

## Примерен вариант за курсова работа по

### БАЗИ ДАННИ

**Задача 1:** Проектирайте базата данни и начертайте ER – диаграма по следните изисквания:

Трябва да се направи база данни, която ще съхранява информация за филми-заглавие, година на издаване, дължина на филма, филмово студио, в което е направен, кой го е продуцирал, бюджет, както и участващи в него актьори. За актьорите, участващи във филмите е достатъчно да се знае : адрес, пол и рождена дата, а за студиото, в което е записан и за продуцента – просто адрес и булстат на фирмите им.

**Задача 2:** Напишете заявка, която извежда имената на всички актьори, които са мъже или живеят в “София”.

**Задача 3:** Напишете заявка, която извежда топ 3 на най-високо бюджетните филми, излезнали между 1990 и 2000 година.

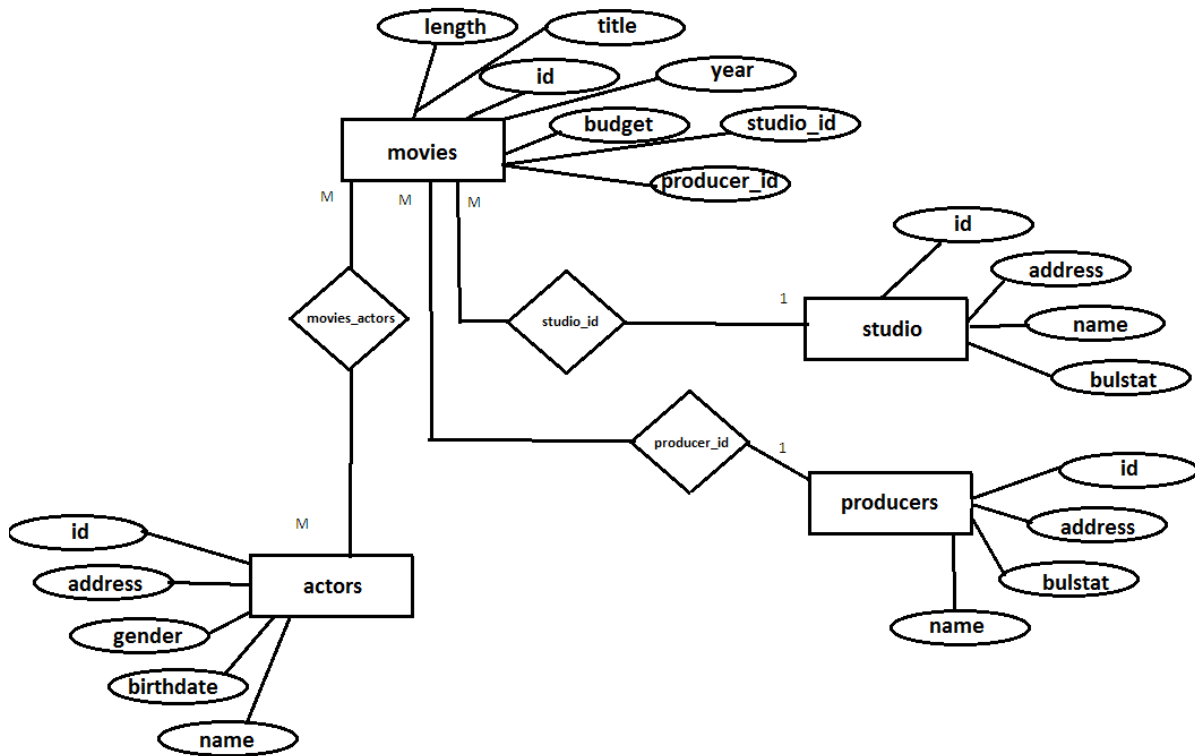
**Задача 4:** Напишете заявка, която извежда имена на филми и имената на актьорите, които участват в тях, но само за филми, които са продуцирани от продуцент „John Smith”.

**Задача 5:** Напишете заявка, която извежда име на актьор и средна стойност на дължините на филмите, в който е участвал, но само за филми, имащи дължини, по-големи от средната дължина на филмите преди 2000г.

### Решение:

За решение на задачата ще използваме езика **MySQL**, който се изучава по време на лабораторните упражнения по дисциплината. За проектирането на базата в задача 1 ще използваме модела **ER-диаграма**(Entity Relationship Diagram).

### Задача 1:



### Нека създадем таблица студиа:

```
CREATE TABLE studios(  
  id int NOT NULL AUTO_INCREMENT PRIMARY KEY,  
  address VARCHAR(255) NOT NULL,  
  name VARCHAR(255) NOT NULL UNIQUE,  
  bulstat VARCHAR(255) UNIQUE  
);
```

### Вкарваме примерни данни за студиа:

```
INSERT INTO `studios` (`address`, `name`, `bulstat`)  
VALUES ('Sofia, Boris str. 56', 'Studio Diana', '1112221'),
```

```
('London, Everest str. 355', 'Studio Express', '1457458'),  
( 'Berlin, Kernigan str. 455', 'Studio Europa', '1114452');
```

	id	address	name	bulstat
▶	1	Sofia, Boris str. 56	Studio Diana	1112221
	2	London, Everest str. 355	Studio Express	1457458
	3	Berlin, Kernigan str. 455	Studio Europa	1114452
*	NULL	NULL	NULL	NULL

### Нека създадем таблица продуценти:

```
CREATE TABLE producers(  
  id int PRIMARY KEY not null AUTO_INCREMENT,  
  address VARCHAR(255) NOT NULL,  
  bulstat VARCHAR(255) NOT NULL UNIQUE,  
  name VARCHAR(255) NOT NULL  
);
```

### Вкарваме примерни данни за продуценти:

```
INSERT INTO producers(address, bulstat, name)  
VALUES("London, Safary str. 477", "12124545", "John Smith"),  
("Sofia, Opalchenska str. 78", '12541250',"Erna & Dina EOOD"),  
("Paris, Earoh str. 626", '14514520',"Exctravaganzza Trading");
```

	id	address	bulstat	name
▶	1	London, Safary str. 477	12124545	John Smith
	2	Sofia, Opalchenska str. 78	12541250	Erna & Dina EOOD
	3	Paris, Earoh str. 626	14514520	Exctravaganzza Trading
*	NULL	NULL	NULL	NULL

### Нека създадем таблица актьори:

```
CREATE TABLE actors(  
  id SERIAL PRIMARY KEY,  
  name VARCHAR(255) NOT NULL,
```

```
address VARCHAR(255) NULL DEFAULT NULL,  
gender ENUM('M', 'F') NOT NULL,  
birthday DATE NULL DEFAULT NULL  
);
```

### Вкарваме примерни данни за актьори:

```
INSERT INTO actors(name, address, gender, birthday)  
VALUES("Robert de Niro", "London, JMS str. 346", 'M','1943-08-17'),  
("Arnold Schwarzenegger", "London, Monaco str. 478", 'M','1947-07-30'),  
("Stoyanka Mutafova", "Sofia, Tsar Boris str. 45", 'F','1922-02-02');
```

	id	name	address	gender	birthday
▶	1	Robert de Niro	London, JMS str. 346	M	1943-08-17
	2	Arnold Schwarzenegger	London, Monaco str. 478	M	1947-07-30
	3	Stoyanka Mutafova	Sofia, Tsar Boris str. 45	F	1922-02-02
*	NULL	NULL	NULL	NULL	NULL

### Нека създадем таблица филми:

```
CREATE TABLE movies(  
id int AUTO_INCREMENT PRIMARY KEY,  
title VARCHAR(255) NOT NULL UNIQUE,  
year YEAR NOT NULL,  
budget double NOT NULL,  
length INT UNSIGNED NOT NULL,  
studio_id int,  
producer_id int,  
FOREIGN KEY(studio_id) REFERENCES studios(id),  
FOREIGN KEY(producer_id) REFERENCES producers(id)  
);
```

### Вкарваме примерни данни за филми:

```
INSERT INTO MOVIES(title, year, budget, length, studio_id, producer_id)
VALUES("Greetings",1968,5000000,120,2,3),
("Junior",1994,50000000,115,3,1),
("If somebody loves you",2010,500000,90,1,2)
```

id	title	year	budget	length	studio_id	producer_id
1	Greetings	1968	5000000	120	2	3
2	Junior	1994	50000000	115	3	1
3	If somebody loves you	2010	500000	90	1	2
▶*	NULL	NULL	NULL	NULL	NULL	NULL

### Нека създадем свързваща таблица за актьори и филми:

```
CREATE TABLE movies_actors(
  actor_id BIGINT UNSIGNED NOT NULL REFERENCES actors(id),
  movie_id BIGINT UNSIGNED NOT NULL REFERENCES movies(id),
  PRIMARY KEY(actor_id, movie_id)
);
```

### Вкарваме примерни данни за актьори, участващи във филми:

```
INSERT INTO movies_actors(actor_id,movie_id)
VALUES(1,1),
(2,2),
(1,2),
(3,3);
```

**Задача 2:** Напишете заявка, която извежда имената на всички актьори, които са мъже или живеят в “Sofia”.

```
SELECT actors.name
FROM actors
WHERE actors.address LIKE '%Sofia%' OR gender="m";
```

name
Arnold Schwarzenegger
Robert de Niro
Stoyanka Mutafova

**ЗАДАЧА 3:** Напишете заявка, която извежда топ 3 на най-високо бюджетните филми, излезнали между 1990 и 2000 година.

```
SELECT *
FROM movies
WHERE year BETWEEN 1990 AND 2000
ORDER BY budget desc
LIMIT 3;
```

id	title	year	budget	length	studio_id	producer_id
2	Junieur	1994	50000000	115	3	1
*	NULL	NULL	NULL	NULL	NULL	NULL

**Задача 4:** Напишете заявка, която извежда имена на филми и имената на актьорите, които участват в тях, но само за филми, които са продуцирани от продуцент „John Smith”.

```
SELECT movies.title, actors.name
FROM movies JOIN actors
ON actors.id IN(
SELECT actor_id
FROM movies_actors
WHERE movies_actors.movie_id = movies.id)
WHERE producer_id=(
```

```
SELECT id
FROM producers
WHERE name="John Smith");
```

	title	name
▶	Juniour	Robert de Niro
	Juniour	Arnold Schwarzenegger

**Задача. 5** Напишете заявка, която извежда име на актьор и средна стойност на дължините на филмите, в който е участвал, но само за филми, имащи дължини, по-големи от средната дължина на филмите преди 2000г.

```
SELECT actors.name, AVG(movies.length) AS avglength
FROM movies JOIN actors
ON actors.id IN(
SELECT actor_id
FROM movies_actors
WHERE movies_actors.movie_id = movies.id)
WHERE movies.length > (
SELECT AVG(movies.length)
FROM movies
WHERE movies.year > 2000)
GROUP BY actors.name;
```

	name	avglength
▶	Arnold Schwarzenegger	115.0000
	Robert de Niro	117.5000