

About the Presentations

- The presentations cover the objectives found in the opening of each chapter.
- All chapter objectives are listed in the beginning of each presentation.
- You may customize the presentations to fit your class needs.
- Some figures from the chapters are included. A complete set of images from the book can be found on the Instructor Resources disc.

Oracle 12c: SQL

Chapter 6

Additional Database Objects

Objectives

- Define the purpose of a sequence and state how it can be used in a database
- Explain why gaps may appear in the integers generated by a sequence
- Use the `CREATE SEQUENCE` command to create a sequence
- Call and use sequence values
- Identify which options cannot be changed by the `ALTER SEQUENCE` command
- Delete a sequence

Objectives (continued)

- Create indexes with the CREATE INDEX command
- Explain the main index structures: B-tree and bitmap
- Verify index use with the explain plan
- Introduce variations on conventional indexes, including a function-based index and an index organized table

Objectives (continued)

- Verify index existence via the data dictionary
- Rename an index with the ALTER INDEX command
- Remove an index using the DELETE INDEX command
- Create and remove a public synonym

Database Objects

- An object is anything that has a name and defined structure
- Includes:
 - Table – stores data
 - Sequence – generates sequential integers
 - Index – allows users to quickly locate specific records
 - Synonym – alias for other database objects

Sequences

- Used for internal control purposes by providing sequential integers for auditing
- Used to generate unique value for primary key column
 - Surrogate key = no correlation with actual row contents

Creating a Sequence

- Use the CREATE SEQUENCE command
- Various intervals are allowed – Default: 1
- You can specify the starting number – Default: 1

```
CREATE SEQUENCE sequencename  
[INCREMENT BY value]  
[START WITH value]  
[{MAXVALUE value | NOMAXVALUE}]  
[{MINVALUE value | NOMINVALUE}]  
[{CYCLE | NOCYCLE}]  
[{ORDER | NOORDER}]  
[{CACHE value | NOCACHE}];
```


Creating a Sequence (continued)

- Can specify MINVALUE for decreasing sequence and MAXVALUE for increasing sequence
- Numbers can be reused if CYCLE is specified
- ORDER clause is used in application cluster environment
- Use CACHE to pregenerate integers – Default: 20

Creating a Sequence (continued)

Enter SQL Statement:

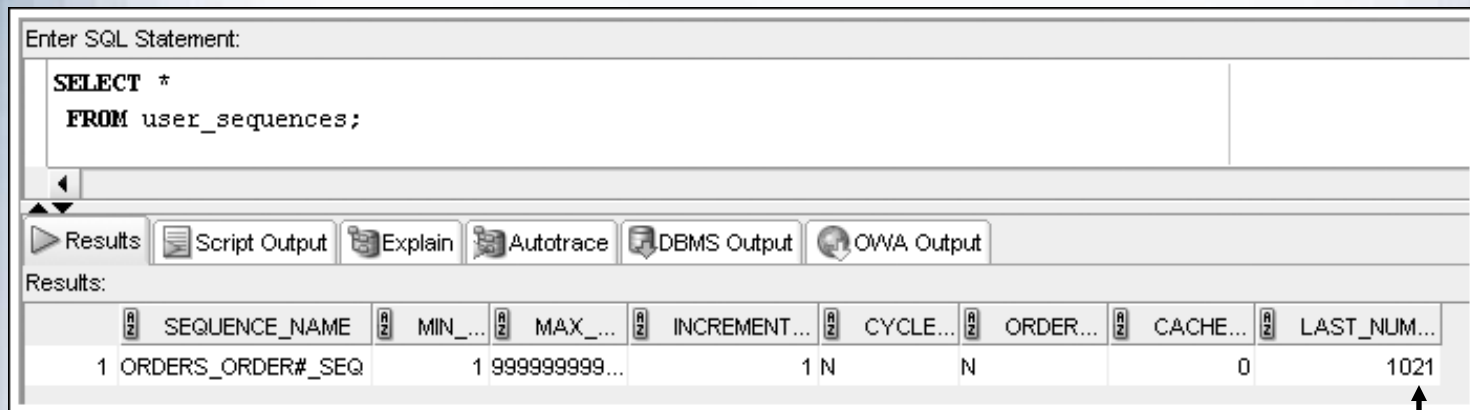
```
CREATE SEQUENCE orders_order#_seq  
  INCREMENT BY 1  
  START WITH 1021  
  NOCACHE  
  NOCYCLE;
```

Results Script Output Explain Autotrace DBMS Output OWA Output

CREATE SEQUENCE succeeded.

Creating a Sequence (continued)

- To verify the settings for options of a sequence, query USER_SEQUENCES data dictionary view



The screenshot shows a SQL query interface with the following components:

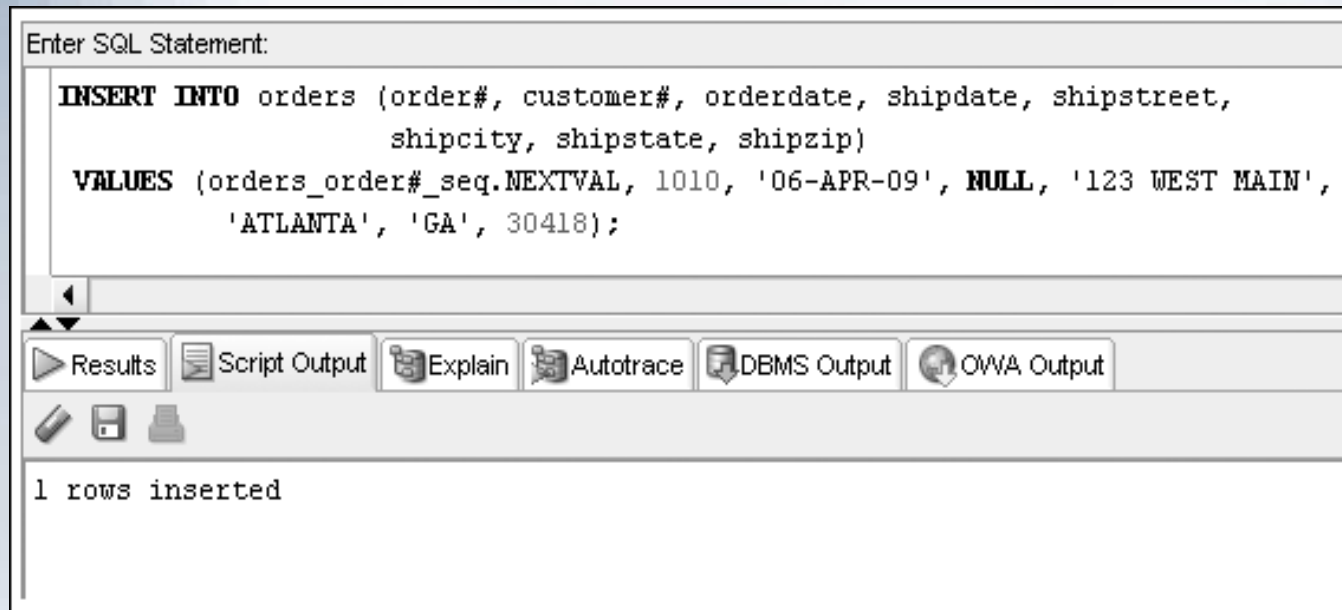
- Enter SQL Statement:** A text area containing the query: `SELECT * FROM user_sequences;`
- Results:** A tabbed interface with buttons for Results, Script Output, Explain, Autotrace, DBMS Output, and OWA Output. The Results tab is selected.
- Results:** A table displaying the results of the query. The table has 10 columns: SEQUENCE_NAME, MIN_..., MAX_..., INCREMENT..., CYCLE..., ORDER..., CACHE..., and LAST_NUM... (the last column is truncated). The first row of data shows the sequence 'ORDERS_ORDER#_SEQ' with a minimum value of 1, a maximum value of 999999999, an increment of 1, and a last number of 1021.

	SEQUENCE_NAME	MIN_...	MAX_...	INCREMENT...	CYCLE...	ORDER...	CACHE...	LAST_NUM...
1	ORDERS_ORDER#_SEQ	1	999999999...	1	N	N	0	1021

Next Number to issue

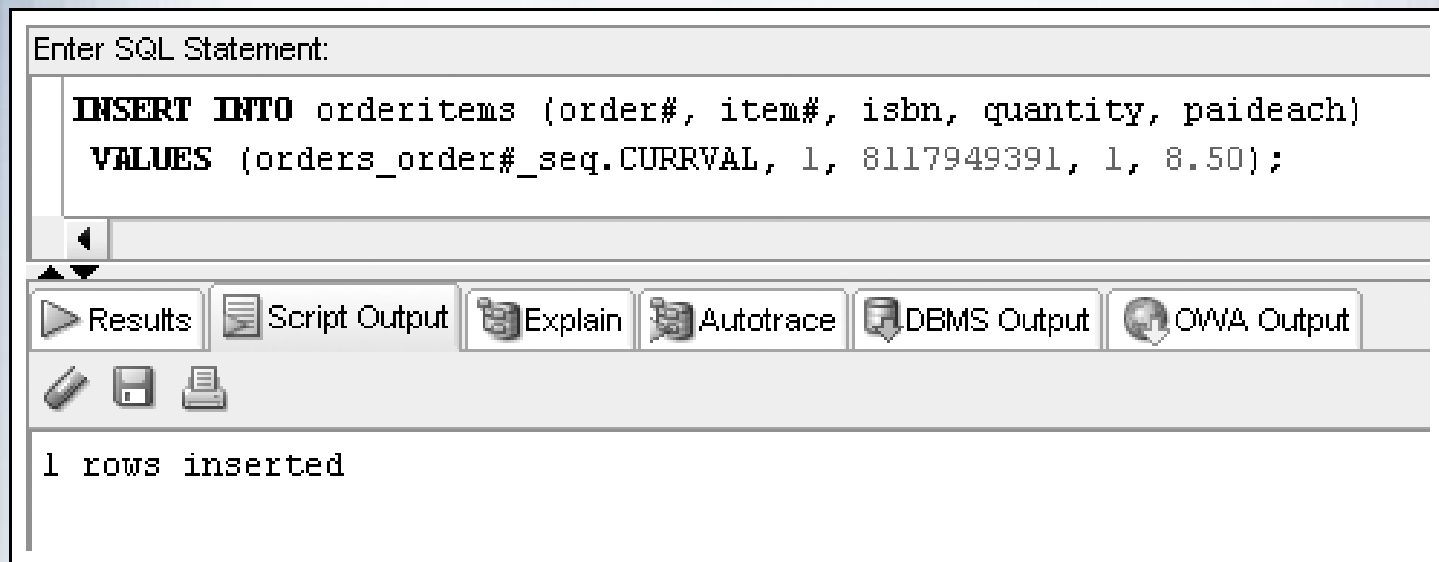
Using Sequence Values

- NEXTVAL – generates integer



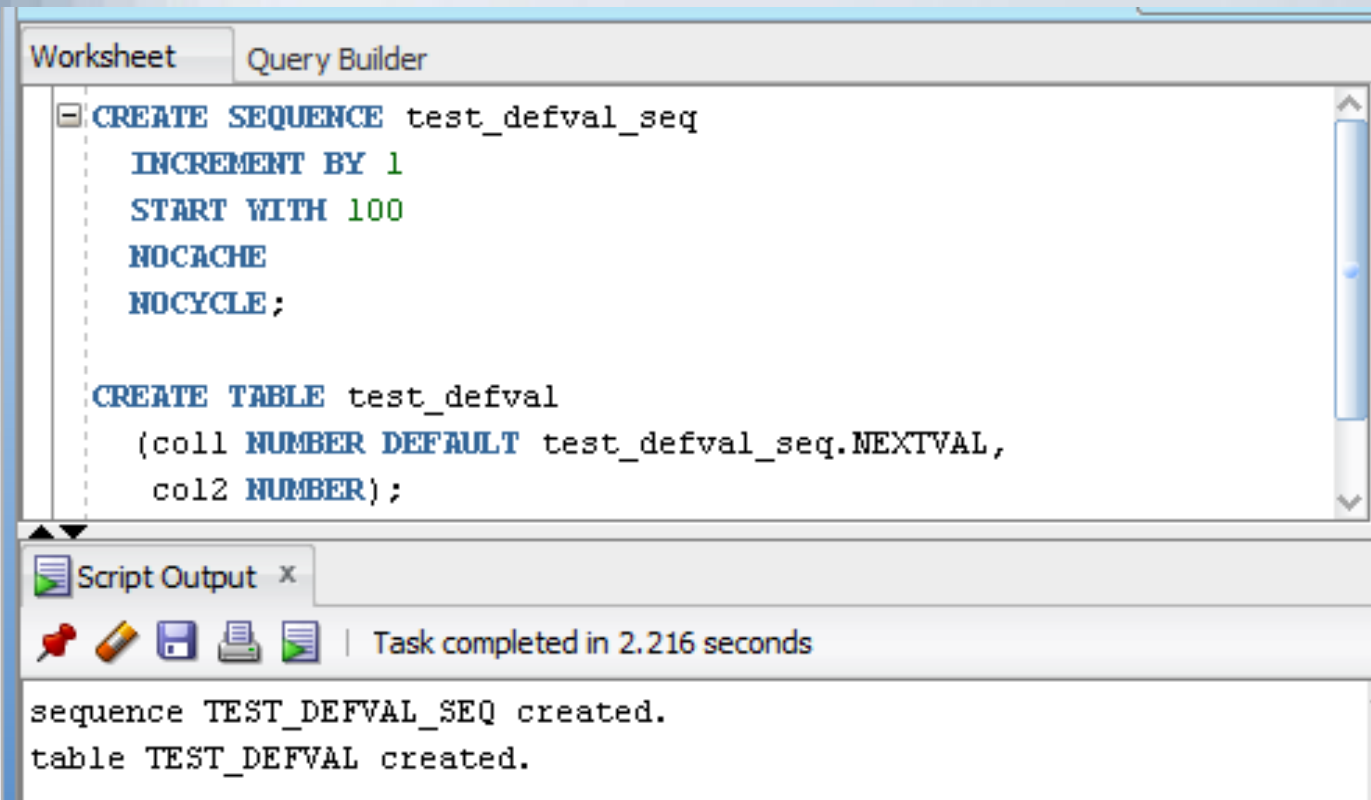
Using Sequence Values (continued)

- CURRVAL – contains last integer generated by NEXTVAL



Using Sequence Values (continued)

- Set column DEFAULT value



Altering Sequence Definitions

- Use ALTER SEQUENCE command to change the settings for a sequence
- START WITH value cannot be altered – drop the sequence and re-create it
- Changes cannot make current integers invalid

ALTER SEQUENCE Command Example

Enter SQL Statement:

```
ALTER SEQUENCE orders_order#_seq  
INCREMENT BY 10;
```

Results Script Output Explain Autotrace DBMS Output OWA Output

ALTER SEQUENCE orders_order#_seq succeeded.

Removing a Sequence

- Use the DROP SEQUENCE command to delete a sequence
- Previous values generated are not affected by removing a sequence from a database

Removing a Sequence (continued)

Enter SQL Statement:

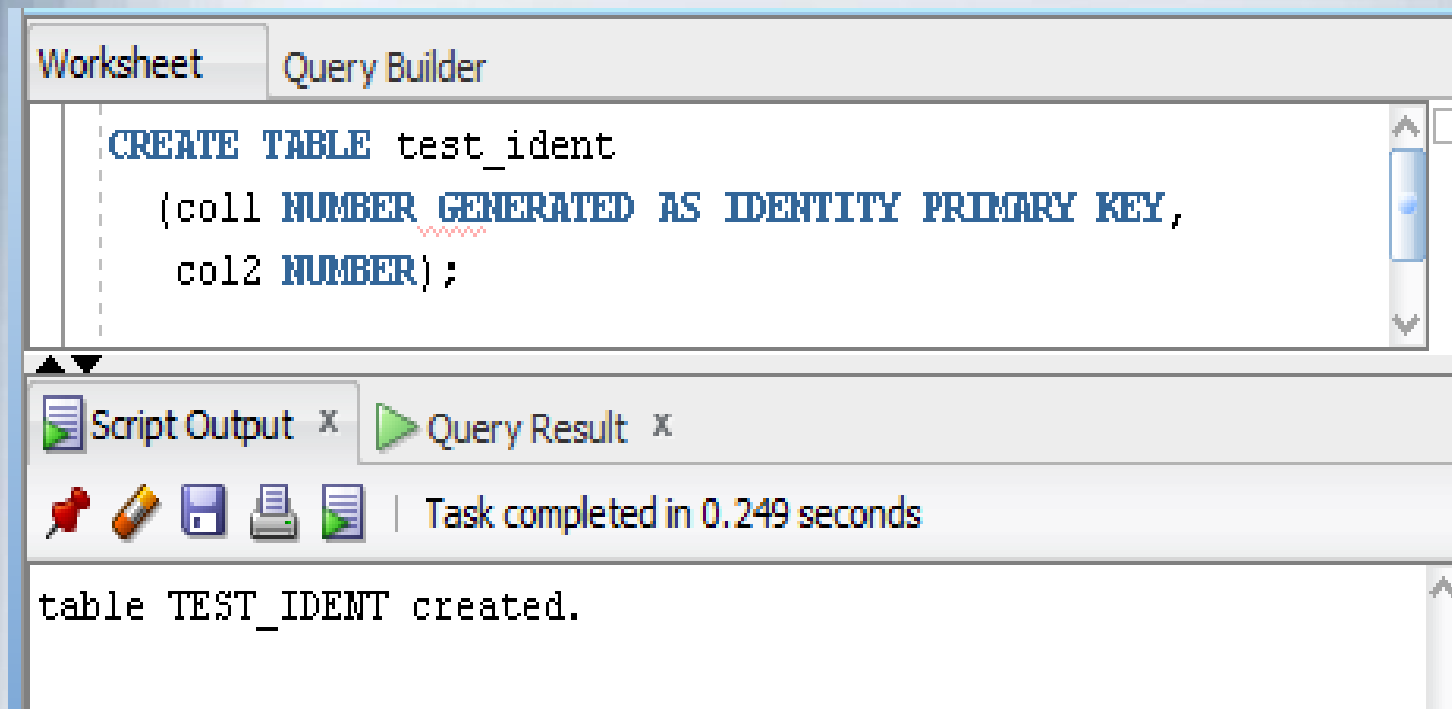
```
DROP SEQUENCE orders_order#_seq;
```

Results Script Output Explain Autotrace DBMS Output OWA Output

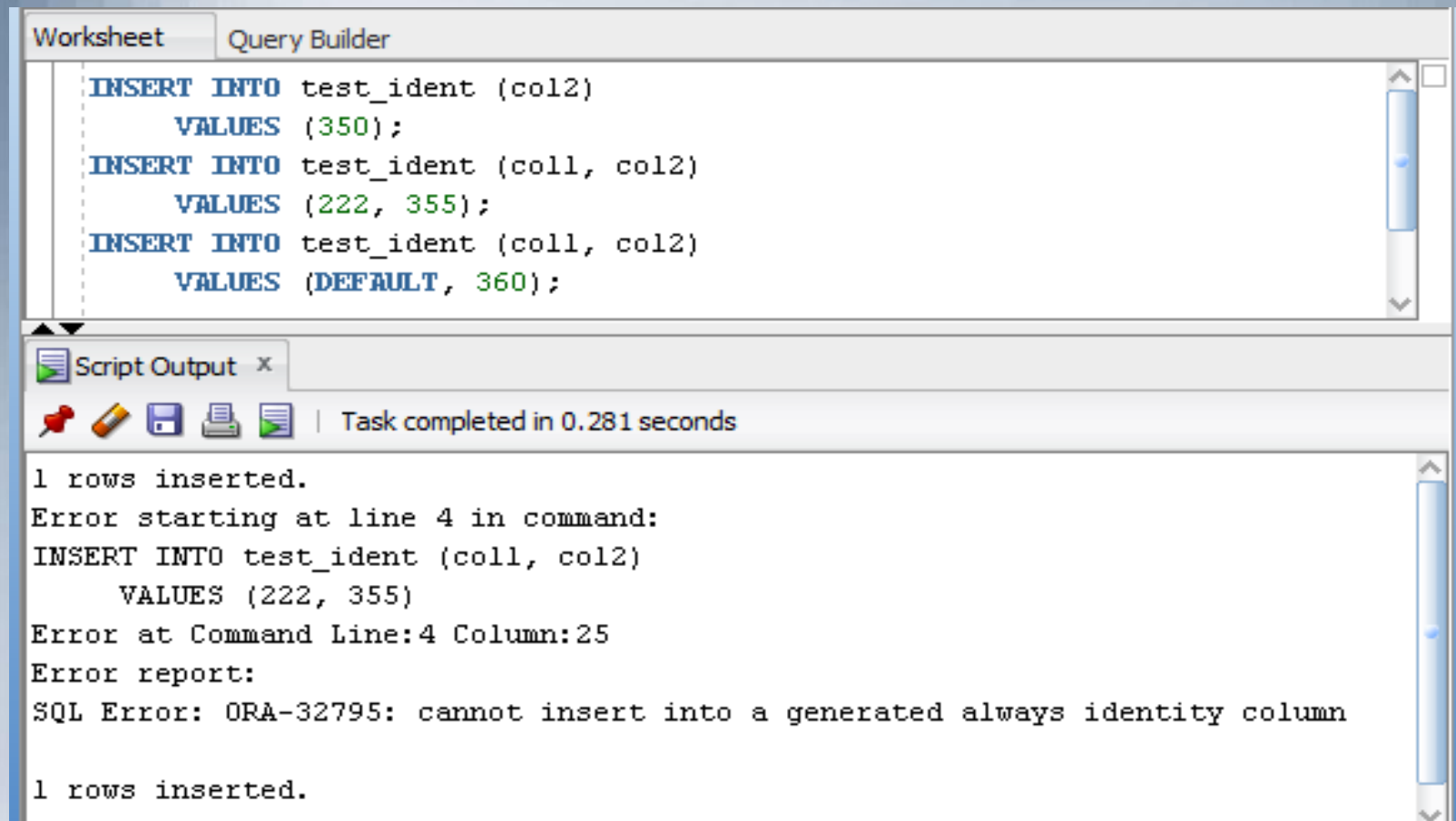
DROP SEQUENCE orders_order#_seq succeeded.

Create an Identity Column

- Alternative to using sequences to populate primary key columns



Using an Identity Column



The screenshot displays a database query builder interface. The top section, labeled 'Query Builder', contains three SQL INSERT statements for a table named 'test_ident'. The first statement inserts a single value (350) into 'col2'. The second statement inserts two values (222, 355) into 'col1' and 'col2'. The third statement inserts a default value and 360 into 'col1' and 'col2'. Below the query builder is a 'Script Output' window. It shows the execution of the script, indicating that 1 row was inserted successfully. However, it then reports an error starting at line 4 in the command: 'INSERT INTO test_ident (col1, col2) VALUES (222, 355)'. The error message is 'SQL Error: ORA-32795: cannot insert into a generated always identity column', indicating that the column 'col1' is an identity column and the value 222 is not within its generated range.

```
Worksheet | Query Builder
```

```
INSERT INTO test_ident (col2)
VALUES (350);
INSERT INTO test_ident (col1, col2)
VALUES (222, 355);
INSERT INTO test_ident (col1, col2)
VALUES (DEFAULT, 360);
```

Script Output x

Task completed in 0.281 seconds

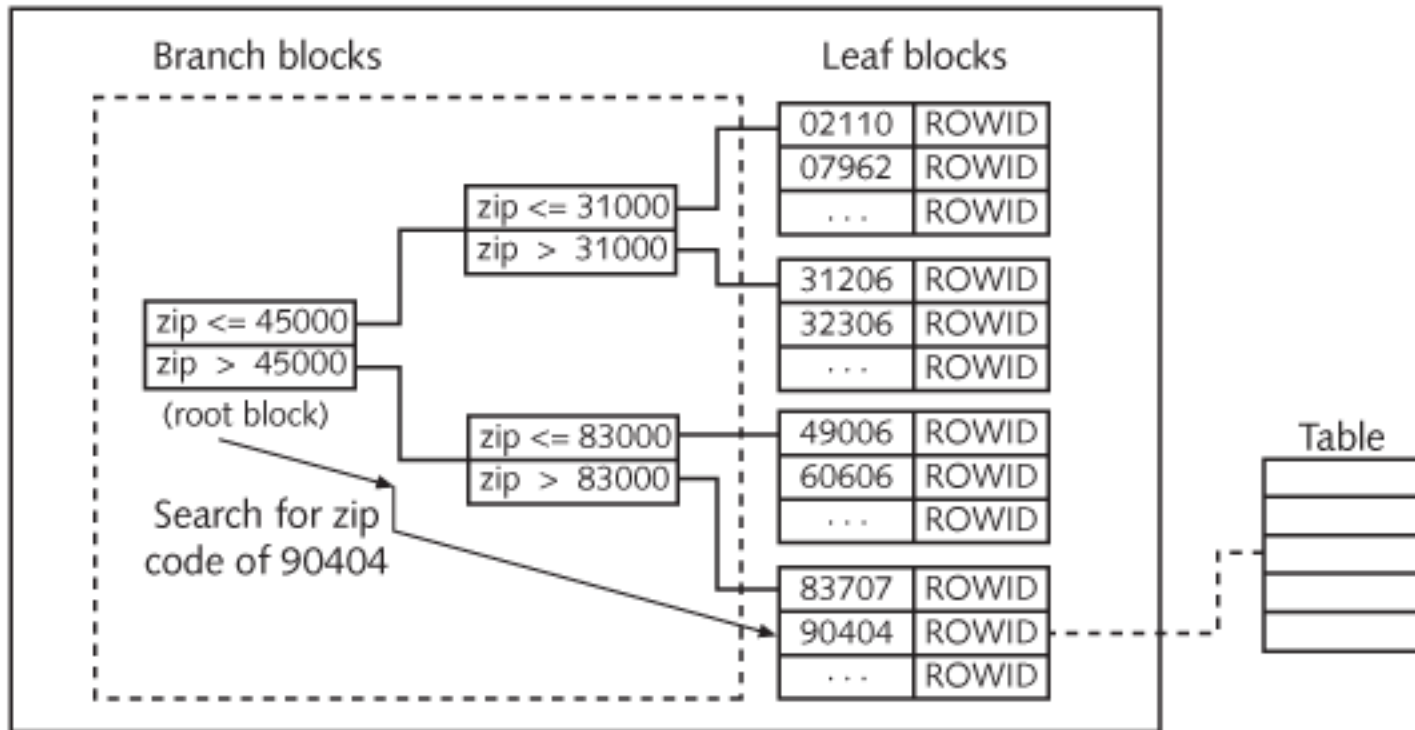
```
1 rows inserted.
Error starting at line 4 in command:
INSERT INTO test_ident (col1, col2)
VALUES (222, 355)
Error at Command Line:4 Column:25
Error report:
SQL Error: ORA-32795: cannot insert into a generated always identity column
1 rows inserted.
```

Indexes

- An index stores frequently referenced values and ROWIDs
- Can be based on one column, multiple columns, functions, or expressions

B-Tree Index

B-tree index



B-Tree Index (continued)

- Implicitly create an index by PRIMARY KEY and UNIQUE constraints
- Explicitly create an index by using the CREATE INDEX command

CREATE INDEX Command Examples

Enter SQL Statement:

```
CREATE INDEX customers_zip_idx  
ON customers (zip);
```

Results Script Output Explain Autotrace DBMS Output OWA Output

CREATE INDEX succeeded.

Enter SQL Statement:

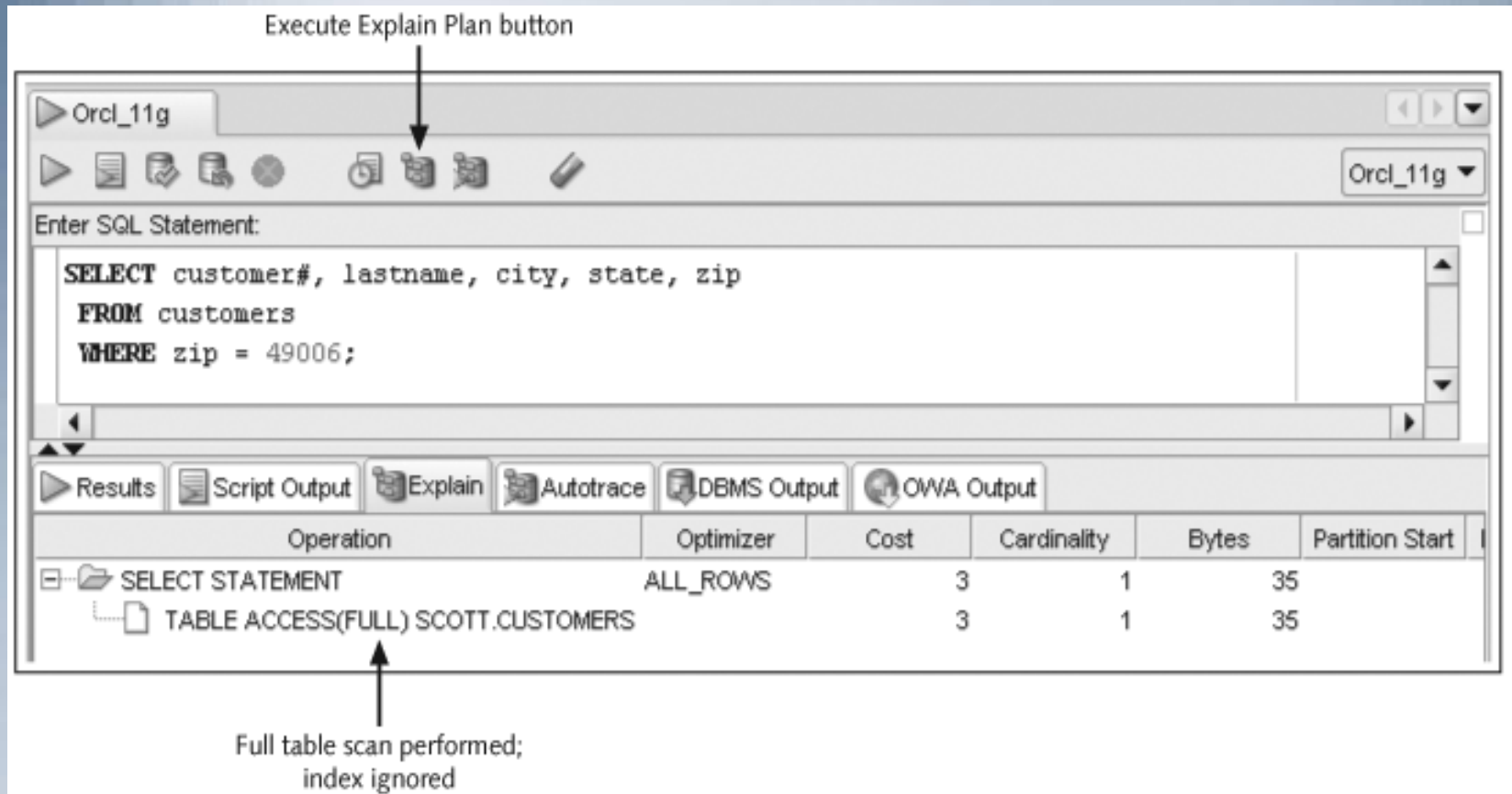
```
CREATE INDEX customer_name_idx  
ON customers (lastname, firstname);
```

Results Script Output Explain Autotrace DBMS Output OWA Output

CREATE INDEX succeeded.

The Explain Plan

Execute Explain Plan button



Orcl_11g

Enter SQL Statement:

```
SELECT customer#, lastname, city, state, zip
FROM customers
WHERE zip = 49006;
```

Results | Script Output | Explain | Autotrace | DBMS Output | OWA Output

Operation	Optimizer	Cost	Cardinality	Bytes	Partition Start
SELECT STATEMENT	ALL_ROWS	3	1	35	
TABLE ACCESS(FULL) SCOTT.CUSTOMERS		3	1	35	

Full table scan performed;
index ignored

Bitmap Indexes

Enter SQL Statement:

```
CREATE BITMAP INDEX customers_region_idx  
ON customers (region);
```

Results Script Output Explain Autotrace DBMS Output OWA Output

CREATE BITMAP succeeded.

Bitmap index
Region

N	NW	NE	S	SE	SW	W	E
0	1	0	0	0	0	0	0
0	0	0	0	1	0	0	0
1	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0
0	0	0	0	0	1	0	0

Function-Based Indexes

Enter SQL Statement:

```
CREATE INDEX books_profit_idx  
ON books (retail-cost);
```



Results



Script Output



Explain



Autotrace



DBMS Output



OWA Output




CREATE INDEX succeeded.

```
CREATE INDEX orders_shipdate_idx  
ON orders(NVL(shipdate,'null'));
```

Index Organized Tables

- An IOT stores table contents in a B-tree index structure
- Use the “ORGANIZATION INDEX” option in a CREATE TABLE statement to build an IOT



```
CREATE TABLE books2
(ISBN VARCHAR2(10),
 title VARCHAR2(30),
 pubdate DATE,
 pubID NUMBER (2),
 cost NUMBER (5,2),
 retail NUMBER (5,2),
 category VARCHAR2(12),
 CONSTRAINT books2_isbn_pk PRIMARY KEY(isbn))
 ORGANIZATION INDEX;
```

Verifying an Index

- Use the USER_INDEXES data dictionary view to determine that the index exists
- Use the USER_IND_COLUMNS data dictionary view to determine the column index information

Verifying an Index (continued)

Enter SQL Statement:

```
SELECT table_name, index_name, index_type
FROM user_indexes
WHERE table_name = 'CUSTOMERS';
```

Results Script Output Explain Autotrace DBMS Output OWA Output

Results:

	TABLE_NAME	INDEX_NAME	INDEX_TYPE
1	CUSTOMERS	CUSTOMERS_REGION_IDX	BITMAP
2	CUSTOMERS	CUSTOMER_NAME_IDX	NORMAL
3	CUSTOMERS	CUSTOMERS_ZIP_IDX	NORMAL
4	CUSTOMERS	CUSTOMERS_CUSTOMER#_PK	NORMAL

Index type; NORMAL is the default B-tree structure

USER_IND_COLUMNS

Enter SQL Statement:

```
SELECT table_name, index_name, column_name
FROM user_ind_columns
WHERE table_name = 'CUSTOMERS';
```



Results



Script Output



Explain



Autotrace



DBMS Output



OWA Output

Results:


	TABLE_NAME	INDEX_NAME	COLUMN_NAME
1	CUSTOMERS	CUSTOMERS_CUSTOMER#_PK	CUSTOMER#
2	CUSTOMERS	CUSTOMERS_ZIP_IDX	ZIP
3	CUSTOMERS	CUSTOMER_NAME_IDX	LASTNAME
4	CUSTOMERS	CUSTOMER_NAME_IDX	FIRSTNAME
5	CUSTOMERS	CUSTOMERS_REGION_IDX	REGION


Removing an Index


- Use the DROP INDEX command to remove an index


Enter SQL Statement:


```
DROP INDEX books_profitcalc_idx;
```


 Results




 Script Output

 Explain

 Autotrace

 DBMS Output

 OWA Output

```
DROP INDEX books_profitcalc_idx succeeded.
```


Synonyms

- Synonyms serve as permanent aliases for database objects
- Simplify object references
- Can be private or public
 - Private synonyms are only available to the user who created them
 - PUBLIC synonyms are available to all database users

CREATE SYNONYM Command Syntax

```
CREATE [PUBLIC] SYNONYM synonymname  
FOR objectname;
```

CREATE SYNONYM Command

Enter SQL Statement:

```
CREATE SYNONYM orderentry
FOR orders;
```

Results Script Output Explain Autotrace DBMS Output OWA Output

CREATE SYNONYM succeeded.

Synonym used in place of table name

Enter SQL Statement:

```
SELECT *
FROM orderentry
WHERE customer# = 1010;
```

Results Script Output Explain Autotrace DBMS Output OWA Output

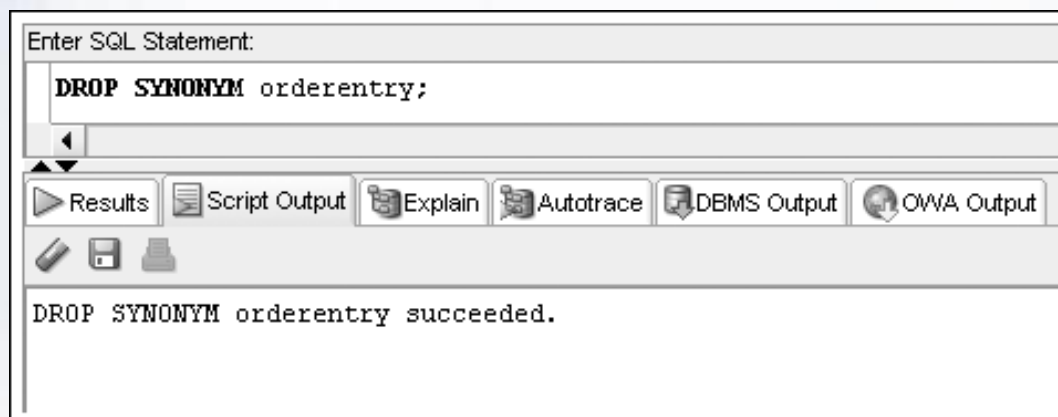
Results:

	ORDER#	CUSTOMER#	ORDERDATE	SHIPDATE	SHIPSTREET	SHIPCITY	SHIPSTATE	SHIPZIP	SHIPCOST
1	1001	1010	31-MAR-09	01-APR-09	114 EAST SAVANNAH	ATLANTA	GA	30314	3
2	1011	1010	03-APR-09	05-APR-09	102 WEST LAFAYETTE	ATLANTA	GA	30311	2
3	1021	1010	06-APR-09	(null)	123 WEST MAIN	ATLANTA	GA	30418	(null)

Deleting a SYNONYM

- A private synonym can be deleted by its owner
- A PUBLIC synonym can only be deleted by a user with DBA privileges

```
DROP [PUBLIC] SYNONYM synonymname;
```



The screenshot shows a web-based SQL execution interface. At the top, there is a text input field labeled "Enter SQL Statement:" containing the command `DROP SYNONYM orderentry;`. Below the input field is a horizontal toolbar with several icons and labels: a play button for "Results", a document icon for "Script Output", a magnifying glass for "Explain", a document with a checkmark for "Autotrace", a document with a checkmark for "DBMS Output", and a globe for "OWA Output". Below the toolbar are three small icons: a pencil, a floppy disk, and a printer. At the bottom of the interface, a text area displays the message `DROP SYNONYM orderentry succeeded.`

Summary

- A sequence can be created to generate a series of integers
- The values generated by a sequence can be stored in any table
- A sequence is created with the CREATE SEQUENCE command
- Gaps in sequences might occur if the values are stored in various tables, if numbers are cached but not used, or if a rollback occurs
- A value is generated by using the NEXTVAL pseudocolumn
- The CURRVAL pseudocolumn is NULL until a value is generated by NEXTVAL
- The USER_OBJECTS data dictionary object can be used to confirm the existence of all schema objects
- The USER_SEQUENCES data dictionary object is used to view sequence settings
- A sequence may be set as a column DEFAULT value
- An identity column can be created to manage primary key population as an alternative to using sequences

Summary (continued)

- The ALTER SEQUENCE command is used to modify an existing sequence; the only settings that can't be modified are the START WITH option and any option that would be invalid because of previously generated values
- The DUAL table is helpful for testing sequence value generation
- The DROP SEQUENCE command deletes an existing sequence
- An index can be created to speed up the query process
- DML operations are always slower when indexes exist
- Oracle 11g creates an index for PRIMARY KEY and UNIQUE constraints automatically
- An explicit index is created with the CREATE INDEX command
- An index can be used by Oracle 11g automatically if a query criterion or sort operation is based on a column or an expression used to create the index

Summary (continued)

- The two main structures for indexes are B-tree and bitmap
- The explain plan can verify whether an index is used in a query
- Function-based indexes are used to index an expression or the use of functions on a column or columns
- An index organized table is a table stored in a B-tree structure to combine the index and table into one database object
- Information about an index can be retrieved from the `USER_INDEXES` and `USER_IND_COLUMNS` views
- An index can be dropped with the `DROP INDEX` command
- An index can be renamed with the `ALTER INDEX` command

Summary (continued)

- Except for a name change, an index can't be modified; it must be deleted and then re-created
- A synonym provides a permanent alias for a database object
- A public synonym is available to any database user
- A private synonym is available only to the user who created it
- A synonym is created by using the CREATE SYNONYM command
- A synonym is deleted by using the DROP SYNONYM command
- Only a user with DBA privileges can drop a public synonym