

Київ-2021 рік

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ «КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ
ІНСТИТУТ ІМЕНІ ІГОРЯ СІКОРСЬКОГО»

Факультет Інформатики та обчислювальної
техніки

(повна назва)

Кафедра Інформатики та програмної
інженерії

(повна назва)

Дисципліна Бази даних
на

Курс 2 Група ІІІ-02 Семестр 1

ЗАВДАННЯ

НА КУРСОВУ РОБОТУ СТУДЕНТУ

Полішко Андрію Олексійовичу

(прізвище, ім'я, по батькові)

1. Тема роботи *База даних для підтримки діяльності рекламного
агентства*

сесії

керівник роботи *старший викладач Марченко Олена Іванівна*

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

2. Строк подання студентом роботи 24.12.2021

3. Вихідні дані до роботи

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно

розробити) *Вступ, аналіз предметної області, постановка завдання,*

концептуальна модель бази даних, логічна модель бази даних, реалізація бази

даних, висновки

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)

6. Дата *31.10.2021*
видачі
завдання

№ з/п	Назва етапів виконання курсового проекту	Строк виконання етапів проекту	Примітка
1	Вивчення літератури	<i>05.11.2021</i>	
2	Аналіз предметного середовища	<i>10.11.2021</i>	
3	Побудова ER-діаграми	<i>12.11.2021</i>	
4	Побудова реляційної схеми	<i>22.11.2021</i>	
5	Створення бази даних	<i>26.11.2021</i>	
6	Створення користувачів бази даних	<i>05.12.2021</i>	
7	Імпорт даних з використанням засобів СУБД	<i>10.12.2021</i>	
8	Створення запитів до розробленої БД	<i>17.12.2021</i>	

9	Оптимізація роботи запитів	20.12.2021	

КАЛЕНДАРНИЙ ПЛАН

Студент

Полішко А. О.

(підпис)

(прізвище та ініціали)

Керівник роботи

Марченко О. І.

(підпис)

(прізвище та ініціали)

Вступ

У сьогоденні людям доводиться оперувати величезними об'ємами інформації, яка без належної структури та подання, буде не доступна для використання. Інформація має бути не тільки доступна для використання, але швидко аналізуватися, підлягати зміні та зручній імплементації з технологіями у яких ця інформація використовується. Для таких цілей використовують бази даних (надалі - БД). БД широко використовуються як у колосальних комерційних проектах так і в невеликих рішеннях, як та, що буде моделюватись протягом цієї курсової роботи. При виконанні поставлених задач використовувались такі інструменти :

- MySQL – швидка та сучасна система управління базами даних (надалі - СУБД), використання якої широко поширене;
- draw.io – інструмент для створення діаграм, що надає необхідних функціонал безкоштовно.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

Потрібно розробити базу даних для підтримки діяльності рекламного агентства.

Основною сутністю є рекламне агентство, яке має наступні атрибути:

- Унікальний номер
- Тип(творче або медійне)
- Мінімальна ціна проекту для співпраці
- Середня ціна однієї години роботи
- Кількість працівників
- Дата заснування
- Розташування головного офісу
- ФІО виконувача обов'язків голови агентства
- Основне середовище публікації реклами(ТБ, інтернет,

друковані матеріали і т.д.)

Замовник — юридична чи фізична особа, що замовляє послуги агентства. Його атрибути:

- Унікальний номер
- ПІБ замовника
- Тип замовника (юридична чи фізична особа)
- Максимальна сума, що виділяється на замовлення

Замовлена послуга — продукт або інша послуга, яку замовляє замовник. Її атрибути:

- Унікальний номер
- Ціна
- Дата початку виконання послуги
- Прогнозований дата закінчення виконання послуги

Офіс, що виконує замовлення — той офіс, в який делегується виконання послуги. Його атрибути:

- Унікальний номер
- Розташування
- Кількість працівників
- Середня ціна години роботи

Керівник виконання замовленого проекту — особа, що є відповідальною за виконання проекту та делегує процес. Його атрибути:

- Унікальний номер
- ПІБ

Посада керівника проект. Її атрибути:

- Унікальний номер
- Назва
- Заробітна плата

Середовище публікації реклами — середовище, в якому планується реалізувати рекламу:

- Унікальний номер
- Прогнозована охоплена аудиторія
- Тип середовища(ТБ, інтернет, друковані матеріали т.д.)

Мета замовленої послуги — описує цілі, для яких послуга рекламується. Її атрибути:

- Унікальний номер
- Тип (благодійність, комерційні наміри тощо)

Додаткові залучені компанії — у процесі реклами можуть залучатись допоміжні компанії. Їх атрибути:

- Унікальний номер
- Назва
- Тип
- Мета, з якою компанія була залучена

Аудиторія реклами — у кожного продукту, чи то товар чи будь-що інше, є цільова аудиторія, яку бажають зацікавити. Її атрибути :

- Унікальний номер
- Прогнозована кількість
- Середній прогнозований вік аудиторії.

Варто передбачити наступні обмеження:

1. Максимальна сума виділена на замовлення замовником має бути не менше 1000 і не більше 1000000

2. Дата початку виконання послуги не може бути більш за прогнозовану дату кінця виконання послуги
3. Кількість працівників у офісі, що виконує замовлення не може бути менше 40 і більше 400
4. Ім'я керівника проекту не довше 100 символів.
5. Кількість аудиторії не може бути менше 100 і більше 100000
6. Вік аудиторії не може бути менше 1 року або більше 150 років.
7. Зарплата керівника проекту не може бути менше 1000 і більше 10000

Підсумок розділу: ми проаналізували детально проаналізували предметну область, та сутності, визначили, що вони будуть виконувати та привели можливі обмеження для підтримки нашої БД

2 ПОСТАНОВКА ЗАДАЧІ

Необхідно спроектувати БД для роботи рекламного агентства. Для цього потрібно зробити наступне:

- 1.проаналізувати предметне середовище;
- 2.побудувати ER-діаграму та схему БД;
- 3.створити та заповнити БД записами;
- 4.розробити скрипти та запити на мові SQL, оптимізувати їх роботу.

В результаті ми створюємо базу даних для підтримки роботи рекламного агентства

3 Концептуальна модель бази даних

3.1 Побудувати ER діаграму

Наша БД матиме 10 сутностей. Їх назви та опис зазначені у таблиці 3.1

Таблиця 3.1

Назва	Опис
рекламне агентство(advertising agency)	Інформує про тип агентства; мінімальну ціну проекту для співпраці;середню ціну однієї години роботи;кількість працівників;дату заснування;розташування головного офісу;ФІО голови агентства; основне середовище публікацій реклами
Замовник(Customer)	Інформує про ПІБ замовника;Тип замовника;Максимальна сума, що виділяється на замовлення
Замовлена послуга(Ordered service)	Інформує про Ціну;Дата початку виконання послуги;Прогнозований дата закінчення виконання послуги
Офіс, що виконує замовлення(Ordering office)	Інформує про Розташування;Кількість

	працівників;Середня ціна години роботи
Керівник виконання замовленого проекту(manager)	Інформує про ПІБ керівника проекту
Посада керівника проекту(Position of manager)	Інформує про назву посади;заробітну плату
Середовище публікації реклами(publishing environment)	Інформує про Прогнозовану охоплену аудиторію;Тип середовища(ТБ, інтернет, друковані матеріали т.д.)
Мета замовленої послуги (The purpose)	Інформує про Тип (благодійність, комерційні наміри тощо)
Додаткові залучені компанії(Additional companies)	Інформує про Назву;Тип;Мету, з якою компанія була залучена
Аудиторія реклами(audience)	Інформує про Прогнозовану кількість;Середній прогнозований вік аудиторії.

Зв'язки між сутностями наведені у таблиці 3.2

Таблиця 3.2

рекламне агентство	Замовник	Багато до багатьох
рекламне агентство	Офіс, що виконує замовлення	Один до багатьох
рекламне агентство	Керівник виконання замовленого проекту	Один до багатьох
Замовник	Замовлена послуга	Один до багатьох
Замовлена послуга	Аудиторія реклами	Багато до багатьох
Замовлена послуга	Мета замовленої послуги	Один до багатьох
Замовлена послуга	Середовище публікації реклами	Багато до багатьох
Замовлена послуга	Офіс, що виконує замовлення	Багато до одного
Офіс, що виконує замовлення	Керівник виконання замовленого проекту	Один до одного
Офіс, що виконує замовлення	Додаткові залучені компанії	Один до багатьох
Керівник виконання замовленого проекту	Посада керівника проекту	Багато до одного

З отриманих таблиць можемо побудувати ER діаграму. Вона зображена на рисунку 3.1

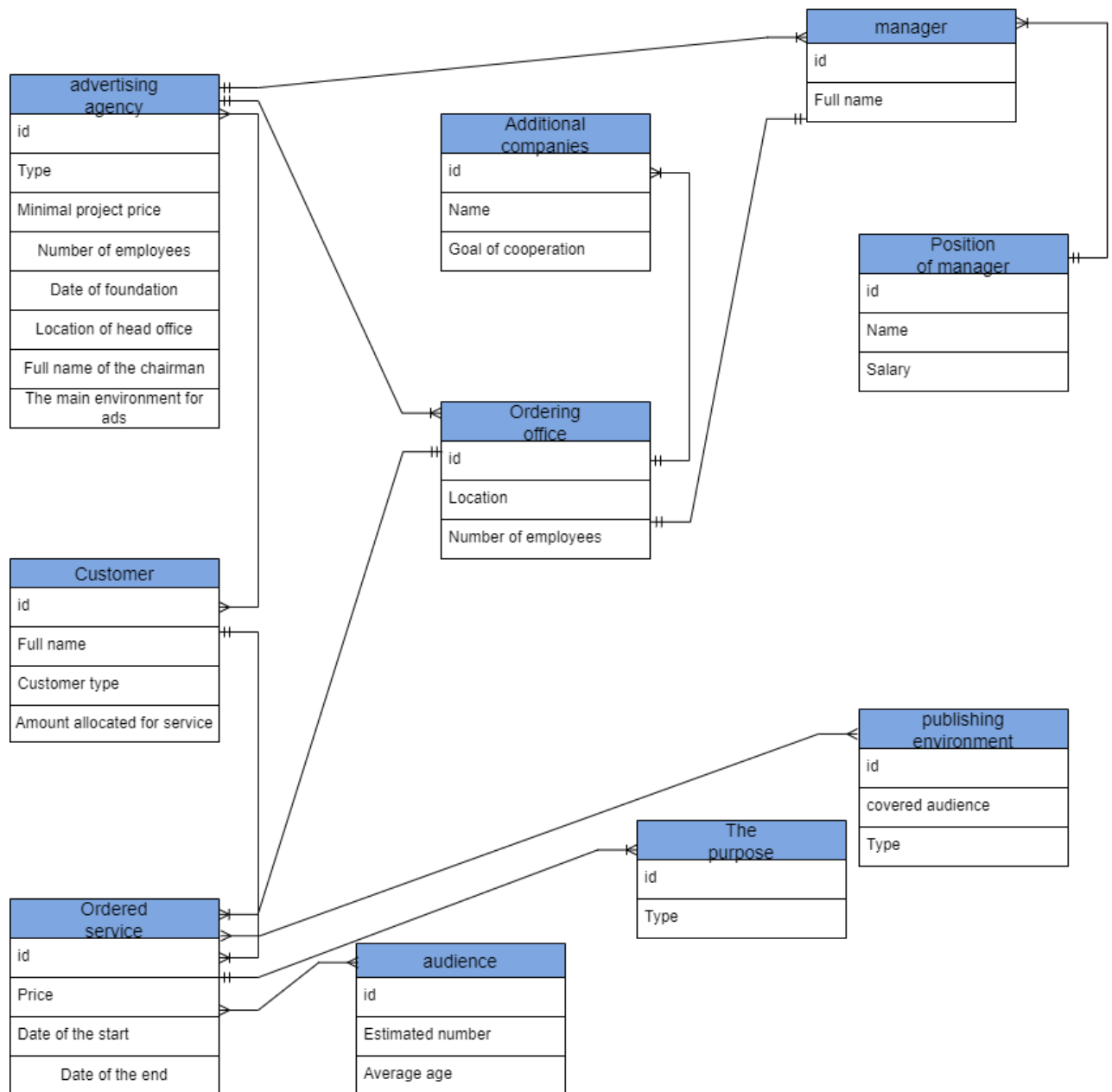


Рис 3.1

Висновок розділу: ми описали таблиці та навели концептуальну ER модель

4 Логічна Модель Бази Даних

Для втілення зв'язків багато до багатьох потрібно додати декілька допоміжних

таблиць:

- company_customer – для зв’язку агентства та замовника
- service_environment – для зв’язку послуги і середовища публікації
- service_audience – для зв’язку послуги і аудиторії

Схеми таблиць зображені на таблицях 4.1.1 -

Таблиця 4.1.1 — Схеми агентства

Атрибут	Тип даних
id(PK)	int
Type	var_char
Minimal_project_price	int
Number of employees	int
Location of head office	var_char
Full_name_of_the_chairman	var_char
The_main_environment_for_ads	var_char
id_customer(FK)	int

Таблиця 4.1.2 — Схеми замовника

Атрибут	Тип даних
id(PK)	int
Full_name	var_char
Customer_type	var_char

Amount_allocated_for_service	int
id_agency(FK)	int

Таблиця 4.1.3 Схема замовленої послуги

Атрибут	Тип даних
id(PK)	int
Price_of_service	int
Date_of_the_start	date
Date_of_the_end	date
id_customer(FK)	int
id_audience(FK)	int
id_office(FK)	int
id_environment(FK)	int

Таблиця 4.1.4 Схема аудиторії

Атрибут	Тип даних
id(PK)	int
Estimated_number	int
Average_age	int
id_service(FK)	int

Таблиця 4.1.5 Схема мети

Атрибут	Тип даних
id(PK)	int

Type	var_char
id_service(FK)	int

Таблиця 4.1.6 Схема середовища публікації

Атрибут	Тип даних
id(PK)	int
Type	var_char
Covered_audience	var_char
id_service(FK)	int

Таблиця 4.1.7 Схема офісу, що виконує замовлення

Атрибут	Тип даних
id(PK)	int
Number_of_employees	int
Location	var_char
id_manager(FK)	int

Таблиця 4.1.8 Схема додаткових компаній

Атрибут	Тип даних
id(PK)	int
Goal_of_cooperation	var_char
Name	var_char
id_orderingOffice(FK)	int

Таблиця 4.1.9 Схема керівника виконання проекту

Атрибут	Тип даних
id(PK)	int
Full_name	var_char
id_orderingOffice(FK)	int
id_position (FK)	int
id_agency(FK)	int

Таблиця 4.1.10 Схема посади керівника проекту

Атрибут	Тип даних
id(PK)	int
name	var_char
salary	double

Допоміжна таблиця 4.1.11 Схема зв'язку замовника і агентства

Атрибут	Тип даних
id_agency(PK)(FK)	int
id_customer(PK)(FK)	int

Допоміжна таблиця 4.1.12 Схема зв'язку замовлення та аудиторії

Атрибут	Тип даних
id_service(PK)(FK)	int
id_audience(PK)(FK)	int

Допоміжна таблиця 4.1.12 Схема зв'язку замовлення та середовища

Атрибут	Тип даних
id_service(PK)(FK)	int
id_environment(PK)(FK)	int

Тоді кінцева ER діаграма виглядатиме так(рис 4.1)

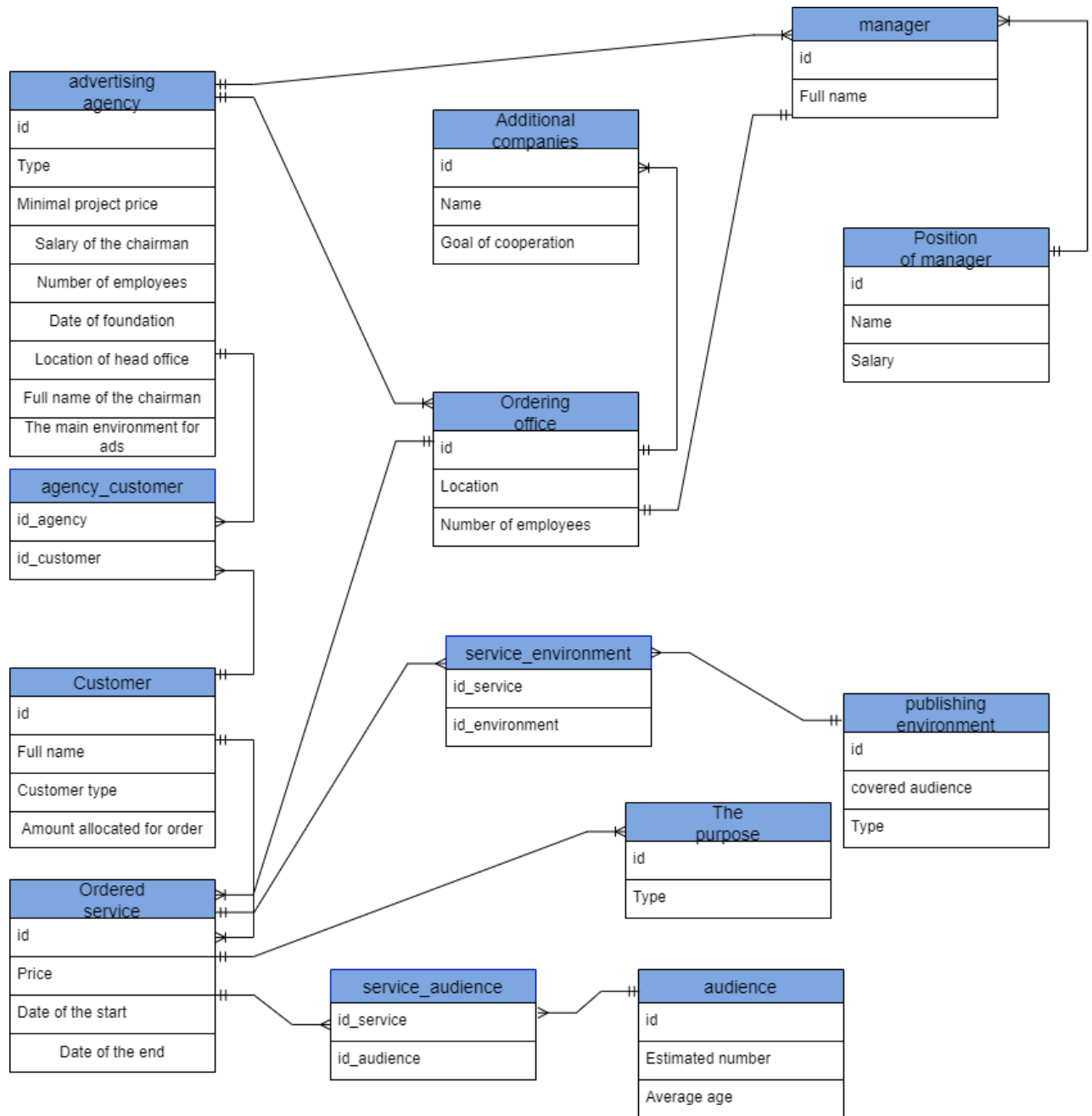


рис 4.1

Висновок розділу: в результаті цього розділу ми детально описали поля таблиць та навели кінцевий вигляд логічної моделі БД з таблицями для підтримки зв'язків багато до багатьох

5 Реалізація Баз Даних

5.1 Проектування БД

База даних була спроектована за допомогою MySQL Workbench через зручний інтерфейс.

Нижче наведений код для створення таблиць та їх полів

```
create table advertesingAgency (  
    id int not null auto_increment,  
    type_ varchar(64),  
    minimalProjectPrice int,  
    numberOfEmployees int not null,  
    dateOfFoundation date,  
    locationOfHeadOffice varchar(64),  
    fullNameOfTheChairman varchar(64),  
    theMainEnvironmentForAds varchar(64),  
    primary key(id),  
    id_customer int not null  
)ENGINE=InnoDB DEFAULT CHARSET=utf8mb4  
COLLATE=utf8mb4_0900_ai_ci;
```

```
create table customer (  
    id int not null auto_increment,  
    fullName varchar(64),  
    customerType varchar(64),  
    amountAllocatedForService int,  
    primary key(id),  
    id_agency int not null
```

**)ENGINE=InnoDB DEFAULT CHARSET=utf8mb4
COLLATE=utf8mb4_0900_ai_ci;**

**create table agency_customer (
 id_agency int not null,
 id_customer int not null,
 primary key(id_agency,id_customer),
 constraint id_agency foreign key(id_agency) references advertesingAgency(id)
on update cascade on delete cascade,
 constraint id_customer foreign key(id_customer) references customer(id) on
update cascade on delete cascade
)ENGINE=InnoDB DEFAULT CHARSET=utf8mb4
COLLATE=utf8mb4_0900_ai_ci;**

**create table orderingOffice(
 id int not null auto_increment,
 location varchar(64),
 numberOfEmployees int,
 primary key(id)
)ENGINE=InnoDB DEFAULT CHARSET=utf8mb4
COLLATE=utf8mb4_0900_ai_ci;**

**create table orderedService (
id int not null auto_increment,
price int not null,
dateOfTheStart date,
dateOfTheEnd date,
primary key(id),**

```
id_customer int not null,  
constraint fk2_id_customer foreign key(id_customer) references customer(id) on  
update cascade on delete cascade,  
id_office int not null,  
constraint fk1_id_office foreign key(id_office) references orderingoffice(id) on  
update cascade on delete cascade  
)ENGINE=InnoDB DEFAULT CHARSET=utf8mb4  
COLLATE=utf8mb4_0900_ai_ci;
```

```
create table audience (  
    id int not null auto_increment,  
    estimatedNumber int,  
    avrageAge int,  
    primary key(id),  
    id_service int not null  
)ENGINE=InnoDB DEFAULT CHARSET=utf8mb4  
COLLATE=utf8mb4_0900_ai_ci;
```

```
create table audience_service (  
    id_service int not null,  
    id_audience int not null,  
    primary key(id_service,id_audience),  
    constraint fk1_id_service foreign key(id_service) references  
orderedService(id) on update cascade on delete cascade,  
    constraint fk1_id_audience foreign key(id_audience) references audience(id)  
on update cascade on delete cascade  
)ENGINE=InnoDB DEFAULT CHARSET=utf8mb4  
COLLATE=utf8mb4_0900_ai_ci;
```



```
create table thePurpose (  
    id int not null auto_increment,  
    type_ varchar(64),  
    primary key(id),  
    id_service int not null,  
    constraint fk2_id_service foreign key(id_service) references  
orderedService(id) on update cascade on delete cascade  
)ENGINE=InnoDB DEFAULT CHARSET=utf8mb4  
COLLATE=utf8mb4_0900_ai_ci;
```

```
create table publishingEnvironment (  
    id int not null auto_increment,  
    coveredAudience int,  
    type_ varchar(64),  
    primary key(id)  
)ENGINE=InnoDB DEFAULT CHARSET=utf8mb4  
COLLATE=utf8mb4_0900_ai_ci;
```

```
create table service_environment (  
    id_service int not null,  
    id_environment int not null,  
    primary key(id_service,id_environment),  
    constraint fk3_id_service foreign key(id_service) references  
orderedService(id) on update cascade on delete cascade,  
    constraint fk1_id_environment foreign key(id_environment) references  
publishingenvironment(id) on update cascade on delete cascade
```

```
)ENGINE=InnoDB DEFAULT CHARSET=utf8mb4  
COLLATE=utf8mb4_0900_ai_ci;
```

```
create table additionalCompanie (  
    id int not null auto_increment,  
    name_ varchar(64),  
    goalOfCooperation varchar(64),  
    primary key(id),  
    id_office int not null,  
    constraint fk2_id_office foreign key(id_office) references  
orderingoffice(id) on update cascade on delete cascade  
)ENGINE=InnoDB DEFAULT CHARSET=utf8mb4  
COLLATE=utf8mb4_0900_ai_ci;
```

```
create table positionOfManager (  
    id int not null auto_increment,  
    name_ varchar(64),  
    salary int,  
    primary key (id)  
)ENGINE=InnoDB DEFAULT CHARSET=utf8mb4  
COLLATE=utf8mb4_0900_ai_ci;
```

```
create table manager (  
    id int not null auto_increment,  
    fullName varchar(64),  
    primary key (id),  
    constraint fk1_id foreign key(id) references orderingoffice(id) on update  
cascade on delete cascade,
```

```
id_agency int not null,  
constraint fk_id_agency foreign key(id_agency) references  
advertisagency(id) on delete cascade on update cascade,  
id_position int not null,  
constraint fk_id_position foreign key(id_position) references  
positionOfManager(id) on delete cascade on update cascade  
)ENGINE=InnoDB DEFAULT CHARSET=utf8mb4  
COLLATE=utf8mb4_0900_ai_ci;
```

На рисунку 5.1 зображена схема побудована в результаті виконання описаного вище коду.

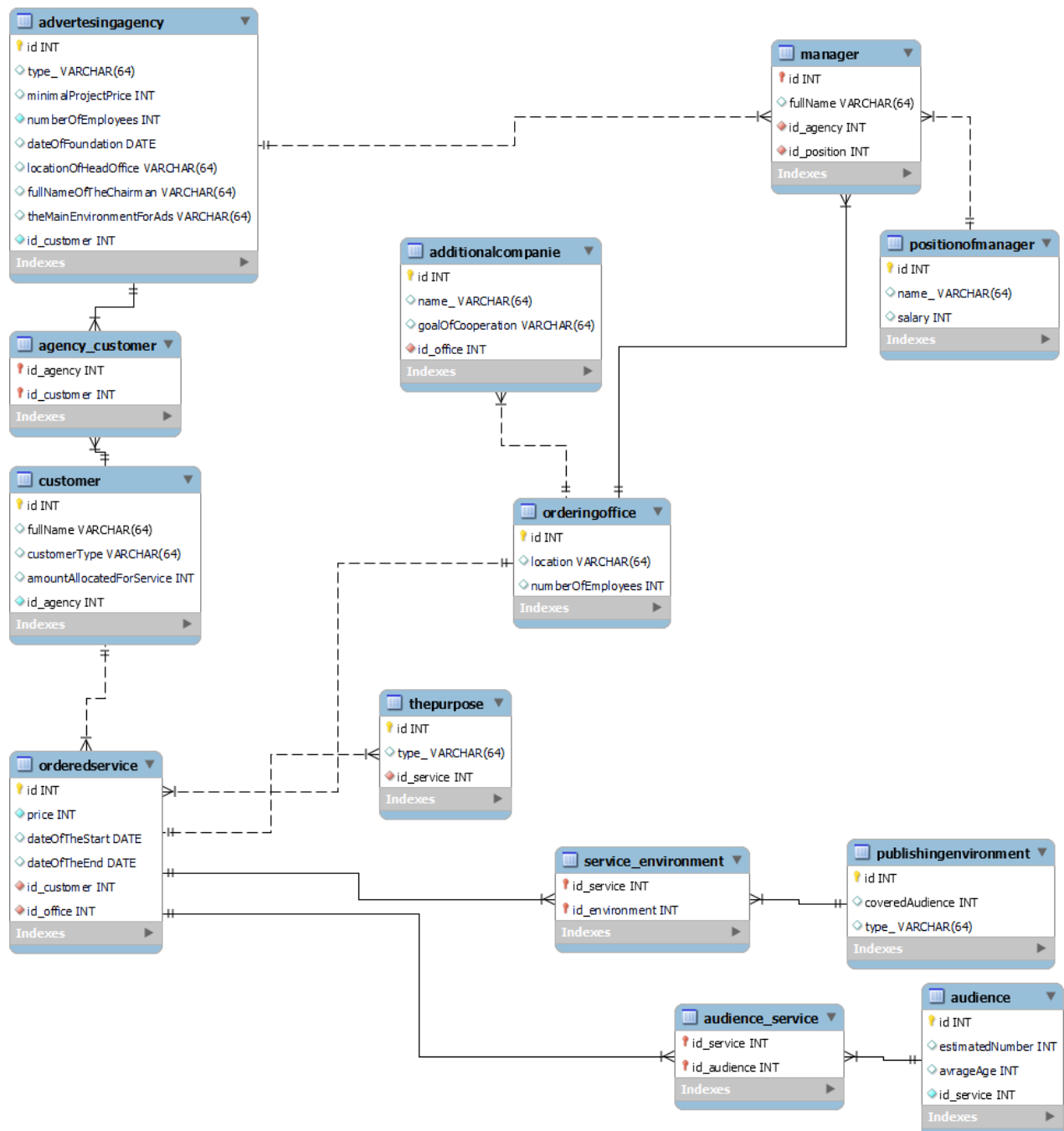


рис 5.1

5.2 Створення обмежень

1)Обмеження на довжину імені керівника проекту

alter table manager

**add constraint checkLongevityOfTheName check
(CHAR_LENGTH(fullName)<100);**

2)Обмеження на мінімальну ціну проекту

alter table advertesingagency

**add constraint checkMinPrice check(minimalProjectPrice>=1000 and
minimalProjectPrice<=1000000);**

3)Обмеження на зарплату керівників проектів

alter table positionOfManager

add constraint checkSallary check(salary>=1000 and salary<=10000);

4)Обмеження на розмір аудиторії

alter table audience

**add constraint checkNumber check(estimatedNumber>=100 and
estimatedNumber<=100000);**

5)Обмеження на кількість працівників

alter table orderingoffice

**add constraint checkNumOfWorkers check(numberOfEmployees>=40 and
numberOfEmployees<=400);**

6)Обмеження на вік аудиторії

alter table audience

add constraint checkAge check(averageAge>=1 and averageAge<=150);

7)Обмеження на дати проектів

alter table orderedService

add constraint checkDate check(dateOfTheEnd>dateOfTheStart);

5.3 Створення користувачів

Для підвищення зручності та безпеки нам потрібно створити користувачів, у кожного з яких буде свій діапазон прав та певні обмеження користування системою. Не кожен користувач має мати права для повної взаємодії з БД, тому ми маємо розподілити ці права по користувачам.

Ми можемо це зробити за допомогою SQL скриптів.

Створимо користувача **kursova_admin1** у якого будуть права на будь які зміни у базі даних **kursova**:

```
create user kursova_admin1@localhost identified by "admin123";
```

Користувача **kursova_customer1** у якого будуть права лише на перегляд даних у базі даних **kursova**:

```
create user kursova_customer1@localhost identified by "customer123";
```

Нам потрібно створити відповіді ролі, яким будуть надані бажані нам права та надати ці ролі користувачам:

```
create role admin, customer;
```

```
grant select on kursova.* to customer;
```

```
grant all on kursova.* to admin;
```

```
grant customer to kursova_customer1@localhost;
```

```
grant admin to kursova_admin1@localhost;
```

Таблиця 5.3.1 з логінами, пароллями та правами користувачів

Логін	Пароль	Права
-------	--------	-------

kursova_admin1	admin123	усі
kursova_customer1	customer123	права на перегляд на перегляд

5.4 Імпорт записів до БД

Обрана СУБД - MySQL підтримує імпорт даних. Завчасно згенеровані файли розширення .csv можна імпортувати за допомогою MySQL Workbench, якщо слідувати наступним крокам.

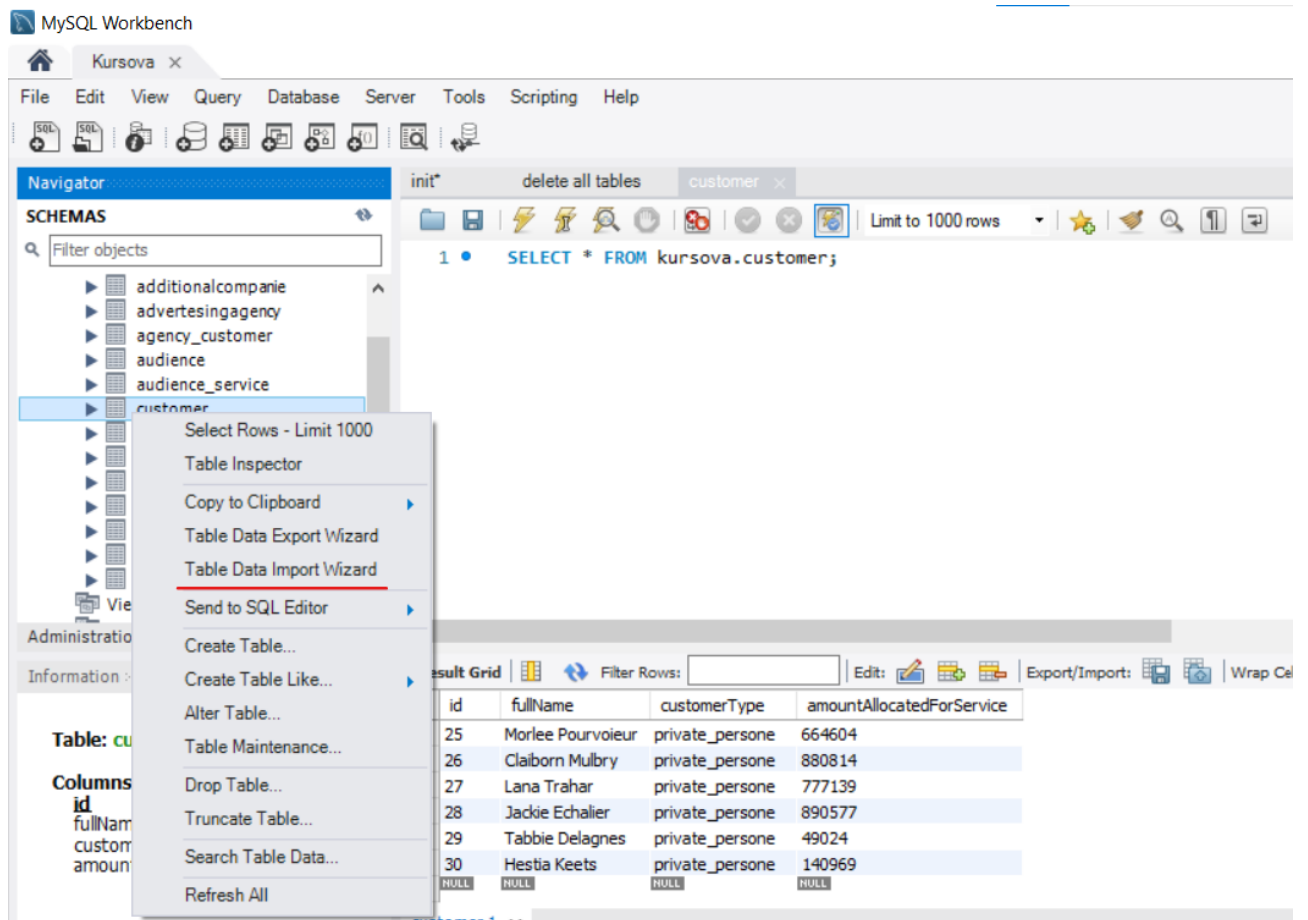


рис. 5.4.1

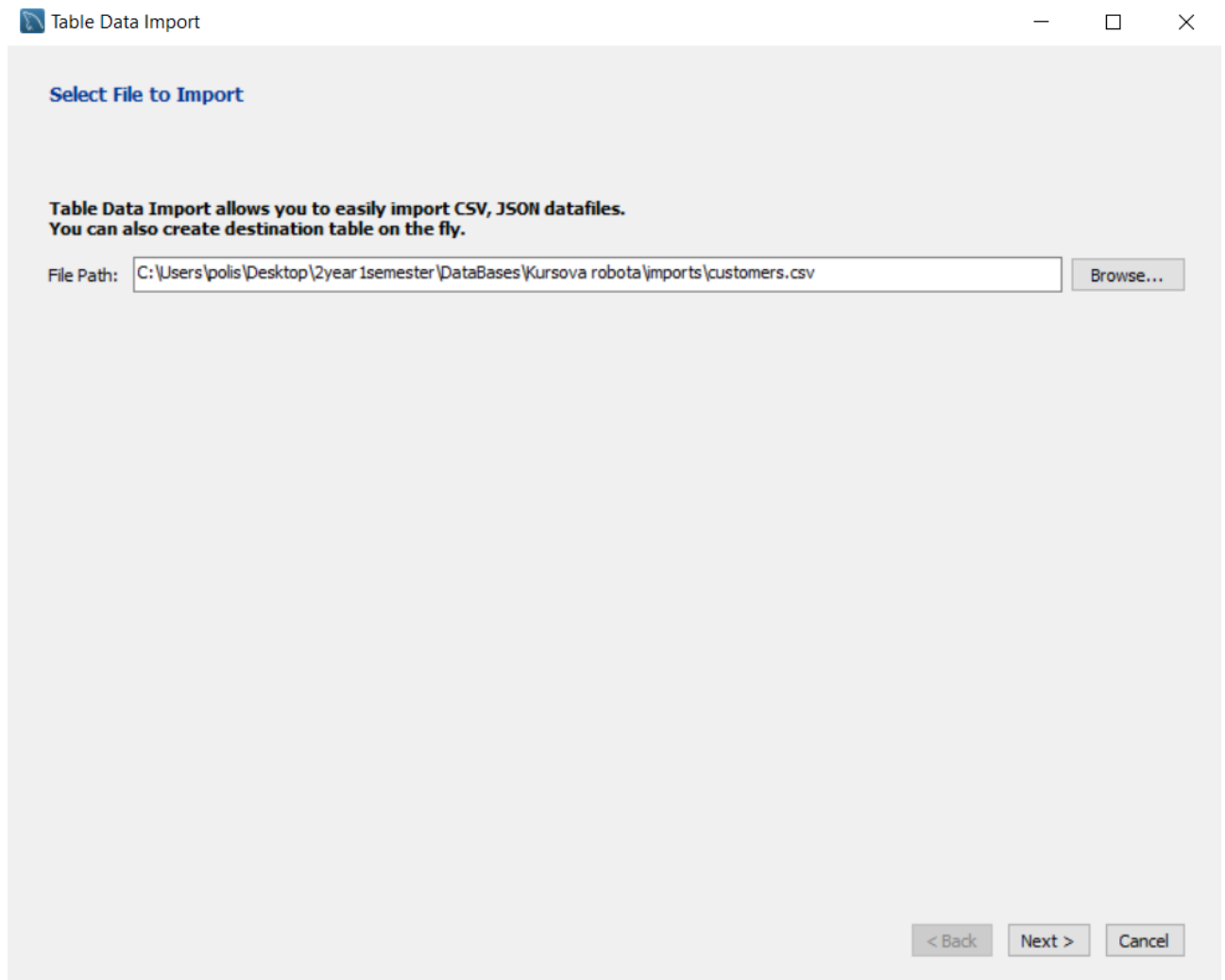


рис 5.4.2

Далі необхідно налаштувати параметри імпорту і запустити його.

5.5 Реалізація запитів

1) Обрати менеджера з конкретним ім'ям

select

advertisingsagency.id,advertisingsagency.agencyName,advertisingsagency.location

OfHeadOffice, customer.fullname as "cutomer`s fullname"

from advertisingsagency

join agency_customer on advertisingsagency.id = agency_customer.id_agency

join customer on customer.id = agency_customer.id_customer

where advertisingsagency.agencyName like("A%")

	id	agencyName	locationOfHeadOffice	customer`s fullname
►	1	Aenean Eget LLC	Luik	Charity Ekless
	22	At Egestas A Corporation	Oregon	Fan Goodere

2)Обрати аудиторію у межах середнього віку

```
select audience.*,orderservice.price as "service price"
from audience
join audience_service on audience_service.id_audience = audience.id
join orderservice on orderservice.id = audience_service.id_service
where audience.averageAge between 18 and 28
```

	id	estimatedNumber	averageAge	service price
►	8	49956	21	31962
	19	47223	26	836873

3)Обрати замовників з сумою, яка може покрити витрати

```
select customer.id,customer.fullName,advertisingagency.agencyName
from customer
join agency_customer on customer.id = agency_customer.id_customer
join advertisingagency on advertisingagency.id = agency_customer.id_agency
where customer.amountAllocatedForService >
advertisingagency.minimalProjectPrice
```

	id	fullName	agencyName
▶	1	Charity Ekless	Aenean Eget LLC
	2	Genovera Croix	Elit Etiam Laoreet LLC
	3	Shayla Antliff	Pretium Aliquet Institute
	8	Zonnya Blanden	Risus Quisque Libero LLP
	15	Andromache Plitz	Euismod Urna Company
	18	Lee Daine	Gravida Aliquam Institute

4)Обрати замовників з конкретним id

```
select customer.fullName as "fullName of customer",
advertesingagency.agencyName
from customer
join agency_customer on customer.id = agency_customer.id_customer
join advertesingagency on advertesingagency.id = agency_customer.id_agency
where customer.id in (1,5,8,20)
```

	fullName of customer	agencyName
▶	Charity Ekless	Aenean Eget LLC
	Sapphira Kermitt	Orci Luctus Incorporated
	Zonnya Blanden	Risus Quisque Libero LLP
	Becca Karolyi	Nulla Facilisis Suspendisse LLC

5)Відобразити скільки разів користувач може оплатити послугу,якщо не 0

```
select
customer.id,floor(customer.amountAllocatedForService/orderedservice.price) as
"Paid services"
from customer
join orderedservice on orderedservice.id_customer = customer.id
where floor(customer.amountAllocatedForService/orderedservice.price) > 0
```

	id	Paid services
▶	1	7
	2	25
	3	7
	8	18
	9	1
	12	6

6)Обрати менеджерів з специфічним ім'ям за допомогою regex
select manager.id,manager.fullName,positionofmanager.salary
from manager
join positionofmanager on positionofmanager.id = manager.id_position
where manager.fullName regexp '^A' and manager.fullName regexp 'n\$'

	id	fullName	salary
▶	1	Alea Osborn	3647
	22	Althea Johnston	4662

7)Обрати менеджера з посадою, що входить в перелічений список
select manager.id,manager.fullName, positionofmanager.name_ as "position of
manager"
from manager
join positionofmanager on manager.id_position = positionofmanager.id
where positionofmanager.name_ in ('Sales Manager','Administrative Services
Manager')

	id	fullName	position of manager
▶	1	Alea Osborn	Sales Manager
	2	Tanner Whitfield	Sales Manager
	7	Vaughan Vasquez	Administrative Services Manager
	8	Wanda Bender	Administrative Services Manager
	9	Pascale Stanley	Sales Manager
	10	Ali Hewitt	Sales Manager
	15	Hedy Saunders	Administrative Services Manager

8)Обрати послуги з некомерційною метою

```
select orderedservice.*  
from orderedservice  
join thepurpose on thepurpose.id_service=orderedservice.id  
where thepurpose.type_ != "commercial"
```

	id	price	dateOfTheStart	dateOfTheEnd	id_customer	id_office
►	3	135178	2020-12-28	2021-05-05	3	3
	4	900158	2020-12-23	2021-09-25	4	4
	7	571158	2020-12-29	2022-12-05	7	7
	8	31962	2020-12-25	2021-05-11	8	8
	11	578739	2020-12-19	2021-09-14	11	11
	12	48014	2020-12-18	2022-09-22	12	12
	15	125751	2021-01-02	2022-04-01	15	15
	16	431659	2020-12-29	2021-12-29	16	16
	19	836873	2020-12-24	2021-08-04	19	19

9)Обрати офіси з певним розташуванням

```
select orderingoffice.id, orderingoffice.location,  
orderingoffice.numberOfWorkers,advertisingagency.agencyName  
from orderingoffice  
join advertisingagency on advertisingagency.id = orderingoffice.id_agency  
where orderingoffice.location = "Boston"
```

	id	location	numberOfWorkers	agencyName
►	2	Boston	147	Elit Etiam Laoreet LLC

10)Обрати розташування офісів менеджерів і відсортувати за ім'ям

```
select manager.id,manager.fullName,orderingoffice.location  
from manager  
join orderingoffice on orderingoffice.id = manager.id  
order by fullName
```

	id	fullName	location
►	1	Alea Osborn	Grand-Manil
	10	Ali Hewitt	Ciplet
	22	Althea Johnston	Hillsboro
	18	Alyssa Ball	Ethe
	13	Basil Stephenson	Ehein
	11	Bianca David	Kenosha
	23	Brynn Deleon	Sioux City
	17	Castor Curtis	Metairie
	14	Ferdinand Hale	Brussel

11)Обрати офіси з специфічною кількістю працівників

```
select orderingoffice.id, orderingoffice.location,
orderingoffice.numberOfWorkers,advertisingagency.agencyName
from orderingoffice
join advertisingagency on advertisingagency.id = orderingoffice.id_agency
where orderingoffice.numberOfWorkers in (201,131,284);
```

	id	location	numberOfWorkers	agencyName
►	1	Grand-Manil	201	Aenean Eget LLC
	3	Knokke-Heist	131	Pretium Aliquet Institute
	16	Buggenhout	284	Duis Mi Institute

12)Обрати офіси з специфічними підсобними компаніями

```
select orderingoffice.*,additionalcompany.name _ as "additionalcompany"
from additionalcompany
join orderingoffice on additionalcompany.id_office = orderingoffice.id
where additionalcompany.goalOfCooperation = "transportation" or
additionalcompany.goalOfCooperation = "security_services"
```

	id	location	numberOfEmployees	id_agency	additionalcompanie
►	1	Grand-Manil	201	1	Nulla Vulputate PC
	2	Boston	147	2	Vivamus Nibh Dolor Corporation
	5	Montignies-sur-Sambre	206	5	Bibendum LLC
	6	Tucson	322	6	Sagittis Semper Foundation
	7	Ramsdonk	164	7	Diam Inc.
	8	Avin	214	8	Tristique Aliquet Inc.
	11	Kenosha	111	11	Cursus In PC
	12	Provo	331	12	Lorem Eu Inc

13) Обрати офіси, які підпорядковуються агентствам, якого типу

select orderingoffice.*,advertisagency.agencyName

from orderingoffice

join advertisagency on advertisagency.id = orderingoffice.id_agency

where advertisagency.type_ = "Full-service"

	id	location	numberOfEmployees	id_agency	agencyName
►	1	Grand-Manil	201	1	Aenean Eget LLC
	2	Boston	147	2	Elit Etiam Laoreet LLC
	7	Ramsdonk	164	7	Sem LLP
	8	Avin	214	8	Risus Quisque Libero LLP
	13	Ehein	370	13	Ipsum Ac LLP
	14	Brussel	216	14	Nec Institute
	19	Sint-Pieters-Kapelle	262	19	Sed Hendrerit A LLC
	20	Tonnerlo	147	20	Nulla Facilisis Suspendisse LLC

14)Отримати імена менеджерів і їх зарплати

```
select manager.id,manager.fullName,positionofmanager.salary  
from manager  
join positionofmanager on manager.id_position = positionofmanager.id
```

	id	fullName	salary
►	1	Alea Osborn	3647
	2	Tanner Whitfield	4052
	3	Giacomo Barton	5626
	4	Kyra Porter	5208
	5	Ivana Mcclure	3485
	6	Pearl Hines	3954
	7	Vaughan Vasquez	7748
	8	Wanda Bender	6277

15)Отримати тільки ті послуги у яких аудиторія буде більше зазначеної

```
select orderedservice.*  
from orderedservice  
join service_environment on orderedservice.id = service_environment.id_service  
join publishingenvironment on publishingenvironment.id =  
service_environment.id_environment  
where coveredAudience>2000
```

	id	price	dateOfTheStart	dateOfTheEnd	id_customer	id_office
►	1	81748	2021-01-07	2022-04-09	1	1
	2	27914	2020-12-18	2022-07-13	2	2
	3	135178	2020-12-28	2021-05-05	3	3
	5	422678	2020-12-18	2021-03-03	5	5
	6	314681	2020-12-18	2022-07-14	6	6
	7	571158	2020-12-29	2022-12-05	7	7
	8	31962	2020-12-25	2021-05-11	8	8
	9	305622	2020-12-31	2022-04-19	9	9
	10	548453	2020-12-24	2021-03-14	10	10
	11	578739	2020-12-19	2021-09-14	11	11

16) Отримати замовлення, де офіси , які над ними працюють, налічують більше зазначеної кількості працівників

```
select orderedservice.id,orderedservice.price,dateOfTheStart,dateOfTheEnd
from orderedservice
join orderingoffice on orderingoffice.id = orderedservice.id_office
where orderingoffice.numberOfEmployees > 100
```

	id	price	dateOfTheStart	dateOfTheEnd	id_customer	id_office
►	1	81748	2021-01-07	2022-04-09	1	1
	2	27914	2020-12-18	2022-07-13	2	2
	3	135178	2020-12-28	2021-05-05	3	3
	5	422678	2020-12-18	2021-03-03	5	5
	6	314681	2020-12-18	2022-07-14	6	6

17)Визначити некомерційні замовлення, що публікуватимуться в інтернеті

```
select orderedservice.*
from orderedservice
join thepurpose on thepurpose.id_service=orderedservice.id
join service_environment on service_environment.id_service = orderedservice.id
join publishingenvironment on publishingenvironment.id =
service_environment.id_environment
where thepurpose.type_ = "noncommercial"
and publishingenvironment.type_ in ('Internet')
```

	id	price	dateOfTheStart	dateOfTheEnd	id_customer	id_office
►	3	135178	2020-12-28	2021-05-05	3	3
	4	900158	2020-12-23	2021-09-25	4	4
	11	578739	2020-12-19	2021-09-14	11	11
	12	48014	2020-12-18	2022-09-22	12	12
	19	836873	2020-12-24	2021-08-04	19	19
	20	341784	2020-12-29	2021-06-12	20	20

18)Визначити сервіси виконання яких починається з певної дати

```
select orderedservice.*,customer.fullName
```


from orderedservice

join customer on customer.id = orderedservice.id_customer

where orderedservice.dateOfTheStart > "2020-12-19"

	id	price	dateOfTheStart	dateOfTheEnd	id_customer	id_office	fullName
►	1	81748	2021-01-07	2022-04-09	1	1	Charity Ekless
	3	135178	2020-12-28	2021-05-05	3	3	Shayla Antliff
	4	900158	2020-12-23	2021-09-25	4	4	Charlton Ellett
	7	571158	2020-12-29	2022-12-05	7	7	Gayel Dessent
	8	31962	2020-12-25	2021-05-11	8	8	Zonnya Blanden
	9	305622	2020-12-31	2022-04-19	9	9	Wolfie Gourlie
	10	548453	2020-12-24	2021-03-14	10	10	Flem Corneille
	14	683807	2020-12-22	2021-11-28	14	14	Griffy Simmans
	15	125751	2021-01-02	2022-04-01	15	15	Andromache Plitz
	16	431659	2020-12-29	2021-12-29	16	16	Inesita Persehouse

19) Визначити який замовник відповідає якому агентству

select customer.*,advertisingagency.*

from customer

join agency_customer on customer.id = agency_customer.id_customer

join advertisingagency on advertisingagency.id = agency_customer.id_agency

	id	fullName	customerType	amountAllocatedForService	id	agencyName	type_	minimalProjectPrice	numberOfEmployees	dateOfFoundat
►	1	Charity Ekless	private_person	644461	1	Aenean Eget LLC	Full-service	455531	353	2022-08-10
	2	Genovera Croix	private_person	725209	2	Elit Etiam Laoreet LLC	Full-service	64121	389	2021-11-26
	3	Shayla Antliff	private_person	968676	3	Pretium Aliquet Institute	Digital	960914	92	2022-08-22
	4	Charlton Ellett	private_person	50851	4	Sit Amet Metus Associates	Digital	90630	269	2022-10-09
	5	Sapphira Kermitt	private_person	254073	5	Orci Luctus Incorporated	Traditional	645746	382	2022-06-27
	6	Rosella Izard	private_person	284836	6	Dignissim Magna Inc.	Traditional	748835	288	2022-03-24
	7	Gayel Dessent	private_person	541978	7	Sem LLP	Full-service	659195	199	2021-05-23
	8	Zonnya Blanden	private_person	581292	8	Risus Quisque Libero LLP	Full-service	198081	47	2022-06-17
	9	Wolfie Gourlie	private_person	378700	9	Tellus Corp.	Digital	564878	290	2021-02-05

20) Визначити відповідну до замовлення аудиторію

select audience.*,orderedservice.*

from audience

join audience_service on audience.id = audience_service.id_audience

join orderedservice on orderedservice.id = audience_service.id_service

	id	estimatedNumber	averageAge	id	price	dateOfTheStart	dateOfTheEnd	id_customer	id_office
▶	1	25825	68	1	81748	2021-01-07	2022-04-09	1	1
	2	34139	66	2	27914	2020-12-18	2022-07-13	2	2
	3	27239	150	3	135178	2020-12-28	2021-05-05	3	3
	4	35064	2	4	900158	2020-12-23	2021-09-25	4	4
	5	31003	133	5	422678	2020-12-18	2021-03-03	5	5
	6	7625	110	6	314681	2020-12-18	2022-07-14	6	6
	7	35874	135	7	571158	2020-12-29	2022-12-05	7	7
	8	49956	21	8	31962	2020-12-25	2021-05-11	8	8
	9	43658	104	9	305622	2020-12-31	2022-04-19	9	9
	10	25978	93	10	548453	2020-12-24	2021-03-14	10	10

5.7 Реалізація тригерів

1) Тригер при оновленні, щоб відсікати небажані призначення компаній

delimiter //

create trigger updateAdditionalComp

before update

on additionalcompanie

for each row

begin

if new.goalOfCooperation not in

("security_services", 'medical_services', 'transportation')

then set new.goalOfCooperation = "different type";

end if;

end; //

delimiter ;

2) Тригер при вставці, щоб виправляти зарплату

delimiter //

create trigger checkPositionSalary before insert on positionofmanager

for each row if new.salary <= 0

then set new.salary = 1000;

end if; //

delimiter ;

3) Тригер для підрахунку кількості вствлених користувачів

set @num = 0;

#drop trigger custNum;

create trigger custNum after insert on customer

for each row set @num = @num + 1;

```
select @num as "number of added customers" ;
```

5.8 Реалізація індексів

1)Індекс для назв агентств

```
create index agencyName_idx  
on advertesingagency(agencyName);
```

```
|      8 | 0.00063325 | select agencyName  
from advertesingagency
```

рис 5.8.1 До оптимізації

```
|     10 | 0.00061475 | select agencyName  
from advertesingagency
```

рис 5.8.2 Після оптимізації

2)Індекс на ім'я замовника

```
create index fullname_idx  
on customer(fullName);
```

```
|     11 | 0.00024675 | set profiling = 1  
|     12 | 0.00135350 | select fullName  
from customer
```

рис 5.8.1 До оптимізації

```
|     13 | 0.00015850 | set profiling = 1  
|     14 | 0.00097225 | select fullName  
from customer
```

рис 5.8.2 Після оптимізації

5.9 Реалізація функцій

1)Перевірка чи бюджет більше за мінімальну ціну проекту

create function checkPrices(minimalP int,budget int)

returns bool

return minimalP<=budget;

2)Перевірка назв агентств

delimiter //

create function agencyNameCheck(agencyName varchar(64))

returns varchar(64)

begin

declare result varchar(64);

if agencyName like('A%') then set result = agencyName;

else set result = "wrong";

end if;

return (result);

end//

delimiter ;

3)Перевірка чи входить індекс в сукупність

create function checkId (id int)

returns bool

return id in (1,5,8,20)

5.10 Реалізація представлень

Для кожного запиту було реалізоване представлення

1)

```
create view agencyName  
as  
select  
advertisagency.id,advertisagency.agencyName,advertisagency.location  
OfHeadOffice, customer.fullname as "cutomer`s fullname"  
from advertisagency  
join agency_customer on advertisagency.id = agency_customer.id_agency  
join customer on customer.id = agency_customer.id_customer  
where advertisagency.agencyName like("A%")
```

2)

```
create view audieceAgeRange  
as  
select audience.*,orderservice.price as "service price"  
from audience  
join audience_service on audience_service.id_audience = audience.id  
join orderservice on orderservice.id = audience_service.id_service  
where audience.averageAge between 18 and 28
```

3)

```
create view cutomers_That_Can_Pay  
as  
select customer.id,customer.fullName,advertisagency.agencyName  
from customer  
join agency_customer on customer.id = agency_customer.id_customer
```

**join advertesingagency on advertesingagency.id = agency_customer.id_agency
where customer.amountAllocatedForService >
advertesingagency.minimalProjectPrice**

4)

create view specific_id_cutomer

as

select customer.fullName as "fullName of customer",

advertesingagency.agencyName

from customer

join agency_customer on customer.id = agency_customer.id_customer

join advertesingagency on advertesingagency.id = agency_customer.id_agency

where customer.id in (1,5,8,20)

5)

create view number_paid_services_per_customer

as

select

customer.id, floor(customer.amountAllocatedForService/orderedservice.price) as

"Paid services"

from customer

join orderedservice on orderedservice.id_customer = customer.id

where floor(customer.amountAllocatedForService/orderedservice.price) > 0

6)

create view specific_managers_name

as

select manager.id,manager.fullName,positionofmanager.salary

from manager

join positionofmanager on positionofmanager.id = manager.id_position

where manager.fullName regexp '^A' and manager.fullName regexp 'n\$'

7)

create view select_managers_with_certain_position

as

select manager.id,manager.fullName, positionofmanager.name_ as "position of manager"

from manager

join positionofmanager on manager.id_position = positionofmanager.id

where positionofmanager.name_ in ('Sales Manager','Administrative Services Manager')

8)

create view select_non_comercial_services

as

select orderedservice.*

from orderedservice

join thepurpose on thepurpose.id_service=orderedservice.id

where thepurpose.type_ != "commercial"

9)

create view select_office_of_manager

as

select manager.id,manager.fullName,orderingoffice.location


```
from manager  
join orderingoffice on orderingoffice.id = manager.id  
order by fullName
```

10)

```
create view select_office_with_certain_location  
as  
select orderingoffice.id, orderingoffice.location,  
orderingoffice.numberOfWorkers,advertisingagency.agencyName  
from orderingoffice  
join advertisingagency on advertisingagency.id = orderingoffice.id_agency  
where orderingoffice.location = "Boston"
```

Висновок розділу:

У цьому розділі ми продемонстрували перенесення БД у код, створення обмежень, функцій, запитів, індексів та представлення

6 Висновки

При виконанні цієї курсової роботи було спроектовано базу даних рекламного агентства, були реалізовані та проаналізовані такі аспекти мови програмування SQL, як індекси, представлення, робота з функціями, обмеження та dml запити. Як результат ми створили робочу базу даних для підтримки рекламного агентства

7 Список використаної літератури

1. <https://www.mysqltutorial.org/> - опис великого об'єму функціоналу
2. <https://generatedata.com/generator> - сервіс для генерації даних у .csv
3. <https://stackoverflow.com/> - використане для знаходження деяких рішень
4. <https://dev.mysql.com/doc/> - документація використаного СУБД