PUNCH=to_date(I_OT_Date||' '|| v_maxpunchtime,'DD/MM/RRRR HH24:MI:SS')
where empid= I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');
commit;

end if;

```

```

end;

elsif check_reg_shift_name = '2 Shift-B' then

begin

v_minpunchtime_t:="";
v_r_shift_out_t:="";
V_PUNCHTIME:="";

select to_date(to_char(max(PUNCHTIME),'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
into v_maxpunchtime_t from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS') ;

select to_date(to_char(min(PUNCHTIME),'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
into v_minpunchtime_t from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS') ;

select to_date(to_char(R_SHIFT_in,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
into v_r_shift_in_t from att_in.TBL_OT_ENTRY
where empid=I_EmpId
and OT_OR_REG_DATE = to_date(to_char(I_OT_Date,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS');

select nvl(to_char(min(punchtime),'HH24:MI:SS'),'0')
into v_minpunchtime from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY') ;

select nvl(to_char(max(punchtime),'HH24:MI:SS'),'0')
into v_maxpunchtime from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY') ;

select nvl(to_number(to_char(min(punchtime),'HH24')),0),
nvl(to_number(to_char(min(punchtime),'MI')),0)
into v_minpunchtimehh,v_minpunchtimemi from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY') ;

select nvl(to_number(to_char(max(punchtime),'HH24')),0) ,
nvl(to_number(to_char(max(punchtime),'MI')),0)
into v_maxpunchtimehh,v_maxpunchtimemi from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY') ;

```



```

select R_SHIFT,to_char(COMPCARDID),to_char(OT_OR_REG_DATE,'DD/MM/YYYY'),
to_number(to_char(R_SHIFT_IN,'HH24')) , to_number(to_char(R_SHIFT_OUT,'HH24'))
into v_r_shift,v_compcardid,v_otdate,v_r_shift_in,v_r_shift_out
from att_in.TBL_OT_ENTRY
where empid=I_EmpId
and OT_OR_REG_DATE = to_date(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');

select REG_STATUS into V_REG_STATUS from att_in.TBL_OT_ENTRY
where empid=I_EmpId
and OT_OR_REG_DATE = to_date(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');

select count(punchtime)into V_PUNCHTIME from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY') ;

Select NVL(SUM(datediff('hh',v_r_shift_in_t,v_maxpunchtime_t)),0) * 60
into diff_in_Ashift_t
from dual;

Select NVL(SUM(datediff('hh',v_minpunchtime_t,v_maxpunchtime_t)),0) * 60
into diff_in_Ashift_tt
from dual;

if V_REG_STATUS='W' and V_PUNCHTIME>=2 then

if diff_in_Ashift_tt >360 and diff_in_Ashift_tt<510 and diff_in_Ashift_t <360 then

update att_in.TBL_OT_ENTRY
set
REG_STATUS = 'P',
R_IN_TIME_PUNCH=to_date(I_OT_Date||' '||v_minpunchtime,'DD/MM/RRRR HH24:MI:SS'),
R_OUT_TIME_PUNCH=to_date(I_OT_Date||' '|| v_maxpunchtime,'DD/MM/RRRR HH24:MI:SS')
where empid= I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');
commit;
elsif
diff_in_Ashift_tt>510 and diff_in_Ashift_t <360 then
update att_in.TBL_OT_ENTRY
set
REG_STATUS = 'P',
R_SHIFT_IN=to_date(I_OT_Date||' '||'06:00:00','DD/MM/RRRR HH24:MI:SS'),
R_SHIFT_OUT=to_date(I_OT_Date||' '|| '14:00:00','DD/MM/RRRR HH24:MI:SS'),
R_IN_TIME_PUNCH=to_date(I_OT_Date||' '||v_minpunchtime,'DD/MM/RRRR HH24:MI:SS'),
R_OUT_TIME_PUNCH=to_date(I_OT_Date||' '|| v_maxpunchtime,'DD/MM/RRRR HH24:MI:SS')
where empid= I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');
commit;

if v_otapprove=1 then
select nvl(to_number(to_char(max(punchtime),'HH24')),0) ,
nvl(to_number(to_char(max(punchtime),'MI')),0)
into v_maxpunchtimehh,v_maxpunchtimemi
from att_in.TBL_RAW_DATA_CMIS

```

```

where COMPCARDID = comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY') ;

select nvl(to_number(to_char(R_SHIFT_OUT),'HH24'),'0'),
nvl(to_number(to_char(R_SHIFT_OUT),'MI'),'0')
into v_r_shift_out,v_r_shift_out_min
from att_in.TBL_OT_ENTRY
where empid=I_EmpId
and OT_OR_REG_DATE = to_date(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');

v_r_shift_out:=v_r_shift_out*60+v_r_shift_out_min;
v_maxpunchtimehh:=v_maxpunchtimehh*60+v_maxpunchtimemi;

if v_maxpunchtimehh>v_r_shift_out then
v_max_total_time:=(v_maxpunchtimehh -v_r_shift_out);
end if;

if v_max_total_time>480 then
v_max_total_time:=480;
else
v_max_total_time:=v_max_total_time;
end if;
ot_min := v_max_total_time mod 60;
ot_hour := floor((v_max_total_time) / 60);
if v_max_total_time >29 then
if ot_min<=9 and ot_hour<=9 then
v_ot_time:= to_char('0'||ot_hour)||':'||to_char('0'||(ot_min))||':00';
else
v_ot_time:= to_char('0'||ot_hour)||':'||to_char(ot_min)||':00';
end if;
else
v_ot_time:= '00:00:00';
end if;
else
v_ot_time:= '00:00:00';
end if;

update att_in.TBL_OT_ENTRY
set OT_HOUR = v_ot_time,
R_IN_TIME_PUNCH=to_date(I_OT_Date||' '||v_minpunchtime,'DD/MM/RRRR HH24:MI:SS'),
R_OUT_TIME_PUNCH=to_date(I_OT_Date||' '|| v_maxpunchtime,'DD/MM/RRRR HH24:MI:SS')
where empid= I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');
commit;
else
update att_in.TBL_OT_ENTRY
set OT_HOUR = "",
REG_STATUS = 'A'
where empid= I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');
commit;
end if;

elsif diff_in_Ashift_t <360 or diff_in_Ashift_tt <360 then

```

```

update att_in.TBL_OT_ENTRY
set OT_HOUR = ",
REG_STATUS = 'A', LATE_ON_REG_SHIFT = '00:00:00',
R_IN_TIME_PUNCH=to_date(I_OT_Date||' '||v_minpunchtime,'DD/MM/RRRR HH24:MI:SS'),
R_OUT_TIME_PUNCH=to_date(I_OT_Date||' '|| v_maxpunchtime,'DD/MM/RRRR HH24:MI:SS')
where empid= I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');
commit;
else

r_shift_cilling_time:=(v_r_shift_in*60)+30;
r_p_time_hour:=(v_minpunchtimehh*60)+v_minpunchtimemi;
if r_shift_cilling_time >= r_p_time_hour then
update att_in.TBL_OT_ENTRY
set REG_STATUS = 'P', LATE_ON_REG_SHIFT = '00:00:00'
where empid= I_EmpId and OT_OR_REG_DATE = to_date(otDate,'DD/MM/YYYY');
commit;
elsif r_p_time_hour > r_shift_cilling_time then
r_shift_late_by := r_p_time_hour - r_shift_cilling_time;

if r_shift_late_by >60 then
r_late_hour := (r_shift_late_by / 60);
r_late_min := (r_shift_late_by mod 60);
if r_late_min<=9 then
update att_in.TBL_OT_ENTRY
set REG_STATUS = 'L', LATE_ON_REG_SHIFT = to_char('0'||to_char(r_late_hour)||':'||'0'||
r_late_min||':00')
where empid= I_EmpId and OT_OR_REG_DATE = to_date(otDate,'DD/MM/YYYY');
commit;
else
update att_in.TBL_OT_ENTRY
set REG_STATUS = 'L', LATE_ON_REG_SHIFT = to_char('0'||to_char(r_late_hour)||':'|| r_late_min||':00')
where empid= I_EmpId
and OT_OR_REG_DATE = to_date(otDate,'DD/MM/YYYY');
commit;
end if;
elsif r_shift_late_by>9 and r_shift_late_by < 60 then

update att_in.TBL_OT_ENTRY
set REG_STATUS = 'L', LATE_ON_REG_SHIFT = to_char('00:'||r_shift_late_by||':00')
where empid= I_EmpId
and OT_OR_REG_DATE = to_date(otDate,'DD/MM/YYYY');
commit;
elsif r_shift_late_by<=9 then
update att_in.TBL_OT_ENTRY
set REG_STATUS = 'L', LATE_ON_REG_SHIFT = to_char('00:'||'0'||r_shift_late_by||':00')
where empid= I_EmpId
and OT_OR_REG_DATE = to_date(otDate,'DD/MM/YYYY');
commit;
end if;

end if;

if v_otapprove=1 then

```

```

v_r_shift_in:=v_r_shift_in*60;
v_minpunchtimehh:=(v_minpunchtimehh)*60+v_minpunchtimemi;

if v_minpunchtimehh<v_r_shift_in then
v_max_total_time:=(v_r_shift_in -v_minpunchtimehh);
end if;

if v_max_total_time>480 then
v_max_total_time:=480;
else
v_max_total_time:=v_max_total_time;
end if;
ot_min := v_max_total_time mod 60;
ot_hour := floor((v_max_total_time) / 60);
if v_max_total_time >29 then
if ot_min<=9 and ot_hour<=9 then
v_ot_time:= to_char('0'||ot_hour)||':'||to_char('0'||(ot_min))||':00';
else
v_ot_time:= to_char('0'||ot_hour)||':'||to_char(ot_min)||':00';
end if;
else
v_ot_time:= '00:00:00';
end if;
else
v_ot_time:= '00:00:00';
end if;

update att_in.TBL_OT_ENTRY
set OT_HOUR = v_ot_time,
R_IN_TIME_PUNCH=to_date(I_OT_Date||' '||v_minpunchtime,'DD/MM/RRRR HH24:MI:SS'),
R_OUT_TIME_PUNCH=to_date(I_OT_Date||' '|| v_maxpunchtime,'DD/MM/RRRR HH24:MI:SS')
where empid= I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');
commit;

end if;

end;

elsif check_reg_shift_name = '2 Shift-C' then
begin
v_minpunchtime_t:="";
v_r_shift_out_t:="";

select to_date(to_char(min(PUNCHTIME),'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
into v_minpunchtime_t
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID =comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS') ;

```

```

select to_date(to_char(max(PUNCHTIME),'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
into v_maxpunchtime_t
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS') ;

```

```

select to_date(to_char(R_SHIFT_OUT,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
into v_r_shift_out_t
from att_in.TBL_OT_ENTRY
where empid=I_EmpId
and OT_OR_REG_DATE = to_date(to_char(I_OT_Date,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS');

```

```

select to_date(to_char(R_SHIFT_in,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
into v_r_shift_in_t
from att_in.TBL_OT_ENTRY
where empid=I_EmpId
and OT_OR_REG_DATE = to_date(to_char(I_OT_Date,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS');

```

```

select REG_STATUS
into V_REG_STATUS
from att_in.TBL_OT_ENTRY
where empid=I_EmpId
and OT_OR_REG_DATE = to_date(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');

```

```

select count(punchtime)
into V_PUNCHTIME
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY') ;

```

```

select nvl(to_char(min(punchtime),'HH24:MI:SS'),'0')
into v_minpunchtime
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY') ;

```

```

select nvl(to_char(max(punchtime),'HH24:MI:SS'),'0')
into v_maxpunchtime
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY') ;

```

```

select nvl(to_number(to_char(min(punchtime),'HH24'),'0'),
nvl(to_number(to_char(min(punchtime),'MI'),'0')
into v_minpunchtimehh,v_minpunchtimemi
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY') ;

```

```

select nvl(to_number(to_char(max(punchtime),'HH24'),'0') ,

```

```

nvl(to_number(to_char(max(punchtime),'MI')),0')
into v_maxpunchtimehh,v_maxpunchtimemi
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY') ;

select R_SHIFT,to_char(COMPCARDID),to_char(OT_OR_REG_DATE,'DD/MM/YYYY'),
to_number(to_char(R_SHIFT_IN,'HH24')) , to_number(to_char(R_SHIFT_OUT,'HH24'))
into v_r_shift,v_compcardid,v_otdate,v_r_shift_in,v_r_shift_out
from att_in.TBL_OT_ENTRY
where empid=I_EmpId
and OT_OR_REG_DATE = to_date(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');

diff_in_Ashift_t:="";

Select NVL(SUM(datediff('hh',v_minpunchtime_t,v_r_shift_out_t)),0) * 60 --for normal duty
into diff_in_Ashift_t
from dual;

Select NVL(SUM(datediff('hh',v_minpunchtime_t,v_maxpunchtime_t)),0) * 60 ---for weekly off day
duty
into diff_in_Ashift_tt
from dual;

if V_REG_STATUS='W' and V_PUNCHTIME>=2 then
if diff_in_Ashift_tt >360 and diff_in_Ashift_tt <450 and diff_in_Ashift_t<360 then

update att_in.TBL_OT_ENTRY
set R_SHIFT='2 Shift-D',
R_SHIFT_VIEW='2 Shift-D',
REG_STATUS = 'P',
R_IN_TIME_PUNCH=to_date(I_OT_Date||' '||v_minpunchtime,'DD/MM/RRRR HH24:MI:SS'),
R_OUT_TIME_PUNCH=to_date(I_OT_Date||' '|| v_maxpunchtime,'DD/MM/RRRR HH24:MI:SS')
where empid= I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');
commit;

elsif diff_in_Ashift_tt >=450 and diff_in_Ashift_t<360 then

update att_in.TBL_OT_ENTRY
set R_SHIFT='2 Shift-D',
R_SHIFT_VIEW='2 Shift-D',
REG_STATUS = 'P',
R_SHIFT_IN=to_date(I_OT_Date||' '||'13:30:00','DD/MM/RRRR HH24:MI:SS'),
R_SHIFT_OUT=to_date(I_OT_Date||' '|| '20:30:00','DD/MM/RRRR HH24:MI:SS'),
R_IN_TIME_PUNCH=to_date(I_OT_Date||' '||v_minpunchtime,'DD/MM/RRRR HH24:MI:SS'),
R_OUT_TIME_PUNCH=to_date(I_OT_Date||' '|| v_maxpunchtime,'DD/MM/RRRR HH24:MI:SS')
where empid= I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');

```

```

commit;
if v_otapprove=1 then
select nvl(to_number(to_char(min(punchtime),'HH24')),0'),
nvl(to_number(to_char(min(punchtime),'MI')),0')
into v_minpunchtimehh,v_minpunchtimemi
from att_in.TBL_RAW_DATA_CMIS
where  COMPCARDID = comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY') ;

select nvl(to_number(to_char(min(R_SHIFT_IN),'HH24')),0'),
nvl(to_number(to_char(min(R_SHIFT_IN),'MI')),0')
into v_r_shift_in,v_r_shift_in_min
from att_in.TBL_OT_ENTRY

where empid= I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');

v_r_shift_in:= v_r_shift_in*60+v_r_shift_in_min;
v_minpunchtimehh:=(v_minpunchtimehh)*60+v_minpunchtimemi;

if v_minpunchtimehh<v_r_shift_in then
v_max_total_time:=(v_r_shift_in -v_minpunchtimehh);
end if;

if v_max_total_time>330 then
v_max_total_time:=330;
else
v_max_total_time:=v_max_total_time;
end if;

ot_min := v_max_total_time mod 60;
ot_hour := floor((v_max_total_time) / 60);
if v_max_total_time >29 then
if ot_min<=9 and ot_hour<= 9 then
v_ot_time:= to_char('0'||ot_hour)||':'||to_char('0'|(ot_min))||':00';
else
v_ot_time:= to_char('0'||ot_hour)||':'||to_char(ot_min)||':00';
end if;
else
v_ot_time:= '00:00:00';
end if;
else
v_ot_time:= '00:00:00';
end if;

update att_in.TBL_OT_ENTRY
set OT_HOUR = v_ot_time,
R_IN_TIME_PUNCH=to_date(I_OT_Date||' '||v_minpunchtime,'DD/MM/RRRR HH24:MI:SS'),
R_OUT_TIME_PUNCH=to_date(I_OT_Date||' '|| v_maxpunchtime,'DD/MM/RRRR HH24:MI:SS')
where empid= I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');
commit;
else
update att_in.TBL_OT_ENTRY

```

```

set OT_HOUR = "",
REG_STATUS = 'A'
where empid= I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');
commit;
end if;

elsif diff_in_Ashift_t <360 or diff_in_Ashift_tt <360 then
update att_in.TBL_OT_ENTRY
set OT_HOUR = "",
REG_STATUS = 'A', LATE_ON_REG_SHIFT = '00:00:00',
R_IN_TIME_PUNCH=to_date(I_OT_Date||' '||v_minpunchtime,'DD/MM/RRRR HH24:MI:SS'),
R_OUT_TIME_PUNCH=to_date(I_OT_Date||' '|| v_maxpunchtime,'DD/MM/RRRR HH24:MI:SS')
where empid= I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');
commit;
else

r_shift_cilling_time:=(v_r_shift_in*60)+30;
r_p_time_hour:=(v_minpunchtimehh*60)+v_minpunchtimemi;
if r_shift_cilling_time >= r_p_time_hour then
update att_in.TBL_OT_ENTRY
set REG_STATUS = 'P', LATE_ON_REG_SHIFT = '00:00:00'
where empid= I_EmpId and OT_OR_REG_DATE = to_date(otDate,'DD/MM/YYYY');
commit;
elsif r_p_time_hour > r_shift_cilling_time then
r_shift_late_by := r_p_time_hour - r_shift_cilling_time;

if r_shift_late_by >60 then
r_late_hour := (r_shift_late_by / 60);
r_late_min := (r_shift_late_by mod 60);
if r_late_min<=9 then
update att_in.TBL_OT_ENTRY
set REG_STATUS = 'L', LATE_ON_REG_SHIFT = to_char('0'||to_char(r_late_hour)||':'||'0'||
r_late_min||':00')
where empid= I_EmpId
and OT_OR_REG_DATE = to_date(otDate,'DD/MM/YYYY');
commit;
else
update att_in.TBL_OT_ENTRY
set REG_STATUS = 'L',
LATE_ON_REG_SHIFT = to_char('0'||to_char(r_late_hour)||':'|| r_late_min||':00')
where empid= I_EmpId
and OT_OR_REG_DATE = to_date(otDate,'DD/MM/YYYY');
commit;
end if;
elsif r_shift_late_by>9 and r_shift_late_by < 60 then
update att_in.TBL_OT_ENTRY
set REG_STATUS = 'L',
LATE_ON_REG_SHIFT = to_char('00:'||r_shift_late_by||':00')
where empid= I_EmpId
and OT_OR_REG_DATE = to_date(otDate,'DD/MM/YYYY');
commit;
elsif r_shift_late_by<=9 then
update att_in.TBL_OT_ENTRY

```



```

set REG_STATUS = 'L',
LATE_ON_REG_SHIFT = to_char('00:00'||r_shift_late_by||':00')
where empid= I_EmpId
and OT_OR_REG_DATE = to_date(otDate,'DD/MM/YYYY');
commit;
end if;
end if;

if v_otapprove=1 then
v_r_shift_out:=v_r_shift_out*60;
v_maxpunchtimehh:=(v_maxpunchtimehh)*60+v_maxpunchtimemi;

if v_maxpunchtimehh>v_r_shift_out then
v_max_total_time:=(v_maxpunchtimehh -v_r_shift_out);
end if;

if v_max_total_time>330 then
v_max_total_time:=330;
else
v_max_total_time:=v_max_total_time;
end if;

ot_min := v_max_total_time mod 60;
ot_hour := floor((v_max_total_time) / 60);
if v_max_total_time >29 then
if ot_min<=9 and ot_hour<= 9 then
v_ot_time:= to_char('0'||ot_hour)||':00'||to_char('0'||(ot_min))||':00';
else
v_ot_time:= to_char('0'||ot_hour)||':00'||to_char(ot_min)||':00';
end if;
else
v_ot_time:= '00:00:00';
end if;
else
v_ot_time:= '00:00:00';
end if;

update att_in.TBL_OT_ENTRY
set OT_HOUR = v_ot_time,
R_IN_TIME_PUNCH=to_date(I_OT_Date||' '||v_minpunchtime,'DD/MM/RRRR HH24:MI:SS'),
R_OUT_TIME_PUNCH=to_date(I_OT_Date||' '|| v_maxpunchtime,'DD/MM/RRRR HH24:MI:SS')
where empid= I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');
commit;
end if;

end;

elsif check_reg_shift_name = '2 Shift-D' then
begin

v_minpunchtime_t:="";
v_r_shift_out_t:="";

```

```

select to_date(to_char(min(PUNCHTIME),'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
into v_minpunchtime_t
from att_in.TBL_RAW_DATA_CMIS
where  COMPCARDID = comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS') ;

```

```

select to_date(to_char(max(PUNCHTIME),'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
into v_maxpunchtime_t
from att_in.TBL_RAW_DATA_CMIS
where  COMPCARDID = comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS') ;

```

```

select to_date(to_char(min(PUNCHTIME),'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
into v_minpunchtime_t
from att_in.TBL_RAW_DATA_CMIS
where  COMPCARDID = comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS') ;

```

```

select REG_STATUS into V_REG_STATUS
from att_in.TBL_OT_ENTRY
where empid=I_EmpId
and OT_OR_REG_DATE = to_date(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');

```

```

select count(punchtime)
into V_PUNCHTIME
from att_in.TBL_RAW_DATA_CMIS
where  COMPCARDID = comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY') ;

```

```

select to_date(to_char(R_SHIFT_in,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
into v_r_shift_in_t
from att_in.TBL_OT_ENTRY
where empid=I_EmpId
and OT_OR_REG_DATE = to_date(to_char(I_OT_Date,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS');

```

```

select to_date(to_char(R_SHIFT_out,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
into v_r_shift_out_t
from att_in.TBL_OT_ENTRY
where empid=I_EmpId
and OT_OR_REG_DATE = to_date(to_char(I_OT_Date,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS');

```

```

select nvl(to_char(min(punchtime),'HH24:MI:SS'),'0')
into v_minpunchtime
from att_in.TBL_RAW_DATA_CMIS
where  COMPCARDID = comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY') ;

```

```

select nvl(to_char(max(punchtime),'HH24:MI:SS'),'0')
into v_maxpunchtime
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY') ;

select nvl(to_number(to_char(min(punchtime),'HH24')),'0'),
nvl(to_number(to_char(min(punchtime),'MI')),'0')
into v_minpunchtimehh,v_minpunchtimemi
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY') ;

select nvl(to_number(to_char(max(punchtime),'HH24')),'0') ,
nvl(to_number(to_char(max(punchtime),'MI')),'0')
into v_maxpunchtimehh,v_maxpunchtimemi
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY') ;

select R_SHIFT,to_char(COMPCARDID),to_char(OT_OR_REG_DATE,'DD/MM/YYYY'),
to_number(to_char(R_SHIFT_IN,'HH24')) ,to_number(to_char(R_SHIFT_IN,'MI')),
to_number(to_char(R_SHIFT_OUT,'HH24'))
into v_r_shift,v_compcardid,v_otdate,v_r_shift_in,v_r_shift_in_min,v_r_shift_out
from att_in.TBL_OT_ENTRY
where empid=I_EmpId
and OT_OR_REG_DATE = to_date(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');

diff_in_Ashift_t:="";

Select NVL(SUM(datediff('hh',v_r_shift_in_t,v_maxpunchtime_t)),0) * 60
into diff_in_Ashift_t
from dual;

Select NVL(SUM(datediff('hh',v_minpunchtime_t,v_maxpunchtime_t)),0) * 60
into diff_in_Ashift_tt
from dual;

if V_REG_STATUS='W' and V_PUNCHTIME>=2 then

if diff_in_Ashift_tt >360 and diff_in_Ashift_tt <450 and diff_in_Ashift_t<360 then

update att_in.TBL_OT_ENTRY
set R_SHIFT='2 Shift-C',
R_SHIFT_VIEW='2 Shift-C',
REG_STATUS = 'P',
R_IN_TIME_PUNCH=to_date(I_OT_Date||' '||v_minpunchtime,'DD/MM/RRRR HH24:MI:SS'),
R_OUT_TIME_PUNCH=to_date(I_OT_Date||' '|| v_maxpunchtime,'DD/MM/RRRR HH24:MI:SS')
where empid= I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');
commit;

```

```

elseif diff_in_Ashift_tt >=450 or diff_in_Ashift_t<360 then
update att_in.TBL_OT_ENTRY
set R_SHIFT='2 Shift-C',
R_SHIFT_VIEW='2 Shift-C',
REG_STATUS = 'P',
R_SHIFT_IN=to_date(I_OT_Date||' '||'08:30:00','DD/MM/RRRR HH24:MI:SS'),
R_SHIFT_OUT=to_date(I_OT_Date||' '||'15:00:00','DD/MM/RRRR HH24:MI:SS'),
R_IN_TIME_PUNCH=to_date(I_OT_Date||' '||v_minpunchtime,'DD/MM/RRRR HH24:MI:SS'),
R_OUT_TIME_PUNCH=to_date(I_OT_Date||' '|| v_maxpunchtime,'DD/MM/RRRR HH24:MI:SS')
where empid= I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');
commit;

```

```

if v_otapprove=1 then
select nvl(to_number(to_char(max(punchtime),'HH24'),'0') ,
nvl(to_number(to_char(max(punchtime),'MI'),'0')
into v_maxpunchtimehh,v_maxpunchtimemi
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY') ;

```

```

select nvl(to_number(to_char(R_SHIFT_OUT),'HH24'),'0'),
nvl(to_number(to_char(R_SHIFT_OUT),'MI'),'0')
into v_r_shift_out,v_r_shift_out_min
from att_in.TBL_OT_ENTRY
where empid=I_EmpId
and OT_OR_REG_DATE = to_date(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');

```

```

v_r_shift_out:=v_r_shift_out*60+v_r_shift_out_min;
v_maxpunchtimehh:=v_maxpunchtimehh*60+v_maxpunchtimemi;

```

```

if v_maxpunchtimehh>v_r_shift_out then
v_max_total_time:=(v_maxpunchtimehh -v_r_shift_out);
end if;

```

```

if v_max_total_time>330 then
v_max_total_time:=330;
else
v_max_total_time:= v_max_total_time;
end if;
ot_min := v_max_total_time mod 60;

```

```

ot_hour := floor((v_max_total_time) / 60);
if v_max_total_time >29 then
if ot_min<=9 and ot_hour<=9 then
v_ot_time:= to_char('0'||ot_hour)||':'||to_char('0'|(ot_min))||':00';
else
v_ot_time:= to_char('0'||ot_hour)||':'||to_char(ot_min)||':00';
end if;
else

```

```

v_ot_time:= '00:00:00';
end if;

else
v_ot_time:= '00:00:00';
end if;

update att_in.TBL_OT_ENTRY
set OT_HOUR = v_ot_time,
R_IN_TIME_PUNCH=to_date(I_OT_Date||' '||v_minpunchtime,'DD/MM/RRRR HH24:MI:SS'),
R_OUT_TIME_PUNCH=to_date(I_OT_Date||' '|| v_maxpunchtime,'DD/MM/RRRR HH24:MI:SS')
where empid= I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');
commit;
else
update att_in.TBL_OT_ENTRY
set OT_HOUR = "",
REG_STATUS = 'A'
where empid= I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');
commit;

end if;

elsif diff_in_Ashift_t <360 or diff_in_Ashift_tt <360 then
update att_in.TBL_OT_ENTRY
set OT_HOUR = "",
REG_STATUS = 'A', LATE_ON_REG_SHIFT ='00:00:00',
R_IN_TIME_PUNCH=to_date(I_OT_Date||' '||v_minpunchtime,'DD/MM/RRRR HH24:MI:SS'),
R_OUT_TIME_PUNCH=to_date(I_OT_Date||' '|| v_maxpunchtime,'DD/MM/RRRR HH24:MI:SS')
where empid= I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');
commit;

else

r_shift_cilling_time:=(v_r_shift_in*60)+30;
r_p_time_hour:=(v_minpunchtimehh*60)+v_minpunchtimemi;
if r_shift_cilling_time >= r_p_time_hour then
update att_in.TBL_OT_ENTRY
set REG_STATUS = 'P', LATE_ON_REG_SHIFT ='00:00:00'
where empid= I_EmpId and OT_OR_REG_DATE = to_date(otDate,'DD/MM/YYYY');
commit;
elsif r_p_time_hour > r_shift_cilling_time then
r_shift_late_by := r_p_time_hour - r_shift_cilling_time;

if r_shift_late_by >60 then
r_late_hour := (r_shift_late_by / 60);
r_late_min := (r_shift_late_by mod 60);
if r_late_min<=9 then
update att_in.TBL_OT_ENTRY
set REG_STATUS = 'L', LATE_ON_REG_SHIFT = to_char('0'||to_char(r_late_hour)||':'||'0'||
r_late_min||':00')
where empid= I_EmpId

```

```

and OT_OR_REG_DATE = to_date(otDate,'DD/MM/YYYY');
commit;
else
update att_in.TBL_OT_ENTRY
set REG_STATUS = 'L', LATE_ON_REG_SHIFT = to_char('0'||to_char(r_late_hour)||':'|| r_late_min||':00')
where empid= I_EmpId
and OT_OR_REG_DATE = to_date(otDate,'DD/MM/YYYY');
commit;
end if;
elsif r_shift_late_by>9 and r_shift_late_by < 60 then

update att_in.TBL_OT_ENTRY
set REG_STATUS = 'L', LATE_ON_REG_SHIFT = to_char('00:'||r_shift_late_by||':00')
where empid= I_EmpId
and OT_OR_REG_DATE = to_date(otDate,'DD/MM/YYYY');
commit;
elsif r_shift_late_by<=9 then
update att_in.TBL_OT_ENTRY
set REG_STATUS = 'L', LATE_ON_REG_SHIFT = to_char('00:'||'0'||r_shift_late_by||':00')
where empid= I_EmpId
and OT_OR_REG_DATE = to_date(otDate,'DD/MM/YYYY');
commit;
end if;
end if;

if v_otapprove=1 then
v_r_shift_in:=v_r_shift_in*60+v_r_shift_in_min;
v_minpunchtimehh:=(v_minpunchtimehh)*60+v_minpunchtimemi;

if v_minpunchtimehh<v_r_shift_in then
v_max_total_time:=(v_r_shift_in -v_minpunchtimehh);
end if;

if v_max_total_time>330 then
v_max_total_time:=330;
else
v_max_total_time:= v_max_total_time;
end if;
ot_min := v_max_total_time mod 60;

ot_hour := floor((v_max_total_time) / 60);
if v_max_total_time >29 then
if ot_min<=9 and ot_hour<=9 then
v_ot_time:= to_char('0'||ot_hour)||':'||to_char('0'|(ot_min))||':00';
else
v_ot_time:= to_char('0'||ot_hour)||':'||to_char(ot_min)||':00';
end if;
else
v_ot_time:= '00:00:00';
end if;

else
v_ot_time:= '00:00:00';
end if;

```

```

update att_in.TBL_OT_ENTRY
set OT_HOUR = v_ot_time,
R_IN_TIME_PUNCH=to_date(I_OT_Date||' '||v_minpunchtime,'DD/MM/RRRR HH24:MI:SS'),
R_OUT_TIME_PUNCH=to_date(I_OT_Date||' '|| v_maxpunchtime,'DD/MM/RRRR HH24:MI:SS')
where empid= I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');
commit;
end if;
end;

elsif check_reg_shift_name = '2 Shift-E' then

begin

v_minpunchtime_t:="";
v_r_shift_out_t:="";

select to_date(to_char(min(PUNCHTIME),'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
into v_minpunchtime_t
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS') ;

select to_date(to_char(R_SHIFT_out,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
into v_r_shift_out_t
from att_in.TBL_OT_ENTRY
where empid=I_EmpId
and OT_OR_REG_DATE = to_date(to_char(I_OT_Date,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS');

select nvl(to_char(min(punchtime),'HH24:MI:SS'),'0')
into v_minpunchtime
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY') ;

select nvl(to_char(max(punchtime),'HH24:MI:SS'),'0')
into v_maxpunchtime
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY') ;

select nvl(to_number(to_char(min(punchtime),'HH24'),'0'),
nvl(to_number(to_char(min(punchtime),'MI'),'0'))
into v_minpunchtimehh,v_minpunchtimemi
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY') ;

select nvl(to_number(to_char(max(punchtime),'HH24'),'0') ,

```

```

nvl(to_number(to_char(max(punchtime),'MI')),0')
into v_maxpunchtimehh,v_maxpunchtimemi
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY') ;

select R_SHIFT,to_char(COMPCARDID),to_char(OT_OR_REG_DATE,'DD/MM/YYYY'),
to_number(to_char(R_SHIFT_IN,'HH24')) , to_number(to_char(R_SHIFT_OUT,'HH24'))
into v_r_shift,v_compcardid,v_otdate,v_r_shift_in,v_r_shift_out
from att_in.TBL_OT_ENTRY
where empid=I_EmpId
and OT_OR_REG_DATE = to_date(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');

diff_in_Ashift_t:="";

Select NVL(SUM(datediff('hh',v_minpunchtime_t,v_r_shift_out_t)),0) * 60
into diff_in_Ashift_t
from dual;

if diff_in_Ashift_t <360 then
update att_in.TBL_OT_ENTRY
set OT_HOUR = ",
REG_STATUS = 'A', LATE_ON_REG_SHIFT ='00:00:00',
R_IN_TIME_PUNCH=to_date(I_OT_Date||' '||v_minpunchtime,'DD/MM/RRRR HH24:MI:SS'),
R_OUT_TIME_PUNCH=to_date(I_OT_Date||' '|| v_maxpunchtime,'DD/MM/RRRR HH24:MI:SS')
where empid= I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');
commit;

else

r_shift_cilling_time:=(v_r_shift_in*60)+30;
r_p_time_hour:=(v_minpunchtimehh*60)+v_minpunchtimemi;

if r_shift_cilling_time >= r_p_time_hour then
update att_in.TBL_OT_ENTRY
set REG_STATUS = 'P', LATE_ON_REG_SHIFT ='00:00:00'
where empid= I_EmpId and OT_OR_REG_DATE = to_date(otDate,'DD/MM/YYYY');
commit;
elsif r_p_time_hour > r_shift_cilling_time then
r_shift_late_by := r_p_time_hour - r_shift_cilling_time;

if r_shift_late_by >60 then
r_late_hour := (r_shift_late_by / 60);
r_late_min := (r_shift_late_by mod 60);
if r_late_min<=9 then
update att_in.TBL_OT_ENTRY
set REG_STATUS = 'L', LATE_ON_REG_SHIFT = to_char('0'||to_char(r_late_hour)||':'||'0'||
r_late_min||':00')
where empid= I_EmpId and OT_OR_REG_DATE = to_date(otDate,'DD/MM/YYYY');
commit;
else

```



```

update att_in.TBL_OT_ENTRY
set REG_STATUS = 'L', LATE_ON_REG_SHIFT = to_char('0'||to_char(r_late_hour)||':'|| r_late_min||':00')
where empid= I_EmpId and OT_OR_REG_DATE = to_date(otDate,'DD/MM/YYYY');
commit;
end if;
elsif r_shift_late_by>9 and r_shift_late_by < 60 then

update att_in.TBL_OT_ENTRY
set REG_STATUS = 'L', LATE_ON_REG_SHIFT = to_char('00:'||r_shift_late_by||':00')
where empid= I_EmpId
and OT_OR_REG_DATE = to_date(otDate,'DD/MM/YYYY');
commit;
elsif r_shift_late_by<=9 then
update att_in.TBL_OT_ENTRY
set REG_STATUS = 'L', LATE_ON_REG_SHIFT = to_char('00:'||'0'||r_shift_late_by||':00')
where empid= I_EmpId
and OT_OR_REG_DATE = to_date(otDate,'DD/MM/YYYY');
commit;
end if;

end if;

if v_otapprove=1 then
v_r_shift_out:=v_r_shift_out*60;
v_maxpunchtimehh:=(v_maxpunchtimehh)*60+v_maxpunchtimemi;

if v_maxpunchtimehh>v_r_shift_out then
v_max_total_time:=(v_maxpunchtimehh -v_r_shift_out);
end if;

if v_max_total_time>360 then
v_max_total_time:=360;
else
v_max_total_time:= v_max_total_time;
end if;

ot_min := v_max_total_time mod 60;
ot_hour := floor((v_max_total_time) / 60);
if v_max_total_time >29 then
if ot_min<=9 and ot_hour<= 9 then
v_ot_time:= to_char('0'||ot_hour)||':'||to_char('0'|(ot_min))||':00';
else
v_ot_time:= to_char('0'||ot_hour)||':'||to_char(ot_min)||':00';
end if;
else
v_ot_time:= '00:00:00';
end if;
else
v_ot_time:= '00:00:00';
end if;

update att_in.TBL_OT_ENTRY
set OT_HOUR = v_ot_time,

```

```

R_IN_TIME_PUNCH=to_date(I_OT_Date||' '||v_minpunchtime,'DD/MM/RRRR HH24:MI:SS'),
R_OUT_TIME_PUNCH=to_date(I_OT_Date||' '|| v_maxpunchtime,'DD/MM/RRRR HH24:MI:SS')
where empid= I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');
commit;

end if;

end;

elsif check_reg_shift_name = '2 Shift-F' then
v_minpunchtime_t := "";

begin
select to_date(to_char(min(PUNCHTIME),'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
into v_minpunchtime_t
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS') ;

select to_date(to_char(R_SHIFT_out,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
into v_r_shift_out_t
from att_in.TBL_OT_ENTRY
where empid=I_EmpId
and OT_OR_REG_DATE = to_date(to_char(I_OT_Date,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS');

select nvl(to_char(min(punchtime),'HH24:MI:SS'),'0')
into v_minpunchtime
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY') ;

select nvl(to_char(max(punchtime),'HH24:MI:SS'),'0')
into v_maxpunchtime
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY') ;

select nvl(to_number(to_char(min(punchtime),'HH24')),0),
nvl(to_number(to_char(min(punchtime),'MI')),0)
into v_minpunchtimehh,v_minpunchtimemi
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY') ;

select nvl(to_number(to_char(max(punchtime),'HH24')),0) ,
nvl(to_number(to_char(max(punchtime),'MI')),0)
into v_maxpunchtimehh,v_maxpunchtimemi
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY') ;

```

```

select R_SHIFT,to_char(COMPCARDID),to_char(OT_OR_REG_DATE,'DD/MM/YYYY'),
to_number(to_char(R_SHIFT_IN,'HH24')) , to_number(to_char(R_SHIFT_OUT,'HH24'))
into v_r_shift,v_compcardid,v_otdate,v_r_shift_in,v_r_shift_out
from att_in.TBL_OT_ENTRY
where empid=I_EmpId
and OT_OR_REG_DATE = to_date(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');

Select NVL(SUM(datediff('hh',v_minpunchtime_t,v_r_shift_out_t)),0) * 60
into diff_in_shift_t
from dual;

if diff_in_shift_t <360 then

update att_in.TBL_OT_ENTRY
set OT_HOUR = ",
REG_STATUS = 'A', LATE_ON_REG_SHIFT = '00:00:00',
R_IN_TIME_PUNCH=to_date(I_OT_Date||' '||v_minpunchtime,'DD/MM/RRRR HH24:MI:SS'),
R_OUT_TIME_PUNCH=to_date(I_OT_Date||' '|| v_maxpunchtime,'DD/MM/RRRR HH24:MI:SS')
where empid= I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');
commit;

else

r_shift_cilling_time:=(v_r_shift_in*60)+30;
r_p_time_hour:=(v_minpunchtimehh*60)+v_minpunchtimemi;
if r_shift_cilling_time >= r_p_time_hour then
update att_in.TBL_OT_ENTRY
set REG_STATUS = 'P', LATE_ON_REG_SHIFT = '00:00:00'
where empid= I_EmpId
and OT_OR_REG_DATE = to_date(otDate,'DD/MM/YYYY');
commit;
elsif r_p_time_hour > r_shift_cilling_time then
r_shift_late_by := r_p_time_hour - r_shift_cilling_time;

if r_shift_late_by >60 then
r_late_hour := (r_shift_late_by / 60);
r_late_min := (r_shift_late_by mod 60);
if r_late_min<=9 then
update att_in.TBL_OT_ENTRY
set REG_STATUS = 'L', LATE_ON_REG_SHIFT = to_char('0'||to_char(r_late_hour)||':'||'0'||
r_late_min||':00')
where empid= I_EmpId
and OT_OR_REG_DATE = to_date(otDate,'DD/MM/YYYY');
commit;
else
update att_in.TBL_OT_ENTRY
set REG_STATUS = 'L', LATE_ON_REG_SHIFT = to_char('0'||to_char(r_late_hour)||':'|| r_late_min||':00')
where empid= I_EmpId
and OT_OR_REG_DATE = to_date(otDate,'DD/MM/YYYY');
commit;
end if;

```

```

elsif r_shift_late_by>9 and r_shift_late_by < 60 then

update att_in.TBL_OT_ENTRY
set REG_STATUS = 'L', LATE_ON_REG_SHIFT = to_char('00:'||r_shift_late_by||':00')
where empid= I_EmpId
and OT_OR_REG_DATE = to_date(otDate,'DD/MM/YYYY');
commit;
elsif r_shift_late_by<=9 then
update att_in.TBL_OT_ENTRY
set REG_STATUS = 'L', LATE_ON_REG_SHIFT = to_char('00:'||'0'||r_shift_late_by||':00')
where empid= I_EmpId
and OT_OR_REG_DATE = to_date(otDate,'DD/MM/YYYY');
commit;
end if;
end if;

if v_otapprove=1 then
v_r_shift_out:=v_r_shift_out*60;
v_maxpunchtimehh:=(v_maxpunchtimehh)*60+v_maxpunchtimemi;

if v_maxpunchtimehh>v_r_shift_out then
v_max_total_time:=(v_maxpunchtimehh -v_r_shift_out);
end if;

if v_max_total_time>300 then
v_max_total_time:=300;
else
v_max_total_time:= v_max_total_time;
end if;

ot_min := v_max_total_time mod 60;
ot_hour := floor((v_max_total_time) / 60);
if v_max_total_time >29 then
if ot_min<=9 and ot_hour<= 9 then
v_ot_time:= to_char('0'||ot_hour)||':'||to_char('0||(ot_min))||':00';
else
v_ot_time:= to_char('0'||ot_hour)||':'||to_char(ot_min)||':00';
end if;
else
v_ot_time:= '00:00:00';
end if;

else
v_ot_time:= '00:00:00';
end if;

update att_in.TBL_OT_ENTRY
set OT_HOUR = v_ot_time,
R_IN_TIME_PUNCH=to_date(I_OT_Date||' '||v_minpunchtime,'DD/MM/RRRR HH24:MI:SS'),
R_OUT_TIME_PUNCH=to_date(I_OT_Date||' '|| v_maxpunchtime,'DD/MM/RRRR HH24:MI:SS')
where empid= I_EmpId and OT_OR_REG_DATE =
TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');
commit;

end if;

```

```

end;

--elsif
elsif check_reg_shift_name = '2 Shift-G' then
begin

v_minpunchtime_t:="";
v_r_shift_out_t:="";

select to_date(to_char(min(PUNCHTIME),'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
into v_minpunchtime_t
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS') ;

select to_date(to_char(max(PUNCHTIME),'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
into v_maxpunchtime_t
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS') ;

select to_date(to_char(min(PUNCHTIME),'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
into v_minpunchtime_t
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID =comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS') ;

select REG_STATUS into V_REG_STATUS
from att_in.TBL_OT_ENTRY
where empid=I_EmpId
and OT_OR_REG_DATE = to_date(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');

select count(punchtime)
into V_PUNCHTIME
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY') ;

select to_date(to_char(R_SHIFT_in,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
into v_r_shift_in_t
from att_in.TBL_OT_ENTRY
where empid=I_EmpId
and OT_OR_REG_DATE = to_date(to_char(I_OT_Date,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR
HH24:MI:SS');

select to_date(to_char(R_SHIFT_out,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS')
into v_r_shift_out_t
from att_in.TBL_OT_ENTRY
where empid=I_EmpId

```

```
and OT_OR_REG_DATE = to_date(to_char(I_OT_Date,'DD/MM/RRRR HH24:MI:SS'),'DD/MM/RRRR HH24:MI:SS');
```

```
select nvl(to_char(min(punchtime),'HH24:MI:SS'),'0')
into v_minpunchtime
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY') ;
```

```
select nvl(to_char(max(punchtime),'HH24:MI:SS'),'0')
into v_maxpunchtime
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY') ;
```

```
select nvl(to_number(to_char(min(punchtime),'HH24')),'0'),
nvl(to_number(to_char(min(punchtime),'MI')),'0')
into v_minpunchtimehh,v_minpunchtimemi
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY') ;
```

```
select nvl(to_number(to_char(max(punchtime),'HH24')),'0') ,
nvl(to_number(to_char(max(punchtime),'MI')),'0')
into v_maxpunchtimehh,v_maxpunchtimemi
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY') ;
```

```
select R_SHIFT,to_char(COMPCARDID),to_char(OT_OR_REG_DATE,'DD/MM/YYYY'),
to_number(to_char(R_SHIFT_IN,'HH24')) ,to_number(to_char(R_SHIFT_IN,'MI')),
to_number(to_char(R_SHIFT_OUT,'HH24'))
into v_r_shift,v_compcardid,v_otdate,v_r_shift_in,v_r_shift_in_min,v_r_shift_out
from att_in.TBL_OT_ENTRY
where empid=I_EmpId
and OT_OR_REG_DATE = to_date(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');
```

```
diff_in_Ashift_t:="";
```

```
Select NVL(SUM(datediff('hh',v_r_shift_in_t,v_maxpunchtime_t)),0) * 60
into diff_in_Ashift_t
from dual;
```

```
Select NVL(SUM(datediff('hh',v_minpunchtime_t,v_maxpunchtime_t)),0) * 60
into diff_in_Ashift_tt
from dual;
```

```
if V_REG_STATUS='W' and V_PUNCHTIME>=2 then
```

```

if diff_in_Ashift_tt >360 and diff_in_Ashift_tt <450 then

update att_in.TBL_OT_ENTRY
set R_SHIFT='2 Shift-C',
R_SHIFT_VIEW='2 Shift-C',
REG_STATUS = 'P',
R_IN_TIME_PUNCH=to_date(I_OT_Date||' '||v_minpunchtime,'DD/MM/RRRR HH24:MI:SS'),
R_OUT_TIME_PUNCH=to_date(I_OT_Date||' '|| v_maxpunchtime,'DD/MM/RRRR HH24:MI:SS')
where empid= I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');
commit;

elsif diff_in_Ashift_tt >=450 then
update att_in.TBL_OT_ENTRY
set R_SHIFT='2 Shift-C',
R_SHIFT_VIEW='2 Shift-C',
REG_STATUS = 'P',
R_SHIFT_IN=to_date(I_OT_Date||' '||'08:30:00','DD/MM/RRRR HH24:MI:SS'),
R_SHIFT_OUT=to_date(I_OT_Date||' '|| '15:00:00','DD/MM/RRRR HH24:MI:SS'),
R_IN_TIME_PUNCH=to_date(I_OT_Date||' '||v_minpunchtime,'DD/MM/RRRR HH24:MI:SS'),
R_OUT_TIME_PUNCH=to_date(I_OT_Date||' '|| v_maxpunchtime,'DD/MM/RRRR HH24:MI:SS')
where empid= I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');
commit;

if v_otaprove=1 then
select nvl(to_number(to_char(max(punchtime),'HH24')),'0') ,
nvl(to_number(to_char(max(punchtime),'MI')),'0')
into v_maxpunchtimehh,v_maxpunchtimemi
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY') ;

select nvl(to_number(to_char(R_SHIFT_OUT),'HH24'),'0'),
nvl(to_number(to_char(R_SHIFT_OUT),'MI'),'0')
into v_r_shift_out,v_r_shift_out_min
from att_in.TBL_OT_ENTRY
where empid=I_EmpId
and OT_OR_REG_DATE = to_date(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');

v_r_shift_out:=v_r_shift_out*60+v_r_shift_out_min;
v_maxpunchtimehh:=v_maxpunchtimehh*60+v_maxpunchtimemi;

if v_maxpunchtimehh>v_r_shift_out then
v_max_total_time:=(v_maxpunchtimehh -v_r_shift_out);
end if;

if v_max_total_time>330 then
v_max_total_time:=330;
else
v_max_total_time:= v_max_total_time;

```

```

end if;
ot_min := v_max_total_time mod 60;

ot_hour := floor((v_max_total_time) / 60);
if v_max_total_time > 29 then
if ot_min <= 9 and ot_hour <= 9 then
v_ot_time := to_char('0' || ot_hour) || ':' || to_char('0' || (ot_min)) || ':00';
else
v_ot_time := to_char('0' || ot_hour) || ':' || to_char(ot_min) || ':00';
end if;
else
v_ot_time := '00:00:00';
end if;

else
v_ot_time := '00:00:00';
end if;
--else
update att_in.TBL_OT_ENTRY
set OT_HOUR = v_ot_time,
R_IN_TIME_PUNCH = to_date(I_OT_Date || ' ' || v_minpunchtime, 'DD/MM/RRRR HH24:MI:SS'),
R_OUT_TIME_PUNCH = to_date(I_OT_Date || ' ' || v_maxpunchtime, 'DD/MM/RRRR HH24:MI:SS')
where empid = I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date, 'DD/MM/YYYY'), 'DD/MM/YYYY');
commit;
else
update att_in.TBL_OT_ENTRY
set OT_HOUR = "",
REG_STATUS = 'A'
where empid = I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date, 'DD/MM/YYYY'), 'DD/MM/YYYY');
commit;

end if;

elseif diff_in_Ashift_t < 360 then

update att_in.TBL_OT_ENTRY
set OT_HOUR = "",
REG_STATUS = 'A', LATE_ON_REG_SHIFT = '00:00:00',
R_IN_TIME_PUNCH = to_date(I_OT_Date || ' ' || v_minpunchtime, 'DD/MM/RRRR HH24:MI:SS'),
R_OUT_TIME_PUNCH = to_date(I_OT_Date || ' ' || v_maxpunchtime, 'DD/MM/RRRR HH24:MI:SS')
where empid = I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date, 'DD/MM/YYYY'), 'DD/MM/YYYY');
commit;

else

r_shift_cilling_time := (v_r_shift_in * 60) + 30;
r_p_time_hour := (v_minpunchtimehh * 60) + v_minpunchtimemi;
if r_shift_cilling_time >= r_p_time_hour then
update att_in.TBL_OT_ENTRY

```



```

set REG_STATUS = 'P', LATE_ON_REG_SHIFT = '00:00:00'
where empid= I_EmpId and OT_OR_REG_DATE = to_date(otDate,'DD/MM/YYYY');
commit;
elsif r_p_time_hour > r_shift_cilling_time then
r_shift_late_by := r_p_time_hour - r_shift_cilling_time;

if r_shift_late_by >60 then
r_late_hour := (r_shift_late_by / 60);
r_late_min := (r_shift_late_by mod 60);
if r_late_min<=9 then
update att_in.TBL_OT_ENTRY
set REG_STATUS = 'L', LATE_ON_REG_SHIFT = to_char('0'||to_char(r_late_hour)||':'||'0'||
r_late_min||':00')
where empid= I_EmpId
and OT_OR_REG_DATE = to_date(otDate,'DD/MM/YYYY');
commit;
else
update att_in.TBL_OT_ENTRY
set REG_STATUS = 'L', LATE_ON_REG_SHIFT = to_char('0'||to_char(r_late_hour)||':'|| r_late_min||':00')
where empid= I_EmpId
and OT_OR_REG_DATE = to_date(otDate,'DD/MM/YYYY');
commit;
end if;
elsif r_shift_late_by>9 and r_shift_late_by < 60 then

update att_in.TBL_OT_ENTRY
set REG_STATUS = 'L', LATE_ON_REG_SHIFT = to_char('00:'||r_shift_late_by||':00')
where empid= I_EmpId
and OT_OR_REG_DATE = to_date(otDate,'DD/MM/YYYY');
commit;
elsif r_shift_late_by<=9 then
update att_in.TBL_OT_ENTRY
set REG_STATUS = 'L', LATE_ON_REG_SHIFT = to_char('00:'||'0'||r_shift_late_by||':00')
where empid= I_EmpId
and OT_OR_REG_DATE = to_date(otDate,'DD/MM/YYYY');
commit;
end if;
end if;

if v_otapprove=1 then
v_r_shift_in:=v_r_shift_in*60+v_r_shift_in_min;
v_minpunchtimehh:=(v_minpunchtimehh)*60+v_minpunchtimemi;

if v_minpunchtimehh<v_r_shift_in then
v_max_total_time:=(v_r_shift_in -v_minpunchtimehh);
end if;

if v_max_total_time>330 then
v_max_total_time:=330;
else
v_max_total_time:= v_max_total_time;
end if;
ot_min := v_max_total_time mod 60;

```

```

ot_hour := floor((v_max_total_time) / 60);
if v_max_total_time > 29 then
if ot_min <= 9 and ot_hour <= 9 then
v_ot_time := to_char('0' || ot_hour) || ':' || to_char('0' || (ot_min)) || ':00';
else
v_ot_time := to_char('0' || ot_hour) || ':' || to_char(ot_min) || ':00';
end if;
else
v_ot_time := '00:00:00';
end if;

else
v_ot_time := '00:00:00';
end if;

update att_in.TBL_OT_ENTRY
set OT_HOUR = v_ot_time,
R_IN_TIME_PUNCH = to_date(I_OT_Date || ' ' || v_minpunchtime, 'DD/MM/RRRR HH24:MI:SS'),
R_OUT_TIME_PUNCH = to_date(I_OT_Date || ' ' || v_maxpunchtime, 'DD/MM/RRRR HH24:MI:SS')
where empid = I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date, 'DD/MM/YYYY'), 'DD/MM/YYYY');
commit;
end if;
end;

end if;

begin
select EMPLOYEE_TYPE, EMPLOYEE_CATEGORY
into v_employee_type, v_employee_category
from att_in.TBL_EMP_PERSONAL_INFO
where EMPID = I_EmpId;
EXCEPTION
when no_data_found then
null;
end;

begin
select count(ID) into special_ot_count
from att_in.TBL_SPECIAL_OT_DAYS
where STARTDATE <= TO_DATE(to_char(I_OT_Date, 'DD/MM/YYYY'), 'DD/MM/YYYY')
and ENDDATE >= TO_DATE(to_char(I_OT_Date, 'DD/MM/YYYY'), 'DD/MM/YYYY');
EXCEPTION
when no_data_found then
null;
end;

```

```

If v_employee_type in('P','G') and v_employee_category='Permanent' and v_otapprove=1 then

if chk_p_exist >0 and special_ot_count > 0 then

select count(R_SHIFT_ID),R_SHIFT
into ot_count,reg_shift_name
from att_in.TBL_OT_ENTRY
where empid= I_EmpId
and OT_OR_REG_DATE = to_date(otDate,'DD/MM/YYYY')
group by R_SHIFT;

if ot_count > 0 and (reg_shift_name = '2 Shift-A' or reg_shift_name = '2 Shift-B'
OR reg_shift_name = '2 Shift-C' OR reg_shift_name = '2 Shift-D'
or reg_shift_name = '2 Shift-E' or reg_shift_name= '2 Shift-F' or reg_shift_name= '2 Shift-G' ) then

select nvl(to_number(to_char(min(punchtime),'HH24')),'0'),
nvl(to_number(to_char(min(punchtime),'MI')),'0')
into v_minpunchtimehh,v_minpunchtimemi
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY') ;

select nvl(to_number(to_char(max(punchtime),'HH24')),'0') ,
nvl(to_number(to_char(max(punchtime),'MI')),'0')
into v_maxpunchtimehh,v_maxpunchtimemi
from att_in.TBL_RAW_DATA_CMIS
where COMPCARDID = comp_cardid
and PUNCHDATE = to_date(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY') ;

ot_in_time:=v_minpunchtimehh*60+v_minpunchtimemi;
ot_punch_out:=v_maxpunchtimehh*60+v_maxpunchtimemi;

if ot_punch_out>ot_in_time then

difrence_In_minutes :=(ot_punch_out- ot_in_time);

if difrence_In_minutes>1920 then
difrence_In_minutes:='1920';
else
difrence_In_minutes:=difrence_In_minutes;
end if;

ot_min := difrence_In_minutes mod 60;
ot_hour := (difrence_In_minutes - ot_min) / 60;
if ot_hour > 9 then

if ot_min >= 0 and ot_min < 10 then
total_ot_hour := to_char(ot_hour)||':0'||to_char(ot_min)||':00';
elsif ot_min >=10 and ot_min <=59 then
total_ot_hour := to_char(ot_hour)||':'||to_char(ot_min)||':00';
end if;
elsif ot_hour <=9 then
if ot_min >= 0 and ot_min < 10 then

```

```

total_ot_hour := to_char('0'||to_char(ot_hour)||':0'||to_char(ot_min)||':00');
elsif ot_min >=10 and ot_min <=59 then
total_ot_hour := to_char('0'||to_char(ot_hour)||':'||to_char(ot_min)||':00');
end if;

end if;

update att_in.TBL_OT_ENTRY
set OT_HOUR = to_char(total_ot_hour),REG_STATUS = 'P', LATE_ON_REG_SHIFT ='00:00:00'
where empid= I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');
commit;
end if;

else

update att_in.TBL_OT_ENTRY
set OT_HOUR = ",REG_STATUS = 'P', LATE_ON_REG_SHIFT ='00:00:00'
where empid= I_EmpId
and OT_OR_REG_DATE = TO_DATE(to_char(I_OT_Date,'DD/MM/YYYY'),'DD/MM/YYYY');
commit;
end if;
end if;
end if;
end if;
end if;
end if;
end if;
End;
End;
/

```

=====

[illegible][illegible][illegible][illegible][illegible][illegible][illegible][illegible][illegible]

**SEQUENCE: SEQ\_PLANT\_ID**

Min Value	Max Value	Increment	Last Value	Cycle	Order
1	99999999999999999999999999999999	1	141	No	No

**SEQUENCE: SEQ\_SEASONID**

[illegible]

**SEQUENCE: SEQ\_SHIFTID**

[illegible]

**SEQUENCE: SEQ\_SHIFTSETUPID**

[illegible]

**SEQUENCE: SEQ\_USER\_TYPE\_ID**

[illegible]

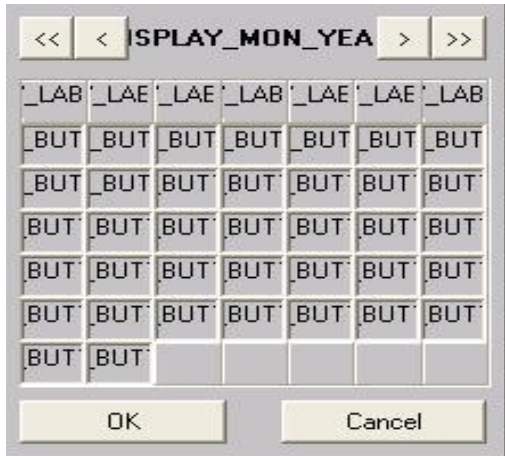
**SEQUENCE: SEQ\_USERID**

[illegible]**SEQUENCE: SEQ\_WEEKLYHOLIID**[illegible]

## Description of Application

### Technical Explanations of Forms used in the application

#### \*\*\* List of Common Controls used all over the application

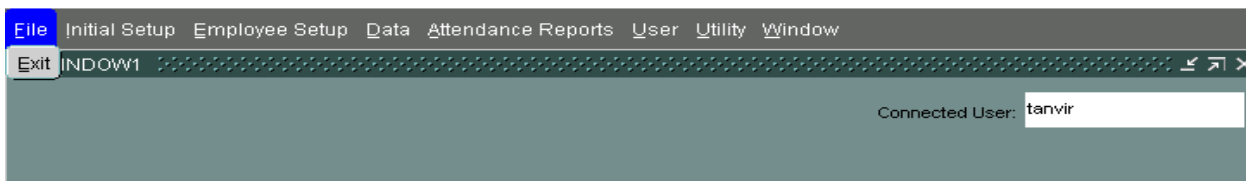


DATE\_CONTROL\_BLOCK,  
DATE\_BUTTON\_BLOCK

#### **Menu Form:**

- A. Form name MENU\_FORM\_user\_test
  - a. Menu Name → MENU

#### **Main Menu Bar Items:**



- 1) File
- 2) Initial Setup
- 3) Employee Setup
- 4) Data
- 5) Attendance Report
- 6) User
- 7) Utility

#### **Submenu Items under main manu:**

8. File

- Exit → To Exit from the Application.

## 9. Initial Setup

SL NO	Submenu Items Name	Form Name
1	Company Name	FMB_COMPANY_SETUP
2	Financial Year Steup	FMB_FINANCIAL_YEAR
3	Shift Name Setup	FRB_SHIFT
4	Shift Time Setup	FMB_SHIFT_TIME_SETUP
5	Leave Type Setup	FMB_LEAVE_TYPE_ENTRY
6	Labour OT Entry	FMB_LABOUR_ENTRY
7	Weekly Holiday Setup	FMB_WEEKLY_HOLIDAY_SETUP
8	Govt Holiday Setup	FMB_GOVTHD_ENTRY_FORM
9	Special Holiday Setup	FMB_SPECIAL_OT_DAYS
10	OCS /Official Work Setup	FMB_OSD_ENTRY_FORM
11	Check Designation	FMB_CHECK

## 10.Employee Setup

SLNO	Submenu Items Name	Form Name
1	Employee Entry	FMB_NEW_EMPLOYEE_ENTRY
2	Leave Apply Form	FMB_LEAVE_APPLY
3	2 Shift Duty Roster Entry	FMB_2SHIFTOT_ENTRY
4	4 Shift Duty Roster Entry	FMB_4SHIFTOT_ENTRY
5	OT Approve Day Wise	FMB_OT_APPROVED
6	Manual OT Edit	FMB_OT_EDIT

## 11.Data

SLNO	Submenu Items Name	Form Name
1	Data Collection and Process	FMB_DATA_COLLECTION
2	Data Manual Entry	FMB_MANUAL_ENTRY
3	Text File Generate	FMB_TEXT_FILE_GENERATE

## 12.Attendance Report

SLN O	Submenu Items Name	Form Name	Call Report
1	Daily Report	FMB_DAILY_REPORT	RptDalyreport_datewise.jsp
2	Job Card Report	FMB_JOB_CARD	rptJobCardReport.jsp
3	MonthlyReport Empcode Range	FMB_MONTHLY_REPORT_EM PRANGE	rptMonthlySummaryEmpcodeRa nge.jsp
4	Monthly Report	FMB_MONTHLY_REPORT	rptMonthlySummary.jsp
5	ManualEntry Report	FMB_MANUALLY_REPORT	rptManualEntryReport.jsp
6	Leave Report	FMB_LEAVE_REPORT	RptLeaveReport.jsp
7	Over Time Edit Report	FMB_OTEDIT_REPORT	RptOTEditReport.jsp



<b>8</b>	User Info Report	<b>FMB_USER_REPORT</b>	<b>RptUserList.jsp</b>
<b>9</b>	Machine Entry Report	<b>FMB_MACHINE_INFO</b>	<b>RptMachineInfoReport.jsp</b>

### 13. User

<b>SLNO</b>	<b>Submenu Items Name</b>	<b>Form Name</b>
<b>1</b>	Create User Type	<b>FMB_CREATE_USER_TYPE</b>
<b>2</b>	Create User	<b>FMB_CREATE_USER</b>
<b>3</b>	Assign User Role	<b>FMB_USER_ROLE</b>
<b>4</b>	Update User Role	<b>FMB_USER_ROLE_UPDATE</b>
<b>5</b>	User Role Grant And Revoke	<b>USER_ROLE</b>
<b>6</b>	Change User Password	<b>USER_PASSWORD_CHANGE</b>

### 14. Utility

<b>SLNO</b>	<b>Submenu Items Name</b>	<b>Form Name</b>
<b>1</b>	Form Entry	<b>FMB_FORM_ENTRY</b>
<b>2</b>	Machine Entry	<b>FMB_MACHINE_INFO</b>

## DETAILS SUBMENU ITEMS

---

### 1. FILE

- Exit → To Exit from the Application.

## 2. Initial Setup Menu

### Forms Description

#### 1) **FMB\_COMPANY\_SETUP**

The screenshot shows the 'Company Setup' form. It contains several input fields for company details, a search section, and a table of existing companies. Annotations with arrows point to specific parts of the form:

- An arrow points from the 'Company Phone', 'Company Fax', and 'Company Web Address' fields to a box labeled 'Block TBL\_COMPANY\_INFO'.
- An arrow points from the 'Company Code' field in the search section to a box labeled 'Block SEARCH'.
- An arrow points from the table of companies to a box labeled 'Block DISPLAY'.

Code	Company Name	Address	Financial Year	Phone
1	Biman Bangladesh Airlines Ltd.	Dalaka Bhaban, Kurmitola Dhaka.	2012-2012	

### Block Name

- TBL\_COMPANY\_INFO
- SEARCH
- DISPLAY
- List of LOVS used in **FMB\_COMPANY\_SETUP**
  - i) FIN\_LOV used in block TBL\_COMPANY\_INFO at COMPFINYNNAME Text Box to fetch value FINIANTIAL\_YEAR from TBL\_FINANCIAL\_YEAR and display into COMPFINYNNAME Text box.

- ii) COMP\_INFO LOV used in SEARCH block COMPCODE Text Box fetch value COMPCODE from TBL\_COMPANY\_INFO and display into COMPCODE Text box.

❖ **TBL\_COMPANY\_INFO BLOCK**

- i) Item Name: COMPCODE,COMPNAMEE,COMPADDE,COMPEMAIL, COMPPHONE, COMPFAX,COMPWEB,COMPFINYNNAME,BTNSAVE,BTN\_SETUP, BTN\_CANCEL
- ii) Tables Used for TBL\_COMPANY\_INFO:
- a) TBL\_USER\_ROLE\_PERMISSION,
- b) b).TBL\_COMPANY\_INFO
- iii) Procedure Used for TBL\_COMPANY\_INFO:
- a) att\_in.sp\_Company\_Add\_Update.
- Call this procedure under BTN\_SAVE button. Call Procedure for insert or update TBL\_COMPANY\_INFO.

❖ **SEARCH**

Item Name: COMPCODE, SEARCH\_BTN

❖ **DIAPLAY**

**a) Item Name:**

COMPID, COMPCODE, COMPNAMEE, COMPADDE, FINANCIALYEAR, COMPPHONE (All Item is Database Item)

**b) Table Name:**

TBL\_COMPANY\_INFO

2) **FMB\_FINANCIAL\_YEAR**

The screenshot shows a 'Financial Year Setup' form. It contains several input fields and a table. Annotations with arrows point from specific parts of the form to three blocks:

- Block TBL\_FINANCIAL\_YEAR**: Points to the top section of the form containing 'Financial Year Name', 'From Date', 'To Date', and 'CAL...' fields.
- Block SEARCH**: Points to the 'For Search' section containing a 'Financial Year' input field and a 'Search' button.
- Block DISPLAY**: Points to the table at the bottom of the form.

Company Financial Year Name	Start Date	End Date
2013-2013	01/01/2013	31/12/2013
2012-2012	01/01/2012	31/12/2012

**Block Name**

- TBL\_FINANCIAL\_YEAR
- SEARCH
- DISPLAY

- List of LOVS used in **FMB\_FINIANCIAL\_YEAR**
  - FIN\_YEAR Lov used in SEARCH block FINANTIAL\_YEAR Text box fetch value FINANTIAL\_YEAR from TBL\_FINANCIAL\_YEAR and display into FINANTIAL\_YEAR Text box.

#### ❖ **TBL\_FINIANCIAL\_YEAR**

**I. Item name:** TXTCOMPANYNAME, FROM\_DATE, TO\_DATE, CAL1, ITEM75, BTN\_SAVE, BTN\_CANCEL

**II.** Table used for TBL\_FINANCIAL\_YEAR

- TBL\_USER\_ROLE\_PERMISSION
- TBL\_FINANCIAL\_YEAR

**III. Procedure used:** att\_in.sp\_FinantialYear\_Add\_Update

#### ❖ **SEARCH**

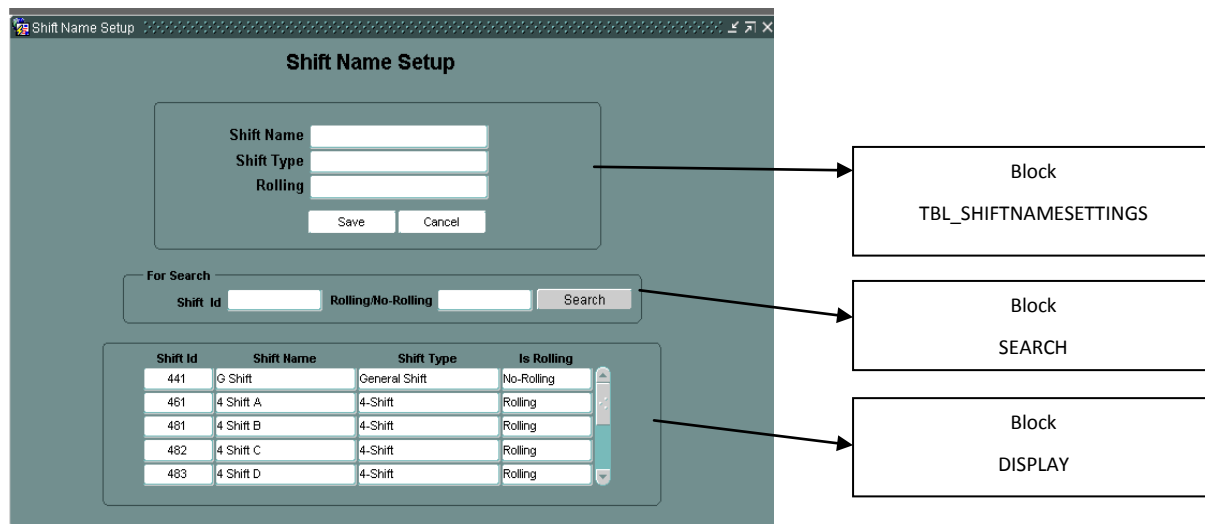
**Item Name:** FINANTIAL\_YEAR, SEARCH\_BTN

#### ❖ **DISPLAY**

**Table Name:** TBL\_FINIANCIAL\_YEAR (All Item is Database Item)

**Item Name:** COMPFINYNNAME, COMPFINSTARTDATE, COMPFINENDDATE

### 3) **FRB\_SHIFT**



#### **Block Name**

- TBL\_SHIFTNAMESETTINGS
- SEARCH
- DISPLAY
- List of LOVS used in **FMB\_SHIFT**
  - SHIFT\_TYPE\_LOV** Lov used in TBL\_SHIFTNAMESETTING block SHIFTTYPE Text box fetch value SHIFTTYPENAME from table TBL\_SHIFTNAMESETTING and display into SHIFTTYPE Text box..

- b) **ROLLING\_LOV** Lov used in TBL\_SHIFTNAMESETTING block ISROLLING Text box fetch value ROLLINGNAME from table TBL\_ROLLING and display into ISROLLING Text box..
- c) **SHIFTINFO** lov used in SEARCH block SHIFTID Text box fetch value SHIFTID from table TBL\_SHIFTNAMESETTINGS and display into SHIFTID Text box.
- d) **ROLLING\_LOV** lov used in SEARCH block ROLLING\_NONROLLING Text box fetch value ROLLINGNAME from table TBL\_ROLLING and display into ROLLING\_NONROLLING Text box.

❖ TBL\_SHIFTNAMESETTINGS

- I. **Item Name:** SHIFTNAME, SHIFTTYPE, ISROLLING, LOGINTIME, LOGOUTTIME, BTNSAVE, BTN\_CANCEL
- II. **Use Table Name:**
  - a) TBL\_USER\_ROLE\_PERMISSION
  - b) TBL\_SHIFTNAMESETTINGS
- III. **Use Procedure Name:**
  - a) att\_in.sp\_ShiftName\_Add\_Update

❖ SEARCH

**Item Name:** SHIFTID, ROLLING\_NONROLLING, SEARCH\_BTN

❖ DISPLAY

- a) **Item Name:** SHIFTID, SHIFTNAME, SHIFTTYPE, ISROLLING, LOGINTIME, LOGOUTTIME, DELETE\_BTTN (All Item is Database Item)
- b) **Use Table Name:** att\_in.TBL\_SHIFT\_NAMESETTINGS

4) **FMB\_SHIFT\_TIME\_SETUP**

The screenshot shows a 'Shift Time Setup' form. Annotations point to the following blocks:

- Block TBL\_SHIFT\_SETUP**: Points to the top section of the form containing fields for Company Name, Directoriet Name, Location Name, Department Name, Season Type, Shift Name, Shift In, Shift Out, Late By, Lunch IN, Lunch Out, and Regular Hour. It also includes Save and Cancel buttons.
- Block SEARCH**: Points to the 'For Search' section containing text boxes for Dept Id, Season Id, and Shift Id, along with a Search button.
- Block TBL\_SHIFT\_SETUP1**: Points to the data table at the bottom of the form.

Company Name	Dept ID	Department Name	Season ID	Season Name	Shift ID	Shift Name	In Time	Out Time	Regular Hour
Biman Bangladesh Airt	60	C M I S	141	SUMMER	441	G Shift	09:00	17:00	08:00
Biman Bangladesh Airt	60	C M I S	141	SUMMER	461	4 Shift A	06:00	14:00	06:00
Biman Bangladesh Airt	60	C M I S	161	RAMADAN	441	G Shift	09:00	15:30	06:30
Biman Bangladesh Airt	60	C M I S	141	SUMMER	481	4 Shift B	14:00	22:00	08:00
Biman Bangladesh Airt	60	C M I S	141	SUMMER	482	4 Shift C	22:00	06:00	08:00

### **Block Name:**

- TBL\_SHIFT\_SETUP
- SEARCH
- TBL\_SHIFT\_SETUP1
- List of LOVS used in **FMB\_SHIFT\_TIME\_SETUP**
  - a) SEARCH\_SHIFT Lov used in SEARCH block SHIFTID Text box fetch value SHIFTID from TBL\_SHIFTNAMESETTINGS and display into SHIFTID Text box.
  - b) SEARCH\_SEASON Lov used in SEARCH block SEASONID Text box fetch value SEASONID from TBL\_SEASON\_SETUP and display into SEASONID Text box.
  - c) SEARCH\_DEPT Lov used in SEARCH block DEPTID Text box fetch value DEPTID from TBL\_DEPARTMENT and display into DEPTID Text box.
  - d) DIR\_LOV Lov used in TBL\_SHIFT\_SETUP block DIRID Text box fetch value BD\_ID from TBL\_BUSINESSSDIVISION and display into DIRID Text box.
  - e) DEPT\_LOV Lov used in TBL\_SHIFT\_SETUP block DEPTID Text box fetch value DEPTID from TBL\_DEPARTMENT and display into DEPTID Text box.
  - f) LOC\_LOV Lov used in TBL\_SHIFT\_SETUP block LOCID Text box fetch value PLANT\_ID from TBL\_PLANTFACTORY and display into LOCID Text box.

### ❖ TBL\_SHIFT\_SETUP

- I. **Item Name:** COMPNAME, DIRID, LOCID, DEPTID, DDLSEC, DDLSHIFT, LOGINTIME, LOGOUTTIME, LATEBY, LUNCHLOGIN, LUNCHLOGOUT, REGULARHOUR, BTN\_SAVE, DELETE\_BTTN, BTN\_CANCEL
- II. **Use Table Name:**
  - a) att\_in.TBL\_USER\_ROLE\_PERMISSION
  - b) TBL\_SHIFT\_SETUP
  - c) TBL\_SHIFT\_NAMESETTINGS
  - d) TBL\_SEASON\_SETUP
  - e) TBL\_DEPARTMENT.
  - f) TBL\_BUSINESSSDIVISION
  - g) TBL\_PLANTFACTORY

h) att\_in.tbl\_company\_info

**III. Procedure Name:**

a) att\_in.sp\_Shift\_Add\_Update

**IV.**

❖ SEARCH

**Item Name:** DEPTID, SEASONID, SHIFTID, SEARCH\_BTN

❖ TBL\_SHIFT\_SETUP1

**a) Item Name:** COMNAME, DEPTID, DEPTNAME, SEASONID, SEASONNAME, SHIFTID, SHIFTNAME, INTIME, OUTTIME, REGULAR (All Item is Database Item)

**b) Use Table Name:** att\_in.TBL\_SHIFT\_SETUP

5) **FMB\_LEAVE\_TYPE\_ENTRY**

The screenshot shows a 'Leave Type Setup' form with three main sections. Arrows point from these sections to three separate boxes on the right:

- Top Section:** Contains 'Leave Type Name' and 'Reporting Name' input fields, and 'Save' and 'Cancel' buttons. An arrow points to a box labeled 'Block TBL\_LEAVE\_TYPE'.
- Middle Section:** Labeled 'For Search', it contains 'Leave Type Id' and 'Leave Type' input fields, and a 'Search' button. An arrow points to a box labeled 'Block SEARCH'.
- Bottom Section:** A table with two columns: 'Leave Type Name' and 'Reporting Name'. It lists: Casual Leave (CL), Sick Leave (SL), Privilege Leave (PL), Earned Leave (EL), and Maternity Leave (ML). An arrow points to a box labeled 'Block DISPLAY'.

Leave Type Name	Reporting Name
Casual Leave	CL
Sick Leave	SL
Privilege Leave	PL
Earned Leave	EL
Maternity Leave	ML

**Block Name:**

- **TBL\_LEAVE\_TYPE**

- **SEARCH**
- **DISPLAY**
- List of LOVS used in **FMB\_LEAVE\_TYPE\_ENTRY**
  - a) LV\_LEAVETYPE lov used in SEARCH block LEAVETYPEID Text box fetch value LEAVETYPEID, LEAVETYPENAME from TBL\_LEAVE\_TYPE and display into Text box LEAVETYPEID, LEAVETYPENAME

❖ TBL\_LEAVE\_TYPE:

- a) **Item Name:** LEAVETYPENAME, REPORTINGNAME, BTN\_SAVE, BTN\_CANCEL
- b) **Use Table Name:**
  - 1) att\_in.TBL\_USER\_ROLE\_PERMISSION
  - 2) TBL\_LEAVE\_TYPE
- c) **Use Procedure Name:**
  - 1) att\_in.sp\_Leave\_Type\_Add\_Update.

❖ SEARCH:

- a) **Item Name:** LEAVETYPEID, LEAVETYPENAME, SEARCH\_BTN.
- b) **User Table Name:** TBL\_LEAVE\_TYPE.

❖ DISPLAY:

- a) Item Name: LEAVETYPENAME, REPORTINGNAME (All Item is Database Item)
- b) Table Name: TBL\_LEAVE\_TYPE

## 6) FMB\_LABOUR\_ENTRY

The screenshot shows the 'Labour Over Time Entry' form. It contains three main sections: a top section for entering labour details, a middle section for searching, and a bottom table for displaying results. Arrows point from these sections to labels on the right: 'Block TBL\_LABOUR' for the top section, 'Block SEARCH' for the middle section, and 'Block DISPLAY' for the bottom table.

Labour Code	Name	OT Status	OT Type
0	UNKNOWN	Yes	ADMIN
1	ADMIN PAY-GROUP I	Yes	ADMIN
2	ADMIN PAY-GROUP II	Yes	ADMIN
3	ADMIN SPL PAY-GRP II	Yes	ADMIN
4	ADMIN PAY-GRP III-1	Yes	ADMIN

**Block Name:**

- **TBL\_LABOUR**
- **SEARCH**
- **DISPLAY**



- List of LOVS used in **FMB\_LABOUR\_ENTRY**
  - a) SEARCH\_LCODE lov used in SEARCH block LABOURCODE Text box fetch value LABOURCODE from TBL\_LABOUR and display into LABOURCODE Text box.
  - b) LOV\_LABOUR Lov used in TBL\_LABOUR block LABOURCODE, Text box fetch value LABOURCODE, NAME from TBL\_LABOUR and display into Text box LABOURCODE, NAME
- ❖ TBL\_LABOUR:
  - a) **Item Name:** LABOURCODE, NAME, OTSTATUS, OTTYPE, BTN\_SAVE, BTN\_CANCEL
  - b) **Use Table Name:**
    - 1) TBL\_USER\_ROLE\_PERMISSION
    - 2) TBL\_LABOUR
  - c) **Use Procedure Name:** ATT\_IN.Sp\_LabourInsertUpdate
- ❖ SEARCH
  - a) **Item Name:** LABOURCODE, OTSTATUS, OTTYPE, SEARCH\_BTN
  - b) **Use table Name:** TBL\_LABOUR
- ❖ DISPLAY:
  - a) **Item Name:** LABOURCODE, NAME, OTSTATUS, OTTYPE (All Item is Database Item)
  - b) **Use Table Name:** TBL\_LABOUR

## 7) FMB\_WEEKLY\_HOLIDAY\_SETUP

The screenshot shows the 'Weekly Holiday Setup' form. It has a title bar 'Weekly Holiday Setup'. Inside, there is a 'Shift Type Name' text box. Below it is a 'Day Name' section containing a list of days: Saturday, Sunday, Monday, Tuesday, Wednesday, Thursday, and Friday. At the bottom are 'Save' and 'Cancel' buttons. Two arrows point from the form to external blocks: one from the 'Shift Type Name' text box to a block labeled 'Block TBL\_WEEKLY\_HOLIDAY', and another from the 'Day Name' list to a block labeled 'Block TBL\_WEEK\_DAY'.

### Block Name:

- **CONTROL**
- **TBL\_WEEKLY\_HOLIDAY**

- **TBL\_WEEK\_DAY**
- List of Lov use in **FMB\_WEEKLY\_HOLIDAY\_SETUP**
  - a) LOV\_SHIFT\_TYPE\_NAME lov used in TBL\_WEEKLY\_HOLIDAY block SHIFTTYPENAME Text box fetch value SHIFTTYPENAME from TBL\_SHIFT\_TYPE and display into SHIFTTYPENAME Text box.

❖ **CONTROL:**

- a) **Item Name:** SAVE, CANCEL
- b) **Use Table Name:**
  - i. TBL\_USER\_ROLE\_PERMISSION
  - ii. TBL\_WEEKLY\_HOLIDAY

❖ **TBL\_WEEKLY\_HOLIDAY:**

- a) **Item Name:** SHIFTTYPENAME
- b) **Use Table Name:** TBL\_SHIFT\_TYPE

❖ **TBL\_WEEK\_DAY**

- a) **Item Name:** DAYNAME, F (All Item is Database Item).
- b) **Use Table Name:** TBL\_WEEKLY\_HOLIDAY

## 8) **FMB\_GOVTHD\_ENTRY\_FORM**

The screenshot shows the 'Govt Holiday Entry Form' interface. It includes input fields for Financial Year, Govt Holiday Name, Govt Holiday Start Date, Govt Holiday End Date, Number Of Days, and Remarks. There are 'Save' and 'Cancel' buttons. Below the input fields is a 'For Search' section with 'Holiday Start Date From' and 'Date To' fields, each with a 'CAL' button, and a 'Search' button. At the bottom is a table with columns: Govt Holiday Name, Holiday Start Date, Holiday End Date, No. Of Days, and Remarks. The table contains three rows of data. Annotations with arrows point from the following blocks to specific parts of the form:

- Block TBL\_GOVTHD\_HOLIDAY\_LIST** points to the main input fields area.
- Block SEARCH** points to the 'For Search' section.
- Block TBL\_GOVTHD\_HOLIDAY\_LIST1** points to the table at the bottom.

**Block Name:**

- **TBL\_GOVTHD\_HOLIDAY\_LIST**
- **SEARCH**
- **TBL\_GOVTHD\_HOLIDAY\_LIST1**

- List of Lov use in **FMB\_GOVTHD\_ENTRY\_FORM**
  - a) FINYEAR\_LV Lov used in TBL\_GOVTHD\_HOLIDAY\_LIST block COMPFINYNNAME Text box fetch value COMPFINYNNAME from TBL\_FINANCIAL\_YEAR and display into COMPFINYNNAME Text box
- ❖ TBL\_GOVTHD\_HOLIDAY\_LIST:
  - a) **Item Name:** COMPFINYNNAME, GOVTHOLINAME, GOVTHOLISTARTDATE, GOVTHOLIENDDATE, GOVTNUMBERDAYS, GOVTREMARKS, FROM\_CAL, TO\_CAL, BTN\_SAVE, EXIT\_BTTN
  - b) **Use Table Name:**
    - i. TBL\_USER\_ROLE\_PERMISSION
    - ii. TBL\_GOVTHD\_HOLIDAY\_LIST
    - iii. TBL\_COMPANY\_INFO
  - c) **Use Procedure Name:** ATT\_IN.sp\_GovtHoliday\_Add\_Update
- ❖ SEARCH
  - a) **Item Name:** FROM\_DATE, TO\_DATE, CAL1, CAL2, SEARCH\_BTN
- ❖ TBL\_GOVTHD\_HOLIDAY\_LIST1:
  - a) **Item Name:** GOVTHOLINAME, GOVTHOLISTARTDATE, GOVTHOLIENDDATE, GOVTNUMBERDAYS, GOVTREMARKS (All Item is Database Item)
  - b) **Use Table Name:** TBL\_GOVTHD\_HOLIDAY\_LIST

## 9) FMB\_SPECIAL\_OT\_DAYS

The screenshot shows the 'Special OT Days' form. It contains three main sections: a top form for adding a new holiday, a search section, and a table of existing holidays. Arrows point from these sections to labels on the right:

- The top form (with fields for Year, Start Date, End Date, Holiday Name, and buttons Save, Delete, Cancel) is labeled **Block TBL\_SPECIAL\_OT\_DAYS**.
- The search section (with fields for Year, Date From, To, and a Search button) is labeled **Block SEARCH**.
- The table of existing holidays (with columns Id, Start Date, End Date, and Holiday Name) is labeled **Block TBL\_SPECIAL\_OT\_DAYS1**.

Id	Start Date	End Date	Holiday Name
1	01/05/2013	01/05/2013	May Day

### Block Name:

- TBL\_SPECIAL\_OT\_DAYS
- SEARCH

- TBL\_SPECIAL\_OT\_DAYS1
- List of Lov use in **FMB\_SPECIAL\_OT\_DAYS**
  - a) SEARCH\_YEAR Lov used in SEARCH block YEAR Text box fetch value YEAR from TBL\_SPECIAL\_OT\_DAYS and display into YEAR Text box
- ❖ TBL\_SPECIAL\_OT\_DAYS:
  - a) **Item Name:** ID, YEAR, STARTDATE, ENDDATE, HOLIDAYNAME, CAL1, CAL2, BTN\_SAVE, DELETE\_BTTN, EXIT\_BTTN
  - b) **Use Table Name:**
    - i. TBL\_USER\_ROLE\_PERMISSION
    - ii. TBL\_SPECIAL\_OT\_DAYS
  - c) **Use Procedure Name:** att\_in.tbl\_special\_ot\_days
- ❖ SEARCH:
  - a) **Item Name:** FROM\_DATE, TO\_DATE, YEAR, CAL1, CAL2, SEARCH\_BTN
- ❖ TBL\_SPECIAL\_OT\_DAYS1:
  - a) **Item Name:** ID, STARTDATE, ENDDATE, HOLIDAYNAME (All Item is Database Item)
  - b) **Use Table Name:** TBL\_SPECIAL\_OT\_DAYS

## 10) FMB\_OSD\_ENTRY\_FORM

The screenshot shows the 'OSD Setup Form' interface. It contains a top section for data entry, a middle section for searching, and a bottom table section. Annotations with arrows point from specific parts of the form to three blocks:

- Block TBL\_OSD\_SETUP:** Points to the top data entry section containing fields for Empid, Osd Start Date, Osd End Date, Number Of Days, Osd Location, and Osd Reason, along with Save and Cancel buttons.
- Block SEARCH:** Points to the 'For Search' section containing fields for Emp Code, From Date, To Date, and a Search button.
- Block TBL\_OSD\_SETUP1:** Points to the table at the bottom of the form.

Empid	Osd start Date	Osd End Date	No. Of Days	Osd Location	Osd Reason
11348	13/03/2013	13/03/2013	1	China	For Buy some product
10508	13/02/2013	14/02/2013	2	Sylhet	For Buy some product
10638	07/02/2013	07/03/2013	1	LONDON	For Buy some product
9111	01/01/2013	01/01/2013	1	Singapur	For Buy some product

Block Name:

- TBL\_OSD\_SETUP
- SEARCH

- TBL\_OSD\_SETUP1
- List of Lov use in **FMB\_OSD\_ENTRY\_FORM**
  - a) LV\_EMP or LV\_EMP\_DEPT Lov used in TBL\_OSD\_SETUP block EMPCODE Text box fetch value EMPID from TBL\_EMP\_PERSONAL\_INFO and display into EMPCODE Text box
  - b) SEARCH\_EMPCODE or SEARCH\_EMPCODE\_DEPT Lov used in SEARCH block EMPCODE Text box and fetch value EMPID from TBL\_EMP\_PERSONAL\_INFO and display into EMPCODE Text box.
- ❖ TBL\_OSD\_SETUP:
  - a) Item Name: EMPCODE, OSDSTARTDATE, FROM\_CAL, OSDENDDATE, TO\_CAL, OSDLOCATION, OSDREASON, BTN\_SAVE, DELETE\_BTTN, EXIT\_BTTN
  - b) Use Table Name:
    - i. TBL\_USER\_ROLE\_PERMISSION
    - ii. TBL\_OSD\_SETUP
  - c) Used Procedure Name: att\_in.sp\_OSD\_Add\_Update
- ❖ SEARCH:
  - a) Item Name: EMPCODE, DEPTID, FROM\_DATE, TO\_DATE, CAL1, CAL2, SEARCH\_BTN
- ❖ TBL\_OSD\_SETUP1:
  - a) Item Name: EMPCODE, OSDSTARTDATE, OSDENDDATE, OSDLOCATION, OSDREASON (All Item is Database Item)
  - b) Use Table Name: TBL\_OSD\_SETUP

## 11) FMB\_CHECK

The screenshot shows the 'Information Check Form' window. It contains several input fields and buttons. Annotations on the right side of the image point to specific blocks in the form:

- Block COMPANY:** Points to the 'Company Name' field, which contains 'Biman Bangladesh Airlines Ltd.'.
- Block TBL\_CHECK:** Points to the section containing fields for 'Designation', 'Empcode', 'Department Id', 'Location Id', 'Labour Code', 'Design\_96\_hour...', and 'Design\_64\_hour...'. It also points to the 'Get not short rest', 'Get milk allowance', 'Get OT 96 hour', and 'Get OT 64 hour' checkboxes.
- Block SEARCH:** Points to the 'For Search' section, which includes fields for 'Designation', 'Empcode', 'Dept Id', 'Location Id', 'Labour Code', 'Max 96 H\_O', and 'Max 64 H\_O', along with a 'Search' button.
- Block DISPLAY:** Points to the table at the bottom of the form, which displays data for 'Designation', 'Empcode', 'Dept Id', 'Location Id', 'Labour Code', 'Max 96 H', and 'Max 64 H'.

Designation	Empcode	Dept Id	Location Id	Labour Code	Max 96 H	Max 64 H
	35231					
	35275					
	50683					

Block Name:

- COMPANY
- TBL\_CHECK
- SEARCH
- DISPLAY
- List of Lov use in **FMB\_CHECK**
  - a) EMP\_LOV or EMP\_DEPT Lov used in TBL\_CHECK block EMPCODE Text box and fetch value EMPID from TBL\_EMP\_PERSONAL\_INFO and display into EMPCODE Text box.
  - b) DESGIN\_LOV Lov used in TBL\_CHECK block DESIGNATION Text box and fetch value DESIGNATIONID from TBL\_DESIGNATION\_SETUP and display into DESIGNATION Text box
  - c) PLANT\_LOV Lov used in TBL\_CHECK block PLANT\_ID Text box and fetch value PLANT\_ID from TBL\_PLANTFACTORY and display into PLANT\_ID Text box
  - d) LABOUR\_LOV Lov used in TBL\_CHECK block LABOURCODE Text box and fetch value LABOURCODE from TBL\_ LABOUR and display into LABOURCODE Text box
  - e) DEPT\_LOV Lov used in TBL\_CHECK block DEPTID Text box and fetch value DEPTID from TBL\_ DEPARTMENT and display into DEPTIDE Text box
- ❖ COMPANY BLOCK:
  - a) Item Name: COMPANY
- ❖ TBL\_CHECK
  - a) Item Name: RADIO\_GROUP7, DESIGNATION, EMPCODE, DEPTID, PLANT\_ID, LABOURCODE, MAX\_96\_HOURS\_OT, MAX\_64\_HOURS\_OT, DES\_YN, EMP\_YN, DEP\_YN, PLANT\_YN, LABOUR\_YN, OT\_HOUR\_96\_YN, OT\_HOUR\_64\_YN, BTN\_SAVE, CLEAR, DELETE, EXIT
  - b) Used Table Name:
    - i. TBL\_EMP\_PERSONAL\_INFO
    - ii. TBL\_DESIGNATION\_SETUP
    - iii. TBL\_PLANTFACTORY
    - iv. TBL\_ LABOUR
    - v. TBL\_ DEPARTMENT
- ❖ SEARCH BLOCK:
  - a) Item Name: DESIGNATION, MAX\_96\_HOURS\_OT, MAX\_64\_HOURS\_OT, EMPCODE, DEPTID, PLANTID, LABOURCODE, SEARCH
  - b) Used Table Name:
    - i. TBL\_EMP\_PERSONAL\_INFO
    - ii. TBL\_DESIGNATION\_SETUP
    - iii. TBL\_PLANTFACTORY
    - iv. TBL\_ LABOUR
    - v. TBL\_ DEPARTMENT
- ❖ DISPLAY BLOCK:
  - a) Item Name: DESGINATION, EMPCODE, DEPTID, PLANT\_ID, LABOURCODE, MAX\_96\_HOURS\_OT, MAX\_64\_HOURS\_OT, DES\_YN, EMP\_YN, DEP\_YN, PLANT\_YN, LABOUR\_YN, OT\_HOUR\_96\_YN, OT\_HOUR\_64\_YN
  - b) Used Table Name:
    - i. TBL\_CHECK

### 3. Employee Setup Menu

#### 1) FMB\_NEW\_EMPLOYEE\_ENTRY

The screenshot shows a 'New Employee Entry Form' with the following fields:

- Employee Code
- Machine ID Number
- Employee Full Name
- Joining Date
- Designation
- Employee Type
- Employee Status
- Labour Code
- Company Name (Biman Bangladesh Airlines)
- Directoriet ID
- Location ID
- Department ID
- Shift
- Shift Start From

Buttons: Save, Delete, Cancel

For Search:

Emp Code, Dept Id, Join Date From, CAL, To, CAL, Shift Id, Search

Emp Code	Emp Name	Joining Date	Dept ID	Shift ID	Labour Code
1	MD NURA ALAM SIDDIK	03/08/2005	67	441	1
2	DR MD ALAMGIR	15/12/2004	79	441	1
6	MD SUJAN MAH	20/05/2003	61	441	1
14	MD SELIM	17/08/2003	78	441	1
15	MD GOLAM MOSTAFA	01/06/2003	24	441	1
16	GOLAM KABIR	23/08/2003	77	441	1
17	MD HASAN ALI	12/06/2004	69	441	1

Annotations:

- EMP\_BLOCK**: Details explanation are given below
- SEARCH**: Search button
- DISPLAY**: Table of employee data

Block Name:

- EMP\_BLOCK
- SEARCH
- DISPLAY
- List of LOVS used in FMB\_NEW\_EMPLOYEE\_ENTRY
  - SEARCH\_EMPCODE or SEARCH\_EMPCODE\_DEPT Lov used in SEARCH block EMPCODE Text box and fetch value EMPID from TBL\_EMP\_PERSONAL\_INFO and display into EMPCODE Text box.
  - SEARCH\_SHIFT Lov used in SEARCH block SHIFTID Text box and fetch value SHIFTID from TBL\_SHIFTNAMESETTINGS and display into SHIFTID Text box.
  - SEARCH\_DEPT Lov used in SEARCH block DEPTID Text box and fetch value DEPTID from TBL\_DEPARTMENT and display into DEPTID Text box
  - DIR\_LOV Lov used in EMP\_BLOCK block TXTDIR Text box and fetch value BD\_ID from TBL\_BUSINESSDIVISION and display into TXTDIR Text box
  - LOC\_LOV Lov used in EMP\_BLOCK block TXTLOCA Text box and fetch value PLANT\_ID from TBL\_PLANTFACTORY and display into TXTLOCA Text box
  - DEPT\_LOV Lov used in EMP\_BLOCK block TXTDEPT Text box and fetch value DEPTID\_ID from TBL\_DEPARTMENT and display into TXTDEPT Text box
  - DEPT\_LOV Lov used in EMP\_BLOCK block TXTDEPT Text box and fetch value DEPTID\_ID from TBL\_DEPARTMENT and display into TXTDEPT Text box

- h) SHIFT\_LOV Lov used in EMP\_BLOCK block SHIFTNAME Text box and fetch value SHIFTNAME from TBL\_SHIFTNAMESETTINGS and display into SHIFTNAME Text box.
- i) LABOUR\_LOV Lov used in EMP\_BLOCK block TXTLABRCODE Text box and fetch value LABOURCODE from TBL\_LABOUR and display into TXTLABRCODE Text box.

❖ EMP\_BLOCK:

- a) ItemName: TXTEMPCODE ,TXTCOMPIDNUM, TXTEMPNAME, TXTJOINDATE, TXTDESIG, TXEMPTYTYPE, TXTEMPSTATUS, TXTLABRCODE, TXTCOMPNAME, TXTDIR, TXTLOCA, TXTDEPT, SHIFTNAME, SHIFTSTARTFROM, TXTEMPID, COMNAME, DIRNAME, LOCNAME, DEPTNAME, TXTSHIFTID, BTN\_SAVE, BTN\_CANCEL, TXTCOMPID
- b) Used Table Name:
  - i. TBL\_USER\_ROLE\_PERMISSION
  - ii. TBL\_SHIFTNAMESETTINGS
  - iii. TBL\_DEPARTMENT
  - iv. TBL\_LABOUR
  - v. TBL\_EMP\_PERSONAL\_INFO
  - vi. TBL\_PLANTFACTORY
- c) Used Procedure: sp\_Employee\_Add\_Update\_Biman

❖ SEARCH BLOCK:

- a) Item Name: FROM\_DATE, TO\_DATE, EMPCODE, DEPTID, SHIFTID, CAL1, CAL2, SEARCH\_BTN
- b) Used Table Name:
  - i. TBL\_EMP\_PERSONAL\_INFO
  - ii. TBL\_SHIFTNAMESETTINGS
  - iii. TBL\_DEPARTMENT

❖ DISPLAY BLOCK:

- a) Item Name: EMPCODE, EMPNAMEE, JOININGDATE, DEPTID, LABOURCODE, DELETE\_BTTN
- b) Used Table Name: TBL\_EMP\_PERSONAL\_INFO



## 2) FMB\_LEAVE\_APPLY

The screenshot shows the 'Employee Leave Application Form' interface. Annotations point to three specific blocks:

- TBL\_LEAVE\_APPLY**: Points to the main form fields including Employee Code, Employee Name, Leave Type (with a 'New' button), Remark, From Date, To Date, Financial Year, and Save/Cancel buttons.
- SEARCH**: Points to the search criteria section with fields for Emp Code, From Date, To Date, and a Search button.
- DISPLAY**: Points to the data table showing application records.

Emp Code	From Date	To Date	Cause
30224	01/07/2012	02/07/2012	Tanvir val Sick
30224	01/07/2012	02/07/2012	Personal
34392	24/07/2012	25/07/2012	cl
34392	17/07/2012	17/07/2012	cl
35285	05/06/2012	06/06/2012	sick

### BLOCK NAME:

- TBL\_LEAVE\_APPLY
- SEARCH
- DISPLAY
- List of LOVS used in FMB\_NEW\_EMPLOYEE\_ENTRY
  - a) SEARCH\_EMPCODE or SEARCH\_EMPCODE\_DEPT Lov used in SEARCH block EMPCODE Text box and fetch value EMPID from TBL\_EMP\_PERSONAL\_INFO and display into EMPCODE Text box.
  - b) LOV\_EMP or LOV\_EMP\_DEPT Lov used in TBL\_LEAVE\_APPLY block EMPCODE Text box and fetch value EMPID from TBL\_EMP\_PERSONAL\_INFO and display into TXTLEAVETYPE Text box.
  - c) LOV\_LEAVE\_TYPE Lov used in TBL\_LEAVE\_APPLY block TXTLEAVETYPE Text box and fetch value LEAVETYPE from TBL\_LEAVE\_TYPE and display into TXTLEAVETYPE Text box.
  - d) SEARCH\_DEPT Lov used in SEARCH block TXTDEPTID Text box and fetch value DEPTID from TBL\_DEPARTMENT and display into TXTDEPTID Text box.
- ❖ TBL\_LEAVE\_APPLY BLOCK
  - a) Item Name:TXTEMPCODE,TXTLEAVETYPE,TXTREMARK,FROM\_DATE,TO\_DATE,FROM\_CAL,TO\_CAL, FINANCIAL\_YEAR, BTN\_SAVE, BTN\_CANCEL
  - b) Table Name:
    - i. TBL\_EMP\_PERSONAL\_INFO
    - ii. TBL\_LEAVE\_TYPE
    - iii. TBL\_DEPARTMENT
  - c) Used Procedure Name: sp\_Leave\_Applied\_Add\_Update
- ❖ SEARCH BLOCK:
  - a) Item Name: EMPCODE, FROM\_DATE, TO\_DATE, CAL1, CAL2, SEARCH\_BTN
  - b) Used Table Name: TBL\_EMP\_PERSONAL\_INFO
- ❖ DISPLAY BLOCK:

- a) Item Name: EMPCODE, FROMDATE, TODATE, LEAVEAPPLYID, DELETE\_BTTN
- b) Used Table Name: TBL\_LEAVE\_APPLY

### 3) FMB\_2SHIFTOT\_ENTRY

The screenshot shows the '2 Shift Duty Roster Entry' form. It includes input fields for Employee Name, Duty Pattern, Regular Shift, Weekly Holiday, and Regular/OT Date From/To. A table lists shifts (A through Night) with their respective login and logout times. Below the form is a search section with fields for Emp Code and Punch Date. At the bottom, a table displays a list of entries with columns for Empcode, Punch Date, Duty Pattern, Regular Shift, and Short Rest. Annotations with arrows point from the form to three blocks: TBL\_OT\_ENTRY (pointing to the shift table), SEARCH (pointing to the search section), and TBL\_OT\_ENTRY1 (pointing to the entry list table).

#### BLOCK NAME:

- TBL\_OT\_ENTRY
- SEARCH
- TBL\_OT\_ENTRY1
- List of LOVS used in FMB\_2SHIFTOT\_ENTRY
  - a) SEARCH\_EMP\_DEPT or SEARCH\_EMP Lov used in SEARCH block EMPCODE Text box and fetch value EMPID from TBL\_EMP\_PERSONAL\_INFO and display into EMPCODE Text box.
  - b) EMP\_LOV or EMP\_DEPT Lov used in TBL\_OT\_ENTRY block EMPCODE Text box and fetch value EMPID from TBL\_EMP\_PERSONAL\_INFO and display into EMPCODE Text box.
- ❖ TBL\_OT\_ENTRY BLOCK:
  - a) Item Name: EMPCODE, EMPNAME, DUTY\_PATERN, R\_SHIFT, WEEKLY\_HOLIDAY, R\_DATE\_FROM, CAL1, R\_DATE\_TO, CAL2, BTN\_SAVE, EXIT\_BTTN
  - b) Used Table Name:
    - i. TBL\_EMP\_PERSONAL\_INFO
    - ii. TBL\_OT\_ENTRY
- ❖ SEARCH BLOCK:
  - a) Item Name: EMPCODE, FROM\_DATE, TO\_DATE, CAL1, CAL2, SEARCH\_BTN
  - b) Used Table Name: TBL\_EMP\_PERSONAL\_INFO
- ❖ TBL\_OT\_ENTRY1 BLOCK:
  - a) Item Name: EMPCODE, OT\_OR\_REG\_DATE, DUTY\_PATERN, R\_SHIFT\_VIEW, DELETE\_BTTN
  - b) Used Table Name: TBL\_OT\_ENTRY

#### 4) FMB\_4SHIFTOT\_ENTRY

The screenshot shows the '4 Shift Duty Roster Entry' form. It includes fields for Employee Name, Roster Date From, Roster Date To, Duty Pattern, First Duty, and Regular Shift. There are 'Save', 'Delete', and 'Cancel' buttons. Below these is a 'For Search' section with 'Emp Code' and 'Punch Date From' fields, and a 'Search' button. At the bottom is a table with columns: Empcode, Shift Date, Duty Pattern, Regular Shift, and Short Rest. The table contains four rows of data. Annotations with arrows point from the form to three blocks: 'TBL\_OT\_ENTRY' (pointing to the main form area), 'SEARCH' (pointing to the search section), and 'TBL\_OT\_ENTRY1' (pointing to the table).

**TBL\_OT\_ENTRY**  
Details explanation are given below

**SEARCH**

**TBL\_OT\_ENTRY1**

Empcode	Shift Date	Duty Pattern	Regular Shift	Short Rest
44	01/05/2013	REG	2 Shift-C	
44	02/05/2013	REG	2 Shift-C	
44	03/05/2013	REG	2 Shift-C	
44	04/05/2013	REG	2 Shift-C	

#### BLOCK NAME:

- TBL\_OT\_ENTRY
- SEARCH
- TBL\_OT\_ENTRY1
- List of LOVS used in FMB\_4SHIFTOT\_ENTRY
  - a) SEARCH\_EMP, SEARCH\_EMP EMP Lov used in SEARCH block EMPCODE Text box and fetch value EMPID from TBL\_EMP\_PERSONAL\_INFO and display into EMPCODE Text box
  - b) EMP\_LOV or EMP\_DEPT Lov used in TBL\_OT\_ENTRY block EMPCODE Text box and fetch value EMPID from TBL\_EMP\_PERSONAL\_INFO and display into EMPCODE Text box.

#### ❖ TBL\_OT\_ENTRY Block:

- a) Item Name: EMPCODE, EMPNAME, OT\_DATE, OT\_DATE\_TO, DUTY\_PATERN, FIRST\_DUTY, BTN\_SAVE, CAL2, CAL1, EXIT\_BTTN
- b) Used Table name:
  - i. TBL\_EMP\_PERSONAL\_INFO
  - ii. TBL\_OT\_ENTRY

#### ❖ SEARCH Block:

- a) Item Name: EMPCODE, FROM\_DATE, TO\_DATE, CAL1, CAL2, SEARCH\_BTN

#### ❖ TBL\_OT\_ENTRY1 Block:

- a) Item Name: EMPCODE, OT\_OR\_REG\_DATE, DUTY\_PATERN, R\_SHIFT\_VIEW, DELETE\_BTTN, SHORT\_REST\_HOUR
- b) Used Table Name: TBL\_OT\_ENTRY

## 5) FMB\_OT\_APPROVED

The screenshot shows the 'Over Time Approved' form. Annotations point to the following components:

- BLOCK3**: Points to the top section of the form containing fields for Company Name, Department, Employee, Employee Name, OT Approve Date From, and OT Approve Date To.
- SEARCH**: Points to the 'Search For Delete' section with fields for Emp Code, Punch Date From, and To.
- TBL\_EMP\_OT**: Points to the table with columns Empid, Emp Name, and OT Date.
- V\_EMP\_OTLI**: Points to the table with columns Emp Code, Emp Name, Labourcode, Otstatus, and App Y/N.

### BLOCK NAME:

- BLOCK3
- SEARCH
- TBL\_EMP\_OT\_APPROVE
- CONTROL
- V\_EMP\_OTLIST
- List of LOVS used in FMB\_OT\_APPROVED
  - a) SEARCH\_EMP or SEARCH\_EMPCODE\_DEPT Lov used in SEARCH block EMPCODE Text box and fetch value EMPID from TBL\_EMP\_PERSONAL\_INFO and display into EMPCODE Text box
  - b) DEPT\_LOV Lov used in SEARCH block DEPARTMENT Text box and fetch value DEPTID from TBL\_DEPARTMENT and display into DEPARTMENTT Text box
  - c) LOV\_EMP or LOV\_EMP\_DEPT Lov used in BLOCK3 block EMPCODE Text box and fetch value EMPID from TBL\_EMP\_PERSONAL\_INFO and display into EMPCODE Text box

### ❖ BLOCK3 Block:

- a) Item Name: COMPNAME, RADIO\_GROUP10, DEPTID, EMP\_NAME, DTSTARTDATE, DTFROMDATE, DATE\_TO, DATETO,
- b) Used Table Name:
  - i. TBL\_EMP\_PERSONAL\_INFO
  - ii. TBL\_DEPARTMENT
- c) Used Procedure Name: sp\_OT\_Approve\_Add\_Update\_new

### ❖ SEARCH Block

- a) Item Name: FROM\_DATE, TO\_DATE, EMPCODE, CAL1, CAL2, SEARCH\_BTN

### ❖ TBL\_EMP\_OT\_APPROVE Block:

- a) Used Item Name: EMPID, OTDATE, EMPNAME, DELETE\_BTTN
- b) Used Table Name: TBL\_EMP\_OT\_APPROVE

- ❖ V\_EMP\_OTLIST Block
  - a) Item Name: EMPNAMEE, LABOURCODE, OTSTATUS, CHKEMP
  - b) Used Table Name: TBL\_EMP\_OT\_APPROVE
- ❖ CONTROL Block:
  - a) Item Name: BTN\_SAVE, BTN\_CANCEL

## 6) FMB\_OT\_EDIT

**EMP\_BLOCK**

Details explanation are given below

**SEARCH**

### BLOCK NAME:

- BLOCK3
- V\_OT\_EDIT
- CONTROL
- List of LOVS used in FMB\_OT\_EDIT
  - a) LOV\_EMP\_DEPT or LOV\_EMP Lov used in BLOCK3 block EMPCODE Text box and fetch value EMPID from TBL\_EMP\_PERSONAL\_INFO and display into EMPCODE Text box
- ❖ BLOCK3 Block:
  - a) Item Name: COMPNAME, DDLEMPLOYEE, EMP\_NAME, FROM\_DATE, CAL1, TO\_DATE, CAL2, OTHOUR, OT\_DATE, UPD\_REMARKS, FIND
  - b) Used Table Name: TBL\_EMP\_PERSONAL\_INFO
- ❖ V\_OT\_EDIT
  - a) Item Name: EMPCODE, PUNCHDATE, OTHOUR, ACTUAL\_OT\_HOUR
  - b) Used Table Name: TBL\_OT\_EDIT
- ❖ CONTROL
  - a) Item Name: BTN\_SAVE, BTN\_CANCEL
  - b) Used Table Name: TBL\_OT\_EDIT

## 4. DATA MENU:

### 1) FMB\_DATA\_COLLECTION

The screenshot shows the 'Data Collection' window with the 'Data Collection' tab selected. The window has a title bar with 'Data Collection' and a menu bar with 'Data Collection' and 'Data Process'. The main area is a light gray rectangle. At the bottom, there is a status bar showing 'Record: 1/1' and '<OSC>'. In the center, there are two empty text input fields, one above the other. To the right of these fields are three buttons: 'Browse...', 'Upload', and 'Cancel'. Below these buttons is a button labeled 'File Transfer All'.

The screenshot shows the 'Data Collection' window with the 'Data Process' tab selected. The window has a title bar with 'Data Collection' and a menu bar with 'Data Collection' and 'Data Process'. The main area is a light gray rectangle. It contains several form fields and buttons. At the top, there is a 'Company Name' field with the value 'Giman Bangladesh Airlines L'. Below this are four radio buttons: 'All' (selected), 'Directorlet', 'Location', and 'Department'. Below these are four text input fields. Below these are two more text input fields labeled 'Employee Name' and 'Season Type' with the value 'SUMMER'. Below these are two text input fields labeled 'Form Date' and 'To Date' with the format '(DD/MM/YYYY)'. At the bottom, there are two buttons: 'Data Process' and 'Cancel'. The status bar at the bottom shows 'Record: 1/1' and '<OSC>'. There is a horizontal line of small, illegible text above the status bar.

Tab Page:

- 1) TABDATACOLLECTION
- 2) TABDATAPROCESS

BLOCK NAME:

- BLOCK5
- EMP
- CONTROL
- List of LOVS used in FMB\_DATA\_COLLECTION
  - a) EMP\_LOV\_DEPT or EMP\_LOV Lov used in BLOCK5 block DDLEMPLOYEE Text box and fetch value EMPID from TBL\_EMP\_PERSONAL\_INFO and display into DDLEMPLOYEE Text box

- b) LOCATION Lov used in BLOCK5 block LOCATION Text box and fetch value PLANT\_ID from TBL\_PLANTFACTORY and display into LOCATION Text box
  - c) DIR\_LOV Lov used in BLOCK5 block DIRID Text box and fetch value BD\_ID from TBL\_BUSINESSDIVISION and display into DIRID Text box
  - d) DEPT\_LOV Lov used in BLOCK5 block DEPTID Text box and fetch value DEPTID from TBL\_DEPARTMENT and display into DEPTID Text box
  - e) SEA\_LOV Lov used in BLOCK5 block TXTSEASONNAME Text box and fetch value SECTION\_ID from TBL\_SEASON\_SETUP and display into TXTSEASONNAME Text box
- ❖ BLOCK5 Block;
    - Item Name: RADIO\_GROUP10, DIRID, LOCID, DEPTID, DDLEMPLOYEE, EMP\_NAME, TXTSEASONNAME, FROMDATE, BTN\_FROMDATE, TODATE, BTN\_TODATE, BTN\_PROCESS, BTN\_CANCEL, BROWSE
  - ❖ EMP Block:
    - a) Item Name: EMP\_NO
  - ❖ CONTROL Block: USER\_NAME, PATH, MSG, PROCESS\_BTN, BTN\_CANCEL\_1, FILE\_TRNSF

## 2) FMB\_MANUAL\_ENTRY

Machine Id	Punch Date	Punch Time	Entry Tag	Remarks
000000003	18/07/2013	18/07/2013 08:00:00	1	official
000000017	19/05/2013	01/01/1900 09:00:00	1	Miss match of finger print

### BLOCK NAME:

- BLOCK3
- SEARCH
- TBL\_RAW\_DATA
- List of LOVS used in FMB\_DATA\_COLLECTION
  - a) EMP\_LOV\_DEPT or EMP\_LOV Lov used in BLOCK3 block DDLEMPLOYEE Text box and fetch value EMPID from TBL\_EMP\_PERSONAL\_INFO and display into DDLEMPLOYEE Text box
  - b) LOCATION Lov used in BLOCK3 block LOCATION Text box and fetch value PLANT\_ID from TBL\_PLANTFACTORY and display into LOCATION Text box

- c) DIR\_LOV Lov used in BLOCK3 block DIRID Text box and fetch value BD\_ID from TBL\_BUSINESSDIVISION and display into DIRID Text box
- d) DEPT\_LOV Lov used in BLOCK3 block DEPTID Text box and fetch value DEPTID from TBL\_DEPARTMENT and display into DEPTID Text box
- e) SEARCH\_EMP\_DEPT or SEARCH\_EMP Lov used in BLOCK5 block EMPCODE Text box and fetch value SECTION\_ID from TBL\_SEASON\_SETUP and display into EMPCODE Text box
- f) DEPT\_LOC\_LOV Lov used in BLOCK3 block LOCATION Text box and fetch value PLANT\_ID from TBL\_PLANTFACTORY and display into LOCATION Text box

❖ BLOCK3 Block:

- a) Item Name: COMPNAME, RADIO\_GROUP10, DIRID, LOCID, DEPTID, LOCATION, DDLEMPLOYEE, EMP\_NAME, REMARK, FROM\_DATE, TO\_DATE, DTFROMDATE, DTTODATE, IN\_TIME, OUT\_TIME, SAVE\_BTN, DELETE\_BTTN, BTN\_CANCEL
- b) Used Table name: TBL\_RAW\_DATA
- c) Used Procedure: sp\_Manual\_Insert

❖ SEARCH Block:

Item Name: EMPCODE, FROM\_DATE, TO\_DATE, CAL1, CAL2, SEARCH\_BTN

❖ TBL\_RAW\_DATA Block:

- a) Item Name: COMPCARDID, PUNCHDATE, PUNCHTIME, MANUALEENTRYTAG, REMARKS, DELETE\_BTTN
- b) Used Table Name: TBL\_RAW\_DATA

### 3) FMB\_TEXT\_FILE\_GENERATE

BLOCK NAME:

- BLOCK3
- List of LOVS used in FMB\_TEXT\_FILE\_GENERATE
  - a) EMP\_LOV\_DEPT or EMP\_LOV Lov used in BLOCK3 block DDLEMPLOYEE Text box and fetch value EMPID from TBL\_EMP\_PERSONAL\_INFO and display into DDLEMPLOYEE Text box



- b) LOCATION Lov used in BLOCK3 block LOCATION Text box and fetch value PLANT\_ID from TBL\_PLANTFACTORY and display into LOCATION Text box
- c) DIR\_LOV Lov used in BLOCK3 block DIRID Text box and fetch value BD\_ID from TBL\_BUSINESSDIVISION and display into DIRID Text box
- d) DEPT\_LOV Lov used in BLOCK3 block DEPTID Text box and fetch value DEPTID from TBL\_DEPARTMENT and display into DEPTID Text box
- e) DEPT\_LOC\_LOV Lov used in BLOCK3 block LOCATION Text box and fetch value PLANT\_ID from TBL\_PLANTFACTORY and display into LOCATION Text box

❖ BLOCK3 Block:

Item Name: COMPNAME, RADIO\_GROUP10, DEPTID, DDLEMPLOYEE, EMP\_NAME, DIRID, LOCID, LOCATION, DTFROMDATE, FROM\_DATE, DTTODATE, TXTDOWNLOADTEXTFILE, BTN\_CANCEL

## 5. USER MENU

### 1) **FMB\_CREATE\_USER\_TYPE**

User Type ID	User Type Name
1	Super Admin
2	Admin
3	User

BLOCK NAME:

- TBL\_USERTYPE
- SEARCH
- DISPLAY
- List of LOVS used in FMB\_ **CREATE\_USER\_TYPE**

LV\_USERTYPE Lov used in SEARCH block USERTYPENAME Text box and fetch value USERTYPENAME from TBL\_USERTYPE and display into USERTYPENAME Text box

❖ TBL\_USERTYPE Block:

- a) Item Name: USERTYPENAME, BTN\_SAVE, BTN\_CANCEL

❖ SEARCH Block:

- a) Item Name: USERTYPEID, USERTYPENAME, SEARCH\_BTN
- b) Used Table Name: TBL\_USERTYPE

❖ DISPLAY Block:

- a) Item Name: USERTYPEID ,USERTYPENAME, DELETE\_BTT

2) **FMB\_CREATE\_USER**

Empcode	User Name	Password	User Type
802	matin	***	Admin
815	rahman	***	Admin
821	mijan	*****	User
827	hai	***	Admin

BLOCK NAME:

- BLOCK3
- TBL\_USER
- SEARCH
- DISPLAY
- ACCOUNT\_UNLOCK

- a) List of LOVS used in FMB\_ **CREATE\_USER**

- b) LV\_USERTYPE Lov used in SEARCH block USERTYPENAME Text box and fetch value USERTYPENAME from TBL\_USERTYPE and display into USERTYPENAME
- c) UESRLOGID Lov used in SEARCH block USERID Text box and fetch value UESRLOGID from TBL\_USER and display into USERID Text box.
- d) SEARCH\_EMPCODE or SEARCH\_EMPCODE\_DEPT Lov used in SEARCH block EMPCODE Text box and fetch value EMPID from TBL\_EMPLOYEE\_PERSONAL\_INFO and display into EMPCODE
- e) AUTH\_DEPT Lov used in TBL\_USER block AUTH\_DEPTID Text box and fetch value DEPTID from TBL\_DEPARTMENT and display into AUTH\_DEPTID Text box.
- f) DEPT\_LOV Lov used in SEARCH block DEPTID Text box and fetch value DEPTID from TBL\_DEPARTMENT and display into DEPTID Text box.
- g) SEARCH\_USER Lov used in ACCOUNT\_UNLOCK block USERNAME Text box and fetch value USERNAME from TBL\_USER and display into USERNAME Text box.

❖ BLOCK3 Block:

- a) Item Name: DDLUSERTYPE, SAVE, CANCEL

❖ TBL\_USER Block:

- a) Item Name: EMPCODE, USERNAME, USERLOGID, USERLOGPASSWORD, DEPTID, AUTH\_DEPTID

- b) Used Table Name:
  - i. TBL\_EMPLOYEE\_PERSONAL\_INFO
  - ii. TBL\_DEPARTMENT
  - iii. TBL\_USER
- ❖ SEARCH Block
  - a) Item Name: EMPCODE, USERID, USERTYPE, SEARCH\_BTN
  - b) Used Table Name:
    - i. TBL\_EMPLOYEE\_PERSONAL\_INFO
    - ii. TBL\_USER
    - iii. TBL\_USERTYPE
- ❖ DISPLAY Block
  - a) Item Name: EMPCODE, USERTYPENAME, USERLOGPASSWORD, USERLOGID, DELETE\_BTTN
- ❖ ACCOUNT\_UNLOCK Block:
  - a) Item Name: USERNAME, ACCOUNT\_UNLOCK\_BTN

### 3) FMB\_USER\_ROLE

Formdescription	Read	Insert	Update	Delete
Company Setup				
Shift Setup				
Shift Time Setup				
Leave Type Entry				
Employee Setup				
New Employee Entry				
Leave Entry				
Data Collection & Process				
Manual Entry				
Daily Report				

#### BLOCK NAME:

- TBL\_FORM
- CONTROL
- List of LOVS used in FMB\_USER\_ROLE
  - a) USERTYPE Lov used in CONTROL block USERTYPENAME Text box and fetch value USERTYPENAME from TBL\_USERTYPE and display into USERTYPENAME
  - b) USERNAME Lov used in CONTROL block USERNAME\_TO Text box and fetch value USERNAME from TBL\_USER and display into USERNAME\_TO
- ❖ TBL\_FORM Block:
  - a) Item name: FORMDESCRIPTION, R,W,U,D,F
  - b) Used Table Name: TBL\_FORM, TBL\_USER\_ROLE\_PERMISSION
- ❖ CONTROL Block:
  - a) Item name: USERNAME, USERNAME\_TO, SAVE, COPY, CANCEL

b) Used Table Name: TBL\_FORM, TBL\_USER\_ROLE\_PERMISSION

#### 4) FMB\_USER\_ROLE\_UPDATE

Formdescription	Read	Insert	Update	Delete
<input type="checkbox"/> Create User Role	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/> Employee Setup	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/> Shift Time Setup	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/> Text File Generate	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/> Overtime Edit	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/> Monthly Report	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/> Weekly Holiday Setup	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/> Update User Role	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/> Form Entry	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/> Daily Report	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

#### BLOCK NAME:

- CONTROL
- TBL\_USER\_ROLE\_PERMISSION
- List of LOVS used in FMB\_USER\_ROLE
  - a) USER\_LOV or USER\_LOV\_DEPT Lov used in CONTROL block USERNAME Text box and fetch value USERNAME from TBL\_USER and display into USERNAM


#### ❖ CONTROL Block:

- a) Item Name: USERNAME, SAVE, CANCEL
- b) Used Table Name: TBL\_USER

#### ❖ TBL\_USER\_ROLE\_PERMISSION Block:

- a) Item name: FORMDESCRIPTION,F,R,I,U,D
- b) Used Table Name: TBL\_FORM, TBL\_USER\_ROLE\_PERMISSION

## 5) USER\_PASSWORD\_CHANGE



The screenshot shows a window titled "User Password Change Form". Inside the window, there is a form with the same title. The form contains five text input fields arranged vertically, each with a label to its left: "Log On Name", "User Name", "Type Previous Password", "New Password", and "ReType New Password". Below these fields, there are two buttons: "Update" and "Cancel".

### BLOCK NAME:

- TBL\_USER
- CONTROL
- List of LOVS used in FMB\_ USER\_ROLE
  - a) USER\_LOV or USER\_LOV\_DEPT Lov used in CONTROL block USERLOGID Text box and fetch value USERLOGID from TBL\_USER and display into USERLOGID Text box.
- ❖ TBL\_USER Block:
  - a) ItemName:USERLOGID,USERNAME,NEW\_PASSWORD\_1,NEW\_PASSWORD\_2, PREVIOUS\_PASSWORD
- ❖ CONTROL Block:
  - a) Item Name: MSG, RESET, CANCEL

## 6) USER\_ROLE

User Authority Form

User Authority Given Form

Empcode

User Name

User Type

Userlogid

Role Name

☐ Grant Role

Authority Given Y/N

Given Date 03-APR-2014 14:57:32

☐ Revoke Role

Revoke Y/N

Withdrawal Date

BLOCK NAME:

- TBL\_USER
  - List of LOVS used in FMB\_USER\_ROLE
    - a) USER\_LOV Lov used in TBL\_USER block EMPCODE Text box and fetch value EMPID, USERNAME, USERTYPENAME, USERLOGINID from TBL\_USER and display into EMPCODE Text box.
- ❖ TBL\_USER Block:
- a) Item Name: EMPCODE, USERTYPENAME, USERLOGID, ROLE\_NAME, ENTRY\_AUTHORITY, ENTRY\_DATE, REVOKE\_ROLE, WITHDRAWAL\_DATE, GRANT, CLEAR, REVOKE, EXIT, RADIO\_GROUP51
  - b) Used Table name: TBL\_USER