Kpi-best

Міністерство освіти і науки України

Національний технічний університет України

“Київський політехнічний інститут імені Ігоря Сікорського”

Факультет інформатики та обчислювальної техніки

Кафедра обчислювальної техніки

Завдання на лабораторну роботу №4 з курсу

«Компоненти програмної інженерії. Якість та тестування програмного забезпечення»

Київ – 2021

Мета: провести інтеграційне тестування модульної системи.

Варіанти:

|  |  |  |  |
| --- | --- | --- | --- |
| N п/п | Nz mod 6 | Input | Output |
| 1 | 0, 3 | [IIG.PasswordHashingUtils (.dll)](https://drive.google.com/drive/u/1/folders/19DgnFmAXtl_Q1qUgVP4xoucQzqLqDzEY)  [IIG.BinaryFlag (project)](https://github.com/IGORGOD/IIG.BinaryFlag) | File ([.dll](https://drive.google.com/drive/u/1/folders/12mKELhYUBLMVmqkItTxemrLNo0w3cQm8))  Database ([.dll](https://drive.google.com/drive/u/1/folders/1znKlrrJsmYBV2wbaFMBwGcqaZekgb6XG) + [.cs](https://drive.google.com/drive/u/1/folders/1D628B-fdROwQxis7muHQ3fQF64i0YEly) + [.bak](https://drive.google.com/drive/u/1/folders/1wRhiEFUiQSOsZ_v8jdD6_Q-ZlLGFPUWV)) |
| 2 | 1, 4 | [IIG.BinaryFlag (.dll)](https://drive.google.com/drive/u/1/folders/1HLsbKf8k27RNVzDhiJkskLX6ypXO5t2u)  [IIG.PasswordHashingUtils (project)](https://github.com/IGORGOD/IIG.PasswordHashingUtils) | File ([.dll](https://drive.google.com/drive/u/1/folders/12mKELhYUBLMVmqkItTxemrLNo0w3cQm8))  Database ([.dll](https://drive.google.com/drive/u/1/folders/1znKlrrJsmYBV2wbaFMBwGcqaZekgb6XG) + [.cs](https://drive.google.com/drive/u/1/folders/1D628B-fdROwQxis7muHQ3fQF64i0YEly) + [.bak](https://drive.google.com/drive/u/1/folders/1wRhiEFUiQSOsZ_v8jdD6_Q-ZlLGFPUWV)) |
| 3 | 2 | [IIG.FileWorker (.dll)](https://drive.google.com/drive/u/1/folders/12mKELhYUBLMVmqkItTxemrLNo0w3cQm8)  [IIG.BinaryFlag (project)](https://github.com/IGORGOD/IIG.BinaryFlag) | Database ([.dll](https://drive.google.com/drive/u/1/folders/1znKlrrJsmYBV2wbaFMBwGcqaZekgb6XG) + [.cs](https://drive.google.com/drive/u/1/folders/1D628B-fdROwQxis7muHQ3fQF64i0YEly) + [.bak](https://drive.google.com/drive/u/1/folders/1wRhiEFUiQSOsZ_v8jdD6_Q-ZlLGFPUWV)) |
| 4 | 5 | [IIG.FileWorker (.dll)](https://drive.google.com/drive/u/1/folders/12mKELhYUBLMVmqkItTxemrLNo0w3cQm8)  [IIG.PasswordHashingUtils (project)](https://github.com/IGORGOD/IIG.PasswordHashingUtils) | Database ([.dll](https://drive.google.com/drive/u/1/folders/1znKlrrJsmYBV2wbaFMBwGcqaZekgb6XG) + [.cs](https://drive.google.com/drive/u/1/folders/1D628B-fdROwQxis7muHQ3fQF64i0YEly) + [.bak](https://drive.google.com/drive/u/1/folders/1wRhiEFUiQSOsZ_v8jdD6_Q-ZlLGFPUWV)) |

Nz – номер залікової книжки студента

Основні кроки виконання:

1. Вибрати вхідні бібліотеки та вихід за допомогою остачі від ділення номеру залікової книжки на 6.
2. Створити солюшен до якого додати проєкт бібліотеки та через посилання на .dll додати інші бібліотеки.
3. Розгорнути БД на Microsoft SQL Server’і, використовуючи .bak файл(-и) та додати до проекту клас(-и) для роботи з конкретною(-ими) БД.
4. Якщо в Output є File – то БД використовується лише для Input (project), а Input (.dll) виводиться до та читається з файлу за допомогою IIG.FileWorker.
5. Провести інтеграційне тестування I/O (інтеграційні тести, manual тестування, etc.).
6. Підготувати звіт про виконану роботу, який буде містити опис написаних тестів, сирцеві коди юніт-тестів та/або посилання на GitHub де розміщено проект, результати тестування і їх аналіз.

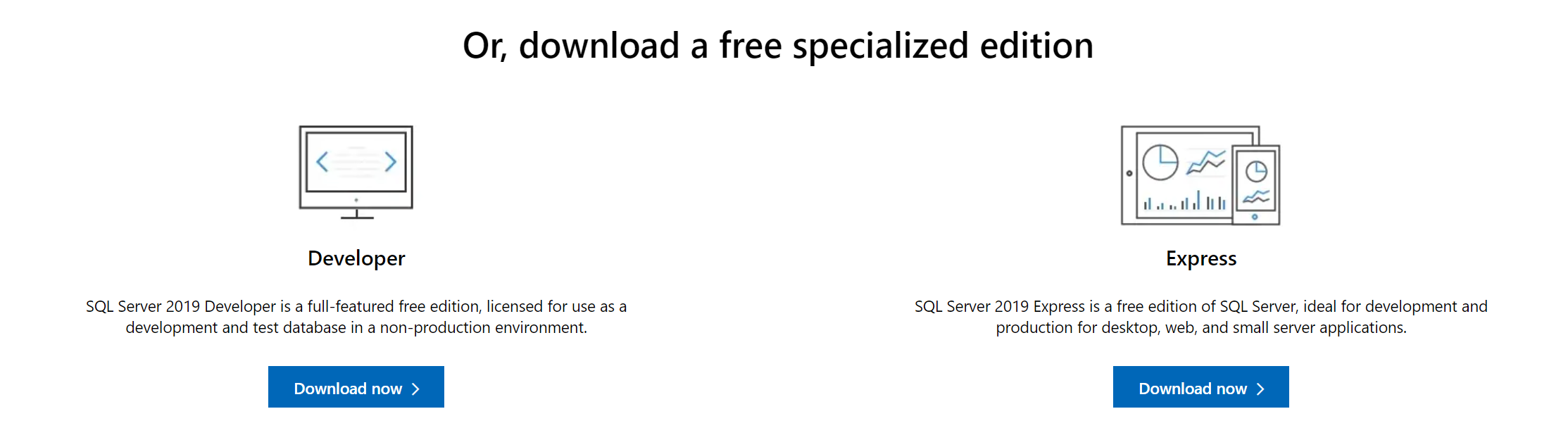
**Додаткові відомості по 2-ому кроку** містяться в завданнях на лаб. роботи 2 та 3.

**Додаткові відомості по 3-ьому кроку:**

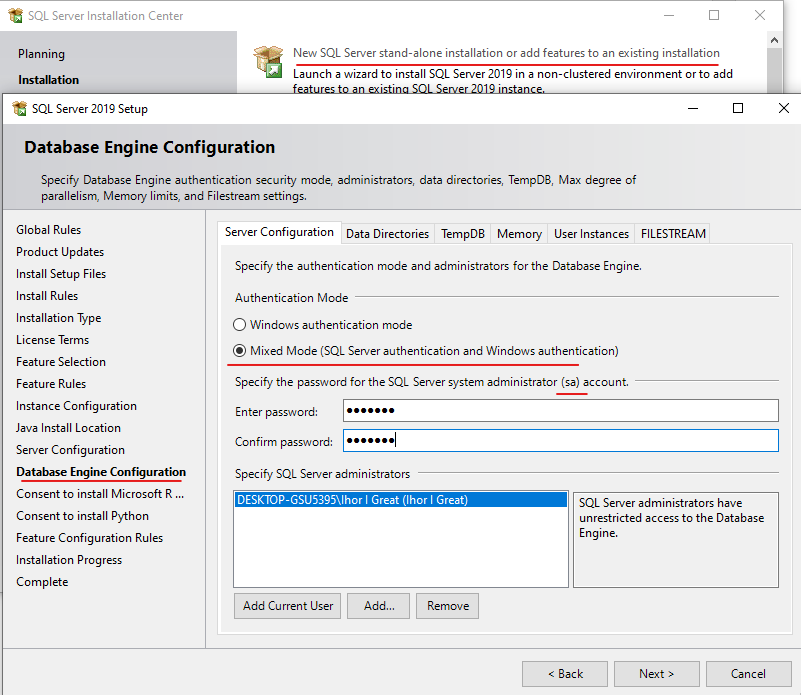
Ви можете використовувати інші СКБД, проте не факт що надані засоби для роботи з БД будуть з ними працювати і вже Вам необхідно буде вирішувати дану проблему, як і перенос міграцію самих БД.

MSSQL:

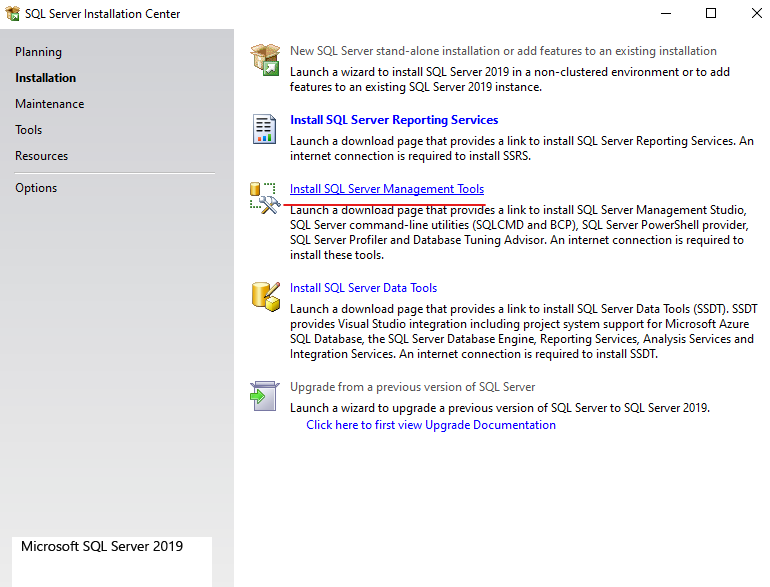
1. Встановити [MSSQL](https://www.microsoft.com/en-us/sql-server/sql-server-downloads) Express/Developer (або більш повну версію)



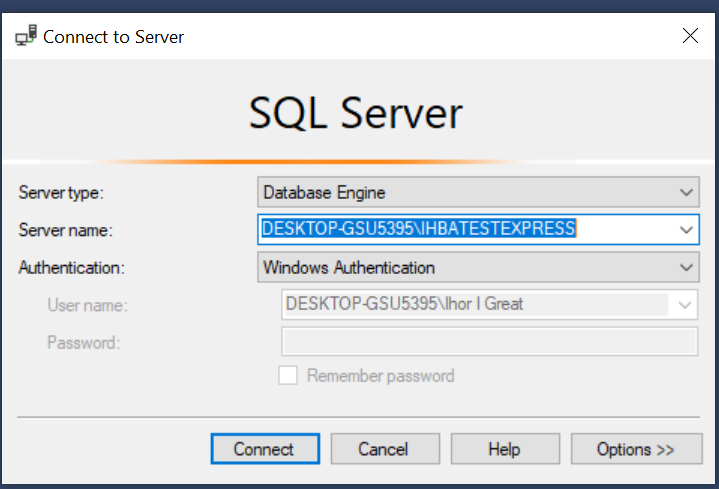
1. Під час інсталяції обов’язково вибрати тип авторизації Mixed, так як для авторизації з ПЗ будемо використовувати логін та пароль, ввести пароль для користувача sa

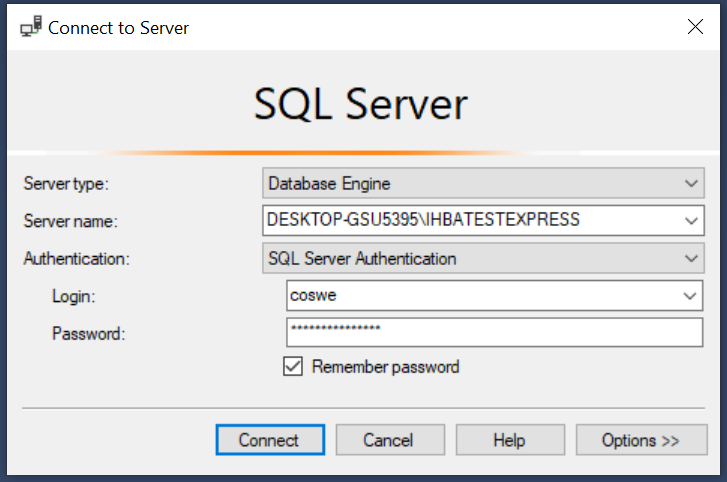


1. Встановити SSMS

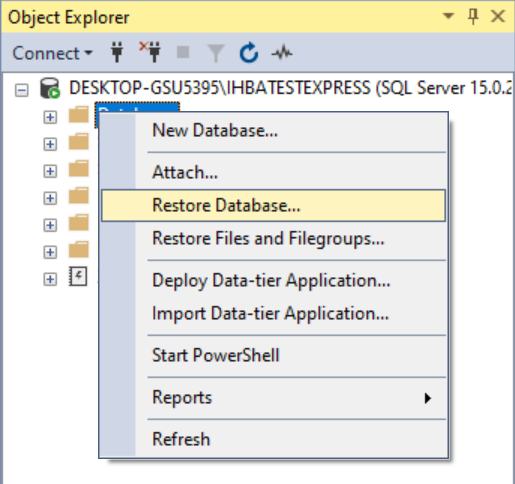


1. Запустити SSMS та заповнити необхідні поля для встановлення з’єднання з Instance’ом (ім’я якого задається під час інсталяції)

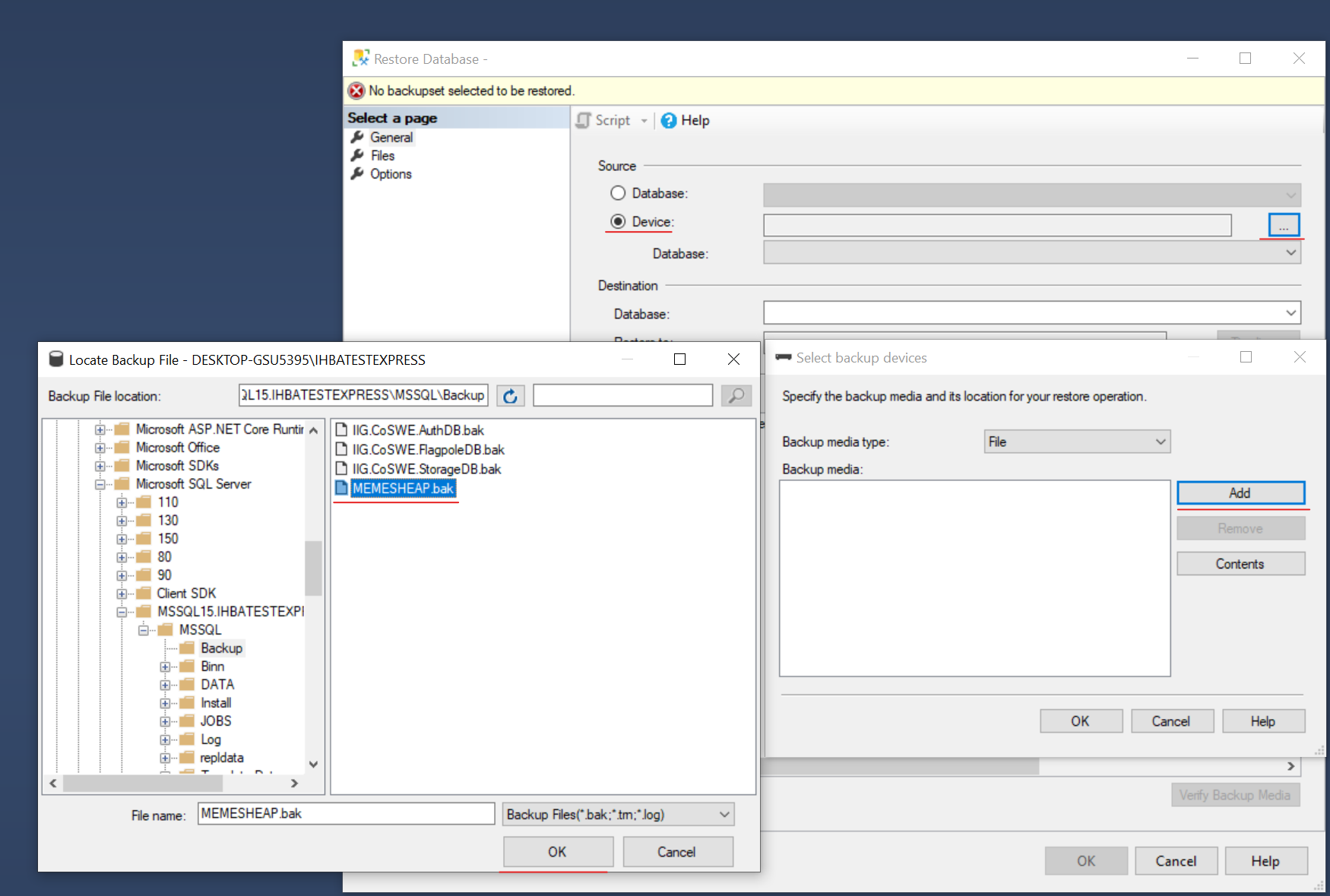




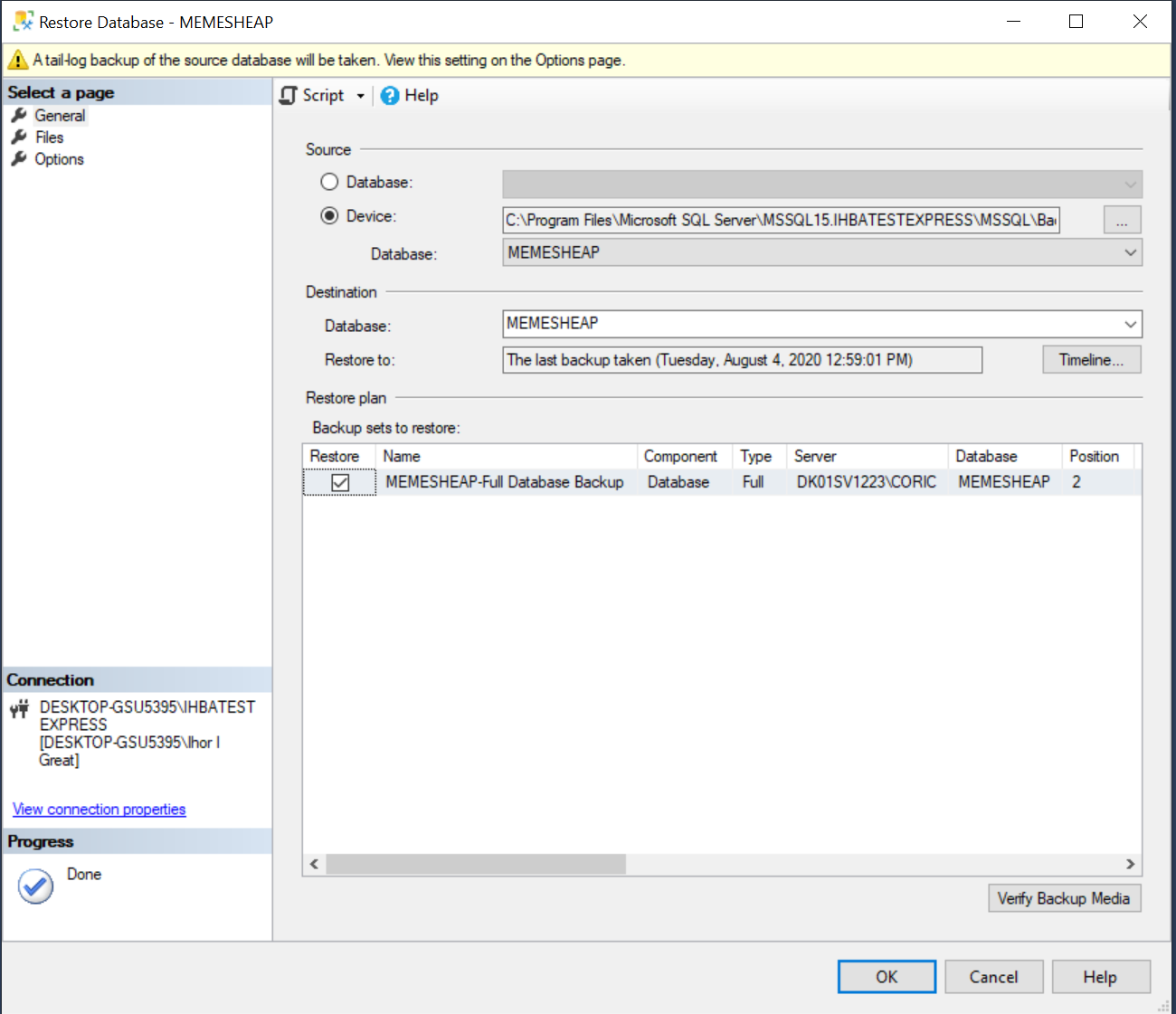
1. ПКМ по Databases та натиснути “Restore Database…”



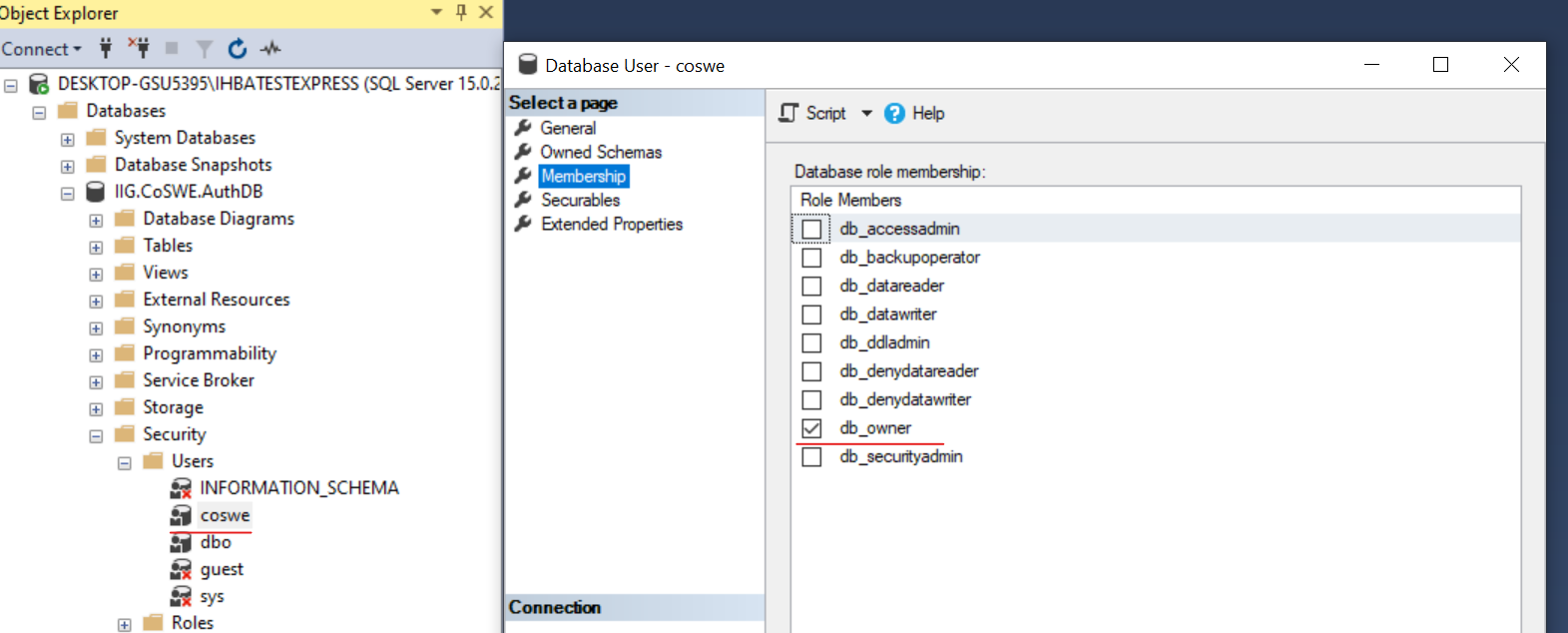
1. Вибрати джерелом Device та додати необхідний .bak файл



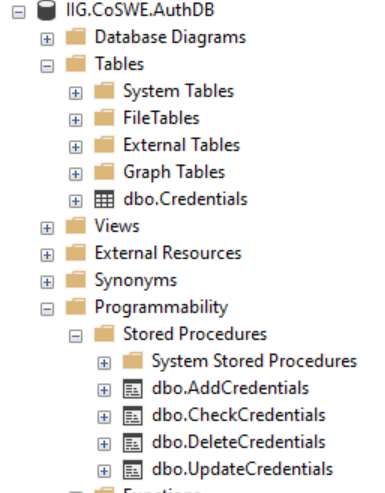
1. Натиснути ОК



1. З БД має підвантажитись і користувач coswe, якого можна використовувати для доступу до БД (пароль: L}EjpfCgru9X@GLj). Якщо користувача не завантажить – для доступу можна використовувати користувача sa



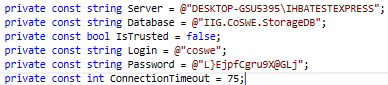
1. Розташування таблиці та процедур на прикладі IIG.CoSWE.AuthDB:



*Зв’язок Input з Output*

|  |  |
| --- | --- |
| Input | Output |
| IIG.PasswordHashingUtils | IIG.CoSWE.AuthDB |
| IIG.FileWorker | IIG.CoSWE.StorageDB |
| IIG.BinaryFlag | IIG.CoSWE.FlagPoleDB |

При підключенні до БД із коду необхідно створити об’єкт необхідного класу передавши в конструктур необхідні параметри:





**Додаткові відомості по 5-ому кроку:**

Ви можете проводити інтеграційне тестування будь-якими методами (запропонованими на лекціях, або взятих із іншого джерела) важливо обгрунтувати вибір (та додати посилання на стороннє джерело, якщо методи не із лекцій).

Я радив би створити проєкт з юніт-тестами, в яких провести тестування шляхом відправки в *Output* (БД/запис в файл) та отримання назад даних, які згенеровані *Input* джерелами, при прогоні тестів мануально перевіряв би вміст *Output* джерел (скріншоти додав би у звіт).