# Laboratory 3 - Creating database and permissions (DDL)

**Author: Piotr Copek**

**Date: 16.04.2025**

## 1. Connect to database UNIVERSITY as the root user

I connected to database using the same technic from 1st and 2nd laboratories.

## 2. Create new tables: teams1 and emp1 that satisfy the following conditions:

- Table `team1`

```
CREATE TABLE teams1 (
    name CHAR(30),
    teamid SMALLINT PRIMARY KEY,
    manid SMALLINT
) ENGINE=InnoDB;
```

- Table `emp1`

```
CREATE TABLE emp1 (
    empid SMALLINT PRIMARY KEY,
    gender CHAR(1),
    birthdate DATETIME NOT NULL,
    name CHAR(15) NOT NULL,
    teamid SMALLINT,
    INDEX (teamid)
) ENGINE=InnoDB;
```

## 3. Inserting data to the tables:

- Insert all rows from the table `TEAMS` into the table `teams1`, using the `INSERT` command

```
INSERT INTO teams1 (name, teamid, manid)
SELECT TEAM_NAME, TEAM_ID, MANAGER_ID
FROM TEAMS;
```

- Insert all rows from the table `EMPLOYEES` to the table `emp1` using the `INSERT` command

```
INSERT INTO emp1 SELECT * FROM EMPLOYEES;
```

## 4. Modify the tables as follows (keep the proper order)

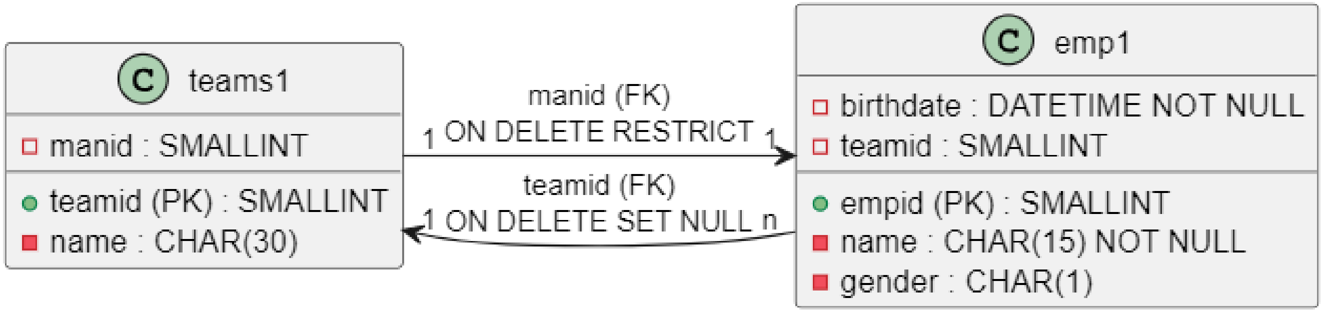
a) In the table team1 should exist a foreign key on manid, which references to the primary key in the emp1 table. Define the necessary constraints that deny deletion of a row in the emp1 table, if there are related rows in the team1 table

```
ALTER TABLE teams1
ADD CONSTRAINT fk_manid
FOREIGN KEY (manid) REFERENCES emp1(empid)
ON DELETE RESTRICT;
```

b) In the table emp1 should exist a foreign key on teamid, which references to primary key in the team1 table. Define the necessary constraints that for any deletion of a row from the team1 table, the related rows of the emp1 table are set to null (in the foreign key).

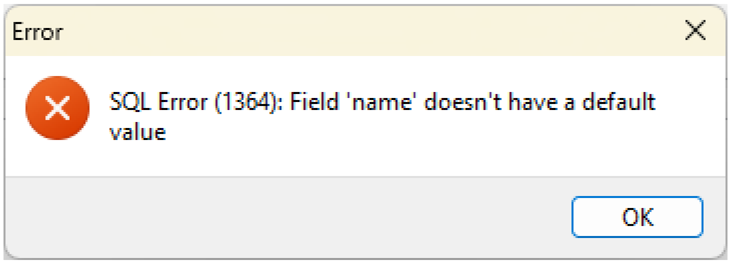
```
ALTER TABLE emp1
ADD CONSTRAINT fk_teamid
FOREIGN KEY (teamid) REFERENCES teams1(teamid)
ON DELETE SET NULL;
```

c) Draw a diagram of the tables



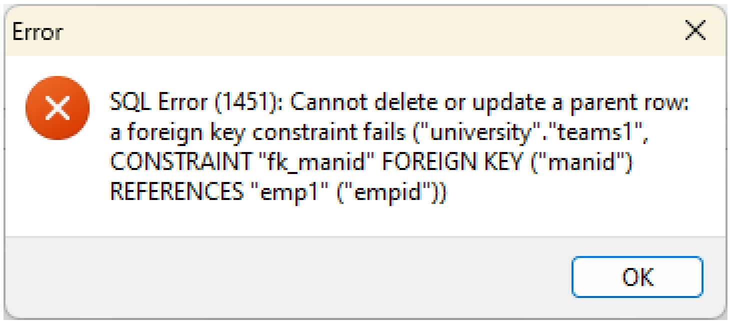
## Task 5. Check the created integrity constraints in the database. (insert Null values, delete dependent/related rows, change values of a primary key or a foreign key in a way that violates the integrity constraints)

```
INSERT INTO emp1 (empid, birthdate) VALUES (100, '2000-01-01');
```



SQL Error (1364): Field 'name' doesn't have a default value

```
DELETE FROM emp1 WHERE empid = 1;
```



SQL Error (1451): Cannot delete or update a parent row: a foreign key constraint fails ("university"."teams1", CONSTRAINT "fk_manid" FOREIGN KEY ("manid") REFERENCES "emp1" ("empid"))

```
DELETE FROM teams1 WHERE teamid = 1;
```

Sets `emp1.teamid` to NULL for affected rows.

## Task 6. Creating users and granting them privileges:

a) Connect to database as the `root` user: Create user `user1@localhost` with password `user1` and grant him the `SELECT` privilege on the `EMPLOYEES` table.

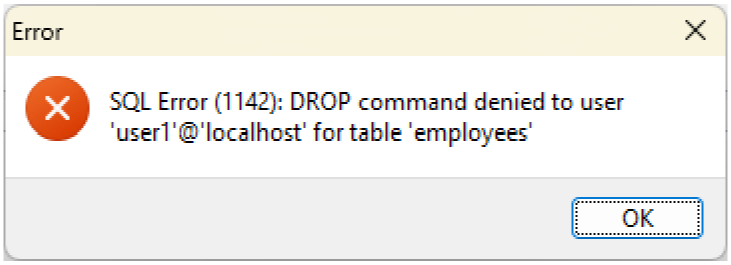
```
CREATE USER 'user1'@'localhost' IDENTIFIED BY 'user1';
GRANT SELECT ON UNIVERSITY.EMPLOYEES TO 'user1'@'localhost';
```

b) Connect as `user1`: Display the `EMPLOYEES` table

```
SELECT * FROM EMPLOYEES;
```

c) Try to drop the `EMPLOYEES` table as `user1`

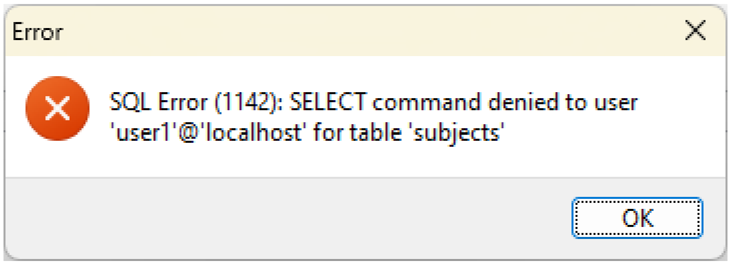
```
DROP TABLE EMPLOYEES;
```



SQL Error (1142): DROP command denied to user 'user1'@'localhost' for table 'employees'

d) Display the `SUBJECTS` table as `user1`

```
SELECT * FROM SUBJECTS;
```



SQL Error (1142): SELECT command denied to user 'user1'@'localhost' for table 'subjects'

e) Connect as root: Grant the `user1` user the delete privilege on the `TEAMS` table

```
GRANT DELETE ON UNIVERSITY.TEAMS TO 'user1'@'localhost';
```

f) Connect as `user1`: Display the `TEAMS` table

```
SELECT * FROM TEAMS;
```

Forgot to make a screenshot. Fails because referenced by foreign key.

g) Drop the team with `team_id=1`

```
DELETE FROM TEAMS WHERE team_id = 1;
```

The `user1` have permissions

h) Connect as `root` : Grant the user `user1` the `SELECT` privilege on the `TEAMS` table.

```
GRANT SELECT ON UNIVERSITY.TEAMS TO 'user1'@'localhost';
```

i) Connect as `user1` : Drop the team with `team_id=1`

```
DELETE FROM TEAMS WHERE team_id = 1;
```

j) Connect as `root` : Create the view called `vCNETWORKS` , which includes the employee ids and names of these employees that belong to the team `COMPUTER NETWORKS` (data from original tables `EMPLOYEES` and `TEAMS` ).

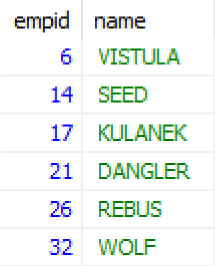
```
CREATE VIEW vCNETWORKS AS
SELECT e.EMPLOYEE_ID AS empid, e.EMP_NAME AS name
FROM EMPLOYEES e
JOIN teams t ON e.TEAM_ID = t.TEAM_ID
WHERE t.TEAM_NAME = 'COMPUTER NETWORKS';
```

k) Grant the `user1` user the `SELECT` privilege on the view `vCNETWORKS`

```
GRANT UPDATE ON UNIVERSITY.vCNETWORKS TO 'user1'@'localhost';
```

l) Connect as `user1` : Display the `vCNETWORKS` view

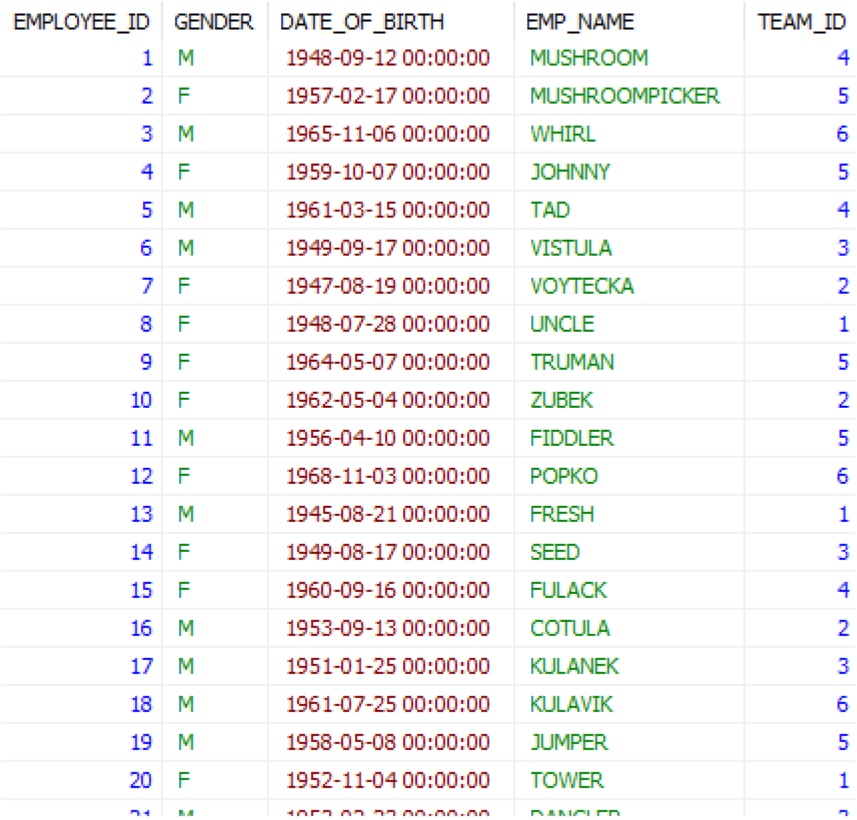
```
SELECT * FROM vCNETWORKS;
```



| empid | name |
|---|---|
| 6 | VISTULA |
| 14 | SEED |
| 17 | KULANEK |
| 21 | DANGLER |
| 26 | REBUS |
| 32 | WOLF |

m) Display the `EMPLOYEES` table

```
SELECT * FROM EMPLOYEES;
```



| EMPLOYEE_ID | GENDER | DATE_OF_BIRTH | EMP_NAME | TEAM_ID |
|---|---|---|---|---|
| 1 | M | 1948-09-12 00:00:00 | MUSHROOM | 4 |
| 2 | F | 1957-02-17 00:00:00 | MUSHROOMPICKER | 5 |
| 3 | M | 1965-11-06 00:00:00 | WHIRL | 6 |
| 4 | F | 1959-10-07 00:00:00 | JOHNNY | 5 |
| 5 | M | 1961-03-15 00:00:00 | TAD | 4 |
| 6 | M | 1949-09-17 00:00:00 | VISTULA | 3 |
| 7 | F | 1947-08-19 00:00:00 | VOYTECKA | 2 |
| 8 | F | 1948-07-28 00:00:00 | UNCLE | 1 |
| 9 | F | 1964-05-07 00:00:00 | TRUMAN | 5 |
| 10 | F | 1962-05-04 00:00:00 | ZUBEK | 2 |
| 11 | M | 1956-04-10 00:00:00 | FIDDLER | 5 |
| 12 | F | 1968-11-03 00:00:00 | POPKO | 6 |
| 13 | M | 1945-08-21 00:00:00 | FRESH | 1 |
| 14 | F | 1949-08-17 00:00:00 | SEED | 3 |
| 15 | F | 1960-09-16 00:00:00 | FULACK | 4 |
| 16 | M | 1953-09-13 00:00:00 | COTULA | 2 |
| 17 | M | 1951-01-25 00:00:00 | KULANEK | 3 |
| 18 | M | 1961-07-25 00:00:00 | KULAVIK | 6 |
| 19 | M | 1958-05-08 00:00:00 | JUMPER | 5 |
| 20 | F | 1952-11-04 00:00:00 | TOWER | 1 |
| 21 | M | [illegible] | DANGLER | [illegible] |

n) Update data using the `vCNETWORKS` view (try change name of a chosen employee)

```
UPDATE vCNETWORKS SET name = 'New Name' WHERE empid = 1;
```

Fails to update privilege on the view.

o) Connect as `root`: Grant the `user1` user the `UPDATE` privilege on the `vCNETWORKS` view

```
GRANT UPDATE ON UNIVERSITY.vCNETWORKS TO 'user1'@'localhost';
```

p) Connect as user1: Update data using the vCNETWORKS view (try change name of chosen employee)

```
UPDATE vCNETWORKS SET name = 'New Name' WHERE empid = 1;
```

Succeeds if the view is updatable and privileges are correct.

q) Connect as `root`: Update data using the `vCNETWORKS` view (try change name of chosen employee)

```sql
UPDATE vCNETWORKS SET name = 'Root Name' WHERE empid = 1;
```

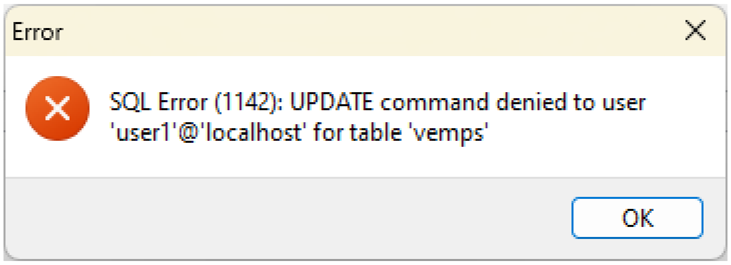
Succeeds: Root has full privileges.

r) Create the view `vEMPS` including the names of employees.

```sql
CREATE VIEW vEMPS AS SELECT emp_name FROM EMPLOYEES;
```

s) Connect as `user1`: Update data using the `vEMPS` view (try change name of the employee)

```sql
UPDATE vEMPS SET name = 'Test' WHERE name = 'Old Name';
```



SQL Error (1142): UPDATE command denied to user 'user1'@'localhost' for table 'vemps'

No `UPDATE` privilege on `vEMPS`.

t) Grant `user1` user the `SELECT` and `UPDATE` privileges on the `vEMPS` view.

```sql
GRANT SELECT, UPDATE ON UNIVERSITY.vEMPS TO 'user1'@'localhost';
```

u) Connect as `user1`: Update data using the view `vEMPS` (try change name of chosen employee having a given id)

```sql
UPDATE vEMPS SET emp_name = 'New Name' WHERE emp_name = 'Old Name';
```

Succeeds if the view allows updates.

v) Update data again using the view `vEMPS` (try change name of chosen employee having a given name)

```
UPDATE vEMPS SET emp_name = 'Another Name' WHERE emp_name = 'New Name';
```

Same situation as above.

w) Connect as `root` : Create user `secnd@localhost` with password `secnd`

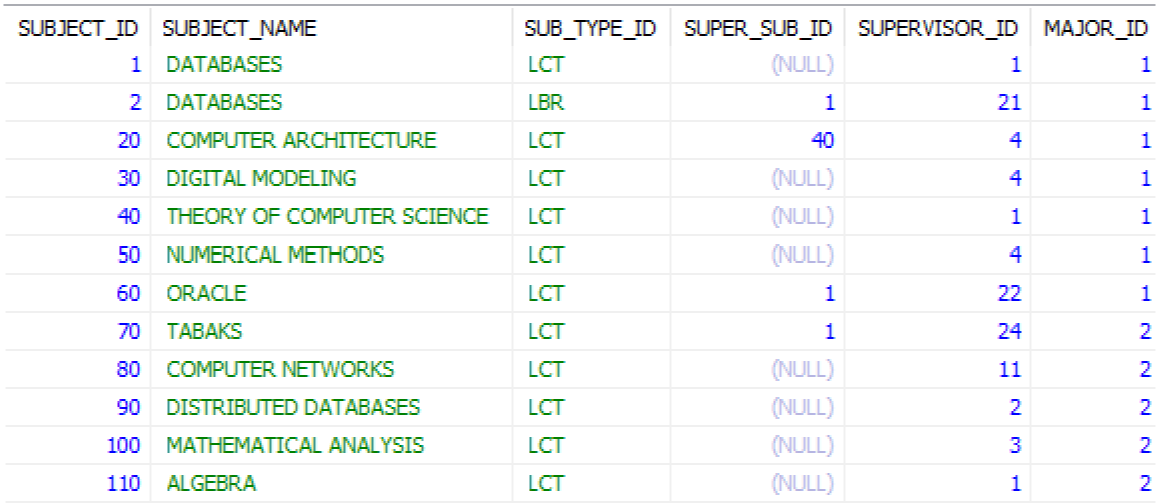
```
CREATE USER 'scend'@'localhost' IDENTIFIED BY 'scend';
```

x) Grant the `secnd` user the `SELECT` privilege on the `SUBJECTS` table.

```
GRANT SELECT ON UNIVERSITY.SUBJECTS TO 'scend'@'localhost';
```

y) Connect as `secnd` : Display the SUBJECTS table

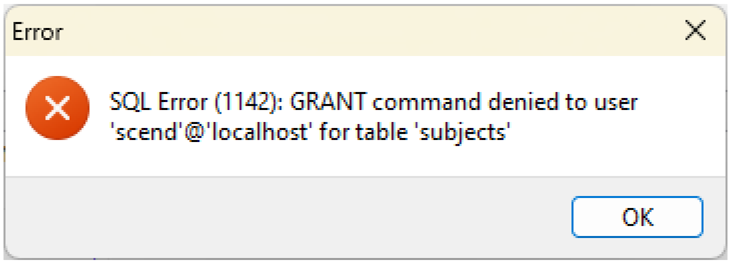
```
SELECT * FROM SUBJECTS;
```



| SUBJECT_ID | SUBJECT_NAME | SUB_TYPE_ID | SUPER_SUB_ID | SUPERVISOR_ID | MAJOR_ID |
|---|---|---|---|---|---|
| 1 | DATABASES | LCT | (NULL) | 1 | 1 |
| 2 | DATABASES | LBR | 1 | 21 | 1 |
| 20 | COMPUTER ARCHITECTURE | LCT | 40 | 4 | 1 |
| 30 | DIGITAL MODELING | LCT | (NULL) | 4 | 1 |
| 40 | THEORY OF COMPUTER SCIENCE | LCT | (NULL) | 1 | 1 |
| 50 | NUMERICAL METHODS | LCT | (NULL) | 4 | 1 |
| 60 | ORACLE | LCT | 1 | 22 | 1 |
| 70 | TABAKS | LCT | 1 | 24 | 2 |
| 80 | COMPUTER NETWORKS | LCT | (NULL) | 11 | 2 |
| 90 | DISTRIBUTED DATABASES | LCT | (NULL) | 2 | 2 |
| 100 | MATHEMATICAL ANALYSIS | LCT | (NULL) | 3 | 2 |
| 110 | ALGEBRA | LCT | (NULL) | 1 | 2 |

z) Grant the `user1` the `SELECT` privilege on the `SUBJECTS` table as `secnd`

```
GRANT SELECT ON UNIVERSITY.SUBJECTS TO 'user1'@'localhost';
```



SQL Error (1142): GRANT command denied to user 'scend'@'localhost' for table 'subjects'

User `secnd` lacks `GRANT OPTION`.

aa) Connect as `root`: Grant the `secnd` user the `SELECT` privilege on the `SUBJECTS` table with grant option.

```
GRANT SELECT ON UNIVERSITY.SUBJECTS TO 'scend'@'localhost' WITH GRANT OPTION;
```

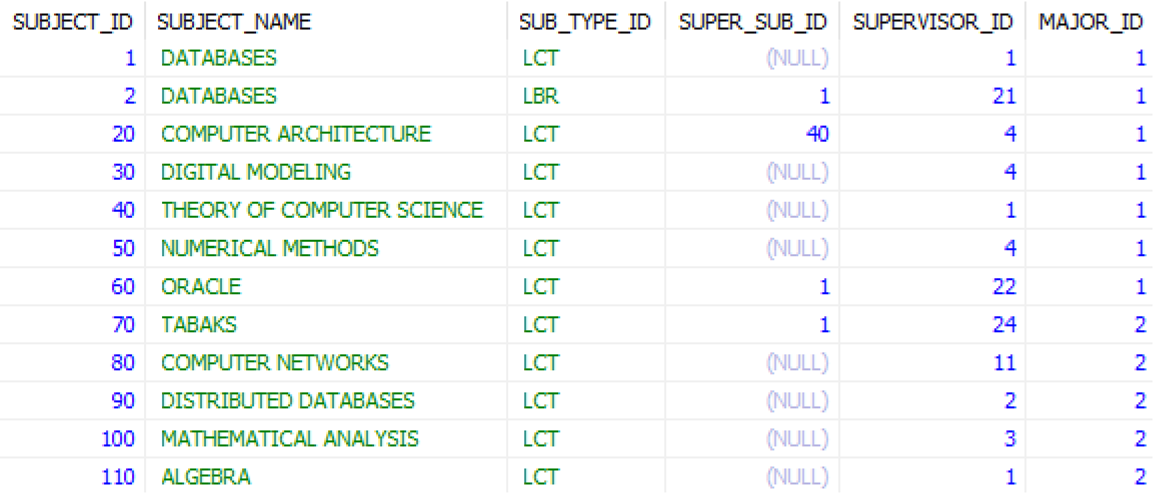
bb) Connect as `secnd`: Grant the `user1` user the `SELECT` privilege on the `SUBJECTS` table

```
GRANT SELECT ON UNIVERSITY.SUBJECTS TO 'user1'@'localhost';
```

User `scend` now has GRANT OPTION.

cc) Connect as user1: Display the SUBJECTS table

```
SELECT * FROM SUBJECTS;
```



| SUBJECT_ID | SUBJECT_NAME | SUB_TYPE_ID | SUPER_SUB_ID | SUPERVISOR_ID | MAJOR_ID |
|---|---|---|---|---|---|
| 1 | DATABASES | LCT | (NULL) | 1 | 1 |
| 2 | DATABASES | LBR | 1 | 21 | 1 |
| 20 | COMPUTER ARCHITECTURE | LCT | 40 | 4 | 1 |
| 30 | DIGITAL MODELING | LCT | (NULL) | 4 | 1 |
| 40 | THEORY OF COMPUTER SCIENCE | LCT | (NULL) | 1 | 1 |
| 50 | NUMERICAL METHODS | LCT | (NULL) | 4 | 1 |
| 60 | ORACLE | LCT | 1 | 22 | 1 |
| 70 | TABAKS | LCT | 1 | 24 | 2 |
| 80 | COMPUTER NETWORKS | LCT | (NULL) | 11 | 2 |
| 90 | DISTRIBUTED DATABASES | LCT | (NULL) | 2 | 2 |
| 100 | MATHEMATICAL ANALYSIS | LCT | (NULL) | 3 | 2 |
| 110 | ALGEBRA | LCT | (NULL) | 1 | 2 |