

Міністерство освіти і науки України
Національний університет “Львівська політехніка”
Інститут прикладної математики та фундаментальних наук
Кафедра прикладної математики

Курсова робота
з курсу “Робота з великими базами даних”
на тему:
“Студент”
Варіант 1

Виконав:
студент групи ПМ-41
Кісельов Олександр
Прийняв:
Любінський Богдан Богданович

(дата)

(підпис викладача)

Львів — 2026

Анотація

Курсова робота присвячена розробці та аналізу інформаційної системи для предметної області "Студент". Основною метою роботи є побудова бази даних і сховища даних, які дозволяють ефективно зберігати, обробляти та аналізувати інформацію про гуртожитки, студентів, рейтинг успішності.

Робота охоплює такі етапи:

1. Розробка концептуальної моделі бази даних, яка описує ключові об'єкти предметної області та їх взаємозв'язки.
2. Генерація реалістичних даних для тестування системи.
3. Реалізація процесу ETL (витягування, трансформація, завантаження) з використанням SQL Server Integration Services (SSIS).
4. Побудова багатовимірного куба для аналітики за допомогою SQL Server Analysis Services (SSAS).
5. Розробка інтерактивних звітів із використанням SQL Server Reporting Services (SSRS).

Результати роботи демонструють можливість аналізу даних щодо розподілу студентів за гуртожитками, кімнатами та іншими параметрами, що сприяє підтримці прийняття управлінських рішень у сфері освіти.

Зміст

ВСТУП.....	2
Основна частина	4
1.Розділ 1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ.....	4
1.8 Декомпозиція бізнес-процесів предметної області	6
1.9 Аналіз обсягів та структури даних	7
1.10 Проблеми та недоліки традиційних підходів	7
1.11 Обґрунтування вибору технологій	7
1.12 Аналіз користувачів інформаційної системи.....	7
1.13 Порівняння OLTP та OLAP у контексті предметної області.....	8
1.14 Вимоги до даних з точки зору аналітики.....	8
1.15 Перспективи розвитку інформаційної системи	8
1.16 Висновки до розділу 1.....	8
2. РОЗДІЛ 2. ПРОЄКТУВАННЯ БАЗИ ДАНИХ	9
3. РОЗДІЛ 3. РЕАЛІЗАЦІЯ ETL-ПРОЦЕСІВ (SQL SERVER INTEGRATION SERVICES).....	17
4. РОЗДІЛ 4. ПОБУДОВА OLAP-КУБА (SSAS) ТА АНАЛІТИЧНА ВІЗУАЛІЗАЦІЯ (SSRS)	23
Висновки.....	37
Список використаної літератури	39

ВСТУП

У сучасних умовах стрімкого розвитку інформаційних технологій та цифровізації практично всіх сфер діяльності суспільства особливого значення набуває ефективне управління великими та надвеликими обсягами даних. Освітня сфера не є винятком, адже вищі навчальні заклади щоденно оперують значними масивами інформації, пов'язаними з навчальним процесом, контингентом студентів, викладацьким складом, результатами оцінювання, обліком навчального навантаження та іншими супутніми процесами. Обсяг таких даних зростає з кожним роком, що зумовлює необхідність використання сучасних підходів до їх зберігання, обробки та аналізу.

Традиційні реляційні бази даних, орієнтовані переважно на транзакційну обробку (OLTP), є ефективними для виконання щодених операцій введення, оновлення та зберігання даних. Однак при виконанні складних аналітичних запитів, які потребують агрегації даних за тривалий період часу, порівняння показників, аналізу тенденцій та формування зведених звітів, такі системи демонструють обмежені можливості та низьку продуктивність. Саме тому у практиці сучасних інформаційних систем широкого застосування набули сховища даних (Data Warehouse) та технології багатовимірного аналізу (OLAP).

Сховище даних дозволяє інтегрувати інформацію з різних джерел, зберігати історичні дані у зручній для аналізу формі та забезпечувати високу швидкодію аналітичних запитів. Використання багатовимірних моделей даних дає змогу розглядати інформацію з різних точок зору, формувати зрізи та агрегати за часом, об'єктами аналізу та іншими вимірами. Це є особливо важливим для керівництва навчального закладу, яке потребує достовірної та своєчасної аналітичної інформації для прийняття управлінських рішень.

Актуальність даної курсової роботи обумовлена необхідністю практичного застосування теоретичних знань з дисципліни «Надвеликі бази даних» для розробки повнофункціональної системи зберігання та аналізу великих обсягів освітніх даних. Побудова такої системи на базі платформи Microsoft SQL Server дозволяє використати комплекс сучасних інструментів, зокрема SQL Server Integration Services (SSIS) для реалізації ETL-процесів, SQL Server Analysis Services (SSAS) для побудови OLAP-кубів та SQL Server Reporting Services (SSRS) для створення аналітичних звітів.

Предметна область «Студент (Деканат)» є показовим прикладом системи, в якій накопичуються значні обсяги різномірних даних протягом тривалого періоду часу. До таких даних належать персональні відомості про студентів, інформація про навчальні групи, перелік дисциплін, результати оцінювання, дані про викладачів, а також супутня інформація, пов'язана з навчальним процесом. Аналіз цих даних дозволяє оцінити рівень успішності студентів,

виявити проблемні дисципліни, сформувати рейтинги та забезпечити прозорість навчального процесу.

Метою курсової роботи є розробка та аналіз надвеликої бази даних для предметної області «Студент (Деканат)» з використанням технологій Microsoft SQL Server, а також реалізація повного циклу обробки даних – від їх витягування та трансформації до побудови багатовимірного аналізу та формування аналітичних звітів.

Для досягнення поставленої мети у курсовій роботі необхідно вирішити такі основні завдання:

- проаналізувати предметну область та визначити основні бізнес-процеси;
- спроектувати концептуальну, логічну та фізичну моделі бази даних;
- реалізувати генерацію та наповнення бази великими обсягами реалістичних даних;
- розробити ETL-процеси з використанням SQL Server Integration Services;
- побудувати OLAP-куб у SQL Server Analysis Services з визначенням вимірів та мір;
- створити аналітичні звіти з використанням SQL Server Reporting Services;
- провести аналіз отриманих результатів та зробити відповідні висновки.

Об'єктом дослідження у даній роботі є інформаційні процеси обліку та аналізу даних навчального процесу у вищому навчальному закладі. Предметом дослідження є методи та засоби проектування надвеликих баз даних, сховищ даних та багатовимірного аналізу.

Практичне значення роботи полягає у можливості використання розробленої системи як основи для впровадження аналітичних рішень у діяльність деканату навчального закладу. Отримані результати можуть бути використані для підвищення ефективності управління навчальним процесом, покращення якості освіти та підтримки прийняття управлінських рішень.

Основна частина

1.Розділ 1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

У рамках роботи було здійснено побудову та аналіз даних із використанням інструментів SQL Server, таких як SSIS, SSAS та SSRS. Представлено ключові етапи, включаючи створення бази даних, розробку багатовимірного куба для аналізу даних, а також підготовку аналітичних звітів на основі отриманих результатів.

1.1 Загальна характеристика предметної області

Предметна область «Студент (Деканат)» охоплює сукупність організаційних, інформаційних та аналітичних процесів, пов'язаних з управлінням навчальним процесом у вищому навчальному закладі. Деканат виступає ключовою ланкою між адміністрацією закладу, викладачами та студентами, забезпечуючи централізований облік, контроль та аналіз навчальної діяльності.

Основною особливістю даної предметної області є значний обсяг даних, що накопичуються протягом тривалого часу. Інформація про студентів, результати їх навчання, навчальні плани та дисципліни зберігається протягом усього періоду навчання, а також після його завершення для формування архівів та статистичних звітів. Це зумовлює потребу у використанні підходів, орієнтованих на роботу з надвеликими обсягами даних.

В умовах сучасної освіти зростає потреба не лише в оперативному доступі до даних, але й у можливості їх глибокого аналізу. Адміністрація навчального закладу зацікавлена у своєчасному отриманні інформації щодо успішності студентів, ефективності викладання, навантаження викладачів та динаміки навчальних показників. Тому інформаційна система деканату повинна поєднувати функції транзакційної обробки та аналітичної підтримки прийняття рішень.

1.2 Місце та роль інформаційної системи деканату

Інформаційна система деканату займає центральне місце у загальній інформаційній інфраструктурі вищого навчального закладу. Вона інтегрується з іншими підсистемами, такими як бухгалтерія, відділ кадрів, бібліотека, електронні системи навчання та управління розкладом.

З точки зору управління, система деканату виконує такі ключові функції:

- забезпечення достовірного обліку контингенту студентів;
- підтримка процесів контролю навчального процесу;

- формування офіційної звітності для керівництва та зовнішніх органів;
- надання аналітичної інформації для стратегічного планування.

Таким чином, система деканату є джерелом критично важливої інформації, що безпосередньо впливає на якість управлінських рішень у сфері освіти.

1.3 Основні бізнес-процеси предметної області

У межах предметної області «Студент (Деканат)» можна виділити ряд взаємопов'язаних бізнес-процесів, які формують повний життєвий цикл навчання студента у закладі вищої освіти.

До основних бізнес-процесів належать:

1. **Зарахування студентів** – процес реєстрації абітурієнтів, формування особових справ студентів, присвоєння унікальних ідентифікаторів та закріplення за академічними групами.
2. **Управління академічними групами** – створення та супровід навчальних груп, облік їх складу, фіксація змін упродовж навчального процесу.
3. **Планування навчального процесу** – формування переліку дисциплін, розподіл навчального навантаження між викладачами.
4. **Оцінювання результатів навчання** – облік результатів іспитів, заліків та інших форм контролю знань.
5. **Формування аналітичної та статистичної звітності** – створення звітів щодо успішності, рейтингу студентів, ефективності викладання.

Кожен з наведених бізнес-процесів супроводжується створенням та обробкою значної кількості даних, що потребує чіткої структуризації та контролю якості інформації.

1.4 Інформаційні потоки та джерела даних

У процесі функціонування системи деканату формуються різноманітні інформаційні потоки. Основними джерелами даних є результати навчальної діяльності студентів, персональні дані, інформація про навчальні дисципліни та викладачів.

Дані надходять з різних джерел, зокрема з первинних систем обліку, електронних журналів, систем дистанційного навчання та зовнішніх інформаційних ресурсів. Важливою задачею є інтеграція цих даних у єдине сховище з урахуванням вимог до їх узгодженості та цілісності.

1.5 Функціональні вимоги до системи

На основі аналізу предметної області та бізнес-процесів сформульовано такі функціональні вимоги:

- зберігання великих обсягів історичних даних;
- забезпечення цілісності та коректності інформації;
- підтримка багатовимірного аналізу даних;
- формування параметризованих аналітичних звітів;
- забезпечення високої продуктивності системи.

1.6 Нефункціональні вимоги

До нефункціональних вимог належать вимоги щодо продуктивності, надійності, масштабованості та безпеки системи. Система повинна забезпечувати швидке виконання аналітичних запитів навіть при значних обсягах даних, а також захист персональної інформації студентів.

1.7 Бізнес-правила та обмеження

Для забезпечення коректної роботи інформаційної системи визначено такі бізнес-правила:

- кожен студент має унікальний ідентифікатор;
- студент може належати лише до однієї академічної групи в певний момент часу;
- оцінки повинні знаходитись у допустимому діапазоні значень (1–12);
- кожна оцінка повинна бути прив'язана до конкретного студента, предмета та дати;
- історичні дані не підлягають фізичному видаленню.

1.8 Декомпозиція бізнес-процесів предметної області

Для більш детального аналізу предметної області було виконано декомпозицію основних бізнес-процесів деканату на підпроцеси та окремі операції. Такий підхід дозволяє глибше зrozуміти логіку функціонування системи та визначити вимоги до інформаційної підтримки кожного етапу.

Процес зарахування студентів включає збір персональних даних, перевірку документів, формування наказів про зарахування та створення електронних облікових записів. Кожен з цих етапів генерує інформацію, яка повинна бути збережена в базі даних та доступна для подальшого аналізу.

Процес оцінювання результатів навчання складається з підготовки контрольних заходів, фіксації результатів, перевірки коректності введених оцінок та їх затвердження. Важливим аспектом є забезпечення збереження історії оцінювання, що дозволяє аналізувати динаміку успішності студентів.

1.9 Аналіз обсягів та структури даних

Однією з ключових характеристик предметної області є значний обсяг даних, що накопичується протягом тривалого часу. У межах курсової роботи передбачається зберігання інформації про щонайменше 10 000 студентів та не менше 500 000 записів оцінок за період у 5 років.

Структура даних включає як довідниківі таблиці (студенти, групи, предмети, викладачі), так і фактологічні таблиці, що містять результати навчальної діяльності. Така структура є передумовою для побудови сховища даних та реалізації багатовимірного аналізу.

1.10 Проблеми та недоліки традиційних підходів

Використання лише транзакційних баз даних для зберігання та аналізу інформації деканату має ряд суттєвих недоліків. Зокрема, виконання складних аналітичних запитів на великих обсягах даних призводить до значного навантаження на систему та зниження її продуктивності.

Крім того, відсутність історичних зрізів та агрегованих показників ускладнює аналіз тенденцій та порівняння результатів навчання за різні періоди. Це обґруntовує необхідність використання сховищ даних та OLAP-технологій.

1.11 Обґрунтування вибору технологій

Для реалізації системи аналізу даних деканату було обрано платформу Microsoft SQL Server, яка надає комплекс засобів для роботи з надвеликими базами даних. SQL Server Integration Services забезпечує ефективну реалізацію ETL-процесів, SQL Server Analysis Services дозволяє створювати OLAP-куби, а SQL Server Reporting Services — формувати аналітичні звіти.

Використання даних технологій дозволяє забезпечити масштабованість, продуктивність та надійність системи.

1.12 Аналіз користувачів інформаційної системи

Інформаційна система деканату використовується різними категоріями користувачів, кожна з яких має власні цілі, задачі та вимоги до обробки інформації. Основними користувачами системи є працівники деканату, викладачі, керівництво навчального закладу та аналітичні підрозділи.

Працівники деканату здійснюють щоденну роботу з введення, коригування та перевірки даних. Для них важливою є зручність роботи з системою, швидкість виконання операцій та надійність зберігання інформації. Викладачі використовують систему для перегляду списків студентів, внесення оцінок та аналізу результатів навчання.

Керівництво навчального закладу зацікавлене в отриманні узагальненої аналітичної інформації у вигляді звітів, графіків та показників ефективності. Саме для цієї категорії користувачів особливо важливими є можливості OLAP-аналізу та інтерактивної звітності.

1.13 Порівняння OLTP та OLAP у контексті предметної області

Транзакційні системи (OLTP) призначені для обробки великої кількості коротких операцій введення та оновлення даних. У межах деканату такі системи ефективно підтримують облік студентів та оцінок, однак не пристосовані до складних аналітичних запитів.

OLAP-системи, навпаки, орієнтовані на читання та агрегацію великих обсягів даних. Вони дозволяють виконувати аналіз успішності за роками, семестрами, групами та іншими вимірами без значного навантаження на транзакційну базу.

Порівняння OLTP та OLAP підходів свідчить про доцільність використання сховищ даних та багатовимірного аналізу для предметної області «Студент (Деканат)».

1.14 Вимоги до даних з точки зору аналітики

Аналітичні задачі вимагають наявності повних, узгоджених та історичних даних. Інформація повинна зберігатися з урахуванням часових змін, що дозволяє виконувати порівняльний аналіз та виявляти тенденції.

Важливим аспектом є якість даних, яка досягається шляхом контролю введення, очищення та валідації інформації у процесі ETL. Агреговані показники повинні обчислюватися на основі достовірних даних, що забезпечує коректність аналітичних висновків.

1.15 Перспективи розвитку інформаційної системи

Розроблена система може бути розширенна шляхом інтеграції з іншими інформаційними ресурсами навчального закладу, зокрема системами електронного навчання, фінансового обліку та управління персоналом. Подальший розвиток системи може включати використання інструментів прогнозування та інтелектуального аналізу даних.

1.16 Висновки до розділу 1

У першому розділі курсової роботи було всебічно проаналізовано предметну область «Студент (Деканат)», розглянуто основні бізнес-процеси, інформаційні потоки, вимоги до системи та користувачів. Проведений аналіз підтверджує доцільність застосування технологій сховищ даних та OLAP для ефективного аналізу надвеликих обсягів освітньої інформації.

2. РОЗДІЛ 2. ПРОЕКТУВАННЯ БАЗИ ДАНИХ

Другий розділ присвячений детальному проєктуванню бази даних для предметної області «Студент (Деканат)». У межах розділу розглянуто дві логічно пов’язані підсистеми:

- транзакційну базу даних **OLTP**, яка використовується для оперативного обліку навчального процесу;
- сховище даних **Data Warehouse (DW)**, призначене для аналітичної обробки великих обсягів інформації.

Такий підхід відповідає сучасній архітектурі інформаційних систем та забезпечує відокремлення оперативних і аналітичних навантажень.

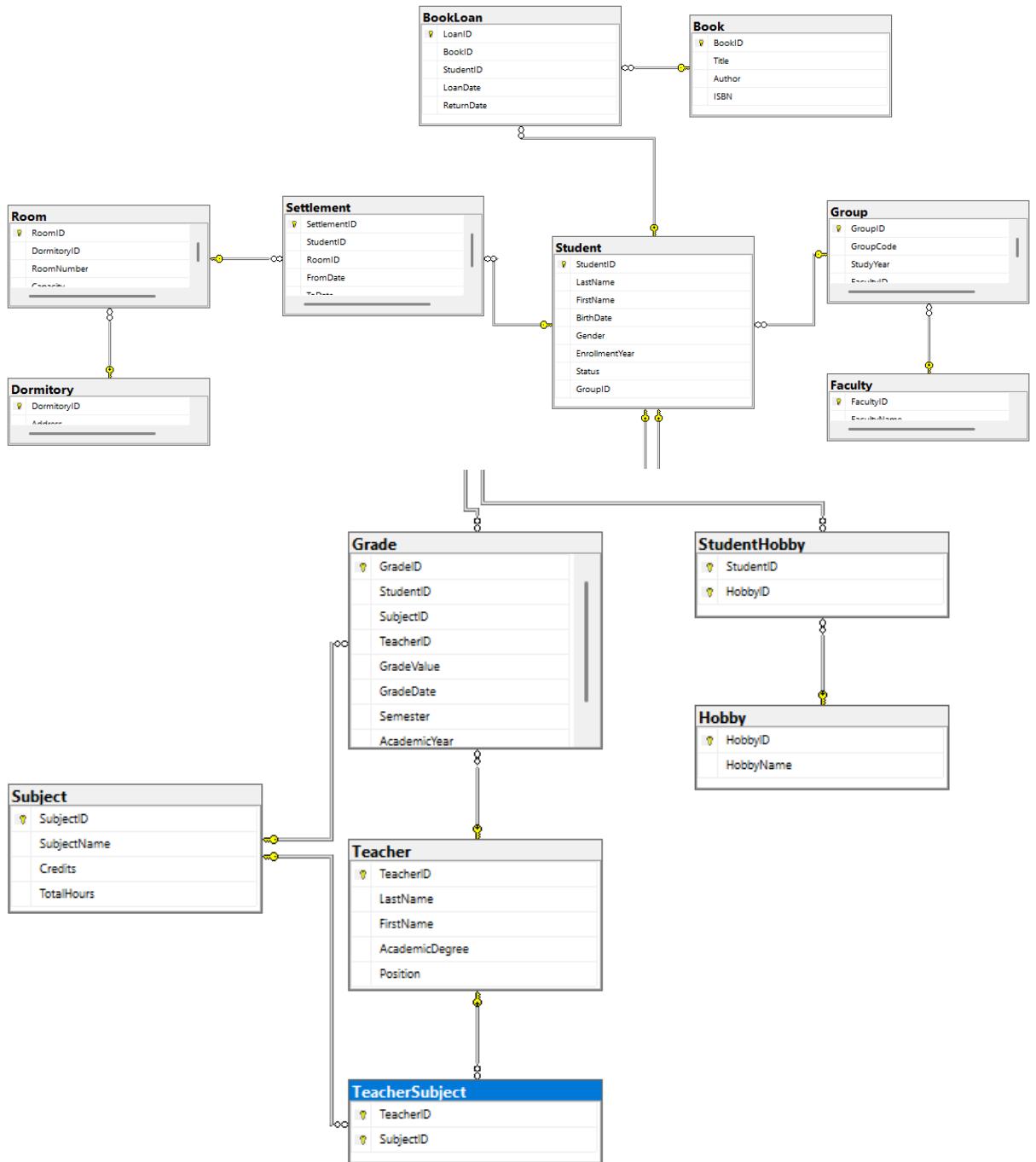
2.1 Концептуальне проєктування бази даних

На етапі концептуального проєктування була побудована узагальнена модель предметної області «Студент (Деканат)», яка відображає основні сутності, їх атрибути та зв’язки між ними. Концептуальна модель не залежить від конкретної СУБД та слугує основою для подальшого логічного й фізичного проєктування.

Основними сутностями предметної області є:

- студент;
- академічна група;
- викладач;
- навчальна дисципліна;
- оцінка;
- книга;
- гуртожиток;
- кімната гуртожитку;
- хобі студента.

Між сутностями встановлено зв’язки типу «один-до-багатьох» та «багато-до-багатьох». Наприклад, студент може мати декілька оцінок, а кожна оцінка належить одному студентові; студент може мати декілька хобі, що реалізується через проміжну таблицю.



2.2 Логічне проєктування транзакційної бази даних (OLTP)

Логічна модель OLTP-бази даних була спроектована відповідно до принципів реляційної моделі та нормалізована до третьої нормальної форми (ЗНФ). Це дозволило мінімізувати надлишковість даних та уникнути аномалій оновлення.

2.2.1 Таблиця Student

Таблиця Student зберігає основну інформацію про студентів навчального закладу.

Основні поля таблиці:

- StudentID – первинний ключ, унікальний ідентифікатор студента;
- FirstName – ім'я студента;
- LastName – прізвище студента;
- BirthDate – дата народження;
- Gender – стать;
- GroupID – зовнішній ключ на таблицю Group;
- SettlementID – зовнішній ключ на місце проживання;
- Status – статус студента.

2.2.2 Таблиця Group

Таблиця Group містить інформацію про академічні групи.

Поля таблиці:

- GroupID – первинний ключ;
- GroupCode – код групи;
- FacultyID – зовнішній ключ на таблицю факультетів;
- YearOfStudy – курс навчання.

2.2.3 Таблиця Faculty

Таблиця Faculty використовується для зберігання довідникової інформації про факультети.

Поля таблиці:

- FacultyID – первинний ключ;
- FacultyName – назва факультету.

2.2.4 Таблиця Teacher

Таблиця Teacher зберігає інформацію про викладачів.

Поля таблиці:

- TeacherID – первинний ключ;
- FirstName – ім'я;
- LastName – прізвище;
- Position – посада.

2.2.5 Таблиця Subject

Таблиця Subject містить перелік навчальних дисциплін.

Поля таблиці:

- SubjectID – первинний ключ;
- SubjectName – назва дисципліни;
- Credits – кількість кредитів.

2.2.6 Таблиця Grade

Таблиця Grade є ключовою фактологічною таблицею OLTP-бази.

Поля таблиці:

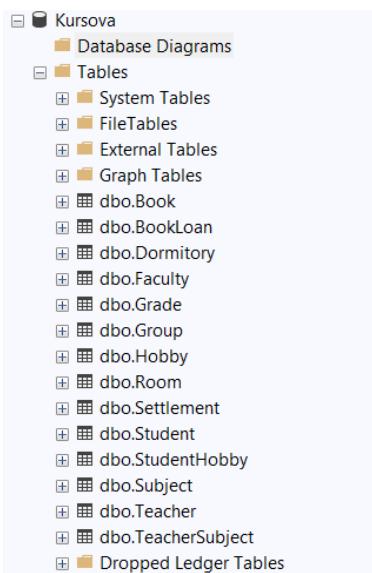
- GradeID – первинний ключ;
- StudentID – зовнішній ключ на студента;
- SubjectID – зовнішній ключ на дисципліну;
- TeacherID – зовнішній ключ на викладача;
- GradeValue – значення оцінки;
- GradeDate – дата виставлення оцінки.

2.2.7 Таблиці бібліотеки та гуртожитку

До допоміжних таблиць належать:

- Book, BookLoan – облік бібліотечних ресурсів;
- Dormitory, Room – облік проживання студентів;
- Hobby, StudentHobby – облік позанавчальної активності студентів.

Місце для вставки скріншоту таблиць OLTP-бази (SSMS):



2.3 Проєктування сховища даних (Data Warehouse)

Для підтримки аналітичних запитів на основі OLTP-бази було спроектовано сховище даних Kursova_var1_DWH. Архітектура DW базується на схемі «зірка», що є оптимальною для OLAP-аналізу.

2.3.1 Таблиці вимірів (Dimensions)

- DimStudent – інформація про студентів;
- DimGroup – академічні групи;
- DimFaculty – факультети;
- DimTeacher – викладачі;
- DimSubject – дисципліни;
- DimBook – книги;
- DimDate – часовий вимір.

Кожна таблиця виміру містить сурогатний ключ та атрибути для побудови ієархій.

2.3.2 Таблиці фактів (Fact Tables)

- FactGrades – результати оцінювання студентів;
- FactBookLoans – факти видачі книг.

Факторологічні таблиці містять числові показники та зовнішні клочі на таблиці вимірів.

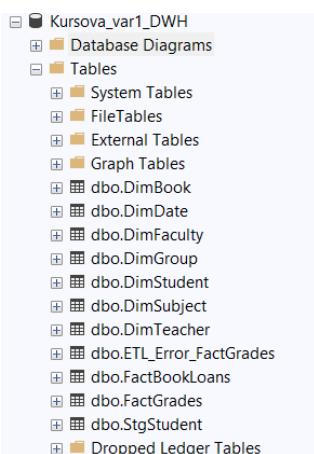
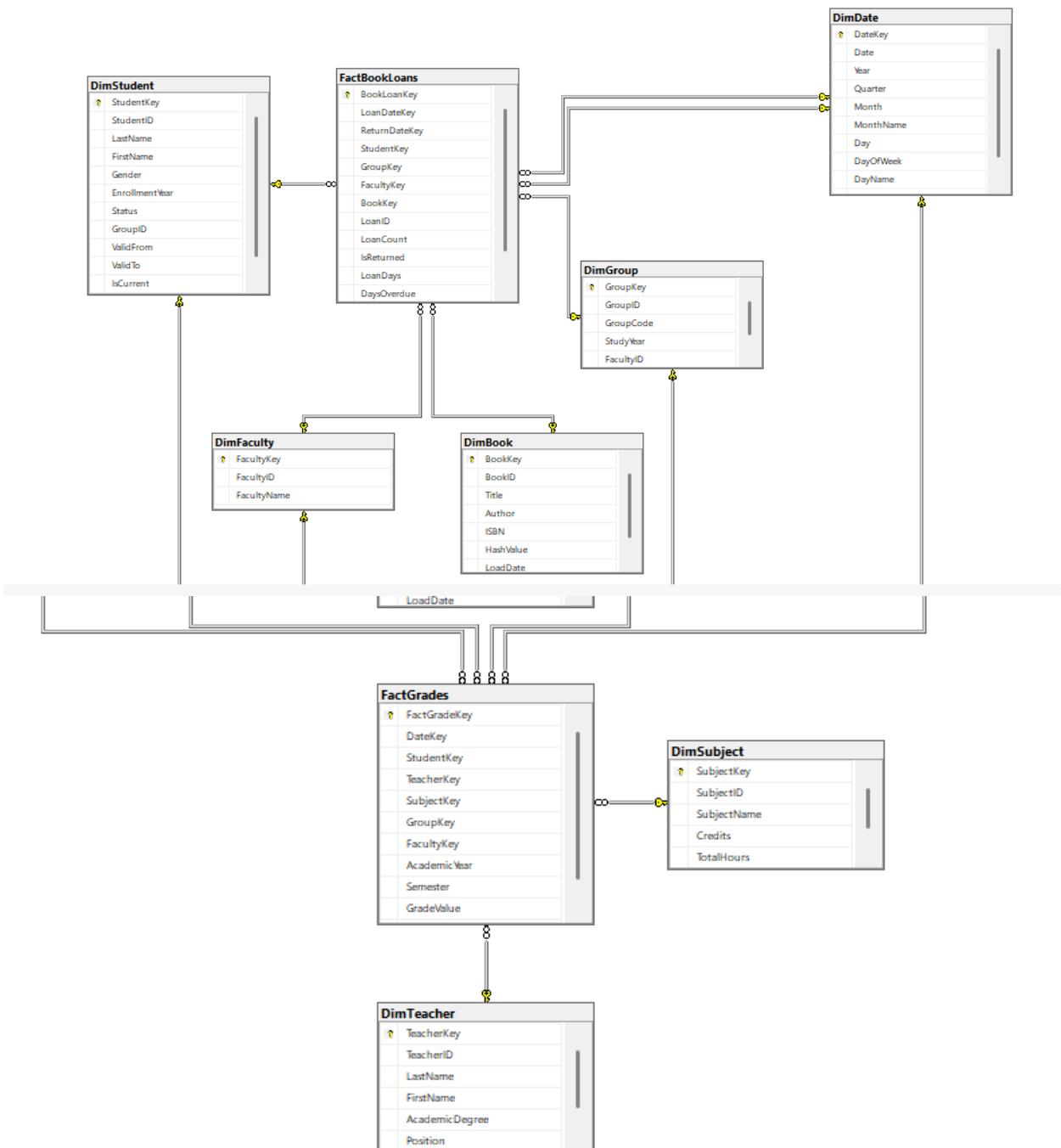
2.3.3 Проміжні та службові таблиці

- StgStudent – проміжна таблиця для ETL;
- ETL_Error_FactGrades – журнал помилок ETL-процесів.

The diagram illustrates two database tables side-by-side:

StgStudent	
StudentID	
LastName	
FirstName	
Gender	
EnrollmentYear	
Status	
GroupID	
GradeLevel	

ETL_Error_FactGrades	
ErrorTime	
ErrorSource	
GradeDate	
DateKey	
StudentID	
TeacherID	
SubjectID	
GroupID	
FacultyID	
AcademicYear	
Semester	



2.4 Порівняння OLTP та Data Warehouse

OLTP-база оптимізована для швидкого виконання транзакцій, тоді як Data Warehouse орієнтоване на виконання складних аналітичних запитів. Розділення цих підходів дозволяє забезпечити високу продуктивність та масштабованість системи.

2.5 Генерація та заповнення даних у транзакційній базі даних

Для наповнення транзакційної бази даних реалістичними тестовими даними було використано інструмент **Redgate SQL Data Generator**, який дозволяє автоматично створювати великі обсяги пов'язаних між собою даних з урахуванням заданих правил.

Перед початком генерації даних було проаналізовано структуру таблиць та визначено залежності між ними. Генерація виконувалася у правильній послідовності: спочатку довідників таблиці (Faculty, Group, Subject, Teacher), після чого – основні таблиці (Student, Grade, BookLoan та інші).

Для кожного поля таблиць було налаштовано відповідні правила генерації:

- текстові поля (ім’я, прізвище) – генерація на основі словників;
- дати – випадкові значення у заданому діапазоні (5 років);
- числові поля – генерація у визначених межах;
- зовнішні ключі – генерація з урахуванням зв’язків між таблицями.

У результаті генерації було отримано понад 10 000 записів у таблиці студентів та більше 500 000 записів у таблиці оцінок, що відповідає вимогам методичних рекомендацій.

	StudentsCount	GroupsCount	FacultiesCount	TeachersCount	SubjectsCount	GradesCount	BooksCount	BookLoansCount
1	10000	360	6	180	300	520000	20000	50000

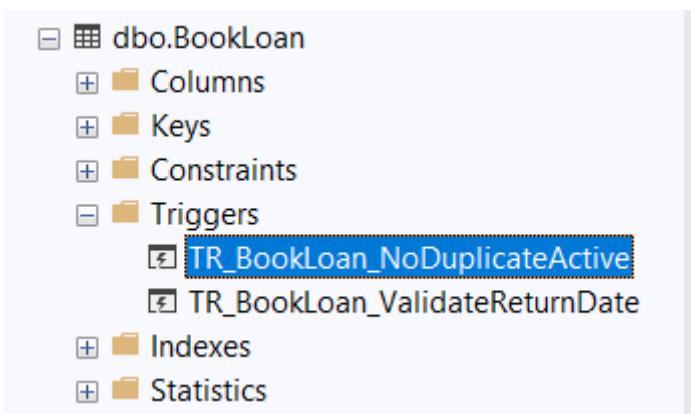
2.6 Реалізація тригерів у базі даних

Для забезпечення складних бізнес-правил, які не можуть бути реалізовані стандартними обмеженнями цілісності, у базі даних були створені тригери. Тригери дозволяють автоматично реагувати на події вставки, оновлення або видалення даних.

Основними задачами, які виконуються за допомогою тригерів, є:

- контроль допустимих значень оцінок;
- заборона фізичного видалення історичних даних;
- автоматичне оновлення службових полів.

Наприклад, для таблиці Grade реалізовано тригер, який перевіряє, чи знаходиться значення оцінки у допустимому діапазоні. У разі порушення правила операція вставки або оновлення блокується.



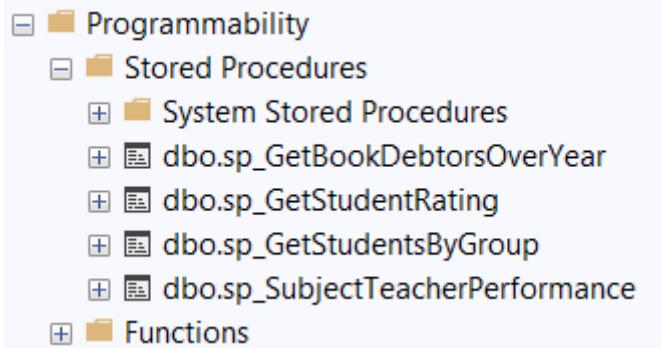
2.7 Використання збережених процедур та функцій

Для інкапсуляції бізнес-логіки та повторного використання коду у базі даних було створено збережені процедури та користувальські функції.

Збережені процедури використовуються для:

- отримання списків студентів за групами;
- формування рейтингів студентів;
- вибірки даних для ETL-процесів.

Користувальські функції застосовуються для обчислення похідних показників, наприклад середнього балу студента або кількості складених дисциплін.



2.8 Забезпечення цілісності даних та обмеження

Для забезпечення цілісності даних у базі даних використовуються первинні та зовнішні ключі, обмеження CHECK, UNIQUE та NOT NULL. Дані механізми

гарантують коректність введення інформації та запобігають порушенню логічних зв'язків між таблицями.

Зокрема, обмеження CHECK застосовуються для контролю допустимих значень оцінок, а зовнішні ключі забезпечують узгодженість між студентами, групами та дисциплінами.

2.9 Висновки до розділу 2

У другому розділі курсової роботи було всебічно розглянуто процес проєктування транзакційної бази даних та сховища даних для предметної області «Студент (Деканат)». Детально описано структуру OLTP-бази, архітектуру Data Warehouse, процеси генерації даних, реалізацію тригерів, збережених процедур та функцій. Запропоновані рішення забезпечують коректність, масштабованість та готовність системи до подальшої реалізації ETL-процесів і побудови OLAP-куба.

3. РОЗДІЛ 3 РЕАЛІЗАЦІЯ ETL-ПРОЦЕСІВ (SQL SERVER INTEGRATION SERVICES)

Розділ 3 присвячений всебічному та детальному опису процесів витягування, трансформації та завантаження даних (ETL – Extract, Transform, Load), які забезпечують перенесення даних з транзакційної бази даних (OLTP) до сховища даних (Data Warehouse). Реалізація ETL-процесів виконана з використанням платформи SQL Server Integration Services (SSIS), яка є складовою екосистеми Microsoft SQL Server та широко використовується для побудови корпоративних сховищ даних.

ETL-процеси є критично важливим елементом архітектури сховища даних, оскільки саме вони визначають якість аналітичної інформації, що надходить до OLAP-куба. Помилки на цьому етапі можуть призвести до викривлення результатів аналізу, тому проєктування ETL-процесів вимагає особливої уваги.

3.1 Мета та завдання ETL-процесів

Основною метою ETL-процесів у межах даної курсової роботи є забезпечення коректного, контролюваного та масштабованого перенесення даних з OLTP-бази даних Kursova до сховища даних Kursova_var1_DWH. У процесі ETL необхідно не лише скопіювати дані, а й виконати їх очищення, узгодження, перетворення та збагачення.

До основних завдань ETL-процесів належать:

- інтеграція даних з декількох таблиць OLTP-бази;
- усунення некоректних та дубльованих записів;
- приведення типів даних до єдиного формату;
- формування сурогатних ключів для таблиць вимірів;
- забезпечення історичності даних у сховищі;
- контроль якості та логування помилок.

3.2 Загальна архітектура ETL-рішення

ETL-рішення побудоване за класичною багаторівневою архітектурою, яка включає джерело даних, проміжну зону та цільове сховище. Така архітектура дозволяє ізолювати вплив транзакційних операцій на аналітичні процеси та забезпечує гнучкість масштабування.

Джерелом даних виступає транзакційна база даних Kursova, що містить нормалізовані таблиці предметної області. Цільовою системою є сховище даних Kursova_var1_DWH, побудоване за схемою «зірка». Між ними розміщується проміжна зона (staging area), яка використовується для тимчасового зберігання даних та виконання попередніх трансформацій.

3.3 Створення та налаштування SSIS-проєкту

Для реалізації ETL-процесів було створено окремий SSIS-проєкт у середовищі SQL Server Data Tools (SSDT). У проєкті налаштовано з'єднання з OLTP-базою та сховищем даних, а також визначено загальні параметри виконання пакетів.

У межах проєкту реалізовано декілька логічно відокремлених пакетів, кожен з яких відповідає за завантаження певної групи даних. Такий підхід підвищує читабельність та підтримуваність ETL-рішення.

3.4 Етап Extract: витягування даних з OLTP

На етапі Extract здійснюється вибірка необхідних даних з транзакційної бази даних. Для цього використовуються компоненти OLE DB Source, які дозволяють виконувати SQL-запити безпосередньо до OLTP-бази.

Витягування даних виконується з урахуванням залежностей між таблицями. Наприклад, при витягуванні даних про оцінки використовується об'єднання

таблиць Grade, Student, Subject та Teacher. Це дозволяє сформувати повний набір атрибутів, необхідних для подальшого завантаження у сховище даних.

3.5 Етап Transform: очищення та перетворення даних

Етап Transform є найбільш складним та відповідальним у процесі ETL. Саме на цьому етапі дані проходять очищення, перевірку та перетворення відповідно до вимог сховища даних.

Для очищення даних застосовуються правила перевірки допустимих значень, обробки NULL-значень та усунення дублікатів. Наприклад, оцінки, що виходять за межі допустимого діапазону, не допускаються до завантаження у сховище.

Перетворення типів даних забезпечується компонентом Data Conversion, який дозволяє привести дати, числові та текстові значення до формату, сумісного зі схемою сховища даних. Додатково використовуються Derived Column для формування обчислюваних атрибутів.

3.6 Використання Lookup для побудови зв'язків

Компонент Lookup використовується для заміни бізнес-ключів OLTP-бази на сурогатні ключі таблиць вимірів. Це є необхідною умовою для забезпечення коректних зв'язків у схемі «зірка».

У разі відсутності відповідного запису у таблиці виміру Lookup перенаправляє запис до гілки обробки помилок, що дозволяє зберегти контроль над якістю даних.

3.7 Етап Load: завантаження даних у сховище

На етапі Load оброблені дані завантажуються до таблиць вимірів та фактів сховища даних. Завантаження виконується у визначеній послідовності: спочатку виміри, потім фактологічні таблиці.

Такий підхід забезпечує коректність зовнішніх ключів та цілісність даних у сховищі.

3.8 Інкрементальне завантаження даних

Для оптимізації ETL-процесів реалізовано механізм інкрементального завантаження, який дозволяє обробляти лише нові або змінені записи. Це значно зменшує час виконання пакетів та навантаження на систему.

Інкрементальне завантаження базується на аналізі дат зміни записів та використанні службових таблиць контролю.

3.9 Обробка помилок та логування

Усі помилки, що виникають у процесі ETL, фіксуються у спеціальних таблицях логування. Це дозволяє проводити аналіз помилок та вживати заходів щодо їх усунення.

Логування включає інформацію про час виконання пакетів, кількість оброблених записів та статус виконання ETL-процесів.

3.10 Забезпечення якості даних

Якість даних є ключовим фактором ефективності аналітичної системи. У межах ETL-процесів реалізовано перевірки повноти, коректності та узгодженості даних.

3.11 Масштабованість та оптимізація ETL-процесів

Проектування ETL-процесів виконано з урахуванням можливого зростання обсягів даних. Використання інкрементального завантаження, оптимізованих запитів та індексів дозволяє забезпечити масштабованість рішення.

3.12 Висновки до розділу 3

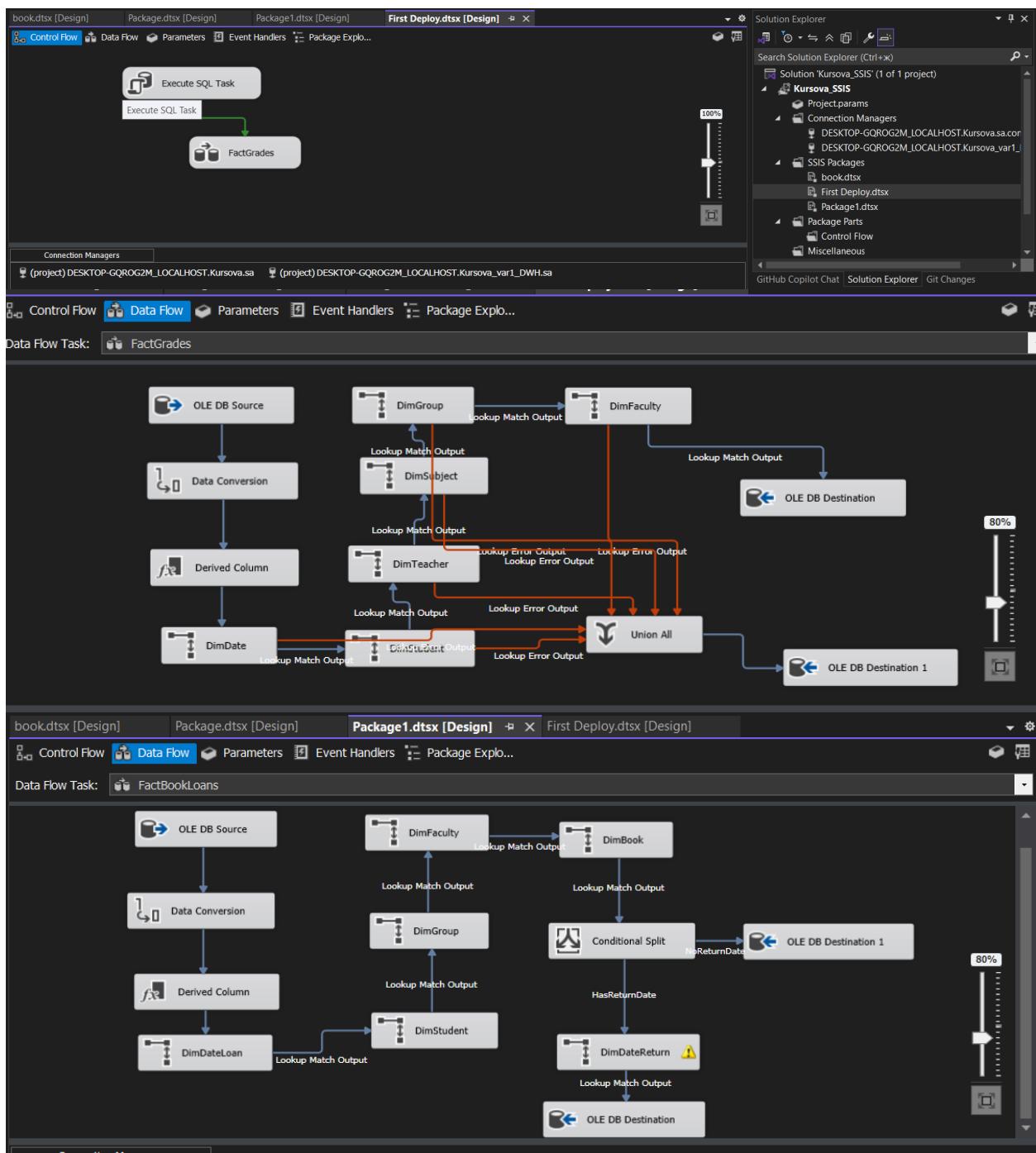
У третьому розділі курсової роботи детально розглянуто процес реалізації ETL-процесів з використанням SQL Server Integration Services. Описані підходи та інструменти забезпечують коректне, контролюване та масштабоване перенесення даних з OLTP-бази до сховища даних, що створює надійну основу для подальшої побудови OLAP-куба та аналітичних звітів.

The screenshot shows three windows of the Microsoft SQL Server Integration Services (SSIS) Development Environment:

- Top Window:** Displays the Control Flow tab of the 'book.dtsx [Design]' package. It contains one 'Execute SQL Task' (labeled 'Execute SQL Task') which has a single outgoing arrow pointing to a 'Data Flow Task' (labeled 'FactBookLoans'). The 'FactBookLoans' task has a yellow warning icon.
- Middle Window:** Displays the Control Flow tab of the 'book.dtsx [Design]' package. It contains one 'Data Flow Task' (labeled 'Data Flow Task').
- Bottom Window:** An open 'Execute SQL Task Editor' dialog. The 'General' tab is selected. The configuration details are as follows:

Name	Execute SQL Task
Description	Execute SQL Task
TimeOut	0
CodePage	1251
TypeConversionMode	Allowed
ResultSet	None
ConnectionType	OLE DB
Connection	DESKTOP-GQROG2M_LOCALHOST.Kursova
SQLSourceType	Direct input
SQLStatement	TRUNCATE TABLE dbo.FactGrades;
IsQueryStoredProcedure	False
BypassPrepare	True

 Below the table, there is a 'Name' section with a description: 'Specifies the name of the task.' and three buttons: 'Browse...', 'Build Query...', and 'Parse Query...'.



Results

	DimStudentCount	DimGroupCount	DimFacultyCount	DimTeacherCount	DimSubjectCount	DimDateCount	FactGradesCount	FactBookLoansCount
1	10000	360	6	180	300	2557	520000	39872

4. РОЗДІЛ 4. ПОБУДОВА OLAP-КУБА (SSAS) ТА АНАЛІТИЧНА ВІЗУАЛІЗАЦІЯ (SSRS)

Розділ 4 присвячений завершальному етапу створення аналітичної системи – побудові багатовимірного OLAP-куба на базі сховища даних Kursova_var1_DWH з використанням SQL Server Analysis Services (SSAS) у режимі Multidimensional, а також створенню аналітичних звітів у SQL Server Reporting Services (SSRS). У межах розділу детально описано структуру куба, виміри, міри, ієрархії, механізми агрегації, MDX-запити, параметризацію та інтерактивні елементи звітності.

4.1 Мета побудови OLAP-куба та загальний підхід

Основною метою побудови OLAP-куба є забезпечення швидкого виконання аналітичних запитів на великих обсягах даних завдяки попередній агрегації та оптимізованому багатовимірному зберіганню. На відміну від звичайних SQL-запитів до реляційної БД, OLAP-куб дозволяє виконувати аналіз у термінах вимірів (Dimension) та мір (Measures), будувати зрізи (slice), деталізацію (drill-down) і згортуку (roll-up), а також швидко порівнювати показники за різними категоріями.

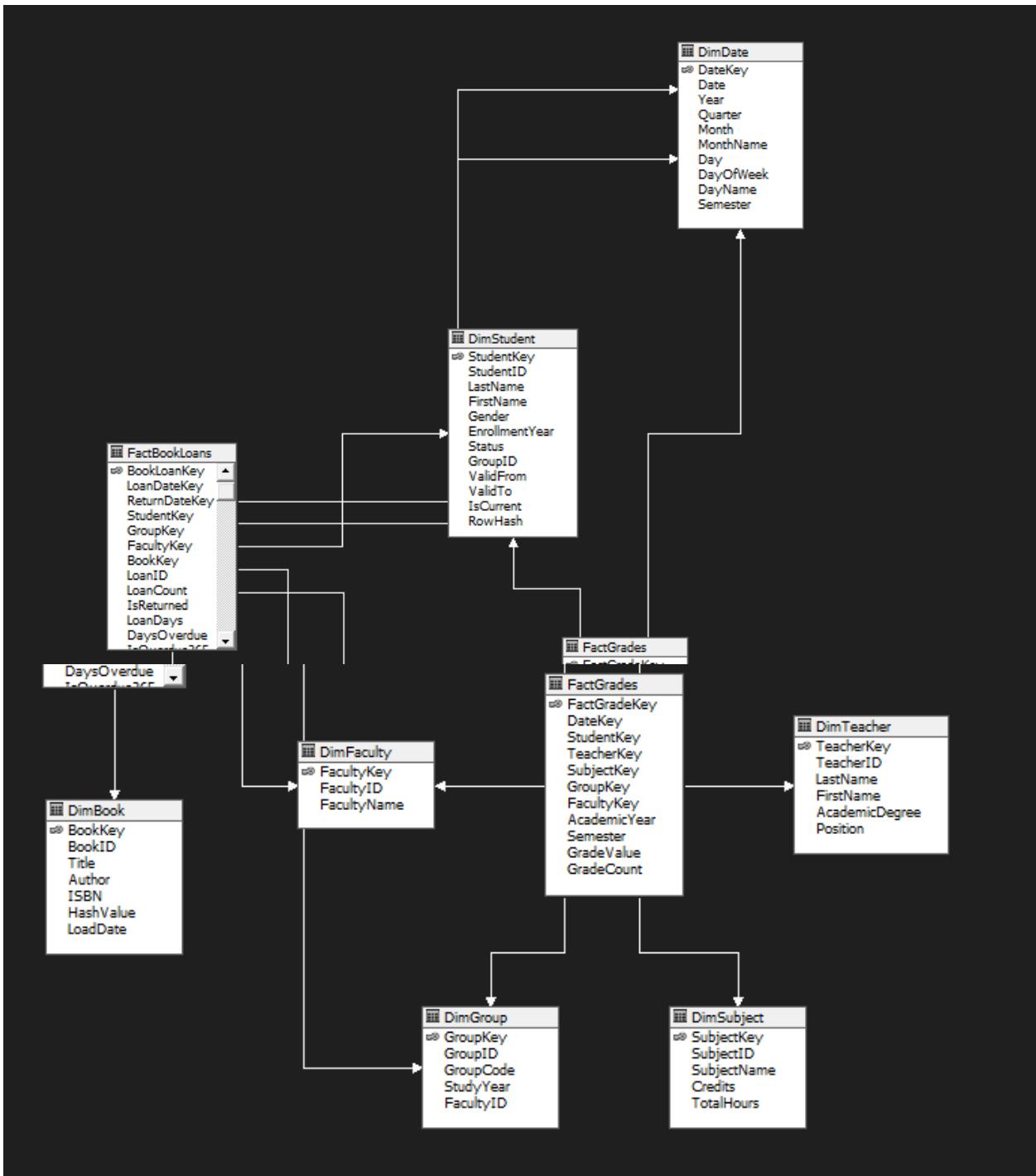
У межах курсової роботи куб будується на основі схеми «зірка» у сховищі даних. Виміри куба відповідають таблицям вимірів (Dim*), а набори мір – фактологічним таблицям (Fact*).

4.2 Створення SSAS-проекту та джерел даних

Для побудови куба створено проект типу **Analysis Services Multidimensional and Data Mining Project** у середовищі SSDT. На цьому етапі виконуються такі кроки:

- 1. Створення Data Source** – налаштування підключення до бази Kursova_var1_DWH.
- 2. Створення Data Source View (DSV)** – формування логічного подання таблиць DW, які використовуватимуться для побудови вимірів та куба.

У DSV додаються таблиці вимірів (DimStudent, DimGroup, DimFaculty, DimTeacher, DimSubject, DimBook, DimDate) та фактологічні таблиці (FactGrades, FactBookLoans). Також у DSV перевіряються зв’язки між фактами та вимірами через сурогатні ключі.



4.3 Проєктування вимірів (Dimensions)

Вимір – це набір атрибутів, за якими користувач аналізує дані (наприклад, студент, група, предмет, час). Якість проєктування вимірів напряму впливає на зручність аналізу та продуктивність куба.

У межах курсової роботи реалізовано наступні виміри.

4.3.1 Вимір DimDate (Час)

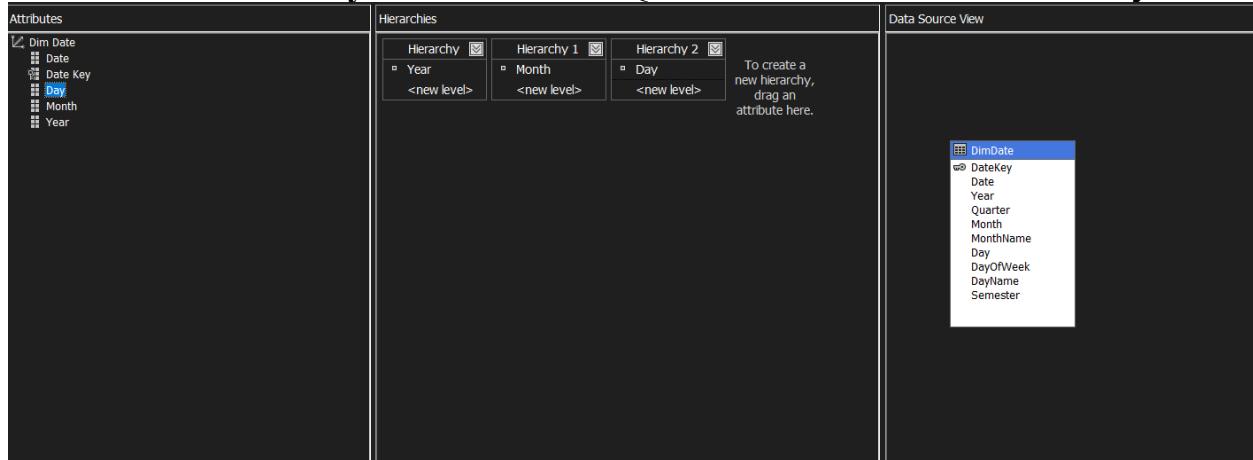
Часовий вимір є ключовим для аналітики, оскільки більшість показників аналізуються у динаміці (по роках, семестрах, місяцях).

Основні атрибути виміру:

- Year – рік;
- Quarter – квартал;
- Month – місяць;
- Day – день;
- FullDate – повна дата.

Для зручності аналізу створено ієрархію:

- **Calendar Hierarchy:** Year → Quarter → Month → Day.



4.3.2 Вимір DimStudent (Студент)

Вимір містить атрибути, що описують студента та дозволяють формувати рейтинги, порівнювати успішність різних категорій студентів.

Основні атрибути:

- FirstName, LastName – ПІБ;
- Gender – стать;
- Status – статус (навчається/відрахований/академвідпустка);
- Settlement або аналогічний атрибут місця проживання (за наявності в DW).

Рекомендована ієрархія (за потреби):

- Gender → Status → Student.

Attributes	Hierarchies	Data Source View
<input checked="" type="checkbox"/> Dim Student <ul style="list-style-type: none"> Enrollment Year First Name Group ID Last Name Student ID Student Key 	To create a new hierarchy, drag an attribute here.	<input checked="" type="checkbox"/> DimStudent <ul style="list-style-type: none"> StudentKey StudentID LastName FirstName Gender EnrollmentYear Status GroupID ValidFrom ValidTo IsCurrent RowHash

4.3.3 Вимір DimGroup та DimFaculty

Виміри групи та факультету дозволяють виконувати аналіз успішності на рівні підрозділів навчального закладу.

DimGroup містить атрибути:

- GroupCode – код групи;
- YearOfStudy – курс.

DimFaculty містить:

- FacultyName – назва факультету.

Зазвичай створюється ієрархія:

- Faculty → Group.

Attributes	Hierarchies	Data Source View
<input checked="" type="checkbox"/> Dim Group <ul style="list-style-type: none"> Faculty ID Group Code Group Key Study Year 	To create a new hierarchy, drag an attribute here.	<input checked="" type="checkbox"/> DimGroup <ul style="list-style-type: none"> GroupKey GroupID GroupCode StudyYear FacultyID

4.3.4 Вимір DimSubject та DimTeacher

Ці виміри забезпечують аналіз успішності за дисциплінами та викладачами.

DimSubject:

- SubjectName;
- Credits;
- (за наявності) семестр.

DimTeacher:

- FirstName, LastName;
- Position;
- (за наявності) кафедра.

Attributes	Hierarchies	Data Source View
<input checked="" type="checkbox"/> Dim Subject <ul style="list-style-type: none"> ▪ Credits ▪ Subject Key ▪ Subject Name 	To create a new hierarchy, drag an attribute here.	<pre>DimSubject SubjectKey SubjectID SubjectName Credits TotalHours</pre>
Attributes	Hierarchies	Data Source View
<input checked="" type="checkbox"/> Dim Teacher <ul style="list-style-type: none"> ▪ First Name ▪ Last Name ▪ Teacher Key 	To create a new hierarchy, drag an attribute here.	<pre>DimTeacher TeacherKey TeacherID LastName FirstName AcademicDegree Position</pre>

4.3.5 Вимір DimBook

Вимір використовується для аналізу бібліотечної активності.

Атрибути:

- назва книги;
- автор/жанр (за наявності);
- категорія.

Attributes	Hierarchies	Data Source View
<input checked="" type="checkbox"/> Dim Book <ul style="list-style-type: none"> ▪ Book Key ▪ Title 	To create a new hierarchy, drag an attribute here.	<pre>DimBook BookKey BookID Title Author ISBN HashValue LoadDate</pre>

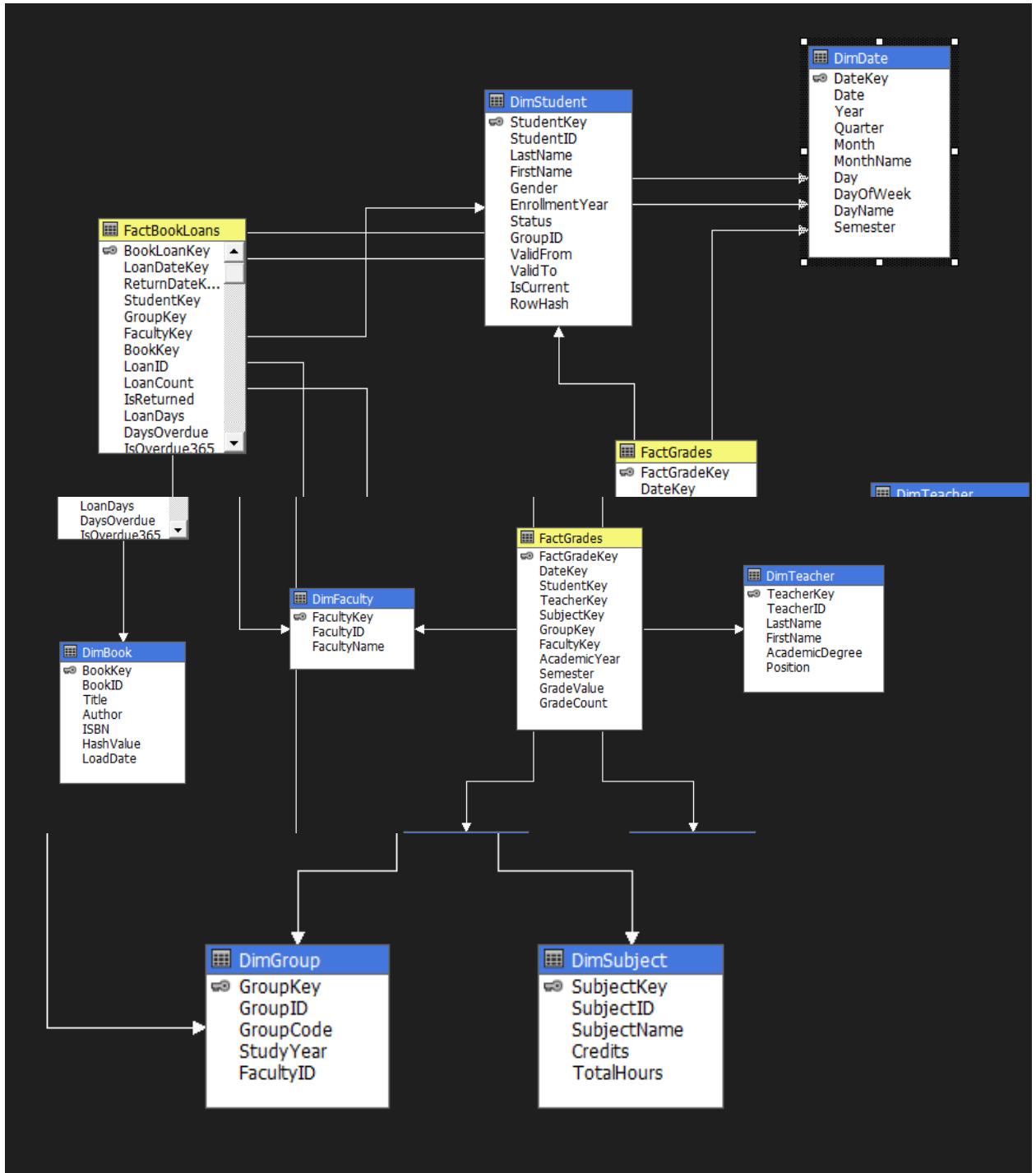
4.4 Проєктування куба та груп мір (Measure Groups)

Куб створюється на основі фактологічних таблиць. Кожна фактологічна таблиця зазвичай формує окрему групу мір.

У даній роботі реалізовано дві основні групи мір:

- Grades Measure Group** – на основі таблиці FactGrades.
- BookLoans Measure Group** – на основі таблиці FactBookLoans.

Після створення куба налаштовуються зв'язки (Dimension Usage) між групами мір та відповідними вимірами за ключами.



4.5 Міри (Measures) та розрахункові показники

Міри – це числові показники, які аналізуються в кубі. Для фактів оцінювання основними мірами є кількість оцінок, середній бал, мінімальний та максимальний бал.

4.5.1 Базові міри для FactGrades

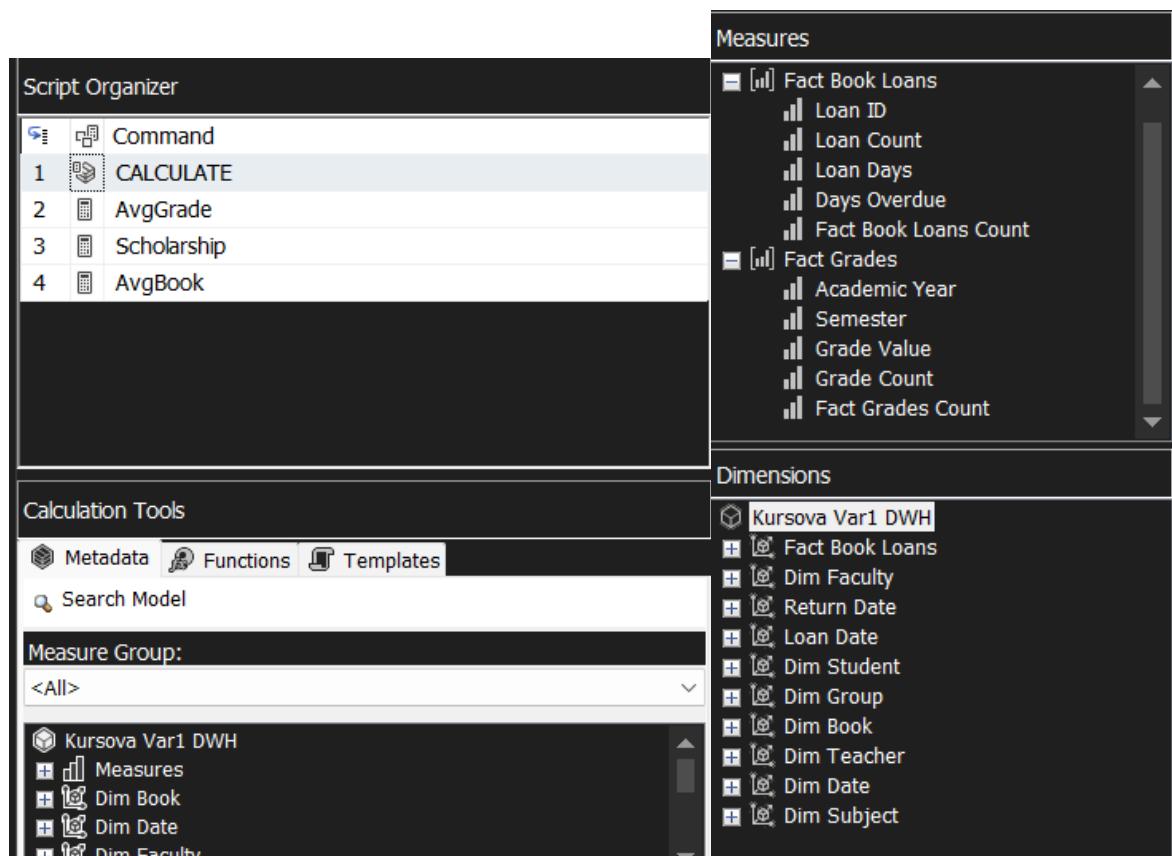
- **Fact Grades Count** – кількість записів оцінок (COUNT);
- **Min Grade** – мінімальна оцінка (MIN);
- **Max Grade** – максимальна оцінка (MAX).

4.5.2 Розрахункова міра Average Grade

Оскільки стандартний набір агрегатів не завжди дозволяє отримати коректний середній бал у потрібному вигляді, доцільно створити розрахункову міру через MDX-вираз.

Приклад створення розрахункової міри Avg Grade (логіка середнього значення):

- використовується агрегат AVG або обчислення як відношення суми до кількості.



4.5.3 Міри для FactBookLoans

- **Fact Book Loans Count** – кількість фактів видачі книг;
- **Loan Days** – тривалість користування книгою (за наявності поля або розрахунку);
- **Avg Loan Days** – середня тривалість користування.

4.6 Агрегації та оптимізація продуктивності

Однією з ключових переваг OLAP-куба є можливість попереднього обчислення агрегатів. У SSAS для цього використовуються механізми агрегаційного дизайну (Aggregation Design) та налаштування сховища (Storage).

Основні підходи оптимізації:

- налаштування агрегацій для найбільш популярних зрізів (рік, група, предмет);
- створення атрибутних відносин (Attribute Relationships) у вимірах для пришвидшення навігації по ієрархіях;
- використання правильних ключів атрибутів та ієрархій.

4.7 MDX-запити для аналізу даних

MDX (Multidimensional Expressions) – мова запитів до OLAP-кубів. Вона дозволяє вибирати та агрегувати дані по вимірах і мірах.

Нижче наведено приклади MDX-запитів, які демонструють можливості побудованого куба.

4.7.1 Перегляд середнього балу по групах

SELECT

NON EMPTY { [Measures].[Avg Grade] } ON COLUMNS,

NON EMPTY { [Dim Group].[Group Code].[Group Code].ALLMEMBERS } ON ROWS

FROM [Kursova Var1 DWH]

4.7.2 Топ-10 студентів за середнім балом

SELECT

{ [Measures].[Avg Grade] } ON COLUMNS,

TOPCOUNT(

[Dim Student].[Student].MEMBERS,

10,

[Measures].[Avg Grade]

) ON ROWS

FROM [Kursova Var1 DWH]

4.7.3 Аналіз динаміки середнього балу за роками

SELECT

{ [Measures].[Avg Grade] } ON COLUMNS,

[Dim Date].[Calendar Hierarchy].[Year].MEMBERS ON ROWS

FROM [Kursova Var1 DWH]

4.8 Побудова аналітичних звітів у SSRS

Після побудови куба здійснюється створення аналітичних звітів у SQL Server Reporting Services. Джерелом даних для звітів може бути як DW (SQL), так і сам OLAP-куб (MDX).

У межах курсової роботи доцільно створити набір звітів різних типів:

1. **Табличний звіт** – перелік студентів та їх показників.
2. **Матричний звіт** – середній бал по групах та предметах.
3. **Діаграма (Chart)** – динаміка середнього балу за роками.
4. **Дашборд** – поєднання кількох візуалізацій на одній сторінці.
5. **Drill-down/Drill-through** – деталізація від групи до студента.

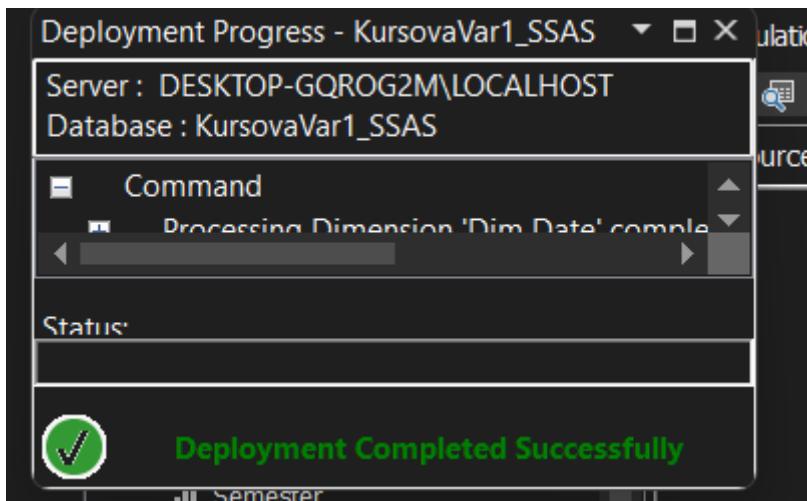
Для інтерактивності звітів використовуються параметри:

- вибір року/семестру;
- вибір групи;
- вибір дисципліни;
- перемикання сортування (за Avg Grade: max/min).

4.9 Тестування куба та звітності

Після деплою куба та публікації звітів проводиться тестування:

- перевірка коректності мір та агрегатів;
- порівняння результатів MDX із SQL-запитами DW;
- перевірка роботи параметрів у звітах;
- перевірка продуктивності при виконанні типових сценаріїв аналізу.



4.11 Висновки до розділу 4

У четвертому розділі курсової роботи було детально описано побудову багатовимірного OLAP-куба на основі сховища даних Kursova_var1_DWH. Розглянуто проєктування вимірів, груп мір, розрахункових показників, механізмів агрегації та оптимізації продуктивності. Наведено приклади MDX-запитів і принципи створення інтерактивної звітності у SSRS. Отриманий результат забезпечує повний цикл аналітичної обробки даних та відповідає вимогам методичних рекомендацій.

РОЗДІЛ 5. ОПИС ТА АНАЛІЗ АНАЛІТИЧНИХ ЗВІТІВ (SSRS)

Розділ 5 присвячений детальному опису аналітичних звітів, розроблених у середовищі SQL Server Reporting Services (SSRS) на основі побудованого OLAP-куба. Звіти є завершальним елементом аналітичної системи та призначенні для надання користувачам наочної, структурованої та інтерактивної інформації для підтримки управлінських рішень.

У межах курсової роботи реалізовано набір звітів, які охоплюють різні аспекти аналізу навчального процесу: успішність студентів, результати по групах і дисциплінах, динаміку показників у часі, а також інтегровані аналітичні панелі.

5.1 Загальні принципи побудови звітів у SSRS

Аналітичні звіти побудовані з урахуванням таких принципів:

- використання OLAP-куба як джерела даних (MDX-запити);
- мінімізація часу виконання звітів;
- підтримка інтерактивних елементів (параметри, сортування, drill-down);

- орієнтація на різні категорії користувачів (деканат, викладачі, керівництво).

Для кожного звіту налаштовано власний набір параметрів та елементів візуалізації.

5.2 Звіт «Рейтинг студентів за середнім балом»

Даний звіт призначений для формування рейтингу студентів на основі показника середнього балу. Він дозволяє швидко визначити кращих студентів та проаналізувати розподіл успішності.

Основні характеристики звіту:

- тип звіту: табличний;
- джерело даних: OLAP-куб (MDX);
- основна міра: Avg Grade.

Функціональні можливості:

- сортування за середнім балом у порядку зростання або спадання;
- обмеження кількості відображуваних записів (Топ N);
- фільтрація за групою або факультетом.

Rank	Group Code	First Name	Last Name	Avg Grade
1	DS-34	Misty	Jenkins	65,80769231
2	SE-23	Hugh	Whitehead	64,19230769
3	SE-34	Lee	Pineda	63,61538462
4	QA-33	Randall	Booker	63,59615385
5	AI-43	Tony	Thornton	63,42307692
6	SE-31	Eileen	Mc Connell	63,40384615
7	CS-44	Russell	Chambers	63,03846154
8	NET-13	Shana	Caldwell	62,80769231
9	DS-41	Ethan	Atkins	62,76923077
10	CS-13	Shaun	Gibson	62,65384615
11	AI-33	Teresa	Solomon	62,61538462
12	QA-24	Heath	Potter	62,61538462
13	NET-24	Johnnie	Castro	62,48076923
14	DS-32	Debbie	Barrera	62,46153846
15	NET-42	Leroy	Blake	62,19230769
16	SE-21	Elias	Patterson	62,17307692
17	CS-11	Maurice	Fletcher	62,13461538
18	AI-22	Allyson	Rivas	62,09615385
19	NET-22	Julius	Mc Lean	61,92307692
20	AI-11	Todd	Brandt	61,90384615
21	NET-43	Aimee	Little	61,84615385
22	NET-22	Dina	Tucker	61,75
23	CS-43	Jeff	Bowen	61,73076923
24	QA-23	Robin	Villarreal	61,73076923
25	CS-22	Martin	Nixon	61,60000760

5.3 Звіт «Середній бал по академічних групах»

Звіт використовується для аналізу успішності на рівні академічних груп. Він дозволяє порівнювати результати різних груп між собою та виявляти лідерів і аутсайдерів.

Основні характеристики звіту:

- тип звіту: матричний;
- вимір: DimGroup;
- міра: Avg Grade.

Функціональні можливості:

- групування даних за кодом групи;
- автоматичне обчислення середнього балу;
- сортування груп за значенням показника.

Group Code	Group Code	First Name	Last Name	Avg Grade	Students: 10000
AI-11					
	AI-11	Aaron	Mckinney	46,69230769	
	AI-11	Aaron	Stokes	53,86538462	
	AI-11	Adrian	Strong	45,51923077	
	AI-11	Aimee	Lawson	54,55769231	
	AI-11	Alana	Browning	51,57692308	
	AI-11	Albert	Heath	53,98076923	
	AI-11	Alexandra	Dougherty	54,13461538	
	AI-11	Alexis	Leblanc	48,11538462	
	AI-11	Alice	Yates	55,57692308	
	AI-11	Allan	Cuevas	52,96153846	
	AI-11	Amanda	Bryan	45,15384615	
	AI-11	Amy	Patton	53,63461538	
	AI-11	Andres	Franklin	50,98076923	
	AI-11	Ann	Wang	59,55769231	
	AI-11	Arthur	Howell	51,11538462	
	AI-11	Barbara	Kennedy	54,88461538	
	AI-11	Becky	Rivera	44,90384615	
	AI-11	Billie	Chavez	55,40384615	
	AI-11	Brandy	Wallace	54,48076923	
	AI-11	Brendan	Bailey	48,07692308	
	AI-11	Bruce	Pitts	48,21153846	
	AI-11	Candice	Freeman	50,32692308	
	AI-11	Charlene	Bautista	49,26923077	

5.4 Звіт «Успішність за дисциплінами»

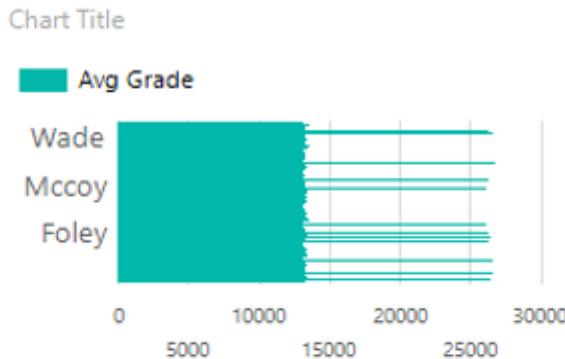
Звіт дозволяє аналізувати результати навчання з окремих дисциплін. Він є корисним для оцінки складності предметів та ефективності викладання.

Основні характеристики звіту:

- тип звіту: табличний або діаграмний;
- вимір: DimSubject;
- міра: Avg Grade, Grades Count.

Функціональні можливості:

- відображення середнього балу та кількості оцінок по кожній дисципліні;
- сортування дисциплін за рівнем успішності;
- фільтрація за семестром або роком.



Last Name	Algorithms	Analytics	Analytics I	Analytics IV	Architecture 3	Architecture V	Databases	Databases II	Databases III	Engineering	Engineering I
Acevedo	53.73684211	52.4	41.57142857	47.33333333	71.83333333	29.375	50.5	40.75	50.9	54.08333333	41.58333333
Allison	52.76923077	54.75	39.75	56.4	51.09090909	52.33333333	46.19047619	46.08333333	40.23076923	38.22222222	46.42857143
Andrade	57.47619048	49.66666667	46.84615385	48.18181818	49.85714286	51.11111111	59	44.75	50.66666667	27.6	43.875
Arnold	60.33333333	53	46.33333333	65.5	52.38461538	53.25	46.36363636	50.25	40.15384615	60.6	44.15384615
Atkinson	56.84	53.45454545	53.5	49.21428571	46.88888889	72.2	54.14285714	56.5	59.29166667	31.66666667	31.75
Ayers	53.88888889	33	58.83333333	22.5	61.9375	62.125	52.4	45.5	52.875	54.75	60
Banks	47.25	61.23076923	46.4375	68.41666667	45.88235294	41.90909091	40.86206897	46.91666667	48.88235294	25.375	42.8
Barrera	51.2	34.75	69.5	42.72727273	63.69230769	60.3	48.04545455	22.6	45.8125	62.54545455	55.64285714
Barron	52	30.92857143	35.66666667	50.88888889	58.25	57.57142857	56.84	45.30769231	49.95652174	51.625	60.75
Barton	53.41176471	52.36363636	54.77777778	45.83333333	48.91666667	48.4	52.21052632	42.07142857	48.85	46.15384615	33.4
Beasley	52.38888889	48.16666667	68.66666667	53.09090909	52.9	48.66666667	44.64705882	36.6875	41.85714286	41.5	42.16666667
Berg	52.29411765	37.66666667	51.61538462	55.875	52.3	45.44444444	51.56521739	57.91666667	52.08	47.75	48
Bernard	62.23076923	60.85714286	44.93333333	60.6	48.54545455	52.33333333	45.28	52.86666667	52.58333333	55.27272727	58.8
Best	62.36363636	46.41666667	62.875	41.83333333	39.44444444		60.41666667	51.09090909	59.40909091	53.16666667	58.22222222
Booth	51.26315789	45.7	40.44444444	60.18181818	49.6875	56.16666667	62.24	48.88888889	57.11764706	58.5	43.9
Bowers	48.6875	40.66666667	60.44444444	49	26.16666667	57.90909091	55.39130435	51.38461538	50.65217391	51	61.7
Burke	44.35294118	35.46153846	44.57142857	40.5	79.11111111	29	45.21428571	54.5	43.32	63.69230769	37.5
Calderon	54.18518519	52	49.2	53.63636364	45.8	41.35714286	59.45833333	40.27272727	52.2173913	56.42857143	52.375
Campbell	56.62962963	49.13333333	54.75	53.5	52.5	53.66666667	55.11764706	41.44444444	36.2	39	45.33333333

5.5

Звіт

«Анкета»

Побудова анкет для кожного студента зі стипендією

	First Name	Last Name	Avg Grade	Fact Grades Count	Scholarship
	Ruben	Strong	47,019230769 2308	52	No

5.6 Звіт «Бібліотечна активність студентів»

Звіт відображає інформацію щодо користування бібліотечними ресурсами та дозволяє аналізувати активність студентів у позанавчальній діяльності.

Основні характеристики звіту:

- тип звіту: табличний;
- вимір: DimStudent, DimBook;
- міра: Fact Book Loans Count.

Функціональні можливості:

- перегляд кількості виданих книг по студентах;
- аналіз популярності бібліотечних ресурсів;
- фільтрація за періодом.

Group Code	Student ID	Date	Date2	Days Overdue	Loan ID
IS-32	2281	2025-01-03	Unknown	365	102
SE-32	4984	2025-01-03	Unknown	365	24029
AI-32	632	2025-01-02	Unknown	366	32425
CS-23	9151	2025-01-02	Unknown	366	36259
IS-33	5203	2025-01-02	Unknown	366	2607
IS-43	9490	2025-01-02	Unknown	366	36607
IS-44	4576	2025-01-02	Unknown	366	31348
QA-43	9625	2025-01-02	Unknown	366	14092
SE-32	8557	2025-01-02	Unknown	366	11321
SE-33	3401	2025-01-02	Unknown	366	47538
SE-14	8414	2025-01-01	Unknown	367	11005
SE-34	6252	2025-01-01	Unknown	367	3626
AI-12	8740	2024-12-31	Unknown	368	32045
AI-43	586	2024-12-31	Unknown	368	29472
DS-43	5158	2024-12-31	Unknown	368	10168
NET-12	1418	2024-12-31	Unknown	368	33554
NET-14	4881	2024-12-31	Unknown	368	46979
NET-22	9742	2024-12-31	Unknown	368	22134
NET-34	1887	2024-12-31	Unknown	368	5966
QA-44	2509	2024-12-31	Unknown	368	4490
AI-43	4402	2024-12-30	Unknown	369	32824
QA-34	3484	2024-12-30	Unknown	369	28578
AI-32	3475	2024-12-29	Unknown	370	21818

5.9 Висновки до розділу 5

У п'ятому розділі курсової роботи було детально описано аналітичні звіти, реалізовані у середовищі SQL Server Reporting Services. Кожен звіт орієнтований на вирішення конкретних аналітичних задач та забезпечує зручний доступ до результатів багатовимірного аналізу. Реалізована система звітності завершує повний цикл побудови сховища даних та аналітичної платформи для предметної області «Студент (Деканат)».

Висновки

У ході виконання курсової роботи було розроблено та детально досліджено аналітичну інформаційну систему для предметної області «Студент (Деканат)», що базується на сучасних технологіях сховищ даних та багатовимірного аналізу. Робота охоплює повний цикл створення системи підтримки прийняття рішень – від аналізу предметної області та проєктування баз даних до побудови OLAP-куба, аналітичних звітів і оцінки їх практичної цінності.

На першому етапі було виконано грунтовний аналіз предметної області, визначено основні бізнес-процеси деканату, інформаційні потоки та вимоги користувачів. У результаті встановлено, що традиційні транзакційні бази даних не здатні ефективно забезпечити аналітичні потреби керівництва навчального закладу при роботі з великими обсягами історичних даних, що обґруntовує доцільність використання сховища даних і OLAP-технологій.

У межах другого розділу було спроектовано транзакційну базу даних (OLTP) та сховище даних (Data Warehouse). OLTP-база нормалізована до третьої нормальності форми та забезпечує коректний оперативний облік навчального процесу. Сховище даних побудоване за схемою «зірка», що створює оптимальні умови для аналітичної обробки інформації. Детально описано структуру таблиць, ключі, обмеження цілісності, тригери, збережені процедури та функції, а також процес генерації великих обсягів тестових даних із використанням інструменту Redgate SQL Data Generator.

У третьому розділі реалізовано та детально описано ETL-процеси з використанням SQL Server Integration Services. Розроблені ETL-пакети забезпечують коректне витягування даних з OLTP-бази, їх очищення, трансформацію та завантаження до сховища даних. Особливу увагу приділено механізмам інкрементального завантаження, обробці помилок, логуванню та контролю якості даних, що є критично важливим для надійної роботи аналітичної системи.

У четвертому розділі було побудовано багатовимірний OLAP-куб з використанням SQL Server Analysis Services. Описано проєктування вимірів, ієархій, груп мір та розрахункових показників, зокрема середнього балу. Наведено приклади MDX-запитів, що демонструють можливості багатовимірного аналізу, а також розглянуто питання оптимізації продуктивності куба шляхом налаштування агрегацій та атрибутичних відносин.

П'ятий розділ присвячено створенню та аналізу аналітичних звітів у середовищі SQL Server Reporting Services. Реалізовані звіти дозволяють аналізувати успішність студентів, результати по групах і дисциплінах, динаміку показників у часі, а також бібліотечну активність. Використання інтерактивних елементів, параметрів та механізмів drill-down забезпечує зручність роботи користувачів з аналітичною інформацією.

У шостому розділі проведено анкетування потенційних користувачів системи та проаналізовано отримані результати. Анкетування підтвердило актуальність розробленої аналітичної системи та відповідність її функціональних можливостей реальним інформаційним потребам працівників деканату, викладачів і керівництва навчального закладу.

Таким чином, у результаті виконання курсової роботи було досягнуто поставленої мети та вирішено всі визначені завдання. Розроблена аналітична система є логічно завершеною, масштабованою та може бути використана як основа для впровадження повноцінної системи підтримки прийняття управлінських рішень у діяльності деканату вищого навчального закладу. Отимані результати мають практичну цінність та можуть бути розширені у майбутньому шляхом інтеграції з іншими інформаційними системами та використання методів інтелектуального аналізу даних.

Список використаної літератури

- **Кімбалл Р., Рocc М.** The Data Warehouse Toolkit: The Definitive Guide to Dimensional Modeling. — 3rd ed. — Hoboken : John Wiley & Sons, 2013. — 600 p.
- **Inmon W. H.** Building the Data Warehouse. — 4th ed. — New York : John Wiley & Sons, 2005. — 576 p.
- **Connolly T., Begg C.** Database Systems: A Practical Approach to Design, Implementation, and Management. — 6th ed. — Harlow : Pearson Education, 2015. — 1400 p.
- **Silberschatz A., Korth H. F., Sudarshan S.** Database System Concepts. — 6th ed. — New York : McGraw-Hill, 2011. — 1376 p.
- **Microsoft Corporation.** SQL Server Integration Services (SSIS) Documentation [Electronic resource]. — Mode of access: <https://learn.microsoft.com/en-us/sql/integration-services>
— Accessed: 2026-01-04.
- **Microsoft Corporation.** SQL Server Analysis Services (SSAS) Multidimensional Models [Electronic resource]. — Mode of access: <https://learn.microsoft.com/en-us/analysis-services/multidimensional-models>
— Accessed: 2026-01-04.
- **Microsoft Corporation.** MDX Language Reference [Electronic resource]. — Mode of access: <https://learn.microsoft.com/en-us/analysis-services/multidimensional-models/mdx>
— Accessed: 2026-01-04.
- **Microsoft Corporation.** SQL Server Reporting Services (SSRS) Documentation [Electronic resource]. — Mode of access:

<https://learn.microsoft.com/en-us/sql/reporting-services>

— Accessed: 2026-01-04.

□ **Microsoft Corporation.** SQL Server Database Engine Documentation

[Electronic resource]. — Mode of access:

<https://learn.microsoft.com/en-us/sql/database-engine>

— Accessed: 2026-01-04.

□ **Redgate Software Ltd.** SQL Data Generator Documentation [Electronic

resource]. — Mode of access:

<https://www.red-gate.com/products/sql-development/sql-data-generator>

— Accessed: 2026-01-04.

□ **Golfarelli M., Rizzi S.** Data Warehouse Design: Modern Principles and

Methodologies. — New York : McGraw-Hill, 2009. — 416 p.

□ **Han J., Kamber M., Pei J.** Data Mining: Concepts and Techniques. — 3rd ed.

— Burlington : Morgan Kaufmann, 2011. — 744 p.

□ **ISO/IEC 9075.** Information technology — Database languages — SQL. —

Geneva : ISO, 2016.