Міністерство освіти і науки України

Тернопільський національний економічний університет

Факультет комп’ютерних інформаційних технологій

Кафедра комп’ютерних наук

Практичне завдання №1

з дисципліни «Засоби програмування баз даних»

на тему «Програмування реляційних баз даних»

Тернопіль – ТНЕУ, 2018

Практична робота №1.

Програмування реляційних баз даних

**Завдання №1.**

1.1. Вибір завдання.

Вибрати з таблиці «Варіанти завдань для студентів» варіант завдання, що відповідає номеру студента в списку навчальної групи. Для всіх наступних практичних робіт варіант залишається незмінним. Кожному студенту надається свій варіант предметної області (ПО), який він буде використовувати в процесі виконання всіх практичних робіт.

1.2. Аналіз предметної області.

На підставі обраного варіанту привести: назву предметнної області, мета діяльності, структура, інформаційні потреби користувачів (коротко).

1.3. Опис основних сутностей ПО.

Тут слід привести опис основних сутностей (об'єктів) ПО. Відбір сутностей проводиться на основі аналізу інформаційних потреб. Необхідно привести таблиці опису сутностей (сутностей має бути не менше 5-х).

Таблиця 1.1.

Список сутностей предметної області.

|  |  |  |
| --- | --- | --- |
| № | Найменування сутності | Короткий опис |
| 1 |  |  |
| 2 |  |  |

Тут же наводиться відбір атрибутів (не менше 5-ти) для кожного екземпляра сутності. Відбираються тільки ті атрибути сутності, які необхідні для формування відповідей на регламентовані і непередбачені запити. Для кожного об'єкта слід привести таблиці його атрибутів.

Таблиця 1.2.

Список атрибутів.

|  |  |  |
| --- | --- | --- |
| № | Найменування атрибутів | Короткий опис |
| 1 |  |  |
| 2 |  |  |

На основі аналізу інформаційних запитів слід виявити зв’язки між сутностями. Для виявлених зв'язків також потрібно заповнити таблицю 1.3.

Таблиця 1.3.

Список атрибутів.

|  |  |  |  |
| --- | --- | --- | --- |
| № | Найменування зв’язку | Сутності, які необхідні для зв’язку | Короткий опис |
| 1 |  |  |  |
| 2 |  |  |  |

1.4. Побудова інфологічної моделі.

На підставі раніше обраного варіанту і таблиць 1.1-1.3:

- описати класи об'єктів (сутностей) і їх властивості;

- розставити існуючі зв'язки між ними;

- на підставі табл. 1.3. в письмовій формі обґрунтувати типи зв'язків (1: 1, 1: Б і т.д.).

При графічній побудові ІМД слід дотримуватися єдиного масштабу для всієї схеми. Всі прямокутники, що позначають класи об'єктів, повинні бути одного розміру. Аналогічно, всі ромби з іменами зв'язків також повинні мати однаковий розмір.

1.5. Побудова даталогічної моделі.

На підставі раніше обраного варіанту і таблиць 1.1-1.3, інфологічної моделі і нормалізації БД необхідно:

- провести відповідність ключів для кожної таблиці 1.1-1.3,

- заповнити для кожної таблиці БД форму, згідно табл. 1.4.

Таблиця 1.4. Структура таблиці для даталогічної моделі

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| № | Найменування  реквізиту | Ідентифікатор | Тип | Довжина | Формат | Обмеження та коментарі |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

Зміст звіту

1. Назва та мета роботи.

2. Словесний і схематичний аналіз предметної області (ПО), включаючи схему структури предметної області.

3. Заповнені таблиці 1.1 - 1.3. з описом основних сутностей ПО.

4. Інфологічна модель БД, відповідно до варіанту.

5. Обґрунтування типів зв'язку інфологічної моделі даних.

6. Даталогічна модель БД (табл. 1.4.).

Таблиця 1. Варіанти завдань для практичної роботи №1

|  |  |
| --- | --- |
| № варіанта | Умови завдання |
| Варіант №1 | На підставі обраного варіанту виконати наступне:  1. Виконати аналіз предметної області досліджуваної області;  2. Описати основні сутності предметної області;  3. Розставити існуючі зв'язки між сутностями: самостійно додати в кожну сутність первинні ключі та встановити зовнішні ключі між сутностями;  4. Побудувати інфологічну модель бази даних;  5. Побудувати даталогічну модель бази даних організації.  БД - успішність студентів ВНЗ. БД складається з наступних таблиць:  факультети, кафедри, навчальні групи, студенти, відомості успішності  Таблиця факультети має наступні атрибути: назва факультету, ПІБ декана, номер кімнати, номер корпусу, телефон.  Таблиця кафедра має наступні атрибути: назва кафедри, факультет, ПІБ завідувача, номер кімнати, номер корпусу, телефон, кіл-ть викладачів.  Таблиця навчальні групи має такі атрибути: назва групи, рік вступу, курс навчання, кількість студентів в групі.  Таблиця студенти має наступні атрибути: студента прізвище, ім'я,батькові, група, рік народження, стать, адреса, місто, телефон.  Таблиця відомості успішності має наступні атрибути: група, студент, предмет, оцінка. |
| Варіант №2 | На підставі обраного варіанту виконати наступне:  1. Виконати аналіз предметної області досліджуваної області;  2. Описати основні сутності предметної області;  3. Розставити існуючі зв'язки між сутностями: самостійно додати в кожну сутність первинні ключі та встановити зовнішні ключі між сутностями;  4. Побудувати інфологічну модель бази даних;  5. Побудувати даталогічну модель бази даних організації.  БД - інформаційна система супермаркету. БД складається з наступних таблиць: відділи, співробітники, товари, продаж товарів, посади.  Таблиця відділи має наступні атрибути: назва відділу, к-ть прилавків, кількість продавців, номер залу.  Таблиця співробітники має наступні атрибути: прізвище, ім'я, батькові, відділ, рік народження, рік прийняття на роботу, стаж, посаду, стать, адреса, місто, телефон.  Таблиця посади має наступні атрибути: назва посади, сума ставки.  Таблиця товари має наступні атрибути: назва товару, відділ, країна виробник, умови зберігання, термін зберігання.  Таблиця продаж товарів має такі атрибути: співробітник, що є продавцем, товару дата, час, кількість, ціна, сума. |
| Варіант №3 | На підставі обраного варіанту виконати наступне:  1. Виконати аналіз предметної області досліджуваної області;  2. Описати основні сутності предметної області;  3. Розставити існуючі зв'язки між сутностями: самостійно додати в кожну сутність первинні ключі та встановити зовнішні ключі між сутностями;  4. Побудувати інфологічну модель бази даних;  5. Побудувати даталогічну модель бази даних.  БД - інформаційна система військового округу. БД складається з наступних таблиць: місця дислокації, вид військ, частини, роти, особовий склад.  Таблиця вид військ має наступні атрибути: назва.  Таблиця місця дислокації має наступні атрибути: країна, місто, адреса, яку займає площу.  Таблиця частини має наступні атрибути: номер частини, місце дислокації, вид військ, кількість рот.  Таблиця роти має наступні атрибути: назва роти, к-ть службовців.  Таблиця особовий склад має наступні атрибути: прізвище, рота, посаду, рік народження, рік надходження на службу, вислуга років, нагороди, участь у військових заходах |
| Варіант №4 | На підставі обраного варіанту виконати наступне:  1. Виконати аналіз предметної області досліджуваної області;  2. Описати основні сутності предметної області;  3. Розставити існуючі зв'язки між сутностями: самостійно додати в кожну сутність первинні ключі та встановити зовнішні ключі між сутностями;  4. Побудувати інфологічну модель бази даних;  5. Побудувати даталогічну модель бази даних.  БД - інформаційна система бібліотеки. БД складається з наступних таблиць: бібліотеки, фонд бібліотеки, тип літератури, співробітники, поповнення фонду.  Таблиця бібліотеки має наступні атрибути: назва, адреса, місто.  Таблиця фонд бібліотеки має наступні атрибути: назва фонду, бібліотека, кількість книг, кількість журналів, кількість газет, кількість  збірок, кількість дисертацій, кількість рефератів.  Таблиця тип літератури має наступні атрибути: назва типу.  Таблиця співробітники має наступні атрибути: прізвище співробітника, бібліотека, посаду, рік народження, рік надходження на роботу, освіту, зарплата.  Таблиця поповнення фонду має такі атрибути: фонд, співробітник, дата, назва джерела літератури, тип літератури, видавництво, дата видання, кількість примірників. |
| Варіант №5 | На підставі обраного варіанту виконати наступне:  1. Виконати аналіз предметної області досліджуваної області;  2. Описати основні сутності предметної області;  3. Розставити існуючі зв'язки між сутностями: самостійно додати в кожну сутність первинні ключі та встановити зовнішні ключі між сутностями;  4. Побудувати інфологічну модель бази даних;  5. Побудувати даталогічну модель бази даних.  БД - інформаційна система туристичного агентства. БД складається з наступних таблиць: пансіонати, тури, клієнти, путівки, вид житла.  Таблиця пансіонати має наступні атрибути: назва пансіонату, адреса, місто, країна, телефон, опис території, кількість кімнат, наявність басейну, наявність медичних послуг, наявність спа-салону, рівень пансіонату, відстань до моря.  Таблиця вид житла має наступні атрибути: назва (будинок, бунгало, квартира, 1-я кімната, 2-я кімната і т.д.), категорія житла (люкс, напівлюкс, і т.д.), пансіонат, опис умов проживання, ціна за номер на добу.  Таблиця тури має наступні атрибути: назва туру (Європа, середня Азія, тибет і т.д.), вид транспорту, категорія житла на ніч. (Готель, готель, намет і т.д.), вид харчування (одноразове, дворазове, триразове, сніданки), ціна туру на добу.  Таблиця клієнти має наступні атрибути: прізвище, ім'я, по батькові, паспортні дані, дата народження, адреса, місто, телефон.  Таблиця путівки має наступні атрибути: клієнт, пансіонату, вид житла, дата заїзду, дата від'їзду, наявність дітей, наявність мед. страховки, Кількість осіб, ціна, сума |
| Варіант №6 | На підставі обраного варіанту виконати наступне:  1. Виконати аналіз предметної області досліджуваної області;  2. Описати основні сутності предметної області;  3. Розставити існуючі зв'язки між сутностями: самостійно додати в кожну сутність первинні ключі та встановити зовнішні ключі між сутностями;  4. Побудувати інфологічну модель бази даних;  5. Побудувати даталогічну модель бази даних.  БД - інформаційна система автопідприємства міста. БД складається з наступних таблиць: автотранспорт, водії, маршрути, обслуговуючий персонал, гаражне господарство.  Таблиця автотранспорт має наступні атрибути: назва транспорту (автобуси, таксі, маршрутні таксі, інший легковий транспорт, вантажний транспорт і т.д.), к-ть напрацювання, пробіг, кількість ремонтів, характеристика.  Таблиця маршрути має наступні атрибути: назва маршруту, транспорт, водій, графік роботи.  Таблиця водії має наступні атрибути: прізвище, ім'я, по батькові, рік народження, рік надходження на роботу, стаж, посада, стать, адреса, місто, телефон.  Таблиця обслуговуючий персонал має наступні атрибути: посаду (техніки, зварювальники, слюсарі, складальники і ін.), прізвище, ім'я, батькові, рік народження, рік прийняття на роботу, стаж, стать, адреса, місто, телефон.  Таблиця гаражне господарство має такі атрибути: назва гаража, транспорт на ремонті, вид ремонту, дата надходження, дата видачі після ремонту, результат ремонту, персонал, що здійснює ремонт. |
| Варіант №7 | На підставі обраного варіанту виконати наступне:  1. Виконати аналіз предметної області досліджуваної області;  2. Описати основні сутності предметної області;  3. Розставити існуючі зв'язки між сутностями: самостійно додати в кожну сутність первинні ключі та встановити зовнішні ключі між сутностями;  4. Побудувати інфологічну модель бази даних;  5. Побудувати даталогічну модель бази даних.  БД - інформаційна система поліклініки. БД складається з наступних таблиць: лікарі, пацієнти, історія хвороб, відділення, обслуговуючий персонал.  Таблиця відділення має такі атрибути: назва відділення (Хірургія, терапія, нервологія і т.д.), поверх, номери кімнат, ПІБ завідувача.  Таблиця лікарі має наступні атрибути: прізвище, ім'я, по батькові, посаду, стаж роботи, наукове звання, адреса, номер відділення, в якому він працює.  Таблиця пацієнти має наступні атрибути: прізвище, ім'я, по батькові, адреса, місто, вік, стать.  Таблиця діагнози має наступні атрибути: назва діагнозу, ознаки хвороби, період лікування, призначення.  Таблиця історія хвороби має наступні атрибути: пацієнт, лікар, діагноз, лікування, дата захворювання, дата виліковування, вид лікування (амбулаторне, стаціонарне). |
| Варіант №8 | На підставі обраного варіанту виконати наступне:  1. Виконати аналіз предметної області досліджуваної області;  2. Описати основні сутності предметної області;  3. Розставити існуючі зв'язки між сутностями: самостійно додати в кожну сутність первинні ключі та встановити зовнішні ключі між сутностями;  4. Побудувати інфологічну модель бази даних;  5. Побудувати даталогічну модель бази даних.  БД - інформаційна система лікарні. БД складається з наступних таблиць: лікарі, пацієнти, історія хвороб, операції, лікарняний лист.  Таблиця лікарі має наступні атрибути :, прізвище, ім'я, по батькові,  посаду, стаж роботи, наукове звання, адреса.  Таблиця пацієнти має наступні атрибути: прізвище, ім'я, по батькові, адреса, місто, вік, стать.  Таблиця історія хвороби має наступні атрибути :, пацієнта лікар, діагноз, дата захворювання, дата виліковування, вид лікування (амбулаторне, стаціонарне), код операції.  Таблиця лікарняний лист має такі атрибути: дата лікування, історія хвороби, ліки, температура, тиск, стан хворого (важке, середнє, і т.д.).  Таблиця операції має такі атрибути: опис операції (Видалення апендициту, пластична операція і т.д.), лікар, дата операції, пацієнт, результат операції. |
| Варіант №9 | На підставі обраного варіанту виконати наступне:  1. Виконати аналіз предметної області досліджуваної області;  2. Описати основні сутності предметної області;  3. Розставити існуючі зв'язки між сутностями: самостійно додати в кожну сутність первинні ключі та встановити зовнішні ключі між сутностями;  4. Побудувати інфологічну модель бази даних;  5. Побудувати даталогічну модель бази даних.  БД - інформаційна система бібліотек міста. БД складається з наступних таблиць: бібліотеки, читальні зали, література, читачі, видача літератури.  Таблиця бібліотеки має наступні атрибути: назва, адреса, місто.  Таблиця читальні зали має наступні атрибути: назва читального залу, бібліотека, кількість одиниць літератури, кількість посадкових місць, час роботи, поверх, к-ть співробітників.  Таблиця читачі має наступні атрибути: прізвище, ім'я, по батькові, категорія читача, місце роботи або навчання, вік, дата реєстрації в бібліотеці.  Таблиця література має наступні атрибути: назва, категорія літератури, автори, видавництво, рік видавництва, кількість сторінок, читальна зала.  Таблиця видача літератури має наступні атрибути: читач, література, дата видачі, термін видачі, вид видачі, наявність застави. |
| Варіант №10 | На підставі обраного варіанту виконати наступне:  1. Виконати аналіз предметної області досліджуваної області;  2. Описати основні сутності предметної області;  3. Розставити існуючі зв'язки між сутностями: самостійно додати в кожну сутність первинні ключі та встановити зовнішні ключі між сутностями;  4. Побудувати інфологічну модель бази даних;  5. Побудувати даталогічну модель бази даних.  БД - інформаційна система автосалону. БД складається з наступних таблиць: автомобілі, марка автомобіля, співробітники, продаж автомобілів, покупці.  Таблиця марка автомобіля має наступні атрибути: назва марки, країна виробник, завод виробник, адреса.  Таблиця автомобіля має наступні атрибути: назва автомобіля, марка, рік виробництва, колір, категорія, ціна.  Таблиця покупці має наступні атрибути: прізвище, ім'я, батькові, паспортні дані, адреса, місто, вік, стать.  Таблиця співробітника має наступні атрибути: прізвище, ім'я, батькові, стаж, зарплата.  Таблиця продаж автомобілів має наступні атрибути: дата, співробітник, автомобіль, покупець. |
| Варіант №11 | На підставі обраного варіанту виконати наступне:  1. Виконати аналіз предметної області досліджуваної області;  2. Описати основні сутності предметної області;  3. Розставити існуючі зв'язки між сутностями: самостійно додати в кожну сутність первинні ключі та встановити зовнішні ключі між сутностями;  4. Побудувати інфологічну модель бази даних;  5. Побудувати даталогічну модель бази даних.  БД - успішність студентів кафедри. БД складається з наступних таблиць: кафедри, дисципліни, викладачі, студенти, відомості успішності.  Таблиця кафедра має наступні атрибути: назва кафедри, факультет, ПІБ завідувача, номер кімнати, номер корпусу, телефон, кіл-ть викладачів.  Таблиця викладачі має наступні атрибути: прізвище, ім'я, батькові, кафедра, рік народження, рік надходження на роботу, стаж, посаду, стать, адреса, місто, телефон.  Таблиця студенти має наступні атрибути: прізвище, ім'я, по батькові, кафедра, рік народження, стать, адреса, місто, телефон.  Таблиця дисципліни має такі атрибути: назва дисципліни, кафедра, яку читає цю дисципліну, к-ть годин, вид підсумкового контроля.  Таблиця відомості успішності має наступні атрибути: викладач, дисципліна, студент, оцінка. |
| Варіант №12 | На підставі обраного варіанту виконати наступне:  1. Виконати аналіз предметної області досліджуваної області;  2. Описати основні сутності предметної області;  3. Розставити існуючі зв'язки між сутностями: самостійно додати в кожну сутність первинні ключі та встановити зовнішні ключі між сутностями;  4. Побудувати інфологічну модель бази даних;  5. Побудувати даталогічну модель бази даних.  БД - торгова організація. БД складається з наступних таблиць: торгова організація, торгова точка, продавці, постачальники, замовлення постачальникам.  Таблиця торгова організація має такі атрибути: назва організації торгівлі, адреса, ПІБ директора, податковий номер.  Таблиця торгова точка має наступні атрибути: назва торгової точки, тип (універмаги, магазини, кіоски, лотки і т.д.), торгова організація, адреса, ПІБ завідувача.  Таблиця продавці має наступні атрибути: прізвище, ім'я, по батькові, торгова точка, посаду, рік народження, стать, адреса проживання, місто.  Таблиця постачальники має наступні атрибути: назва постачальника, тип діяльності, країна, місто, адреса.  Таблиця замовлення постачальникам має наступні атрибути: дата замовлення, торгова точка, постачальник, назва товару, кількість, ціна. |
| Варіант №13 | На підставі обраного варіанту виконати наступне:  1. Виконати аналіз предметної області досліджуваної області;  2. Описати основні сутності предметної області;  3. Розставити існуючі зв'язки між сутностями: самостійно додати в кожну сутність первинні ключі та встановити зовнішні ключі між сутностями;  4. Побудувати інфологічну модель бази даних;  5. Побудувати даталогічну модель бази даних.  БД - проектна організація. БД складається з наступних таблиць: відділи, співробітники, організації, договору, проектні роботи.  Таблиця відділи має наступні атрибути: назва відділу, поверх, телефон, начальник відділу.  Таблиця співробітники має наступні атрибути: ПІБ, посада (Конструктори, інженери, техніки, лаборанти, інший обслуговуючий персонал), номер відділу, в якому працює, стать, адреса, дата народження.  Таблиця організації має наступні атрибути: назва організації, тип діяльності, країна, місто, адреса, ПІБ директора.  Таблиця договору має наступні атрибути: номер договору, дата укладення договору, організація, вартість договору.  Таблиця проектні роботи має наступні атрибути: дата початку проектної роботи, дата завершення проектної роботи, номер договору, відділ, який здійснює розробку. |
| Варіант №14 | На підставі обраного варіанту виконати наступне:  1. Виконати аналіз предметної області досліджуваної області;  2. Описати основні сутності предметної області;  3. Розставити існуючі зв'язки між сутностями: самостійно додати в кожну сутність первинні ключі та встановити зовнішні ключі між сутностями;  4. Побудувати інфологічну модель бази даних;  5. Побудувати даталогічну модель бази даних.  БД - інформаційна система військово-морського флоту. БД складається з наступних таблиць: бази, частини, особовий склад, кораблі, вчення.  Бази військово-морського флоту має наступні атрибути: назва бази, георгафічне розташування, кількість частин.  Таблиця частини має наступні атрибути: номер частини, база флоту, місце базування, вид військ (морська авіація, морська піхота і т.д.).  Таблиця особовий склад має наступні атрибути: прізвище, частина, посада, рік народження, рік надходження на службу, вислуга років, нагороди,  Таблиця кораблі має наступні атрибути: ідентифікаційний номер корабля, назва корабля, тип корабля, дата створення, напрацювання, кількість посадкових місць, пристрій двигуна (вітрильне, гребне, пароплав, теплохід, турбохід, і т.д. ), Тип приводу (самохідне, несамохідне), розміщення корпусу (підводний човен, пірнати,напівзаглибні, і т.д.)  Таблиця навчання: частина, корабель, дата, місце проведення, оцінка. |
| Варіант №15 | На підставі обраного варіанту виконати наступне:  1. Виконати аналіз предметної області досліджуваної області;  2. Описати основні сутності предметної області;  3. Розставити існуючі зв'язки між сутностями: самостійно додати в кожну сутність первинні ключі та встановити зовнішні ключі між сутностями;  4. Побудувати інфологічну модель бази даних;  5. Побудувати даталогічну модель бази даних.  БД - туристична фірма. БД складається з наступних таблиць: туристи, туристичная група, склад груп, готелі, відомості продаж.  Таблиця туристи має наступні атрибути: ПІБ, паспортні дані, стать, вік, діти.  Таблиця тури має наступні атрибути: назва, країна, міста, тип пересування, тип харчування, ціна туру, тип проживання.  Таблиця туристична група має такі атрибути: назва, дата відправлення, дата прибуття, тур, кількість туристів.  Таблиця складу груп має наступні атрибути: дата продажу, турист, група, ціна квитка.  Таблиця готелі має наступні атрибути: назва готелю, країна, місто, адреса, кількість місць, тип готелю.  Таблиця відомість продаж має наступні атрибути: дата, туристична група, готель, загальна вартість. |
| Варіант №16 | На підставі обраного варіанту виконати наступне:  1. Виконати аналіз предметної області досліджуваної області;  2. Описати основні сутності предметної області;  3. Розставити існуючі зв'язки між сутностями: самостійно додати в кожну сутність первинні ключі та встановити зовнішні ключі між сутностями;  4. Побудувати інфологічну модель бази даних;  5. Побудувати даталогічну модель бази даних.  БД - цирк. БД складається з наступних таблиць: працівники цирку,  вистави, розклад гастролей, трупа цирку, програма цирку.  Таблиця працівники цирку має наступні атрибути: прізвище, ім'я, батькові, рік народження, рік надходження на роботу, стаж, посада (Акробат, клоун, гімнаст, музикант, постановник, службовець і т.д.), стать, адреса, місто, телефон.  Таблиця вистава має наступні атрибути: назва, режисер-постановник, художник-постановник, диригент-постановник, автор, жанр, тип.  Таблиця розклад гастролей має наступні атрибути: вистава, дата початку, дата закінчення, місця проведення гастролі.  Таблиця трупа вистави цирку має наступні атрибути:вистава, актор цирку, назва ролі.  Таблиця програма цирку має наступні атрибути: вистава, дата прем'єри, період проведення, дні і час, ціна квитка. |
| Варіант №17 | На підставі обраного варіанту виконати наступне:  1. Виконати аналіз предметної області досліджуваної області;  2. Описати основні сутності предметної області;  3. Розставити існуючі зв'язки між сутностями: самостійно додати в кожну сутність первинні ключі та встановити зовнішні ключі між сутностями;  4. Побудувати інфологічну модель бази даних;  5. Побудувати даталогічну модель бази даних.  БД - аптека. БД складається з наступних таблиць: лікартства, покупці, продавці, рецепти, продаж ліків.  Таблиця ліки має наступні атрибути: назва, тип (готове, виготовляється), вид (таблетки, мазі, настоянки), ціна.  Таблиця покупці має наступні атрибути: прізвище, ім'я, батькові, адреса, місто, телефон.  Таблиця продавці має наступні атрибути: прізвище, ім'я, по батькові, дата надходження, дата народження, освіта.  Таблиця рецепти має наступні атрибути: номер рецепта, дата видачі, ПІБ хворого (покупець), ПІБ лікаря, діагноз пацієнта.  Таблиця продаж ліків має наступні атрибути: дата, ліки, кол-во, рецепт, продавець. |
| Варіант №18 | На підставі обраного варіанту виконати наступне:  1. Виконати аналіз предметної області досліджуваної області;  2. Описати основні сутності предметної області;  3. Розставити існуючі зв'язки між сутностями: самостійно додати в кожну сутність первинні ключі та встановити зовнішні ключі між сутностями;  4. Побудувати інфологічну модель бази даних;  5. Побудувати даталогічну модель бази даних.  БД - міська телефонна мережа. БД складається з наступних таблиць: АТС, абонент, відомість дзвінків, прайс АТС, відомість абонентської плати.  Таблиця АТС має наступні атрибути: назва АТС, вид (Міські, відомчі та установчі), адреса, місто, кількість абонентів.  Таблиця абоненти має наступні атрибути: прізвище, ім'я, по батькові, вид телефону (основний, паралельний або спарений), номер телефону, міжміські (відкритий / закритий), пільга (так / ні), адреса: індекс, район, вулиця, дім квартира.  Таблиця відомість дзвінків має наступні атрибути: абонент, дата дзвінка, час початку, час закінчення, міжміські (так / ні).  Таблиця прайс АТС має наступні атрибути: АТС, ціна на міські, ціна на міжмісто.  Таблиця відомість абонентської плати має такі атрибути: абонент, місяць, рік, кількість хвилин на міські, к-ть хвилин на міжмісто, вартість, сума пільги, загальна вартість. |
| Варіант №19 | На підставі обраного варіанту виконати наступне:  1. Виконати аналіз предметної області досліджуваної області;  2. Описати основні сутності предметної області;  3. Розставити існуючі зв'язки між сутностями: самостійно додати в кожну сутність первинні ключі та встановити зовнішні ключі між сутностями;  4. Побудувати інфологічну модель бази даних;  5. Побудувати даталогічну модель бази даних.  БД - аеропорт. БД складається з наступних таблиць: працівники аеропорту, розклад вильотів, літаки, бригади літаків, відомість продажів квитків.  Таблиця працівники аеропорту має наступні атрибути: прізвище, ім'я, по батькові, рік народження, рік прийняття на роботу, стаж, посада (Пілотів, диспетчерів, техніків, касирів, працівників служби безпеки, довідкової служби та інших,), стать, адреса, місто, телефон.  Таблиця розклад вильотів має наступні атрибути: літак, дата вильоту, час вильоту, місце вибуття, місце прибуття, маршрут (Початковий і кінцевий пункти призначення, пункт пересадки), вартість квитка.  Таблиця літаки має наступні атрибути: номер, рік випуску, кількість посадкових місце, вантажопідйомність.  Таблиця бригади літаків має наступні атрибути: номер бригади, літак, працівник аеропорту (пілоти, техніки і обслуговуючий персонал)  Таблиця відомість продажу квитків має наступні атрибути: дата і час продажу, ПІБ пасажира, паспортні дані, номер рейсу, кількість квитків, наявність пільг (пенсіонери, діти-сироти і т.д.), багаж (Так / ні), вартість. |
| Варіант №20 | На підставі обраного варіанту виконати наступне:  1. Виконати аналіз предметної області досліджуваної області;  2. Описати основні сутності предметної області;  3. Розставити існуючі зв'язки між сутностями: самостійно додати в кожну сутність первинні ключі та встановити зовнішні ключі між сутностями;  4. Побудувати інфологічну модель бази даних;  5. Побудувати даталогічну модель бази даних.  БД - театр. БД складається з наступних таблиць: працівники театру, спектаклі, розклад гастролей, трупа вистави, репертуар театру.  Таблиця працівники театру має наступні атрибути: прізвище, ім'я, батькові, рік народження, рік надходження на роботу, стаж, посада (Акторів, музикантів, постановників і службовців), стать, адреса, місто, телефон.  Таблиця спектаклі має наступні атрибути: назва, режисер постановик, художник-постановник, диригент-постановник, автор, жанр (музична комедія, трагедія, оперета тощо), тип (дитячі, молодіжні та ін.).  Таблиця розклад гастролей має наступні атрибути: назва, дата початку, дата закінчення, місця проведення гастролі, спектакль.  Таблиця трупа вистави має наступні атрибути: спектакль, актор, назва ролі.  Таблиця репертуар театру має наступні атрибути: спектакль, дата прем'єри, період проведення, дні і час, ціна квитка. |
| Варіант №21 | На підставі обраного варіанту виконати наступне:  1. Виконати аналіз предметної області досліджуваної області;  2. Описати основні сутності предметної області;  3. Розставити існуючі зв'язки між сутностями: самостійно додати в кожну сутність первинні ключі та встановити зовнішні ключі між сутностями;  4. Побудувати інфологічну модель бази даних;  5. Побудувати даталогічну модель бази даних.  БД - залізничний вокзал. БД складається з наступних таблиць: працівники ж.д.вокзала, розклад руху поїздів, поїзди, бригади поїздів, відомість продажу квитків.  Таблиця працівники ж.д.вокзала має наступні атрибути: прізвище, ім'я, по батькові, рік народження, рік надходження на роботу, стаж, посада (машиніст, диспетчерів, провідник, ремонтників рухомого складу, шляхів, касирів, працівників служби підготовки складів, довідкової служби та інших,), стать, адреса, місто, телефон.  Таблиця розклад руху поїздів має наступні атрибути: поїзд, дата відправлення, час відправлення, місце відправлення, дата прибуття, час прибуття, місце прибуття, маршрут ((початковий і кінцевий пункти призначення, основні вузлові станції), вартість квитка.  Таблиця поїзда має наступні атрибути: номер, рік випуску, кількість вагонів, тип поїзда (загальний, швидкісний, високошвидкісний).  Таблиця бригади поїздів має наступні атрибути: номер бригади, поїзд, працівник ж.д.вокзала (машиністи, техніки, провідники і обслуговуючий персонал).  Таблиця відомість продажу квитків має наступні атрибути: дата і час продажу, ПІБ пасажира, паспортні дані, номер рейсу, кількість квитків, наявність пільг (пенсіонери, діти-сироти і т.д.), вартість. |
| Варіант №22 | На підставі обраного варіанту виконати наступне:  1. Виконати аналіз предметної області досліджуваної області;  2. Описати основні сутності предметної області;  3. Розставити існуючі зв'язки між сутностями: самостійно додати в кожну сутність первинні ключі та встановити зовнішні ключі між сутностями;  4. Побудувати інфологічну модель бази даних;  5. Побудувати даталогічну модель бази даних.  БД - інформаційна система ВНЗ. БД складається з наступних таблиць: факультети, кафедри, викладачі, дисципліни, навчальне навантаження.  Таблиця факультети має наступні атрибути: назва факультету, ПІБ декана, номер кімнати, номер корпусу, телефон.  Таблиця кафедра має наступні атрибути: назва кафедри, ПІБ завідувача, номер кімнати, номер корпусу, телефон, кількість викладачів.  Таблиця дисципліни має такі атрибути: назва дисципліни, кількість годин, цикл дисциплін.  Таблиця викладачі має наступні атрибути: прізвище, ім'я, батькові, кафедра, рік народження, рік надходження на роботу, стаж, посаду, стать, місто.  Таблиця навчальне навантаження має наступні атрибути: викладач, дисципліна, навчальний рік, семестр, групи, кількість студентів, вид підсумкового контролю. |

**Завдання №2.**

1. Ознайомитись з основними принципами створення бази даних в MS SQL Server. Вивчити операції, що проводяться з базами даних в цілому. Отримати навички використання програми "SQL Server Management Studio" для створення, видалення, реєстрації, підключення, вилучення метаданих, резервного копіювання та відновлення бази даних. Вивчити SQL-оператори для створення, підключення та видалення бази даних. Ознайомитись з основними принципами управління обліковими записами і ролями.

2. Студент отримує індивідуальний варіант вихідних даних з коротким описом предметної області, який використовується при виконанні всіх описаних в даному мануалі практичних робіт. При цьому кожне чергове практичне завдання є продовженням виконаного раніше і тому вони повинні обов’язково виконуватися послідовно.

3. Використовувані програми.

1. Працюючий на комп'ютері сервер "MS SQL Server ".

2. Встановлена платформа .NET Framework 2.

3. Операційна система Microsoft Windows 7 / Windows 8 / Windows 10.

4. Додаток "SQL Server Management Studio ", встановлений на локальному комп'ютері.

4. Завдання.

Практичну роботу слід виконувати в наступному порядку:

1. Створити на сервері pi\_srv (або на локальному комп'ютері, якщо немає сервера) робочу папку для зберігання файлів, одержуваних при виконанні практичної роботи. Ця папка повинна розташовуватися в папці \ Бази даних \ Група \ Студент та відповідати номеру виконуваної практичної роботи.

2. На підставі індивідуального завдання вибрати ім'я файлу створюваної бази даних. Для імені найкраще вибрати одне або кілька англійських слів, відповідних найменуванню предметної області. Використання для імені українських слів, записаних латинськими буквами, не допускається.

3. Відкриття програми "SQL Server Management Studio". Для цього можна або скористатися меню Пуск (Пуск / Програми / SQL Server Management Studio).

4. Створити з'єднання з локальним або віддаленим сервером.

5. Створити базу даних для своєї предметної області за допомогою діалогу, вибравши сервер "pi\_srv" або локальний сервер "ім'я комп'ютера \ SQLEXPRESS"

6. Створити базу даних і вказати в якості імені файлу "\ Бази даних \ Група \ ПІП\_студента \ Назва\_БД ".

7. Витягти метадані для автоматичної генерації команди створення бази даних.

8. Видалити базу даних, виконавши команду "Database / Drop Database" (База даних / Видалити базу даних).

9. Створити базу даних другим способом, виконавши у вікні "Script Executive "оператори, отримані при добуванні метаданих перед попереднім видаленням.

10. Створити резервну копію бази даних.

11. Видалити базу даних.

12. Відновити базу даних з резервної копії.

13. Зберегти файл сценарію на сервері в папці "Студент", давши йому ім'я «Lab.№1» і стандартне розширення "\* .sql".

**Завдання №3.**

1. Створити файл бази даних, згідно з номером варіанту, виданого в практичній роботі №1 за допомогою sql-команди.
2. Створити резервну копію бази даних.
3. Визначити 2-3 посадові особи, які зможуть працювати з таблицями БД. Для кожної посадової особи визначити набір привілеїв, якими він може користуватися.
4. В утиліті SQL Server Management Studio створити під кожну посадову особа відповідну роль, наділити цю роль певними привілеями. Далі створити по одному користувачеві на кожну посаду і присвоїти їм відповідні ролі.
5. Зберегти послідовно SQL-оператори із зазначенням завдань у файлі з назвою SurnameStudent\_Lab\_2.
6. Створити текстовий звіт, в якому відобразити sql-команди розроблених запитів і скріншоти результатів роботи з СУБД SQL Server Management Studio.

**Завдання №4.**

Для раніше створеної бази даних за номером варіанта (створеної в попередніх завданнях) в СУБД SQL Server Management Studio:

1. Створіть всі таблиці бази даних, ключі, обмеження і зв'язки.

2. Кожна таблиця повинна мати обмеження первинного ключа.

3. За допомогою обмежень зовнішнього ключа повинні бути задані всі наші зв'язки між таблицями.

4. В залежності від умов виданого завдання в деяких таблицях можуть бути накладені додаткові обмеження цілісності на стовпці або повинні бути розроблені обчислювані поля.

5. Створіть діаграму бази даних.

6. Заповніть таблиці даними не менше 5 записами в кожній.

7. Створити текстовий звіт, в якому відобразити скріншоти результатів роботи в СУБД SQL Server Management Studio (вікно з базою даних з переліком всіх таблиць, проекти таблиць з переліком стовпців, вікна обмежень зовнішніх ключів (створення), вікно з переліком ключів для кожної таблиці, вікна з даними для кожної таблиці, діаграма бази даних).

**Завдання №5.**

Створити на мові Transact-SQL файл бази даних відповідно до номера варіанта (Присвоїти їй нове ім'я, неспівпадаючі з ім'ям бази даних створеної в завданні 4).

База даних розробляється на основі спроектованої концептуальної моделі даних в завданні №1.

Створити програмно мовою SQL всі таблиці, із зазначенням первинних і зовнішніх ключів і обмеження цілісності. Всі програмні інструкції команд SQL зберігати в файлах з розширенням \* .sql в папці SurnameStudent/Lab1.

Заповнити таблиці даними по 5 записів в кожній. Створити текстовий звіт, в якому відобразити sql-команди розроблених запитів і скріншоти результатів роботи з СУБД SQL Server Management Studio.

**Завдання №6. Створення запитів на вибірку даних.**

Вивчити використовуваний в реляційних СУБД оператор вибору даних з таблиць. Отримати навички роботи з оператором SELECT в програмі 'SQL Server Managmant Studio '.

Початкові дані. Вихідними даними є індивідуальне завдання і результат попередніх практичних робіт.

Використовувані програми. Програма 'SQL Server Managmant Studio'.

Для створеної бази даних, згідно з номером варіанту, самостійно створити на мові Transact-SQL **15 запитів** з вибіркою рядків за умовою:

- 3 найпростіших запит з використанням операторів порівняння;

- 3 запити з використанням логічних операторів AND, OR і NOT;

- 1 запит на використання комбінації логічних операторів;

- 1 запит на використання виразів над стовпцями;

- 2 запити з перевіркою на приналежність множині;

- 2 запити з перевіркою на приналежність діапазону значень;

- 2 запити з перевіркою на відповідність шаблону;

- 1 запит з перевіркою на невизначене значення.

Всі програмні інструкції команд SQL зберігати в файлах з розширенням

\* .sql в папці SurnameStudent/Lab1.

Для кожного запиту сформулювати текстове завдання, яке повинно бути

виконано до бази даних. Створити текстовий звіт, в якому відображено sql-команди розроблених запитів і скріншоти результатів роботи з СУБД SQL Server Management Studio.

**Завдання №7. Створення багатотабличних запитів на вибірку даних.**

Вивчити використовуваний в реляційних СУБД оператор вибірки даних з таблиць. Отримати навички роботи з оператором SELECT в програмі 'SQL Server Managmant Studio '.

Початкові дані. Вихідними даними є індивідуальне завдання і результат попередніх практичних робіт.

Використовувані програми. Програми 'SQL Server Managmant Studio'.

Для створеної бази даних, згідно з номером варіанту, самостійно створити на мовою Transact-SQL **15 багатотабличних запитів**:

- 1 запит з використанням декартового добутку двох таблиць;

- 3 запити з використанням з'єднання двох таблиць по рівності;

- 1 запит з використанням з'єднання двох таблиць по рівності та умовою відбору;

- 1 запит з використанням з'єднання за трьома таблицями;

- створити копії раніше створених запитів на з'єднання з рівності на запити з використанням зовнішнього повного з'єднання таблиць (JOIN).

- 1 запит з використанням лівого зовнішнього з'єднання;

- 1 запит на використання правого зовнішнього з'єднання;

- 1 запит з використанням симетричного з'єднання і видалення надмірності.

Всі програмні інструкції команд SQL зберігати в файлах з розширенням \* .sql в папці SurnameStudent/Lab1.

Для кожного запиту сформулювати текстове завдання, яке повинно бути виконано до бази даних.

Створити текстовий звіт, в якому відобразити sql-команди розроблених запитів і скріншоти результатів роботи з СУБД SQL Server Management Studio.

**Завдання №8. Створення запитів на групування та сортування даних. Запити на зміну. Використання вбудованих функцій.**

Вивчити використовуваний в реляційних СУБД оператор вибірки даних з таблиць SELECT і виконання угруповання і сортування даних. Вивчити синтаксис мови модифікації даних. Навчиться використовувати вбудовані функції в запитах.

Початкові дані. Вихідними даними є індивідуальне завдання і результат попередніх практичних робіт.

Використовувані програми. Програма SQL Server Managmant Studio.

Для створеної бази даних, згідно з номером варіанту, самостійно створити на мовою Transact-SQL **14 багатотабличних запитів**:

- 1 запит з використанням функції COUNT;

- 1 запит з використанням функції SUM;

- 1 запит з використанням функцій UPPER, LOWER;

- 1 запит з використанням тимчасових функцій;

- 1 запит з використанням групування по одніомустовпцю;

- 1 запит на використання групування за кількома стовпцями;

- 1 запит з використанням умови відбору груп HAVING;

- 1 запит з використанням фрази HAVING без фрази GROUP BY;

- 1 запит з використанням сортування по стовпцю;

- 1 запит на додавання нових даних в таблицю;

- 1 запит на додавання нових даних за результатами запиту в якості вставляючого значення;

- 1 запит на оновлення існуючих даних в таблиці;

- 1 запит на оновлення існуючих даних за результатами підзапиту у фразі WHERE;

- 1 запит на видалення існуючих даних.

Всі програмні інструкції команд SQL зберігати в файлах з розширенням \*.sql в папці SurnameStudent/Lab1.

Для кожного запиту сформулювати текстове завдання, яке повинно бути виконано до бази даних.

Створити текстовий звіт, в якому відобразити sql-команди розроблених запитів і скріншоти результатів роботи з СУБД SQL Server Management Studio.

**Завдання №9.** Вивчення призначення представлень баз даних, синтаксису і семантики команд мови Transact-SQL для їх створення, зміни і видалення, системних збережених процедур для отримання інформації про представленн, а також придбання навичок їх створення за допомогою графічних засобів утиліти Enterprise Manager і майстра Create View Wizard.

Завдання 9.1. Створити представлення my, що посилається на основну таблицю за вибором бази згілно з попереднього варіанту, яке містить ідентифікаційний номер my \_lname і телефон phone, при цьому відобразити тільки користувачів за певним критерієм.

Завдання 9.2. Створити представлення report, яке посилається на подання my і дві довільні таблиці бази і в якому виводяться певна назва.

Завдання 9.3. Створити представлення my, розглянуте в першому завданні, за допомогою графічних засобів утиліти Enterprise Manager.

Завдання 9.4. Створити уявлення report, розглянуте в другому завданні, за допомогою майстра Create View Wizard.

Завдання 9.5. Зіставити запити, отримані автоматично в завданнях 3і 4, з запитами відповідно в першому і другому завданнях. Модифікувати запити з допомогою команди ALTER VIEW і отримати довідкову інформацію про ціпредставлення за допомогою процедур sp\_help, sp\_helptexst і sp\_depends.

**Завдання №10.** Вивчити використовуваний в реляційних СУБД вбудовану мову програмування Transact-SQL для написання програм в MS SQL Server. Вивчити правила побудови ідентифікаторів, правила оголошення змінних і їх типів. Вивчити принципи роботи з циклами. Вивчити роботу зі змінними типу Table. Вивчити синтаксис і семантику функцій і процедур, що зберігаються Transact-SQL: способів їх ідентифікації, методів завдання і специфікації параметрів і значень, що повертаються і викликів функцій і процедур, що зберігаються.

Початкові дані. Вихідними даними є індивідуальне завдання і результат попередніх практичних робіт.

Використовувані програми. Програма Microsoft SQL Server Managеmant Studio.

Практичну роботу слід виконувати в наступному порядку:

1. Знайомство з правилами позначення синтаксису команд в довідковій системі MS SQL Server (утиліта Books Online).

2. Вивчення правил написання програм на Transact SQL.

3. Вивчення правил побудови ідентифікаторів, правил оголошення змінних і їх типів.

4. Вивчення роботи з циклами і умовними операторами.

5. Вивчення роботи зі змінними типу Table.

6. Вивчення правил написання збережених процедур і функцій.

7. Опрацювання всіх прикладів, аналіз результатів їх виконання.

8. Виконання індивідуальних завдань за варіантами.

Для створеної бази даних, згідно з номером варіанту, самостійно створити на мовою P-SQL **10 запитів**:

- 1 запит для створення тимчасової таблиці через змінну типу TABLE;

- 1 запит з використанням умовної конструкції IF;

- 1 запит з використанням циклу WHILE;

- 1 запит для створення скалярної функції;

- 1 запит для створення функції, яка повертає табличне значення;

- 1 запит для створення процедури без параметрів;

- 1 запит для створення процедури c вхідним параметром;

- 1 запит для створення процедури c вхідними параметрами і RETURN;

- 1 запит для створення процедури поновлення даних в таблиці бази даних UPDATE;

- 1 запит для створення процедури вилучення даних з таблиць бази даних SELECT;

Всі програмні інструкції команд SQL зберігати в файлах з розширенням

\*.sql в папці SurnameStudent/Lab1.

Для кожного запиту сформулювати текстове завдання, яке повинно бути виконано до бази даних.

Створити текстовий звіт, в якому відобразити sql-команди розроблених запитів і скріншоти результатів роботи з СУБД SQL Server Management Studio.

**Завдання №11.** Вивчення синтаксису та семантики функцій і процедур, що зберігаються Transact- SQL: способів їх ідентифікації, методів завдання і специфікації параметрів і значень, що повертаються, кодування тіла і викликів функцій і процедур, що зберігаються, застосування команд для створення, зміни та видалення системних і призначених для користувача як скалярних, так і табличних (з одного Inline або декількома Multi – statement командами в тілі) функцій, системних, призначених для користувача, тимчасових (локальних або глобальних) і розширених збережених процедур, а також придбання навичок програмування, налагодження, тестування і включення в групу або підключення бібліотеки функцій і процедур, що зберігаються.

Завдання 11.1. Створити функцію для виконання чотирьох арифметичних операцій "+", "-", "\*" і "/" над цілими операндами типу bigint, виконавши кодування і перевірку:

1. Кодування

CREATE FUNCTION Calculator

(@ Opd1 bigint,

@ Opd2 bigint,

@ Oprt char (1) = "\*")

RETURNS bigint

AS

BEGINT

DECLARE @ Result bigint

SET @ Result =

CASE @ Oprt

WHEN "+" THEN @ Opd1 + @ Opd2

WHEN "-" THEN @ Opd1 - @ Opd2

WHEN "\*" THEN @ Opd1 \* @ Opd2

WHEN "/" THEN @ Opd1 / @ Opd2

ELSE 0

END

Return @ Result

END

2. Тестування

SELECT dbo.Calculator (4,5, '+'),

dbo. Calculator (3,7, '\*') - dbo.Calculator (64,4, '/') \* 2.

9 -11

(1 row (s) affected)

Завдання 11.2. Створити функцію, яка повертає таблицю з динамічним набором стовпців, виконавши кодування і тестування:

1. Кодування

CREATE FUNCTION DYNTAB (@ State char (2))

RETURNS Table

AS

RETURNS SELECT au\_id, au\_lname, au\_fname FROM authors

WHERE state = @ state

2. Перевірка

SELECT \* FROM DYNTAB ( "CA")

ORDER BY au\_lname, au\_fname

au\_id au\_lname au\_fname

Завдання 11.3. Створити функцію, розбиває вхідні рядок на підрядки, використовуючи як роздільник пробіли, виконавши кодування і тестування:

1. Кодування

CREATE FUNCTION Parse (@ String nvarchar (500))

RETURNS @ tabl TABLE

(Number int IDENTITY (1,1) NOT NULL,

Substr nvarchar (30))

AS

BEGIN

DECLARE @ Str1 nvarchar (500), @ Pos int

SET @ Str1 = @String

WHILE 1> 0

BEGIN

SET @ Pos = CHARINDEX ( "", @ Str1)

IF @POS> 0

BEGIN

INSERT INTO @ tab1

VALUES (SUBSTRING (@ Str1,1, @ Pos))

END

ELSE

BEGIN

INSERT INTO @ tab1e

VALUES (@ Str1)

BREAK

END

END

RETURN

END

2. Тестування

DECLARE @ Test String nvchar (500) 175

Set @ TestString = ' "SQL Server 2000"

SELECT \* FROM Parse (@ Test String)

-------------------------------------------------

Number Substr

-------------------------------------------------

1 SQL

2 Server

3 2000

(3 row (s)) affected)

Завдання 11.4. Створити зазначені три функції за допомогою утиліти Enterprise Manager і перевірити їх синтаксис, натиснувши кнопку Check Syntax (перевірити синтаксис), потім зберегти їх як шаблон, натиснувши кнопку Save as Template (Зберегти як шаблон).

Завдання 11.5. Створити і відредагувати функцію, використовуючи шаблон, отриманий в завданні №4.

Завдання 11.6. Вибрати новостворену функцію і, двічі клацнувши по ній, відкрити вікно редагування, а потім відредагувати текст цієї функції і права доступу користувачів і ролей до даної функції.

Завдання 11.7. Використовуючи утиліту Enterprise Manager, ознайомитися текстами вихідних модулів системних функцій (якщо це можливо), їх семантикою та способами виклику в вирази, відображаючи при цьому результати їх роботи.

Завдання 11.8. За допомогою Transact-SQL створити три збережені процедури, семантика яких аналогічна розглянутим функціям, ввівши при цьому додатковий параметр для отримання результату роботи відповідної процедури.

Завдання 11.9. Відредагувати зазначені процедури, за допомогою Enterprise Manager.

Завдання 11.10. Створити ці ж процедури з допомогою майстра Create Stored Procedure Wizard і проаналізувати їх властивості. Протестувати новостворені збережені процедури. За допомогою sp\_help отримати довідкові дані по одній з цих процедур.

**Завдання №12.** Вивчення призначення і типів тригерів, умов їх активації, синтаксису і семантики команд мови Transact - SQL для їх створення, модифікації, перейменування, програмування і видалення, а також отримання навичок їх проектування, кодування і налагодження із застосуванням процедур для отримання інформації по триггерах бази даних.

Завдання 12.1. Створити таблицю mytable з таблиці власної бази даних і для нової таблиці запрограмувати тригер my\_del, який буде виводити інформацію про спроби видалення і кількості видалених рядків.

Завдання 12.2. Створити тригер для іншої таблиці, побудований в першому завданні, який буде дозволяти зміну стовпця my\_id цієї таблиці всім, крім власника dbo.

Завдання 12.3. Створити тригер для команд INSERT і UPDATE, що забороняє виробляти зміни для певного користувача.

**Завдання №13.** Вивчення способів забезпечення надійної роботи SQL Server за допомогою механізму транзакцій і контрольних точок, набуття навичок управління локальними і розподіленими транзакціями різних видів, а також ознайомлення з фізичною і логічною архітектурою журналу транзакцій і способами відновлення баз даних.

Завдання 13.1. Перевірити режими автоматичного початку транзакцій і неявного початку транзакцій, використовуючи перемикач IMPLICIT\_TRANSACTION і команду COMMIT.

Завдання 13.2. Створити декількома способами розподілені транзакції і переконатися в коректності їх виконання.

Завдання 13.3. Створити вкладені транзакції, виконавши такі команди:

CREATE TABLE #aaa (cola int) - 0-й рівень

BEGIN TRAN - 1-й рівень

INSERT INTO #aaaVALUES (111)

BEGIN TRAN - 2-й рівень

INSERT INTO #aaaVALUES (222)

BEGIN TRAN - 3-й рівень

INSERT INTO #aaaVALUES (333)

SELECT \* FROM #aaa

SELECT 'Вкладеність транзкцій', @@ TRANCOUNT

ROLLBACK TRAN

SELECT \* FROM #aaa - відкат на 0-й рівень

SELECT 'Вкладеність транзакцій', @@ TRANCOUNT

Проаналізувати отримані результати.

Завдання 13.4. Написати приклад пакету запитів з використанням команд COMMIT і ROLLBACK для автоматичних, неявних і явних транзакцій.

Завдання 13.5. Написати приклад пакета команд, що ілюструють використання засобів оптимізації при відкаті транзакцій.

Завдання 13.6. Використовуючи засоби MS SQL Server, вивчити фізичну і логічну архітектуру журналу транзакцій.

Завдання 13.7. За допомогою системної збереженої процедури sp\_configure змінити інтервал контрольних точок для бази даних.

Завдання 13.8. Уточнити синтаксис команд управління транзакціями і написати приклад пакета з використанням всіх варіантів цих команд.

**Завдання №14.**

Вивчення призначення і типів курсорів, синтаксису і семантики команд мови Transact - SQL для створення і відкриття курсорів, вибірки даних з курсора і зміни рядків таблиць за допомогою курсорів, видалення даних, закриття та звільнення курсорів, а також придбання навичок їх застосування та управління за допомогою команд і системних збережених процедур SQL Server.

Завдання 14.1. Створити курсор довільного типу та призначення, показати можливості його використання.

**Завдання №15.** Вивчення загальних правил розмежування і надання прав доступу користувачам баз даних, архітектури та компонент системи безпеки SQL Server і режимів аутентифікації користувачів, а також придбання навичок адміністрування системи безпеки: створення та управління обліковими записами, управління користувачами, ролями і групами.

Завдання 15.1. Створити обліковий запис SQL сервера, використовуючи графічну утиліту Enterprise Manager, виконавши такі дії:

1. Вибрати потрібний сервер.

2. Відкрити папку Security цього сервера.

3. Вибрати об'єкт Logins, клацнувши по відповідному значку.

4. У правому вікні переглянути список облікових записів даного сервера:

Name - ім'я облікового запису сервера;

Type - походження облікового запису:

User W- користувач Windows;

Group W- група користувачів Windows;

Standard - користувач SQL сервера;

Server Access - доступ до сервера SQL:

Permit - дозволено;

Deny - заборонений;

Default Database - база даних за замовчуванням, до якої підключений користувач (обов'язковий параметр)

User - ім'я користувача бази даних;

Default Language - мова за замовчуванням для цього облікового запису.

5. Для створення нового облікового запису сервера відкрити контекстне меню об'єкта Logins, клацнувши по ньому правою клавішею миші або по значку на панелі інструментів лівою клавішею миші.

6. У діалоговому вікні на вкладці General (загальні) ввести ім'я облікового записи в поле Name.

7. Вибрати тип аутентифікації: Windows Authentication або SQL Server Authentication.

8. Якщо обрана аутентифікація Windows, то задати в поле Domain ім'я домену, в якому враховано користувач або група Windows. Ім'я заданого домену повинно додатися попереду імені користувача також і в поле Name (для вибору домену використовувати кнопку "...").

9. У групі Security Access (безпечний доступ) встановити перемикач Grant Access (доступ дозволений). Установка перемикача Deny Access назавжди заборонить \ реєстрацію користувача або домену Windows.

10. Якщо вибрана аутентифікація SQL Server, то задати тільки пароль для облікового запису.

11. Задавши параметри аутентифікації Windows або SQL Server, вказати в групі Defaults (умовчання) ім'я бази даних в поле Database, до якої користувач буде підключатися автоматично, і мова Language. Якщо ім'я бази даних не задати, то сервер буде автоматично підключати до бази master.

12. Включити створювану обліковий запис в необхідну вбудовану роль сервера: Sysadmin, Serveradmin, Setupadmin, Securityadmin, Processadmin, Dbereator, Diskadmin, Bulkadmin, встановивши прапорець проти цієї ролі на вкладці Server Role.

13. На вкладці Database Access вибрати необхідну базу даних, встановивши прапорець зліва від її імені, і задати ім'я користувача, в яке буде відображатися розглянутий обліковий запис, а в нижній частині вкладки за допомогою прапорця включити користувача в ту чи іншу роль в залежності від його роботи з базою даних.

14. Натиснувши на кнопці Properties (властивості) і переглянути список користувачів, включених в обрану роль даної бази даних.

15. Натиснувши на кнопці Permissions (права) і переглянути список прав, наданих обраній ролі бази даних.

16. Закрити всі вікна.

17. Знову відкрити список облікових записів сервера, двічі клацнути по знову створеному запису і перевірити правильність введених параметрів.

18. Закрити всі вікна.

19. Розпочати роботу з базами даних, використовуючи новий обліковий запис.

Завдання 15.2. Включити обліковий запис користувача або групи користувачів Windows в фіксовану роль сервера SQL за допомогою Enterprise Manager, виконавши такі дії:

1 Вибрати необхідний сервер в лівому вікні Tree.

2. Відкрити об'єкти сервера, клацнувши по його кнопці "+".

3. Відкрити об'єкти Security цього сервера, натиснувши кнопку "+" для Security.

4. Вибрати об'єкт Logins (записи) і клацнути по ньому, при цьому в правому вікні відкриється список облікових записів сервера.

5. Двічі клацнути по необхідному обліковому запису сервера.

6. У вікні на Server Login Properties відкрити вкладку Server role.

7. Встановити прапорці біля тих ролей сервера, в які потрібно включити конфігуровані записи.

8. Закрити всі відкриті вікна, клацаючи по кнопках "OK" цих вікон.

9. Повторити завдання, використовуючи наступний набір команд:

а) Security / Server Rolees;

б) Клацнути лівою клавішею;

в) У правому вікні вибрати необхідну фіксовану роль;

г) Два рази клацнути по обраній ролі;

д) У вікні Server Role Properties клацнути по кнопці Add вкладки General;

е) Додати облікові записи в задану роль;

ж) Закрити вікно зі списком облікових записів;

з) На вкладці Permission вікна Server Role Properties переглянути надані права для даної ролі.

10. Закрити всі відкриті діалогові вікна, клацаючи по кнопках ОК.

11. Перевірити правильність виконаних дій.

Завдання 15.3. Створити нового користувача бази даних для облікового запису Windows за допомогою Enterprise Manager, виконавши такі дії:

1. Вибрати необхідний сервер і необхідну базу даних в лівому вікні Tree.

2. Відкрити об'єкти обраної бази даних, клацнувши по значку "+" цієї бази.

3. Вибрати в списку, що розкрився об'єктів розглянутої бази даних об'єкт Users (користувачі).

4. Клацнути правою клавішею миші і відкрити контекстне меню об'єкта Users (Користувачі).

5. У контекстному меню виконати команду New Database User (новий користувач бази даних).

6. У діалоговому вікні ввести:

а) в поле Login Name - ім'я облікового запису користувача або групи користувачів Windows;

б) в поле User Name - ім'я нового користувача даної бази даних.

7. Включити нового користувача в необхідні ролі бази даних: public, db - owner, db - denydatareader і т.д. Для цього необхідні ролі треба відзначити прапорцями в списку фіксованих ролей бази даних, розташованому в правій частині вікна.

8. Клацнути по кнопці Properties і, переглянувши список всіх користувачів бази даних, переконатися, що новий користувач включений цей список.

9. Клацнути по кнопці Permission і задати права доступу користувача до об'єктів бази даних: SELECT, INSERT, UPDATE, DELET, EXEC, DRI. У вікні знаходиться повний список об'єктів бази даних.

10. Клацнути по кнопці Columns (стовпці) для обраної бази даних і задати права доступу до конкретних стовпчиків таблиці: SELECT і / або UPDATE.

11. Закрити всі відкриті діалогові вікна, клацаючи по кнопках ОК.

12. Перевірити роботу нового користувача з розглянутою базою даних і його права.

Завдання 15.4. Створити обліковий запис SQL сервера, використовуючи майстер Create Login Wizard, виконавши такі дії:

1. Виконати команду в Enterprise Manager Run an Wizard / Create Login Wizard.

2. У вікні, майстра ознайомитися з етапами створення облікового запису сервера:

а) Select an authentication mode - вибрати режим аутентифікації;

б) Grant access to security roles – представити доступ секретним ролям;

в) Grant access to databases - надати доступ до баз даних.

3. Клацнути по кнопці Next.

4. Вибрати режим аутентифікації: Windows або SQL Server.

5. Клацнути по кнопці Next.

6. Якщо Ви вибрали аутентифікації Windows, то у вікні, в полі Windows account задати ім'я облікового запису або групи Windows і домена і вказати тип доступу: Grant access to the server (дозволити доступ до нього) або Deny access to the server (Заборонено доступ до сервера).

7. Якщо Ви вибрали аутентифікації SQL Server, то крім імені облікового записи, що задається в полі Login I вказати пароль в поле Password (пароль) і в полі Confirm Password (підтвердити пароль). Цей пароль користувач буде використовувати при підключенні до SQL сервера.

8. Клацнути по кнопці Next в тому або в іншому випадку.

9. Включити обліковий запис в необхідні фіксовані ролі сервера, встановлюючи проти ролі прапорець.

10. Клацнути по кнопці Next.

11. Дозволити для облікового запису доступ до баз даних, відзначаючи їх прапорцем.

12. Клацнути по кнопці Next.

13. Майстер автоматично створить імена користувачів баз даних.

14. Перевірити зроблені установки.

Завдання 15.5. Створити нову призначену для користувача роль баз даних за допомогою Enterprise Manager, виконавши такі дії:

1. Вибрати необхідну базу даних.

2. Відкрити об'єкти обраної бази даних, клацнувши по значку "+" цієї бази.

3. Вибрати в списку, об'єкт Role (роль).

4. Відкрити контекстне меню об'єкта Role (роль).

5. Виконати команду меню New Database Role (нова роль бази даних).

6. У діалоговому вікні ввести ім'я ролі в поле Name, яке повинно бути унікальним в межах бази даних.

7. Вибрати тип ролі: стандартна роль Standart role або роль додатка Application Role.

8. Якщо обрана стандартна роль, то за допомогою кнопки "додати" Add включити в неї потрібних користувачів.

9. Якщо обрана роль додатка, то ввести в поле Password пароль, який буде використовуватися для ініціалізації даної ролі додатку. Не можна додавати користувачів в роль додатка.

10. Для створеної стандартної ролі або ролі додатка (для користувальницької ролі) задати права доступу до об'єктів бази даних на вкладці Permissions (права), виконавши дії:

а) виділити черговий об'єкт бази даних;

б) для кожного права: SELECT, INSERT, UPDATE, DELETE, EXEC - встановити одне з трьох станів доступу:

V - GRANT - надати,

X - DENI - заборонити,

- REVOKE - неявне відхилення (тобто може мати доступ через членство ролі).

Закрити всі вікна вікна, клацаючи по кнопці "OK" кожного вікна. Перевірте правильність виконаних дій.

**Завдання №16.**  Придбання навичок реєстрації віддалених серверів за допомогою утиліти Enterprise Manager, майстри Register Server Wizard, а також команд Transact-SQL і системної збереженої процедури sp-addserver. Придбання навичок управління основною службою MSSQLServer і допоміжними службами сервера MS SQL Server: завдання режиму автоматичного запуску служби, ручний запуск служби, запуск сервера в режимі одного, з мінімальними вимогами і нестандартної конфігурації, припинення служби та зупинка служб і сервера.

Завдання 16.1. Провести реєстрацію віддаленого сервера за допомогою вікна параметрів реєстрації сервера Register SQL Server Properties утиліти Enterprise Manager, виконавши дії:

1. На дереві об'єктів консолі вибрати одну з груп серверів, де буде зареєстрований віддалений сервер.

2. Відкрити контекстне меню групи серверів і виконати команду New SQL Server Registration.

3. У вікні Register SQL Server Properties задати наступні параметри:

a) Ім'я віддаленого сервера у вигляді такої записи: мережеве ім'я NetBios відповідного комп'ютера, коса риска «\», ім'я копії сервера (для сервера за замовчуванням це ім'я копії можна не ставити);

b) Облікову запис, яка буде використовуватися для встановлення з'єднання з відповідним сервером: або обліковий запис домена Windows NT і її набір прав в SQL Server, або обліковий запис сервера, створена на реєстроване сервера і що включає вхідне ім'я користувача Login Name і його пароль Password, із зазначенням режиму підключення з введенням пароля при кожному з'єднанні або без введення пароля;

c) Ім'я групи серверів з числа наявних або ім'я нової групи, яку можна створити, натиснувши кнопку з трьома крапками в тому ж вікні в області Options;

d) Установіть, якщо необхідно, наступні перемикачі: Display SQL Server state in console - показувати стан сервера у вікні об'єктів Enterprise Manager;

Automatically start SQL Server when connecting - автоматично запускати сервер при з'єднанні; Show system database and system table - відображати системні бази даних і таблиць.

Завдання 16.2. Провести реєстрацію віддаленого сервера за допомогою майстра Register Server Wizard, виконавши такі дії:

1. Клацнути на кнопці Run a Wizard панелі інструментів Enterprise Manager.

2. У вікні, вид якого залежить від лівого вікна Enterprise Manager (Обраний або не обрана сервер або папка групи серверів), вибрати майстер реєстрації сервера Register Server Wizard.

3. Клацнути по кнопці ОК.

4. У першому вікні майстра пропонується наступний порядок роботи:

a) вибрати SQL сервер;

b) вибрати режим аутентифікації;

c) визначити групу SQL серверів.

5. У цьому ж вікні скинути прапорець From now on, I want to perform this task without using a wizard (тепер я хотів би виконати завдання без використання майстра), інакше майстер закінчить свою роботу, відкривши вікно Register SQL Server Properties для ручної реєстрації сервера

6. Клацнути по кнопці Next.

7. У другому вікні вибрати або ввести ім'я реєстрованого сервера в лівій частині вікна.

8. Клацнути по кнопці додати Add. У разі помилки використовувати кнопку видалити Remove. Якщо має бути зареєстрована зразу кілька серверів, то вони будуть включені в одну і ту ж групу з однаковими параметрами і з однієї і тієї ж обліковим записом для встановлення з'єднання.

9. Клацнути по кнопці Next.

10. У третьому вікні необхідно вибрати режим аутентифікації при підключенні до сервера.

11. Якщо використовувати аутентифікацію Windows NT, то можливість підключення до сервера буде залежати від облікового запису, під якою працює SQL Server.

12. Якщо вибрати аутентифікацію SQL Server, то для встановлення з'єднання буде потрібно ім'я та пароль облікового запису, попередньо створене на реєстрованому сервері SQL Server. У цьому випадку відкривається вікно, в якому треба зробити вибір режиму підключення до сервера:

a) з використанням інформації, що зберігається одного і того ж облікового запису, для якого треба в цьому ж вікні ввести ім'я та пароль;

b) з використанням облікового запису, ім'я та пароль який треба вводити кожний раз при підключенні до віддаленого сервера.

13. Клацнути по кнопці Next і перейти до наступного вікна майстра.

14. Вибрати або створити нову групу, в яку включити реєстрований сервер.

15. Клацнути по кнопці Next і перейти в останнє вікно майстра зі списком реєстрованих серверів.

16. Клацнути по кнопці Finish.

17. Якщо реєстрований сервер знайдений, то станеться підключення до нього.

18. Якщо реєстрований сервер не знайдений. То Enterprise Manager видасть запит на повторну реєстрацію сервера.

19. Можливі помилки при реєстрації:

a) реєстрований сервер був зупинений;

b) на комп'ютері, з якого виконується реєстрація використовується інші мережеві бібліотеки і протоколи, ніж на реєстрованому сервері;

c) якщо сервер зареєстрований з використанням аутентифікації Windows NT, а для користувачів не створено відповідного облікового запису на SQL Server (Login failed);

d) якщо використовується аутентифікація SQL Server і ім'я пароль задані невірно.

Завдання 16.3. Провести реєстрацію віддаленого сервера, виконуючи команду:

EXEC sp.addserver @server = 'server',

@local = 'local',

@ Duplicate.ok = 'duplicate.ok'

Тут параметри мають таке призначення:

Server - ім'я реєстрованого сервера, яке складається з мережевого імені NetBios відповідного комп'ютера та імені копії сервера, між якими ставиться роздільник "\", при цьому для копії сервера за замовчуванням друге ім'я не потрібно;

Local - сервер для реєстрації є локальним, інакше - сервер віддалений;

duplicate.ok - дозволяє дублювання імен серверів, що призводить до того, що інформація про новий сервер буде записана поверх старої інформації.

Завдання 16.4. Провести настройку конфігурації утиліти Enterprise Manager, виконавши дії:

- виконати команду Tools / Options утиліти;

- У вікні SQL Server Enterprise Manager Properties вибрати вкладку General.

- в групі Server state pooling (опитування стану сервера) вибрати службу Service і задати кількість секунд, через яке буде проводитися опитування стану відповідної служби, і відображатися цей стан у вигляді відповідного значка.

- для конфігурації одного з серверів в якості центрального сховища інформації необхідно зняти прапорець Read / Store User Independent (незалежне зчитування / зберігання користувачів), а на локальному сервері встановити перемикач Read from remote (зчитувати з віддаленого сервера) і вказати ім'я віддаленого сервера, з якого буде зчитуватися інформація про конфігурацію.

- переконатися, що встановлений прапорець Read / Store User Independent означає колективне використання інформації про конфігурацію, а скинутий - приватне, коли інформація для кожного користувача зберігається окремо.

**Завдання №17.**  Ознайомлення з основними концепціями і технологіями, що лежать в основі функціонування сервера, і реалізують їх компонентами: засобами адміністрування, мережевими бібліотеками, службами, інтерфейсами для створення клієнтських додатків.

Завдання 17.1. Здійснити передачу даних за допомогою майстра Data Transformation Services (DTS), використовуючи спосіб Copy table (s) and view (s) from the source database (копіювати таблицю (таблиці) і уявлення (подання) з джерела), виконавши наступні дії:

1. Запустити майстер: Пуск \ Програми \ Microsoft SQL Server \ Import and Export Data.

2. У першому вікні, яке містить загальну інформацію про роботу майстра, клацнути по кнопці Next.

3. У другому вікні в списку Source (джерело) необхідно вибрати тип джерела даних; в списку Server (сервер) вибрати сервера-джерела; вказати список аутентифікації; в списку Database вибрати базу даних, в яку буде здійснюватися взаємодія. Після цього клацнути по кнопці Next.

4. Для редагування, в разі потреби, параметрів конфігурації клацнути на кнопці Advanced (додатково).

5. У третьому вікні конфігурувати одержувач: в списку Database (база даних) вибрати пункт New (створити) і створити нову базу даних.

6. У четвертому вікні DTS Wizard вибрати спосіб передачі даних Copy table (s) and view (s) from the source database.

7. У п'ятому вікні в стовпці Source Table (таблиця джерело) вибрати одну або більше таблиць для копіювання.

8. Для того, щоб побачити вміст вихідної таблиці, клацніть на кнопці Preview (перегляд).

9. У стовпці Destination (одержувач) вказати ім'я таблиці-одержувача.

10. Якщо необхідно виконати перетворення даних, то в стовпці Transform (перетворення) для відповідної таблиці клацніть на кнопці з двокрапкою. У вікні можна налаштувати процес оновлення не лише самих даних, але і їх типів.

11. Наступне вікно майстра DTS Wizard буде загальним для всіх способів перенесення. У цьому вікні для створеного пакету DTS вказати спосіб його збереження.

12. Якщо вибрали варіант SQL Server, то необхідно встановити параметри:

- в полі Name (ім'я) вказується ім'я, під яким пакет DTS буде збережений в системній базі даних msdb;

- в полі Description (опис) можна ввести опис об'єкта в довільній формі;

- в поле Owner Password (пароль власника), щоб приховати інформацію, зазначену при створенні пакета, від перегляду неавторизованих користувачів, можна встановити пароль власника;

- встановивши в поле User Password (пароль користувача) пароль користувача, можна заборонити виконання пакету користувачами, які не мають на це права.

Тільки ті користувачі, які знають пароль, зможуть виконати пакет DTS.

- в списку Server name (ім'я сервера) вибирається ім'я сервера, на якому буде збережений пакет DTS.

13. При виборі режиму зберігання SQL Server Meta Data Services (служби метаданих SQL Server) майстер виведе вікно, багато в чому нагадує вікно при режимі зберігання SQL Server. Додана лише кнопка Scanning (сканування), за допомогою якої можна встановити взаємозв'язки між об'єктами в джерелі і одержувача даних, які зберігаються в сховищі (первинний і зовнішній ключі, індекси, стовпці, типи даних і т. д.).

14. При виборі двох, що залишилися режимів зберігання майстер виведе вікно, в якому крім імені, опису, пароля користувача і пароля власника необхідно вказати тільки ім'я файлу, в який буде записаний пакет. На цьому робота з майстром DTS Wizard по створенню пакетів для імпорту експорту даних закінчується. Після створення пакета. Після клацання на кнопці Finish (готово) буде створений сам пакет.

15. Якщо при створенні пакета було задано його негайне виконання, то майстер відкриє вікно Executing Package (виконання пакету), що дозволяє стежити за процесом виконання пакету.

Завдання 17.2. Здійснити передачу даних за допомогою майстра Data Transformation Services (DTS), використовуючи спосіб Use a query to specify the data to transfer (використовувати запит для вибірки даних), виконавши такі дії:

1. Виконати з першого по п'ятий пункти завдання 17.1.

2. У четвертому вікні майстра DTS Wizard встановити перемикач Use a query to specify the data to transfer.

3. У вікні, ввести SQL-код запиту; якщо є готовий код, збережений на диску, його можна підключити, скориставшись кнопкою Browse (огляд).

4. Якщо необхідно написати складний запит з перерахуванням таблиць і стовпців і при цьому гарантувати, що вказані правильні імена об'єктів, можна скористатися вбудованим в майстер конструктором запитів. Для виклику конструктора запиту клацнути на кнопці Query Builder (конструктор запитів).

Відкриється вікно, в якому потрібно вибрати, які стовпці, з яких таблиць будуть включені в запит.

5. Клацнути по кнопці Next.

6. У вікні, переносячи імена стовпців з лівої частини вікна в праву, задати порядок сортування, яка ведеться за стовпцями, зазначеним в самому верху списку.

7. Клацнути по кнопці Next.

8. У наступному вікні вказати критерії для вибірки даних: встановити покажчик Only Rows meeting criteria (за виключенням окремих виразів, відповідні критерієм).

9. Якщо необхідності в фільтрації немає, встановіть перемикач All rows (всі рядки). Натисніть на кнопку OK.

10. Після того як редагування запиту закінчено, майстер відкриє вікно, в якому можна налаштувати трансформацію даних. Робота з цим вікном практично нічим не відрізняється від роботи з налаштування трансформації даних при копіюванні між таблицями, описаними в завданінні 17.1.

11. Після настройки трансформації даних необхідно зберегти пакет DTS одним із способів, які вказані в попередньому завданні.

Завдання 17.3. Здійснити передачу даних за допомогою майстра Data Transformation Services (DTS), використовуючи спосіб Copy objects and data between SQL Server databases (копіювати об'єкти і дані між базами даних SQL Server), виконавши такі дії:

1. Виконати з першого по п'ятий пункти завдання 17.1.

2. У четвертому вікні майстра DTS Wizard встановити перемикач Copy objects and data between SQL Server databases.

3. Клацнути на кнопці Next.

4. У вікні вказати, які об'єкти і дані будуть копіюватися: установка прапорця Create destination objects - створення переносяться об'єктів; установка прапорця Drop destination objects first - видалення всіх однойменних об'єктів з кінцевої бази даних; установка прапорця Include all dependent objects - включення всіх залежних об'єктів; установка прапорця Copy data - копіювання тільки структури об'єктів.

5. Встановивши прапорець Copy all objects, виконується копіювання всіх об'єктів.

6. Якщо необхідно скопіювати тільки частина об'єктів, скиньте прапорець Copy all objects і позначити пункти, клацнувши на кнопці Select Objects (вибір об'єктів).

7. Щоб вибрати тільки деякі з них, у вікні майстра скиньте прапорець Use default options (використовувати параметри за замовчуванням). Після клацання на кнопці Options (параметри) у вікні вкажіть об'єкти, які необхідно скопіювати.

8. Після вказівки об'єктів необхідно зберегти пакет DTS одним із способів, які вказані в першому завданні. Після буде створений сам пакет.

Рекомендована література

1. SQL Server Advanced Data Types: JSON, XML, and Beyond Автор:

Peter A. Carter 2018, 392

1. Александр Бондарь. Microsoft SQL Server 2014 (+ файлы примеров) 2015, 589 с.
2. Ицик Бен-Ган. Microsoft SQL Server 2012. Основы T-SQL, 2015, 500 С.
3. Разработка и защита баз данных в Microsoft SQL Server 2005 (2-е изд.), 2016, 157 с.
4. Грошев А.С. Основы работы с базами данных (2-е изд.), 2016, 256 с.