**Лабораторна робота 18.**

**Виконання на клієнті збережених процедур та обробка даних, що вони повертають.**

**Мета роботи:** Навчитись запускати збережені процедури та отримувати стандартний набір рядків, передавати та приймати параметри та код завершення. Виконати обробки множинного повернення даних при використанні збережених процедур.

**Задачі:**

* Запуск збережених процедур без параметрів та отримання стандартного набору рядків
* Передача вхідних параметрів до збереженої процедури
* Обробка вихідних параметрів та коду завершення
* Обробка множинного повернення даних при використанні збережених процедур
* Виконання індивідуального завдання згідно варіанту

1. Створимо процедуру, що повертає список студентів 308 групи (рис.1)

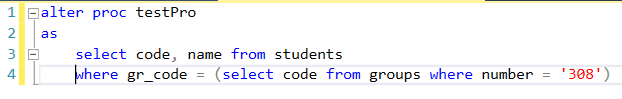
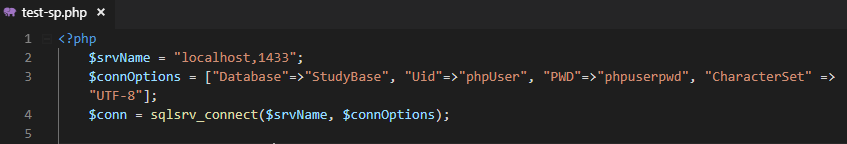


Рис.1. Процедура, що повертає студентів 308 групи

У папці з проектом створимо файл «test-sp.php» та помістимо до нього код, що виконує підключення до БД, виконання створеної збереженої процедури та виведення на екран набору рядків, що повертає процедура (рис.2)



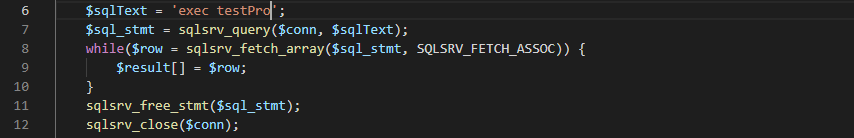




Рис.2. Виконання збереженої процедури

Переглядаємо результат, запустивши у браузері «http://sql-server/test-sp.php» (рис.3)

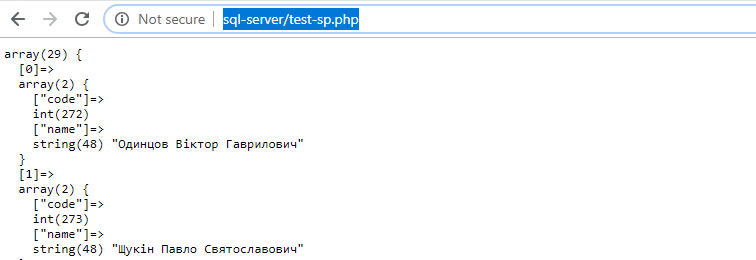


Рис.3. Результат запуску процедури

1. Додамо до нашої процедури вхідний параметр – номер групи для відображення (рис.4)

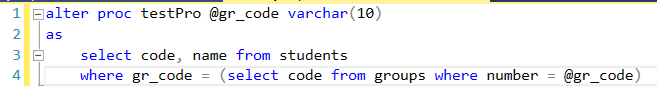


Рис.4. Зміна процедури – додавання вхідного параметру

Модифікуємо код (рис.2) для роботи із новою процедурою (рис.5). Тут і надалі опустимо команди підключення на відключення від БД.

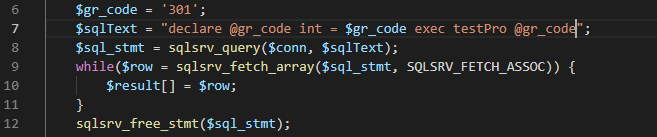


Рис.5. Виклик процедури із параметром

1. Розглянемо використання вихідних параметрів та коду завершення. Виконаємо зміни процедури, щоб вона повертала кількість студентів у групі (рис.6)

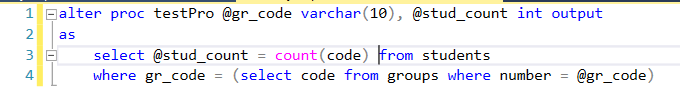


Рис.6. Процедура із вихідним параметром

Модифікуємо код для роботи із новою процедурою (рис.7) та перевіримо результат (рис.8)

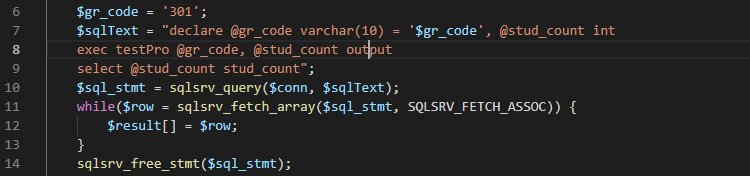


Рис.7. Обробка вихідного параметру процедури

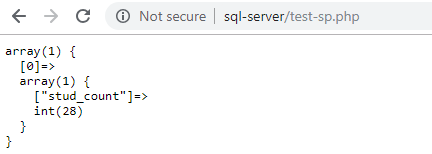


Рис.8. Перевірка обробки вихідного параметра

Аналогічним чином можна обробити код завершення. Реалізуємо процедуру, що виконує перевірку, чи існує група – код повернення 0 або 1 (рис.9), виконаємо її (рис.10) та перевіримо результат роботи (рис.11)

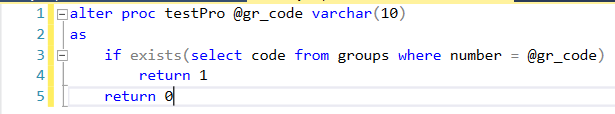


Рис.9. Процедура із кодом повернення

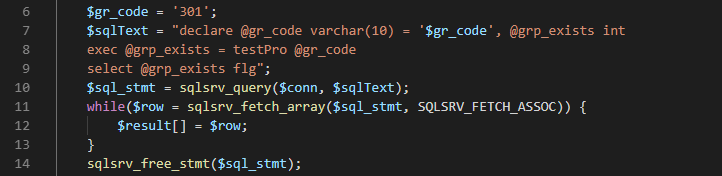


Рис.10. Обробка коду повернення

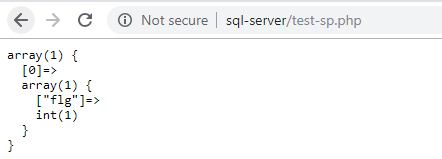


Рис.11. Перевірка результату обробки коду повернення

1. Передати та повернути параметри в збережену процедуру можна ы у інший спосіб. Розглянемо парне використання методів sqlsrv\_prepare() та sqlsrv\_execute(). Повернемо процедуру testPro до стану із вхідним та вихідним параметрами (рис.6) та виконаємо її виклик та обробку вихідного параметру (рис.11)

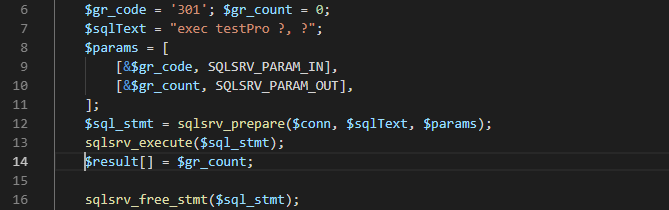


Рис.11. Обробка параметрів через sqlsrv\_prepare() та sqlsrv\_execute()

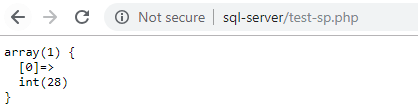


Рис.12. Перевірка роботи процедури

1. Тепер знову змінимо нашу процедуру, щоб вона повертала спеціальність та список студентів певної групи у вигляді набору рядків (рис.13).

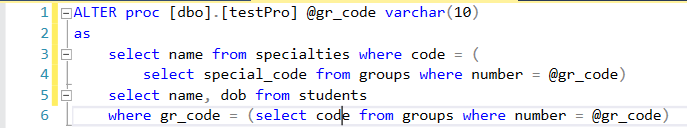


Рис.13. Процедура, що повертає 2 набори рядків

Виконаємо процедуру в SSMS та переглянемо результат (рис.14)

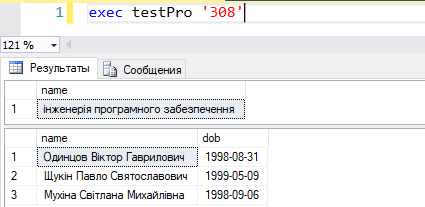


Рис.14. Повернення 2-х результуючих наборів в SSMS

Тепер спробуємо виконати таку процедуру та обробити результат, що вона повертає, використовуючи попередньо розглянутий підхід (рис.15) та переглянемо результат у браузері (рис.16)

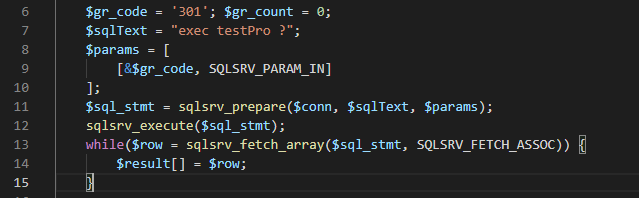


Рис.15. Код виконання процедури

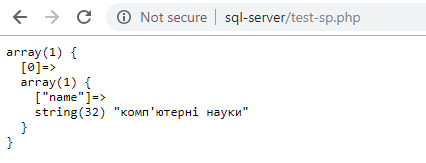


Рис.16. Перевірка результату для 2-х результуючих наборів

Бачимо, що був оброблений лише перший результуючий набір, всі наступні були проігноровані. Даний підхід використовується більшістю драйверів при роботі із БД для запобігання SQL-ін’єкцій (додавання додаткового шкідливого коду до основної команди). Якщо нам потрібно обробити повернення множинного результату, слід скористатись командою sqlsrv\_next\_result. Переробимо попередню обробку результату роботи нашої процедури для коректного повернення всіх результуючих наборів (рис.17) та перевіримо результат (рис.18)

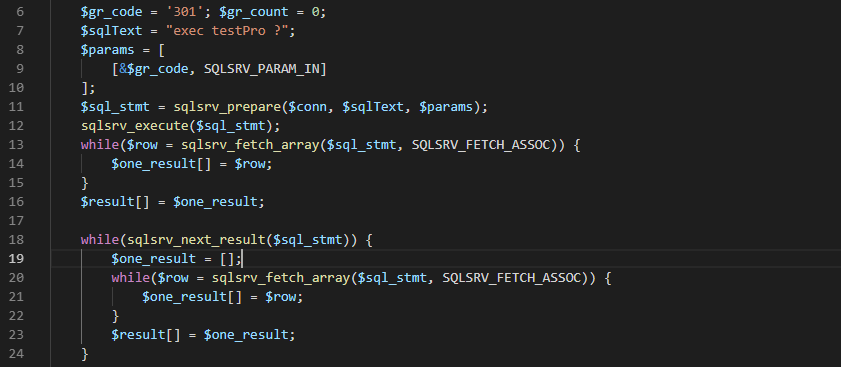


Рис.17. Обробка декількох результуючих наборів



Рис.18. Результат обробки декількох результуючих наборів

1. Тепер реалізуємо декілька наближених до реальності прикладів. Для цього переробимо всі реалізовані на back-end сервісі за попередні 2 заняття запити до БД у вигляді збережених процедур. Почнемо із додавання зміни рядку екзаменаційної відомості. Процедура буде мати 2 вхідні параметри – код рядку і кількість балів та вихідний параметр, який повертає 1 якщо кількість балів в діапазоні [0…100], інакше повертає 0 та процедура не виконує ніяких змін. Для початку реалізуємо саму процедуру на сервері (рис.19)

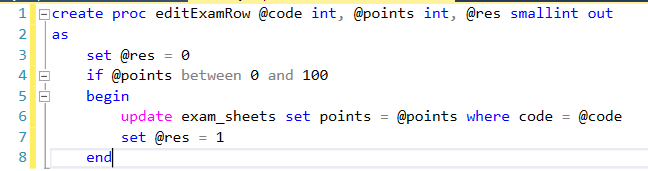


Рис.19. Процедура зміни рядку відомості

Звернемо увагу, що на даний момент у нас не реалізована перевірка кількості балів на допустиме значення при зміні рядку, тобто можливо, наприклад, записати від’ємну кількість балів (рис.20)

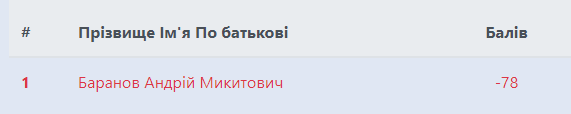
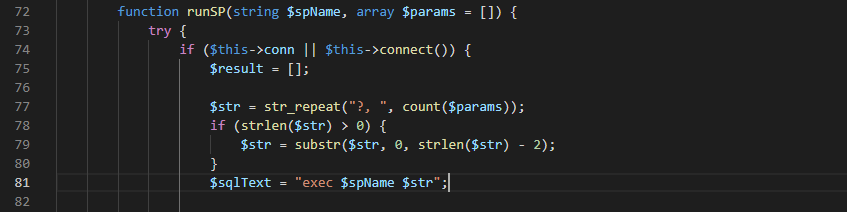
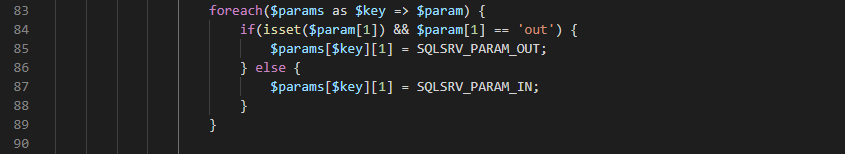
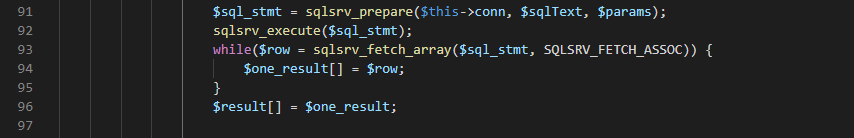


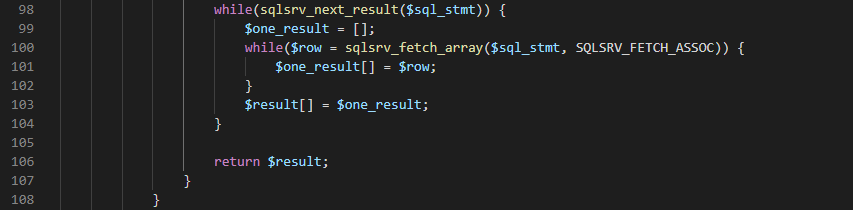
Рис.20. Приклад невірного значення кількості балів

Змінимо бек-енд сервіс, щоб він почав використовувати створену збережену процедуру. Для початку у класі «StudentsDB» (файл «data/db.php») додамо метод для виконання збереженої процедури та обробки вихідних параметрів та результуючих наборів (рис.21)









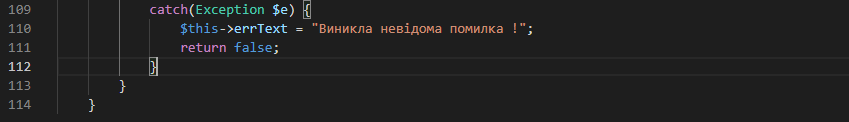


Рис.21. Новий метод «runSP» у класі «StudentsDB»

Далі відкриємо файл «data/exam.php» та у класі «ExamRow» змінимо метод «writeToDB» (рис.22)

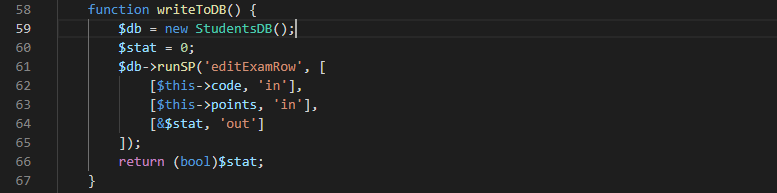


Рис.22. Змінений метод «writeToDB»

Перевіримо результат – тепер некоректна кількість балів не повинна зберігатись.

1. Додамо ще одну процедуру, що повертатиме дані відомості, які потрібно відобразити на сторінці. Як ми пам’ятаємо, у нас є два набори – шапка відомості (викладач, дисципліна, група, дата) і таблична частина із студентами та кількістю балів. Реалізуємо процедуру на сервері БД (рис.23)

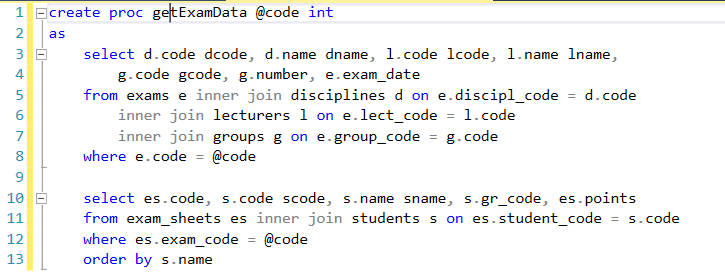


Рис.23. Процедура повернення даних екзаменаційної відомості

Перевіримо роботу процедури в SSMS (рис.24)

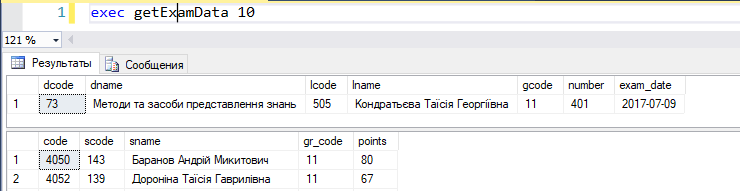
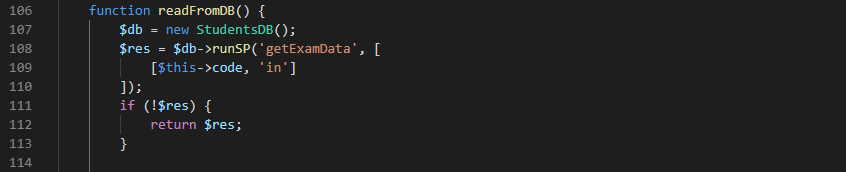
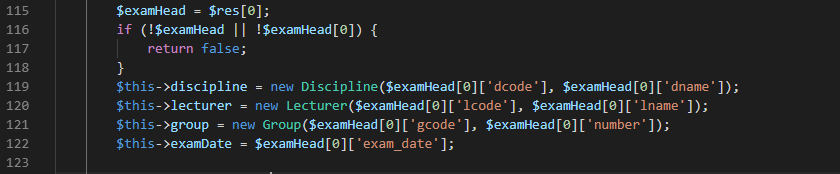


Рис.24. Перевірка роботи процедури повернення даних екзаменаційної відомості

Змінимо метод «readFromDB» у класі «Exam» (файл «data/exam.php»), щоб він використовував створену процедуру (рис.25)





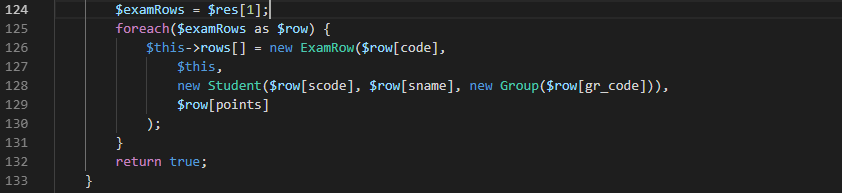


Рис.25. Змінений метод «readFromDB» у класі «Exam»

Перевіряємо роботу сторінки, відображення даних має працювати без змін

1. Кінець роботи

**Завдання для індивідуального виконання.**

Для БД свого варіанту та таблички, обраної при виконанні індивідуального завдання попередніх двох робіт реалізувати у БД процедури додавання, зміни, видалення та вибірки даних та використати їх при реалізації back-end сервісу. Під час реалізації завдання показати використання вхідних і вихідних параметрів процедури та роботу із множинними результуючими наборами.