

## Alerts :

Alerts notify users about critical business events, such as pending approvals, exceptions, or tasks that need action.

alerts can be categorized into two main types: **Periodic Alerts** and **Event Alerts**

## Periodic Alerts

- **Definition:** These alerts are generated at regular intervals like reminder(e.g., daily, weekly, monthly).
- **Example:** An alert that notifies the finance team every month about overdue invoices.

## Event Alerts

- **Definition:** These alerts are triggered by specific actions or events in the system.(like db trigger)
- **Example:** An alert that is sent when a purchase order is approved or when inventory levels drop below a certain threshold.

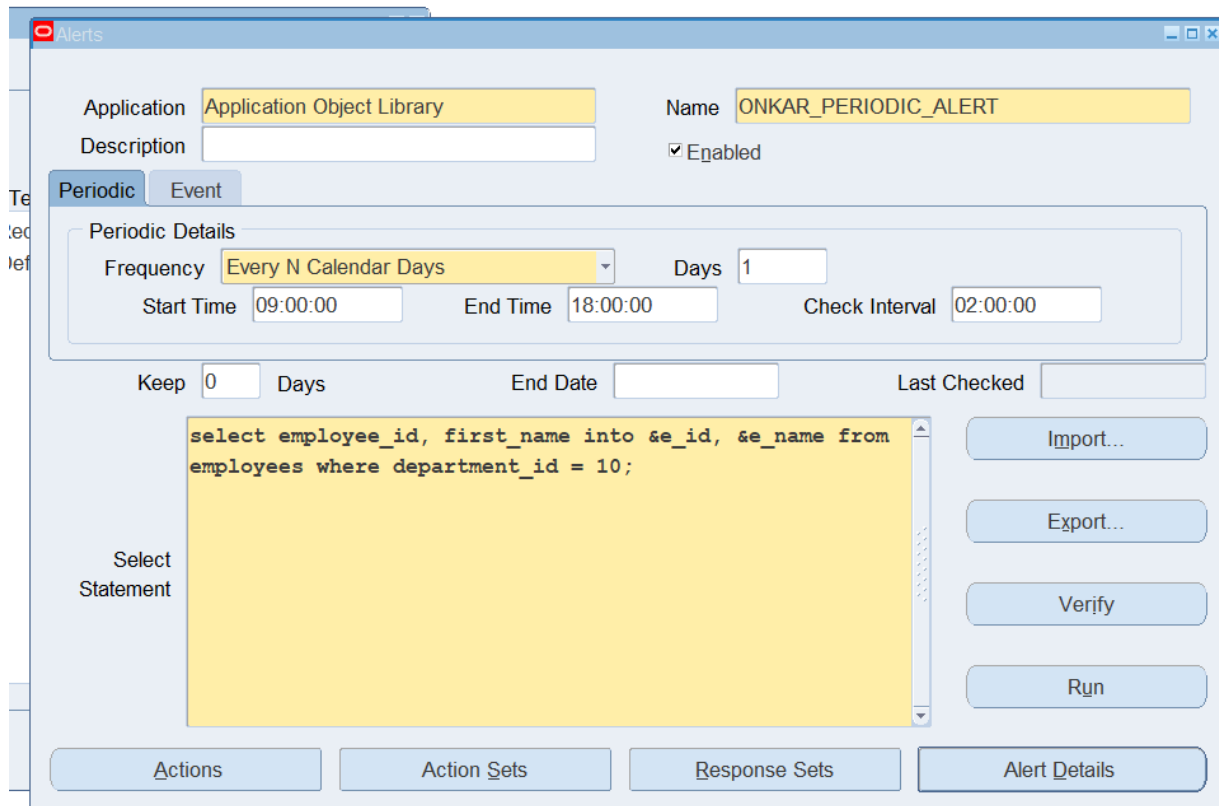
## 1. Periodic Alerts :

Sysadmin > alert manager, vision enterprise (resp)> define

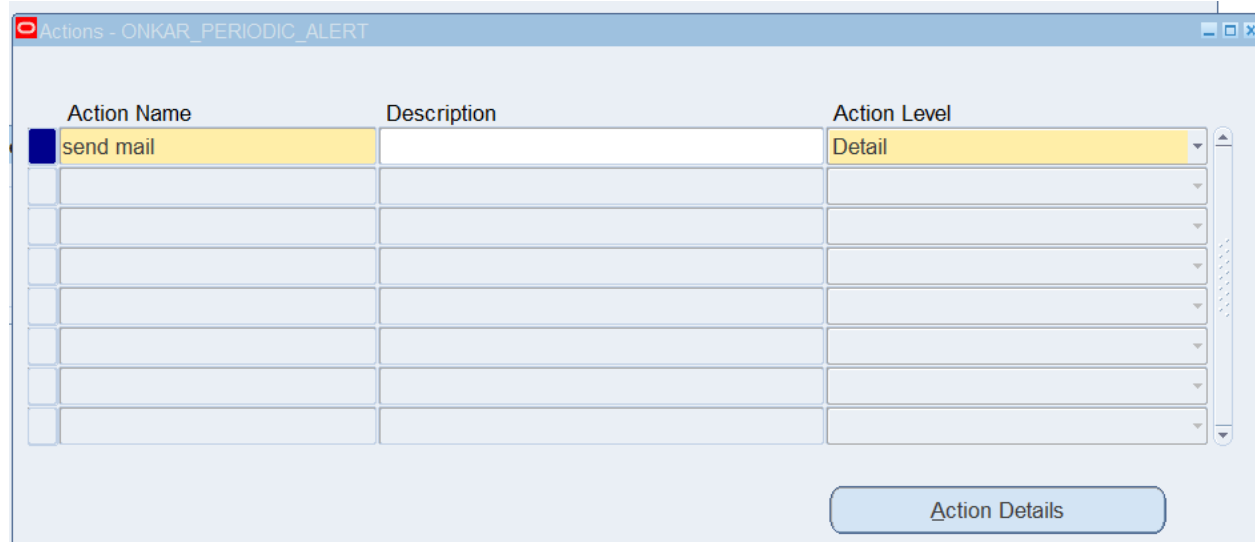
The screenshot displays the 'Alert Manager' interface with the 'Define' dialog open. The dialog is titled 'Alerts' and has tabs for 'Periodic' and 'Event'. The 'Periodic' tab is selected. The 'Periodic Details' section includes a 'Frequency' dropdown set to 'On Demand', 'Start Time', 'End Time', and 'Check Interval' fields. Below this, there are 'Keep' (0) 'Days', 'End Date', and 'Last Checked' fields. A large text area for the 'Select Statement' contains the SQL query: `SELECT COLUMN_NAME INTO &OUTPUT1 FROM TABLE_NAME`. To the right of the text area are buttons for 'Import...', 'Export...', 'Verify', and 'Run'. At the bottom of the dialog are buttons for 'Actions', 'Action Sets', 'Response Sets', and 'Alert Details'. The background shows the 'Alert Manager' window with a tree view on the left and a 'Top Tree' on the right.

FREQUENCY : ON DEMAND (FOR TESTING PURPOSE)

Every n day means all the days in calendar > now click on action



The Alerts configuration window for 'ONKAR\_PERIODIC\_ALERT' is shown. It includes fields for Application (Application Object Library), Name (ONKAR\_PERIODIC\_ALERT), and a checked 'Enabled' checkbox. The 'Periodic' tab is active, showing 'Frequency' as 'Every N Calendar Days' with 'Days' set to 1. 'Start Time' is 09:00:00, 'End Time' is 18:00:00, and 'Check Interval' is 02:00:00. Below these, 'Keep' is set to 0 days and 'End Date' is empty. A 'Select Statement' text area contains the SQL query: `select employee_id, first_name into &e_id, &e_name from employees where department_id = 10;`. To the right of the text area are buttons for 'Import...', 'Export...', 'Verify', and 'Run'. At the bottom are buttons for 'Actions', 'Action Sets', 'Response Sets', and 'Alert Details'.



The Actions configuration window for 'ONKAR\_PERIODIC\_ALERT' is shown. It contains a table with columns 'Action Name', 'Description', and 'Action Level'. The first row is highlighted with 'send mail' in the 'Action Name' column and 'Detail' in the 'Action Level' column. There are 8 rows in total. A vertical scrollbar is on the right. At the bottom right is an 'Action Details' button.

Action Name	Description	Action Level
send mail		Detail

Detail means in only one line while in summary it's like looping and summarize info is passed.

Alerts

Application: Application Object Library Name: ONKAR PERIODIC ALERT

Action Details - [New]

Action Type: Message

List: Reply To:

To: onkarsawant111@gmail.com

Subject: TESTING OF PERIODIC ALERT EBS

Cc:

Bcc:

Print For User: Printer:

Response Set: Response Days:

☐ File (C):

☒ Text (D):

HI TEAM,

&E\_ID | &E\_NAME

Column Overflow: Wrap Max Width:

Import...

SAVE IT.

NOW DEFINE ACTION SETS:

Alerts

Application: Application Object Library      Name: ONKAR\_PERIODIC\_ALERT

Description:      ☒ Enabled

Periodic Action Sets - ONKAR\_PERIODIC\_ALERT

Seq	Action Set Name	Description	Suppress Duplicates	Enabled	End Date
1	SEND MAIL		<input type="checkbox"/>	<input checked="" type="checkbox"/>	
			<input type="checkbox"/>	<input type="checkbox"/>	
			<input type="checkbox"/>	<input type="checkbox"/>	
			<input type="checkbox"/>	<input type="checkbox"/>	
			<input type="checkbox"/>	<input type="checkbox"/>	
			<input type="checkbox"/>	<input type="checkbox"/>	

Statement

Verify      Run

Actions      Action Sets      Response Sets      Alert Details

## ACTION SETS DETAILS > OUTPUT

Alerts

Application: Application Object Library      Name: ONKAR\_PERIODIC\_ALERT

Description:      ☒ Enabled

Periodic Action Sets - ONKAR\_PERIODIC\_ALERT

Seq	Action Set Name	Description	Suppress Duplicates	Enabled	End Date
1	SEND MAIL		<input type="checkbox"/>	<input checked="" type="checkbox"/>	

Action Set Details

Inputs      Outputs      **Members**

Seq	Action	Type	Summary Threshold	On Error Action	Seq	Enabled	End Date
1	send mail	Action: Message		Abort		<input checked="" type="checkbox"/>	
						<input type="checkbox"/>	
						<input type="checkbox"/>	
						<input type="checkbox"/>	
						<input type="checkbox"/>	

Actions      Action Sets      Response Sets      Alert Details

NOW GO TO ALERT DETAILS :

ADD OUR ORGANISATION DETAILS IN OPERATING UNIT :

The screenshot shows the 'Alert Details - ONKAR\_PERIODIC\_ALERT' window. The 'Installations' tab is active, displaying a table with columns: Oracle ID, Operating Unit, and Enabled. The first row is highlighted in yellow, showing 'APPS' in the Oracle ID column and a checked checkbox in the Enabled column. Below the table are buttons for 'Actions', 'Action Sets', 'Response Sets', and 'Alert Details'.

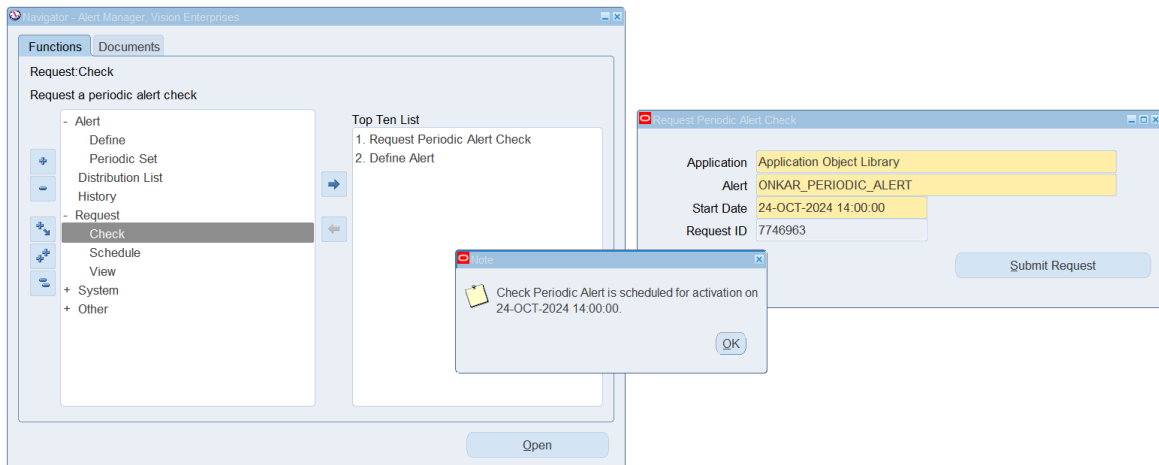
Oracle ID	Operating Unit	Enabled
APPS		<input checked="" type="checkbox"/>
		<input type="checkbox"/>
		<input type="checkbox"/>
		<input type="checkbox"/>
		<input type="checkbox"/>
		<input type="checkbox"/>
		<input type="checkbox"/>
		<input type="checkbox"/>
		<input type="checkbox"/>
		<input type="checkbox"/>

FIRST VERIFY AND THEN RUN .

The screenshot shows the 'Alerts' window with the 'Periodic' tab selected. A 'Note' dialog box is open, displaying the message: '1 rows selected (ROUTINE=aluvss) (FILE=22132029/alr/lib/aluvss.c) (LINE=344)'. The main window shows fields for 'Application' (Application Object Library), 'Name' (ONKAR\_PERIODIC\_ALERT), and 'Description'. Below these are tabs for 'Periodic' and 'Event'. The 'Periodic Details' section includes fields for 'Frequency' (Every N Calendar Days), 'Days' (1), 'Start Time' (09:00:00), 'End Time' (18:00:00), and 'Check Interval' (02:00:00). At the bottom are buttons for 'Import...', 'Export...', 'Verify', and 'Run'.

Application: Application Object Library  
Name: ONKAR\_PERIODIC\_ALERT  
Description:   
Enabled: ☒  
Periodic Details:  
Frequency: Every N Calendar Days  
Days: 1  
Start Time: 09:00:00  
End Time: 18:00:00  
Check Interval: 02:00:00  
Keep: 0 Days  
End Date:   
Last Checked:   
Note: 1 rows selected (ROUTINE=aluvss) (FILE=22132029/alr/lib/aluvss.c) (LINE=344)  
Buttons: Import..., Export..., Verify, Run

ALERT > REQUEST > CHECK



NOW GO TO VIEW> REQUEST CHECK THE ALERT

Requests

Refresh Data Find Requests Submit a New Request Submit New Request Set

☐ Auto Refresh (X) Copy Single Request Copy Request Set

Request ID	Name	Parent	Phase	Status	Parameters
7746963	ONKAR_PERIODIC_ALERT		Completed	Normal	0, 112533, A
7746962	OAM Applications Dashboa		Pending	Scheduled	(None)
7746961	OAM Applications Dashboa		Completed	Normal	(None)
7746960	OAM Applications Dashboa		Completed	Normal	(None)
7746959	OAM Applications Dashboa		Completed	Normal	(None)
7746958	OAM Applications Dashboa		Completed	Normal	(None)
7746957	OAM Applications Dashboa		Completed	Normal	(None)
7746956	OAM Applications Dashboa		Completed	Normal	(None)
7746955	OAM Applications Dashboa		Completed	Normal	(None)
7746954	OAM Applications Dashboa		Completed	Normal	(None)

Hold Request View Details Rerun Request View Output

Cancel Request Diagnostics Reprint/Republish (J) View Log (K)

## 2. Event based alerts :

SYSADMIN > ALERT MANAGER > DEFINE

Alerts

Applicationxxdcs custom app

Description

NameONKAR\_EVENT\_ALERT

☒ Enabled

Periodic

Event

Event Details

Applicationxxdcs custom app

TableXXDCS\_EMP\_DTLS

☒ After Insert (B)

☒ After Update

Keep0

Days

End Date

Last Checked

Select Statement

```
SELECT FIRST_NAME, EMPLOYEE_ID, SALARY INTO &E_NAME,
&E_NO, &E_GRADE FROM XXDCS_EMP_DTLS A WHERE 1=1 AND A.
ROWID = :ROWID;
```

Import...

Export...

Verify

Run

Actions

Action Sets

Response Sets

Alert Details

Alerts

Actions - ONKAR\_EVENT\_ALERT

Action Name	Description	Action Level
<input checked="" type="checkbox"/> SENDING MAIL		Detail
<input type="checkbox"/>		
<input type="checkbox"/>		
<input type="checkbox"/>		
<input type="checkbox"/>		
<input type="checkbox"/>		
<input type="checkbox"/>		
<input type="checkbox"/>		

Action Details

Verify

Run

Actions

Action Sets

Response Sets

Alert Details





Alerts

Action Sets - ONKAR\_EVENT\_ALERT

Action Set Details

Inputs Outputs Members

Seq	Action	Type	Summary Threshold	On Error Action	Seq	Enabled	End Date
1	SENDING MAIL	Action: Message		Abort		<input checked="" type="checkbox"/>	
						<input type="checkbox"/>	
						<input type="checkbox"/>	
						<input type="checkbox"/>	
						<input type="checkbox"/>	

Statement

ebsdb

Worksheet Query Builder

```

SELECT * FROM FND_TABLES
WHERE TABLE_NAME = 'XXDCS_EMP_DTLS';

SELECT * FROM fnd_application;
SELECT * FROM fnd_responsibility;

SELECT user_id, user_name
FROM fnd_user
WHERE user_name = 'SYSADMIN';

SELECT application_id, APPLICATION_SHORT_NAME
FROM fnd_application
WHERE APPLICATION_SHORT_NAME LIKE 'XX%';

SELECT responsibility_id, responsibility_key
FROM fnd_responsibility
WHERE responsibility_key LIKE '%ALERT%';

SELECT * FROM XXDCS_EMP_DTLS;

BEGIN
fnd_global.apps_initialize(55555,20456,0);
INSERT INTO XXDCS_EMP_DTLS (EMPLOYEE_ID, FIRST_NAME, EMAIL, SALARY, DEPARTMENT_ID) VALUES (1000, 'ONKAR', 'ONKAR@', 25000, 10);
COMMIT;
END;

```

Script Output x Query Result x

SQL | All Rows Fetched: 4 in 0.022 seconds

	EMPLOYEE_ID	FIRST_NAME	EMAIL	SALARY	DEPARTMENT_ID
1	101	ORACLE	ORACL123	120000	10
2	102	Diya	DIYAL123	100000	20
3	103	Ram	Ram@rty	90000	50
4	1000	ONKAR	ONKAR@	25000	10

### **CODE FOR CREATING CUSTOM APP AND CUSTOM TABLE :**

```
CREATE TABLE XXDCS_EMP_DTLS AS SELECT EMPLOYEE_ID, FIRST_NAME, EMAIL,  
SALARY, DEPARTMENT_ID FROM EMPLOYEES WHERE 1=0;
```

```
SELECT * FROM XXDCS_EMP_DTLS;
```

```
--REGISTER TABLE
```

```
DECLARE
```

```
CURSOR C1 IS
```

```
SELECT (select APPLICATION_SHORT_NAME from fnd_application where APPLICATION_ID  
=
```

```
(select APPLICATION_ID from fnd_application_tl where APPLICATION_NAME = 'xxdcs  
custom app')) APP_SHORT_NAME,
```

```
A.OBJECT_NAME, 'T' TABLE_T, A.OWNER
```

```
FROM DBA_OBJECTS A
```

```
WHERE 1=1
```

```

AND TRIM(A.OBJECT_NAME) NOT IN (SELECT TABLE_NAME FROM FND_TABLES)
AND A.OBJECT_NAME LIKE 'XXDCS_EMP_DTLS'
--AND A.OBJECT_TYPE NOT IN ('SYNONYM')
;

BEGIN

FOR I IN C1 LOOP

    AD_DD.REGISTER_TABLE (I.APP_SHORT_NAME,I.OBJECT_NAME,I.TABLE_T);

END LOOP;

END;


SELECT * FROM FND_TABLES
WHERE TABLE_NAME ='XXDCS_EMP_DTLS';


--REGISTER COLUMNS

DECLARE

CURSOR C1 IS

SELECT (SELECT APPLICATION_SHORT_NAME FROM FND_APPLICATION WHERE
APPLICATION_ID =

```

```
(SELECT APPLICATION_ID FROM FND_APPLICATION_TL WHERE APPLICATION_NAME =  
'xxdcs custom app')) APP_SHORT_NAME,
```

```
TABLE_NAME,COLUMN_NAME,DATA_TYPE,DATA_LENGTH,COLUMN_ID,NOT_NULLABLE,OWN  
ER
```

```
FROM DBA_TAB_COLUMNS
```

```
WHERE TABLE_NAME IN ( SELECT OBJECT_NAME FROM DBA_OBJECTS A WHERE  
A.OBJECT_NAME LIKE 'XXDCS_EMP_DTLS')
```

```
AND TABLE_NAME NOT IN (SELECT TABLE_NAME FROM FND_TABLES WHERE TABLE_ID  
IN (SELECT TABLE_ID FROM FND_COLUMNS));
```

```
BEGIN
```

```
FOR I IN C1 LOOP
```

```
AD_DD.REGISTER_COLUMN (I.APP_SHORT_NAME, I.TABLE_NAME,I.COLUMN_NAME,  
I.COLUMN_ID, I.DATA_TYPE, I.DATA_LENGTH, 'N', 'N');
```

```
END LOOP;
```

```
END;
```

```
SELECT * FROM FND_COLUMNS
```

```
WHERE TABLE_ID =(SELECT TABLE_ID FROM FND_TABLES
```

```
WHERE TABLE_NAME ='XXDCS_EMP_DTLS');
```

--REGISTER PRIMARY KEY

DECLARE

CURSOR C1 IS

SELECT (SELECT APPLICATION\_SHORT\_NAME FROM FND\_APPLICATION WHERE  
APPLICATION\_ID =

(SELECT APPLICATION\_ID FROM FND\_APPLICATION\_TL WHERE APPLICATION\_NAME =  
'xxdcs custom app')) APP\_SHORT\_NAME,

TABLE\_NAME,COLUMN\_NAME,DATA\_TYPE,DATA\_LENGTH,COLUMN\_ID,NULLABLE,OWN  
ER

FROM DBA\_TAB\_COLUMNS

WHERE TABLE\_NAME IN ( SELECT OBJECT\_NAME FROM DBA\_OBJECTS A WHERE  
A.OBJECT\_NAME LIKE 'XXDCS\_EMP\_DTLS')

AND TABLE\_NAME NOT IN (SELECT TABLE\_NAME FROM FND\_TABLES WHERE TABLE\_ID  
IN (SELECT TABLE\_ID FROM FND\_PRIMARY\_KEYS))

AND COLUMN\_NAME = 'EMPLOYEE\_ID'

;

BEGIN

FOR I IN C1 LOOP

```
AD_DD.REGISTER_PRIMARY_KEY(I.APP_SHORT_NAME,I.COLUMN_NAME,I.TABLE_NAME,I
.DATA_TYPE,'S','Y','Y');
```

```
AD_DD.REGISTER_PRIMARY_KEY_COLUMN(I.APP_SHORT_NAME, I.COLUMN_NAME,
I.TABLE_NAME, I.COLUMN_NAME, 1);
```

```
END LOOP;
```

```
END;
```

```
--Check our event base alert using this query--
```

```
-----  
SELECT REQUEST_ID
```

```
, REQUEST_DATE
```

```
, PHASE_CODE
```

```
, STATUS_CODE
```

```
, ORACLE_ID
```

```
, PROGRAM_APPLICATION_ID
```

```
, CONCURRENT_PROGRAM_ID
```

```
FROM APPLSYS.FND_CONCURRENT_REQUESTS
```

```
where Concurrent_program_id = (select Concurrent_program_id from  
fnd_concurrent_programs_tl where
```

```
USER_CONCURRENT_PROGRAM_NAME = 'Check Event Alert') order by request_date  
desc;
```

