CUNEF
UNIVERSIDAD

DATABASE SYSTEMS COURSE 2021/2022

SESSION 12

CUNEF UNIVERSIDAD

SESSION 12

- **VIEWS**
- ORACLE MATERIALIZED VIEW
- ORACLE SEQUENCES
- MYSQL AUTO_INCREMENT

VIEWS

CUNEF UNIVERSIDAD

CREATE and DROP VIEW Syntax

MySQL/Oracle

```
CREATE VIEW ViewName AS
SELECT .. FROM ... WHERE ...
```

MySQL/Oracle

```
DROP VIEW ViewName
```

CUNEF UNIVERSIDAD

CREATE VIEW Example

MySQL/Oracle

```
CREATE VIEW castfilm AS
SELECT ms.id, ms.name moviename, r.role,a.first_name, a.last_name
FROM moviesmall ms
LEFT JOIN role r ON (ms.id = r.movie_id)
LEFT JOIN actor a ON (r.actor_id = a.id)
ORDER BY ms.name
```

MySQL/Oracle

```
SELECT * FROM castfilm;
```

ID	MOVIENAME	ROLE	FIRST_NAME	LAST_NAME
10920	Aliens	Rebecca 'Newt' Jorden	Carrie	Henn
10920	Aliens	Al Simpson, Colony Officer	Mac (I)	McDonald
10920	Aliens	Amanda Ripley	Elizabeth	Inglis
10920	Aliens	Med Tech	Alibe	Parsons
10920	Aliens	Cpl. Dietrich	Cynthia Dale	Scott

VIEWS

CUNEF UNIVERSIDAD

CREATE VIEW Example

MySQL/Oracle

```
INSERT INTO moviesmall
VALUES (400000,'Transformers: Rise of the Beasts',2022,NULL);
```

MySQL/Oracle

```
SELECT * FROM castfilm WHERE id = 400000;
```

ID	MOVIENAME	ROLE	FIRST_NAME	LAST_NAME
400000	Transformers: Rise of the Beasts	(null)	(null)	(null)

CUNEF UNIVERSIDAD

SESSION 12

- VIEWS
- **ORACLE MATERIALIZED VIEW**
- ORACLE SEQUENCES
- MYSQL AUTO_INCREMENT

ORACLE MATERIALIZED VIEWS

CUNEF UNIVERSIDAD

CREATE MATERIALIZED VIEW Syntax

Oracle

```
CREATE MATERIALIZED VIEW ViewName
BUILD BUILDAT
REFRESH REFRESHTYPE REFRESHON
AS SELECT ... FROM ... WHERE ...
```

REFRESHON:

- ON COMMIT. When commit is done in any internal table.
- ON DEMAND. By executing: `exec dbms_mview.refresh('ViewName');`
- START WITH … NEXT …. START WITH specifies the first refresh, and NEXT specifies the nexts refreshs.

BUILDAT:

- IMMEDIATE. Populated at Creation
- DEFERRED. Populated at first refreshing

REFRESHTYPE:

- FAST. Update changes and insert new rows
- COMPLETE. All the records are inserted again.
- FORCE. Try first FAST and then COMPLETE

ORACLE MATERIALIZED VIEWS

CUNEF UNIVERSIDAD

CREATE MATERIALIZED VIEW Example

Oracle

```
CREATE MATERIALIZED VIEW mcastfilm
BUILD IMMEDIATE
REFRESH FORCE ON COMMIT AS
SELECT ms.id, ms.name moviename, r.role, a.first_name, a.last_name
FROM moviesmall ms, role r , actor a
WHERE ms.id = r.movie_id (+)
  AND r.actor_id = a.id (+);
```

Oracle

```
SELECT * FROM mcastfilm;
```

ID	MOVIENAME	ROLE	FIRST_NAME	LAST_NAME
237431	O Brother, Where Art Thou?	Everett	George	Clooney
238072	Ocean's Eleven	Danny Ocean	George	Clooney
194874	Lost in Translation	Aerobics Instructor	Hugo	Codaro
238695	Office Space	Bill Lumbergh	Gary (I)	Cole

ORACLE MATERIALIZED VIEWS

CUNEF UNIVERSIDAD

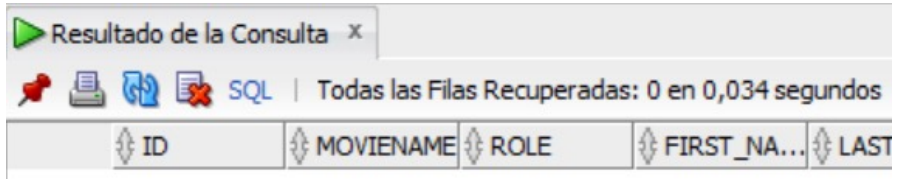
CREATE MATERIALIZED VIEW Example

MySQL/Oracle

```
INSERT INTO moviesmall
VALUES (400000,'Transformers: Rise of the Beasts',2022,NULL);
```

Oracle

```
SELECT * FROM mcastfilm
WHERE id = 400000;
```

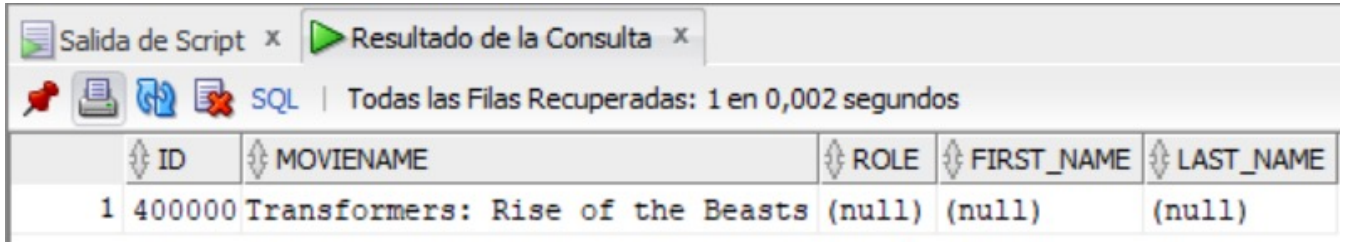


MySQL/Oracle

```
COMMIT;
```

Oracle

```
SELECT * FROM mcastfilm
WHERE id = 400000;
```



ORACLE MATERIALIZED VIEWS

CUNEF UNIVERSIDAD

DROP MATERIALIZED VIEW Syntax

Oracle

```
DROP MATERIALIZED VIEW table
```

DROP MATERIALIZED VIEW Example

Oracle

```
DROP MATERIALIZED VIEW mcastfilm;
```

SESSION 12

- VIEWS
- ORACLE MATERIALIZED VIEW
- **ORACLE SEQUENCES**
- MYSQL AUTO_INCREMENT

ORACLE SEQUENCES

CUNEF UNIVERSIDAD

CREATE SEQUENCE Syntax

Oracle

```
CREATE SEQUENCE SequenceName
START WITH StartValue
INCREMENT BY Value
MAXVALUE MaxValue
MINVALUE MinValue
CYCLE
```

Oracle

```
SequenceName.NEXTVAL
SequenceName.CURRVAL
```

CYCLE:

- CYCLE. The sequence starts again when the MaxValue/MinValue is reached.
- NOCYCLE. The sequence cannot be incremented once the MaxValue/MinValue is reached.

"CREATE SEQUENCE *SequenceName*" creates a sequence that starts with 1, increased by 1 with no upper limit.

ORACLE SEQUENCES

CUNEF UNIVERSIDAD

CREATE SEQUENCE Example

Oracle

```
CREATE SEQUENCE id_seq
     INCREMENT BY 10
     START WITH 10
     MINVALUE 10
     MAXVALUE 100
     NOCYCLE;
```

Oracle

```
CREATE TABLE tabseq
(seqval NUMBER, name VARCHAR2(40), height NUMBER);
```

ORACLE SEQUENCES

CUNEF UNIVERSIDAD

INSERT id value by using SEQUENCE

Oracle

```
INSERT INTO tabseq VALUES (id_seq.NEXTVAL, 'Andy', 1.7);
INSERT INTO tabseq VALUES (id_seq.NEXTVAL, 'Fanny', 1.55);
INSERT INTO tabseq VALUES (id_seq.NEXTVAL, 'Dany', 1.72);
INSERT INTO tabseq VALUES (id_seq.NEXTVAL, 'Camy', 1.69);
```

Oracle

```
SELECT * FROM tabseq;
```

SEQVAL	NAME	HEIGHT
10	Andy	1,7
20	Fanny	1,55
30	Dany	1,72
40	Camy	1,69

ORACLE SEQUENCES

CUNEF UNIVERSIDAD

DROP SEQUENCE Syntax

Oracle

```
DROP SEQUENCE SequenceName
```

DROP SEQUENCE Example

Oracle

```
DROP SEQUENCE id_seq;
```

CUNEF UNIVERSIDAD

SESSION 12

- VIEWS
- ORACLE MATERIALIZED VIEW
- ORACLE SEQUENCES
- **MYSQL AUTO_INCREMENT**

MYSQL AUTO_INCREMENT

CUNEF UNIVERSIDAD

AUTO_INCREMENT Syntax and Example

MySQL

```
CREATE TABLE tabseq
(id INTEGER PRIMARY KEY AUTO_INCREMENT,
 name VARCHAR(40),
 height DECIMAL(5,3))
AUTO_INCREMENT=1;
```

- ***AUTO_INCREMENT*** can be used on PRIMARY KEY columns only.
- ***AUTO_INCREMENT*** = *StartsIn*. Makes the increment starts in "StartsIn" value. If no defined, takes "1" as default.

MySQL

```
INSERT INTO tabseq (name,height) VALUES ('Andy', 1.7);
INSERT INTO tabseq (name,height) VALUES ('Fanny', 1.55);
INSERT INTO tabseq (name,height) VALUES ('Dany', 1.72);
INSERT INTO tabseq (name,height) VALUES ('Camy', 1.69);
```

id *	name	height
1	Andy	1,700
2	Fanny	1,550
3	Dany	1,720
4	Camy	1,690