

```
SELECT * FROM wards;
```

The screenshot shows the Oracle SQL Developer interface. The title bar reads "Oracle SQL Developer : C:\Users\HP\EmergencyTracker\_PDB\_PhaseV\Testing\_queries.sql". The menu bar includes File, Edit, View, Navigate, Run, Source, Team, Tools, Window, and Help. The toolbar has various icons for file operations and database management. On the left is the "Connections" sidebar, which lists "auca" and "EmergencyTracker\_PDB", with "EmergencyTracker\_PDB" expanded to show Tables (ADMISSIONS, BEDS, PATIENTS, WARDS), Views, Indexes, Packages, Procedures, Functions, Operators, Queues, Queues Tables, Triggers, Types, Sequences, and Materialized Views. The main area has tabs for "EmergencyTracker\_PDB.sql", "Insert\_data.sql", "Verification\_queries.sql", and "Testing\_queries.sql". The "Worksheet" tab contains the SQL query: "SELECT \* FROM wards;". Below it is the "Query Result" tab, which displays the results of the query:

WARD_ID	WARD_NAME	TOTAL_BEDS
1	1 Ward_1	11
2	2 Ward_2	12
3	3 Ward_3	13
4	4 Ward_4	14
5	5 Ward_5	15
6	6 Ward_6	16
7	7 Ward_7	17
8	8 Ward_8	18
9	9 Ward_9	19
10	10 Ward_10	20

The status bar at the bottom of the result tab says "All Rows Fetched: 10 in 0.003 seconds".

```
SELECT * FROM beds;
```

Oracle SQL Developer : C:\Users\HP\EmergencyTracker\_PDB\_PhaseV\Testing\_queries.sql

File Edit View Navigate Run Source Team Tools Window Help

Connections

EmergencyTracker\_PDB

Tables (Filtered)

- ADMISSIONS
- BEDS
- PATIENTS
- WARDS

Views

Indexes

Packages

Procedures

Functions

Operators

Queues

Queues Tables

Triggers

Types

Sequences

Materialized Views

Materialized View Logs

Synonyms

Public Synonyms

Database Links

Public Database Links

Directories

Editions

Java

XML Schemas

XML DB Repository

OLAP Option

Worksheet Query Builder

```
SELECT * FROM wards;  
SELECT * FROM beds;
```

Query Result x

SQL | Fetched 50 rows in 0.005 seconds

BED_ID	WARD_ID	STATUS
1	1	2 Occupied
2	2	3 Occupied
3	3	4 Available
4	4	5 Occupied
5	5	6 Occupied
6	6	7 Available
7	7	8 Occupied
8	8	9 Occupied
9	9	10 Available
10	10	1 Occupied
11	11	2 Occupied
12	12	3 Available
13	13	4 Occupied
14	14	5 Occupied
15	15	6 Available
16	16	7 Occupied
17	17	8 Occupied
18	18	9 Available

```
SELECT * FROM patients;
```

Oracle SQL Developer : C:\Users\HP\EmergencyTracker\_PDB\_PhaseV\Testing\_queries.sql

File Edit View Navigate Run Source Team Tools Window Help

Connections

EmergencyTracker\_PDB

Tables (Filtered)

- ADMISSIONS
- BEDS
- PATIENTS
- WARDS

Views

Indexes

Packages

Procedures

Functions

Operators

Queues

Queues Tables

Triggers

Types

Sequences

Materialized Views

Materialized View Logs

Synonyms

Public Synonyms

Database Links

Public Database Links

Directories

Editions

Java

XML Schemas

XML DB Repository

OLAP Option

Worksheet Query Builder

```
SELECT * FROM wards;
SELECT * FROM beds;
SELECT * FROM patients;
```

Query Result x

SQL | Fetched 50 rows in 0.004 seconds

PATIENT_ID	FULL_NAME	CATEGORY
1	1 Patient_1	Regular
2	2 Patient_2	Regular
3	3 Patient_3	Regular
4	4 Patient_4	Emergency
5	5 Patient_5	Regular
6	6 Patient_6	Regular
7	7 Patient_7	Regular
8	8 Patient_8	Emergency
9	9 Patient_9	Regular
10	10 Patient_10	Regular
11	11 Patient_11	Regular
12	12 Patient_12	Emergency
13	13 Patient_13	Regular
14	14 Patient_14	Regular
15	15 Patient_15	Regular
16	16 Patient_16	Emergency

```
SELECT * FROM admissions;
```

The screenshot shows the Oracle SQL Developer interface. The 'Connections' sidebar on the left lists 'auca' and 'EmergencyTracker\_PDB'. Under 'EmergencyTracker\_PDB', the 'Tables (Filtered)' section contains 'ADMISSIONS', 'BEDS', 'PATIENTS', and 'WARDS'. The 'Worksheet' tab in the center contains the following SQL code:

```
SELECT * FROM wards;
SELECT * FROM beds;
SELECT * FROM patients;
SELECT * FROM admissions;
```

The 'Query Result' tab at the bottom shows the output of the last query, which is a table with 12 rows of data:

ADMISSION_ID	PATIENT_ID	BED_ID	ADMISSION_DATE	DISCHARGE_DATE
1	1	2	2 17-DEC-25	(null)
2	2	3	3 16-DEC-25	(null)
3	3	4	4 15-DEC-25	(null)
4	4	5	5 14-DEC-25	(null)
5	5	6	6 13-DEC-25	(null)
6	6	7	7 12-DEC-25	(null)
7	7	8	8 11-DEC-25	(null)
8	8	9	9 10-DEC-25	(null)
9	9	10	10 09-DEC-25	(null)
10	10	11	11 08-DEC-25	(null)
11	11	12	12 07-DEC-25	(null)
12	12	13	13 06-DEC-25	(null)

## Indexes

The screenshot shows the Oracle SQL Developer interface. The 'Connections' sidebar on the left lists 'auca' and 'EmergencyTracker\_PDB'. Under 'EmergencyTracker\_PDB', the 'Indexes' section contains 'IDX\_BEDS\_WARD', 'PK\_ADMISSIONS', 'PK\_BEDS', 'PK\_PATIENTS', 'PK\_WARDS', and 'SYS\_C008254'. The 'IDX\_BEDS\_WARD' entry is selected, and the 'Actions...' button is highlighted. The 'Columns' tab in the center displays the following table:

INDEX_OWNER	INDEX_NAME	TABLE_OWNER	TABLE_NAME	COLUMN_NAME	COLUMN_POSITION	DESCEND
PDB_ADMIN	IDX_BEDS_WARD	PDB_ADMIN	BEDS	WARD_ID	1	ASC

## Join Query

Oracle SQL Developer : EmergencyTracker\_PDB (Unshared)\_3

File Edit View Navigate Run Source Team Tools Window Help

Connections

EmergencyTracker\_PDB

Tables (Filtered)

- ADMISSIONS
- BEDS
- PATIENTS
- WARDS

Views

Indexes

Packages

Procedures

Functions

Operators

Queues

Queues Tables

Triggers

Types

Sequences

Materialized Views

Materialized View Logs

Synonyms

Public Synonyms

Database Links

Directories

Editions

Java

XML Schemas

XML DB Repository

OLAP Option

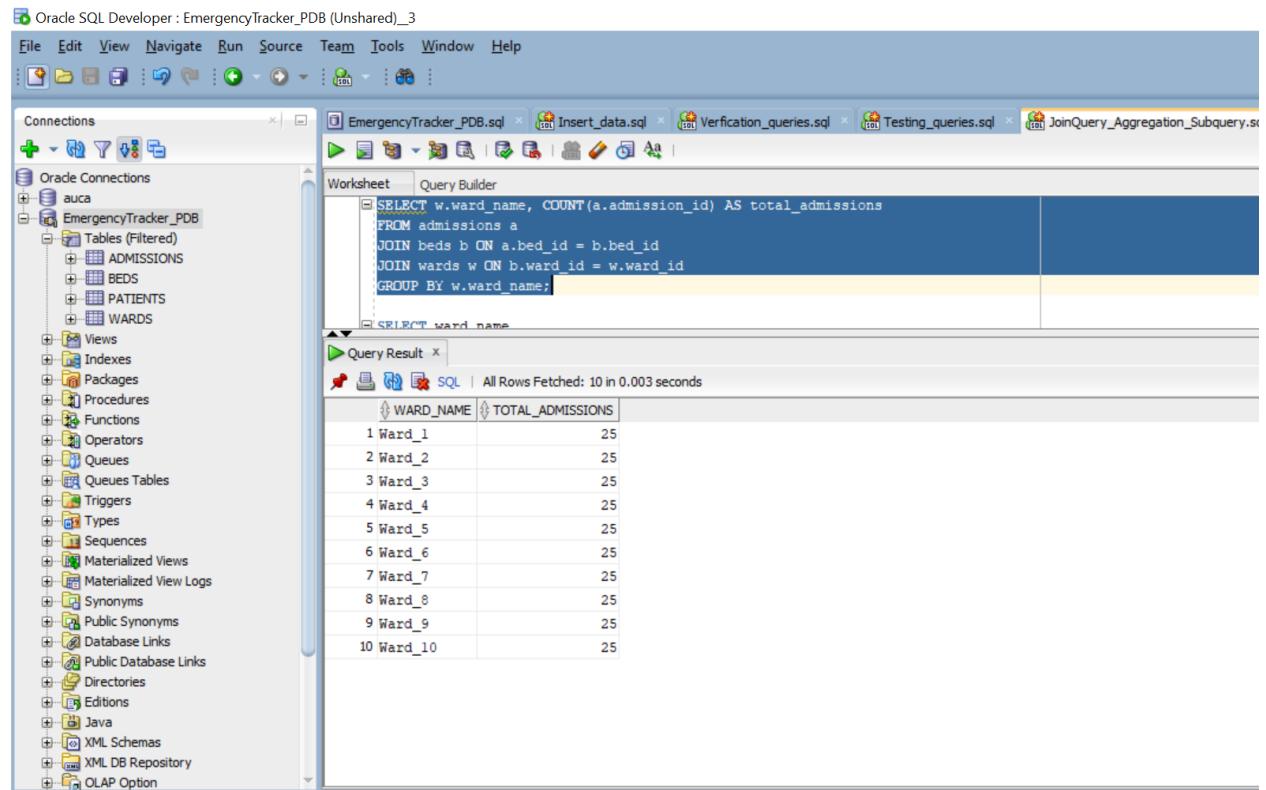
Worksheet Query Builder

```
SELECT p.full_name, w.ward_name, b.status
FROM admissions a
JOIN patients p ON a.patient_id = p.patient_id
JOIN beds b ON a.bed_id = b.bed_id
JOIN wards w ON b.ward_id = w.ward_id;
```

Query Result Fetched 50 rows in 0.004 seconds

FULL_NAME	WARD_NAME	STATUS
Patient_10	Ward_1	Occupied
Patient_20	Ward_1	Occupied
Patient_30	Ward_1	Available
Patient_40	Ward_1	Occupied
Patient_50	Ward_1	Occupied
Patient_60	Ward_1	Available
Patient_70	Ward_1	Occupied
Patient_80	Ward_1	Occupied
Patient_90	Ward_1	Available
Patient_100	Ward_1	Occupied
Patient_110	Ward_1	Occupied
Patient_120	Ward_1	Available
Patient_130	Ward_1	Occupied
Patient_140	Ward_1	Occupied
Patient_150	Ward_1	Available
Patient_160	Ward_1	Occupied

## Aggregation



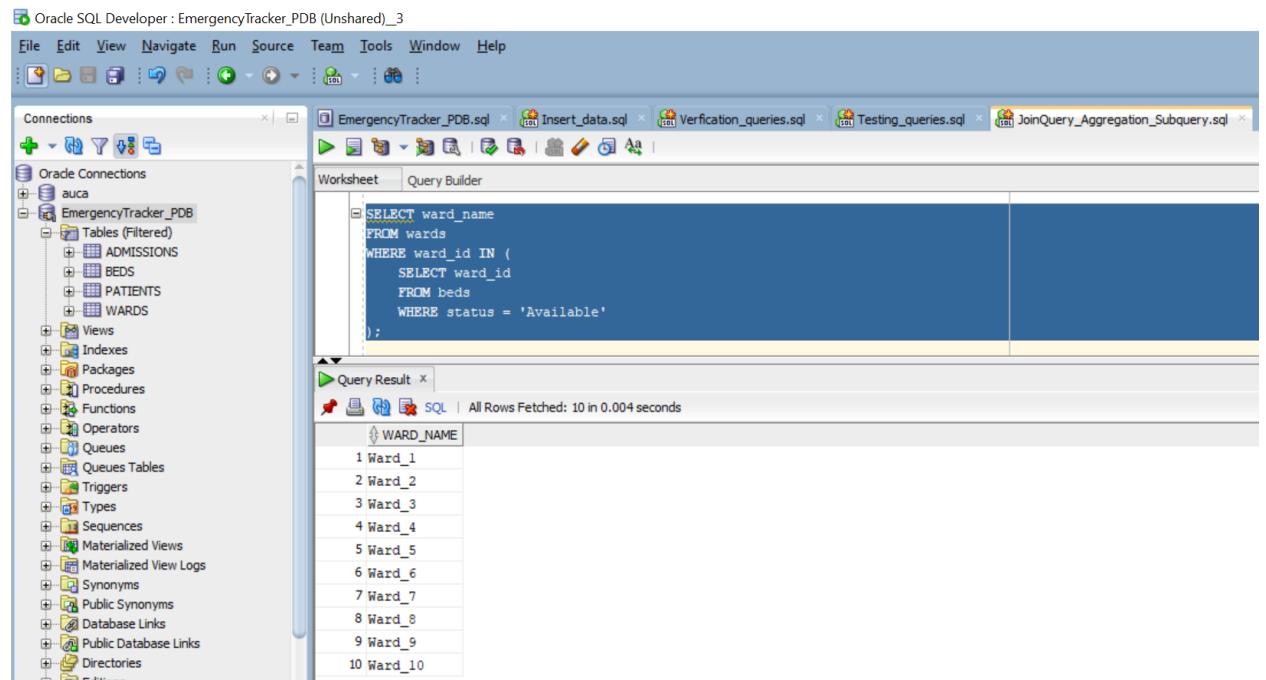
The screenshot shows the Oracle SQL Developer interface. The left pane displays the database connections and schema for 'EmergencyTracker\_PDB'. The right pane shows a worksheet with the following SQL query:

```
SELECT w.ward_name, COUNT(a.admission_id) AS total_admissions
FROM admissions a
JOIN beds b ON a.bed_id = b.bed_id
JOIN wards w ON b.ward_id = w.ward_id
GROUP BY w.ward_name;
```

The query result is displayed in the 'Query Result' tab, showing the following data:

WARD_NAME	TOTAL_ADMISSE
1 Ward_1	25
2 Ward_2	25
3 Ward_3	25
4 Ward_4	25
5 Ward_5	25
6 Ward_6	25
7 Ward_7	25
8 Ward_8	25
9 Ward_9	25
10 Ward_10	25

## Subquery



The screenshot shows the Oracle SQL Developer interface. The left pane displays the database connections and schema for 'EmergencyTracker\_PDB'. The right pane shows a worksheet with the following SQL query:

```
SELECT ward_name
FROM wards
WHERE ward_id IN (
    SELECT ward_id
    FROM beds
    WHERE status = 'Available'
);
```

The query result is displayed in the 'Query Result' tab, showing the following data:

WARD_NAME
1 Ward_1
2 Ward_2
3 Ward_3
4 Ward_4
5 Ward_5
6 Ward_6
7 Ward_7
8 Ward_8
9 Ward_9
10 Ward_10