**ДНІпровський національний університет**

**імені Олеся Гончара**

**ФАКУЛЬТЕТ ПРИКЛАДНОЇ МАТЕМАТИКИ**

**КАФЕДРА КОМП’ЮТЕРНИХ ТЕХНОЛОГІЙ**

**ЗВІТ**

**З ПЕРЕДДИПЛОМНОЇ ПРАКТИКИ**

**Освітньо-професійна програма**

**Комп’ютерне моделювання та технології програмування**

**Спеціальність 113 Прикладна математика**

**Галузь знань 11 Математика і статистика**

**Рівень вищої освіти перший (бакалаврський)**

Студента 4 курсу групи ПА-22у-1

Гречаний Олег Ігорович

(прізвище та ініціали)

Керівник канд. фіз.-мат. Наук  
 Сафронова І.А.   
(посада, вчене звання, науковий ступінь, прізвище та ініціали)

Кількість балів \_\_\_\_\_\_\_\_\_\_\_

Національна шкала \_\_\_\_\_\_\_

Члени комісії:

Зайцев В.Г.

(підпис) (прізвище та ініціали)

Хижа О.Л.

(підпис) (прізвище та ініціали)

Степанова Н.І.

(підпис) (прізвище та ініціали)

м. Дніпро, 2024 р.

ЗМІСТ

[**ВСТУП** 3](#_Toc160120928)

[**ПОСТАНОВКА ЗАДАЧІ** 5](#_Toc160120929)

[1. ВИБІР СЕРЕДОВИЩА ДЛЯ РЕАЛІЗАЦІЇ ІНФОРМАЦІЙНОЇ СИСТЕМИ 6](#_Toc160120930)

[2. РЕАЛІЗАЦІЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ 8](#_Toc160120931)

[2.1. Проектування та створення бази даних: 8](#_Toc160120932)

[2.2. Реалізація імпорту та експорту даних: 10](#_Toc160120933)

[2.3. Вихідні дані: 15](#_Toc160120934)

[2.4. Тестування та налагодження: 16](#_Toc160120935)

[3. АНАЛІЗ ОТРИМАНИХ РЕЗУЛЬТАТІВ 17](#_Toc160120936)

[ВИСНОВКИ 18](#_Toc160120937)

[СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ 19](#_Toc160120938)

# **ВСТУП**

Сучасний розвиток освітніх технологій вимагає постійного удосконалення методів та засобів управління навчальним процесом. У цьому контексті виникає необхідність у використанні ефективних інформаційних систем, спрямованих на оптимізацію роботи навчальних закладів та їх ЗВО. Актуальність проблеми полягає в потребі вдосконалення інструментів управління навчальним процесом, щоб забезпечити якісну та ефективну освіту в умовах сучасного світу.

**Об'єктом дослідження** є інформаційна система, розроблена для навчальних закладів.

**Предметом дослідження** є функціональні можливості даної інформаційної системи, зокрема надання файлів у форматі Excel, форматування звітів, а також додавання, зміна та видалення даних в тому числі, додавання даних із файлів.

**Метою дослідження** є вивчення та науково обґрунтований аналіз можливостей та ефективності запропонованої інформаційної системи для оптимізації навчального процесу.

Впровадження запропонованої інформаційної системи сприятиме покращенню організації навчального процесу та може збільшити ефективність роботи навчальних закладів.

Під час проходження практики з метою досягнення поставленої мети необхідно виконати такі завдання:

* Провести аналіз функціональних можливостей інформаційної системи.
* Оцінити ефективності використання запропонованої системи в навчальному процесі ДНУ.
* Визначити переваги та недоліки інформаційної системи для подальшого її удосконалення.

**Методи дослідження** включають аналіз літературних джерел, порівняльний аналіз існуючих систем, а також спостереження за використанням системи в реальних умовах навчального процесу.(21-43)

**Наукова новизна** дослідження полягає у тому, що воно спрямоване на вивчення конкретних можливостей інформаційної системи для оптимізації навчального процесу та визначення його ефективності.

**Практичне значення** дослідження полягає у розробці та впровадженні нових інструментів, які допоможуть в управлінні навчальним процесом, і дозволять підвищити якість освіти та забезпечити більш ефективну роботу навчального закладу.

# **ПОСТАНОВКА ЗАДАЧІ**

У рамках навчально-практичної діяльності передбачається створення комплексної інформаційної системи, що включатиме різноманітні функціональні можливості. По-перше, необхідно розробити можливість спрощенного перегляду даних для обробки шляхом створення макросів на основі бази даних , що охоплюватиме їх додавання, видалення та редагування. Далі, база даних має забезпечити можливість виконання різноманітних операцій над цими даними, включаючи сортування, фільтрацію, групування і т. д. та створення звітів і форм для забезпечення зручного інтерфейсу.

Також, система повинна вміти обробляти запити користувачів до бази даних для отримання конкретних наборів інформації відповідно до їхніх вимог. Крім того, вона має забезпечити можливість формування звітів на основі зібраних даних, щоб користувачі могли аналізувати та використовувати інформацію для прийняття рішень.

Після створення скелету програми, програма повинна мати інтерфейс для зручного користування юзерів.

Однією з ключових функцій системи повинна бути можливість виявлення даних, що не відповідають заданим правилам або критеріям якості. Після виявлення таких даних система повинна пропонувати можливі шляхи їх виправлення або усунення, наприклад, за допомогою автоматичної корекції або надання рекомендацій користувачеві для втручання в процес.

Таким чином, завдання полягає у розробці системи, яка забезпечить повний цикл управління даними, від їх збору та обробки до аналізу та звітності, одночасно надаючи інструменти для виявлення та виправлення потенційних невідповідностей чи помилок у даних.

1. ВИБІР СЕРЕДОВИЩА ДЛЯ РЕАЛІЗАЦІЇ ІНФОРМАЦІЙНОЇ СИСТЕМИ

Для реалізації інформаційної системи була використана мова програмування С# і додаткові фреймворки для забезпечення функціональності системи, також була впроваджена система управління базами даних (СУБД) MySQL.

C# (C Sharp) [1] - це сучасна об'єктно-орієнтована мова програмування, розроблена компанією Microsoft. Вона призначена для розробки різноманітних програм, від десктопних до веб-додатків і мобільних застосунків. C# має сильну типізацію, підтримує кілька парадигм програмування, включаючи об'єктно-орієнтоване, функціональне та аспектно-орієнтоване програмування. Вона є однією з основних мов програмування для платформ .NET, яка надає розробникам широкі можливості для створення потужних та надійних додатків.

Excel [2]- це програмне забезпечення для створення електронних таблиць це дуже гнучкий і потужний інструмент для роботи з даними, і він може бути легко інтегрований з іншими програмними продуктами та мовами програмування, такими як C#.

MySQL [3]- це відкрита система управління базами даних (СУБД), яка використовує мову запитів SQL (Structured Query Language) для взаємодії з даними. MySQL розроблена, підтримується та розповсюджується компанією Oracle Corporation.

Ця система баз даних широко використовується для зберігання та управління різноманітною інформацією, від простих наборів даних до складних веб-додатків та корпоративних систем. MySQL підтримує багатокористувацький доступ до баз даних, транзакції, реплікацію даних для забезпечення високої доступності та інші функції, які роблять її популярним вибором для багатьох типів проектів. Вона також відома своєю швидкодією та надійністю. MySQL є вільно доступною та має велике спільнотою користувачів і розробників, що сприяє її активному розвитку та підтримці.

Фреймворки:

* Entity Framework є рішенням для зручного взаємодії з базами даних, що дозволяє спростити роботу з реляційно-об'єктною моделлю даних, що лежить в основі багатьох програм.[4]
* Pomelo.EntityFramework - це конкретний інструмент, призначений для забезпечення підключення та взаємодії з базами даних MySQL, що забезпечує надійність і ефективність роботи з цією платформою.[5]
* EPPlus - це набір інструментів для зручної роботи з Excel-файлами, який дозволяє здійснювати різноманітні операції з даними у форматі Excel з використанням .NET платформи.[6]
* WPF (Windows Presentation Foundation) - це потужне середовище для створення графічного інтерфейсу, яке надає розробникам широкі можливості для створення модерних та привабливих додатків для платформи Windows.[7]
* Microsoft.Extensions.Hosting та Microsoft.Extensions.DependencyInjection - це набори інструментів, які допомагають впроваджувати в проект контейнер залежностей та забезпечувати управління сервісами в додатках .NET, що допомагає підтримувати структуру проекту та забезпечити його масштабованість.

Обрання цього набору технологій сприятиме ефективній реалізації декстопної інформаційної системи, забезпечивши інтерфейс користувача та серверну частину з можливістю взаємодії.

1. РЕАЛІЗАЦІЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ

Відповідно до поставлених вимог необхідно розробити інформаційну систему десктопного формату, яка буде синтаксично аналізувати (парсити) вхідні дані з Excel формату та видавати структурований файл. Також інформаційній системі необхідно надати зручний інтерфейс, і впровадити можливість додавання, змінювання та видалення даних.

Для реалізації головного функціоналу інформаційної системи, потрібно створити серверну частину. Серверна частина виступає як централізоване сховище даних, відповідаючи моїм вимогам, а саме додавання, видалення, зберігання та модифікування даних.

Приступаємо до реалізації інформаційної системи на мові програмування С# (та додаткових фреймворків) з використанням СУБД MySQL. Нижче наведено етапі, які були виконані для реалізації цього проекту:

## Проектування та створення бази даних:

Було створено інфологічну модель бази даних (рис. 2.1) із структури таблиць для збереження даних для їх подальшого використання. Також розроблену базу даних було підлюченно до інформаційної системи завдяки фреймворкам EntityFramework та Pomelo.EntityFramework.

Збереження інформації реалізовано завдяки створенню класів, наприклад клас «дисципліна» наведено у табл. 2.1 та табл. 2.2.

Таблиця 2.1 - Атрибути класу

| **Атрибут** | **Опис** |
| --- | --- |
| [Index] | Цей атрибут вказує, що в базі даних має бути створений індекс для властивості CodeNumber і вимагає, щоб ця властивість була унікальною. |
| [Key] | Позначає властивість Id як первинний ключ. |
| [Required] | Позначає властивості CodeNumber та Name як обов'язкові для заповнення. |
| [ForeignKey] | Вказує на зовнішній ключ, який зв'язує цю властивість з таблицею Department. |
| [Range] | Встановлює діапазон значень для властивостей, вказуючи мінімальне та максимальне значення. |

Зображення, що містить текст, знімок екрана, Шрифт, число

Автоматично згенерований опис

Рис. 2.1 – Інфологічна модель БД

Таблиця 2.2 - Властивості класу Discipline

| **Дані** | **Опис** |
| --- | --- |
| Id | Первинний ключ дисципліни |
| CodeNumber | Код дисципліни |
| Назва | Назва дисципліни |
| DepartmentId | Зв'язок з таблицею факультетів (departments) |
| Department | Зв'язок з таблицею факультетів (departments) |
| LectureHours | Кількість годин лекцій |
| PracticeHours | Кількість годин практики |
| LaboratoryHours | Кількість годин лабораторних робіт |
| AmountOfStudents | Кількість студентів |
| MaxAmountOfStudents | Максимальна кількість студентів |
| MaxAmountOfStudentsInGroup | Максимальна кількість студентів у групі |
| MaxAmountOfStudentsInSubGroup | Максимальна кількість студентів у підгрупі |
| Student | Колекція студентів, пов'язаних із цією дисципліною |
| Lecturers | Колекція викладачів, які викладають цю дисципліну |

## Реалізація імпорту та експорту даних:

Одна з обов’язкових умов даної роботи, це можливість надати користувачам вхідні дані в Excel форматі та, після проведення відповідного аналізу та впорядкування, можливість генерувати новий Excel файл з находженням помилок.

Для реалізації парсеру [8], я використав фреймворк EPPlus, саме він надає змогу зчитувати та зберігати необхідні дані у файлах формату Excel. У рамках C# та Excel файлів, парсер є програмним засобом, який використовується для синтаксичного аналізу (парсингу) даних, що зберігаються у файлі Excel. Парсер може витягти інформацію з окремих комірок, рядків або стовпців у файлі Excel, обробити її та використати для подальшої обробки або виведення.

Розроблений алгоритм наведено далі у вигляді коду мовою С#:

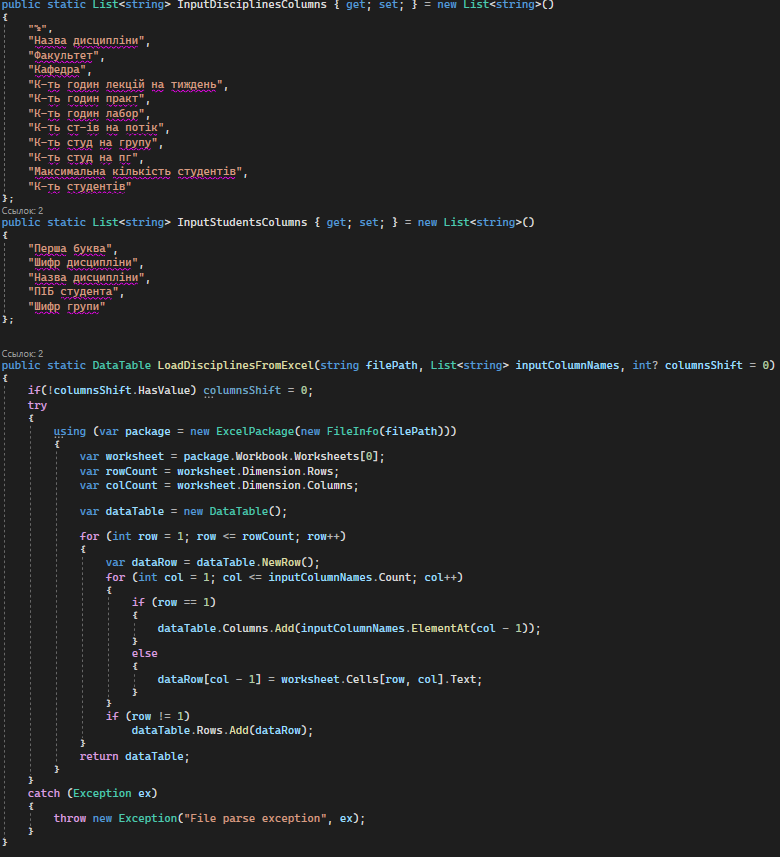


Рис. 2.2 – Алгоритм парсеру

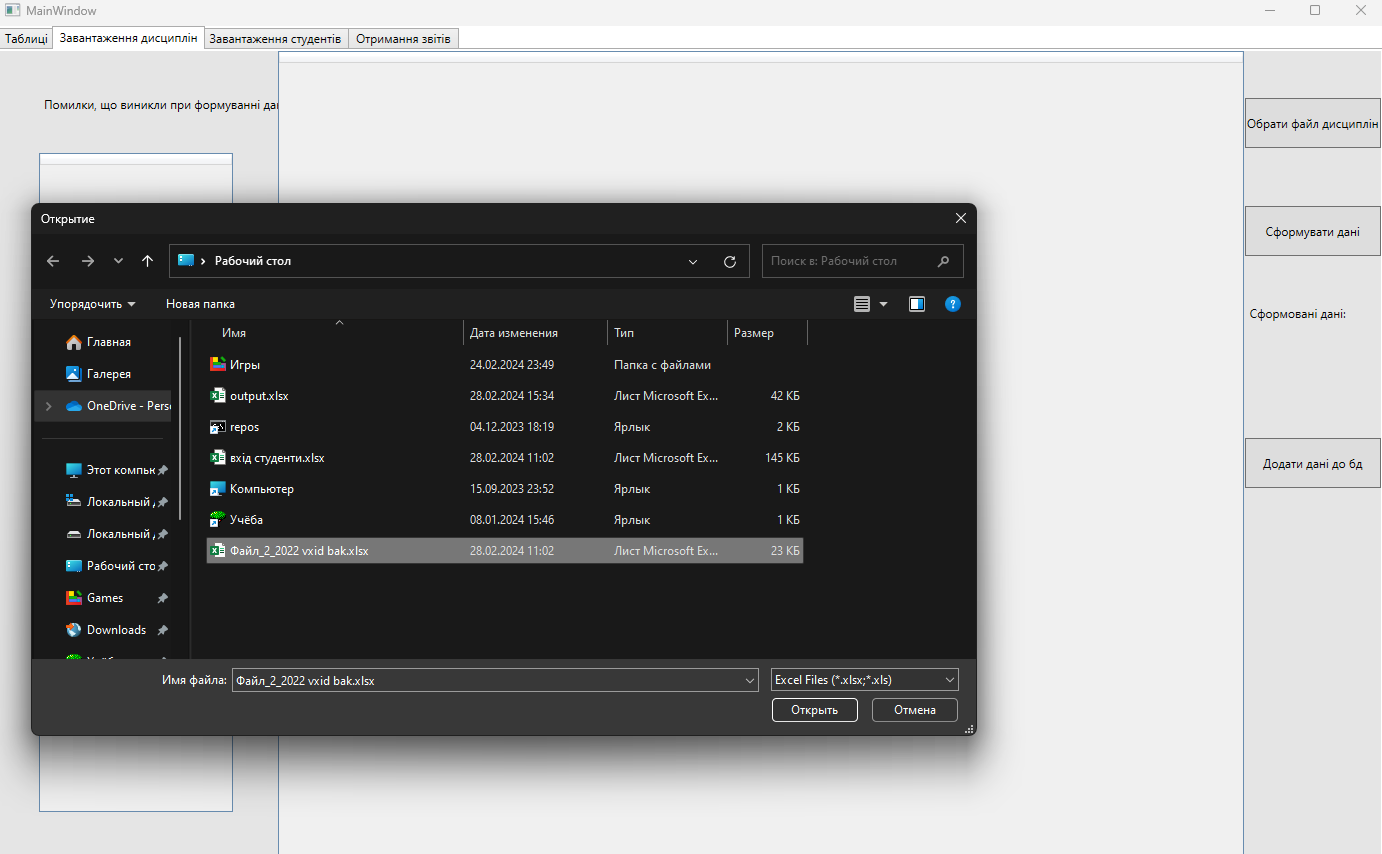
У табл. 2.3 наведено структуру коду, для більш зручного розуміння елементів:

Таблиця 2.3 – Структура коду

| **Елемент коду** | **Опис** |
| --- | --- |
| using directives | Директиви using, що вказують на простори імен, які використовуються у програмі. |
| namespace WpfApp1 | Простір імен, який містить усі класи програми. |
| class ExcelParser | Клас, який містить методи для роботи з Excel-файлами. |
| InputDisciplinesColumns | Список назв стовпців для даних про дисципліни в Excel. |
| InputStudentsColumns | Список назв стовпців для даних про студентів в Excel. |
| LoadDisciplinesFromExcel | Метод для завантаження даних про дисципліни з Excel-файлу. |
| try...catch блок | Блок обробки винятків під час роботи з Excel-файлом. |

1. Інтерфейс користувача:

Для створення інтерфейсу я використав фреймворк WPF (Windows Presentation Foundation) (рис. 2.3):

Рис. 2.3 – Завантаження дисциплін

Далі наведено (рис. 2.4 – 2.6) процес завантаження даних формату Excel для подальшого використання.

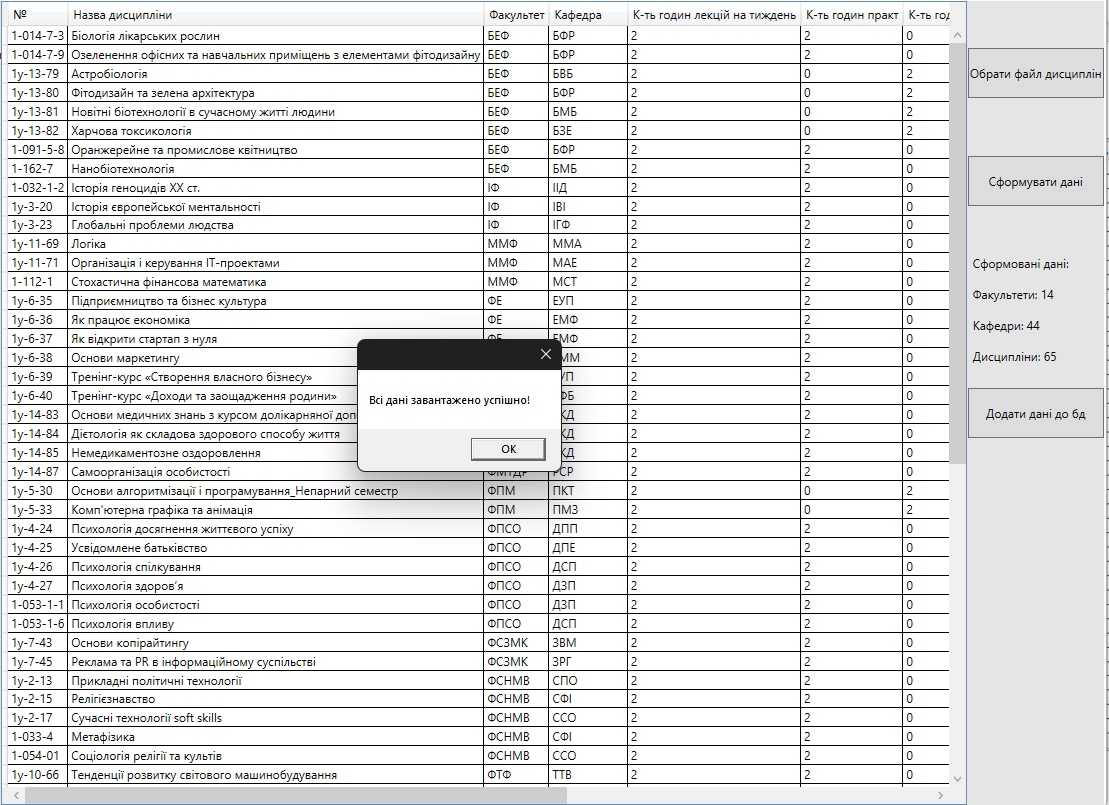


Рис 2.4 – Додавання та зберігання даних у БД

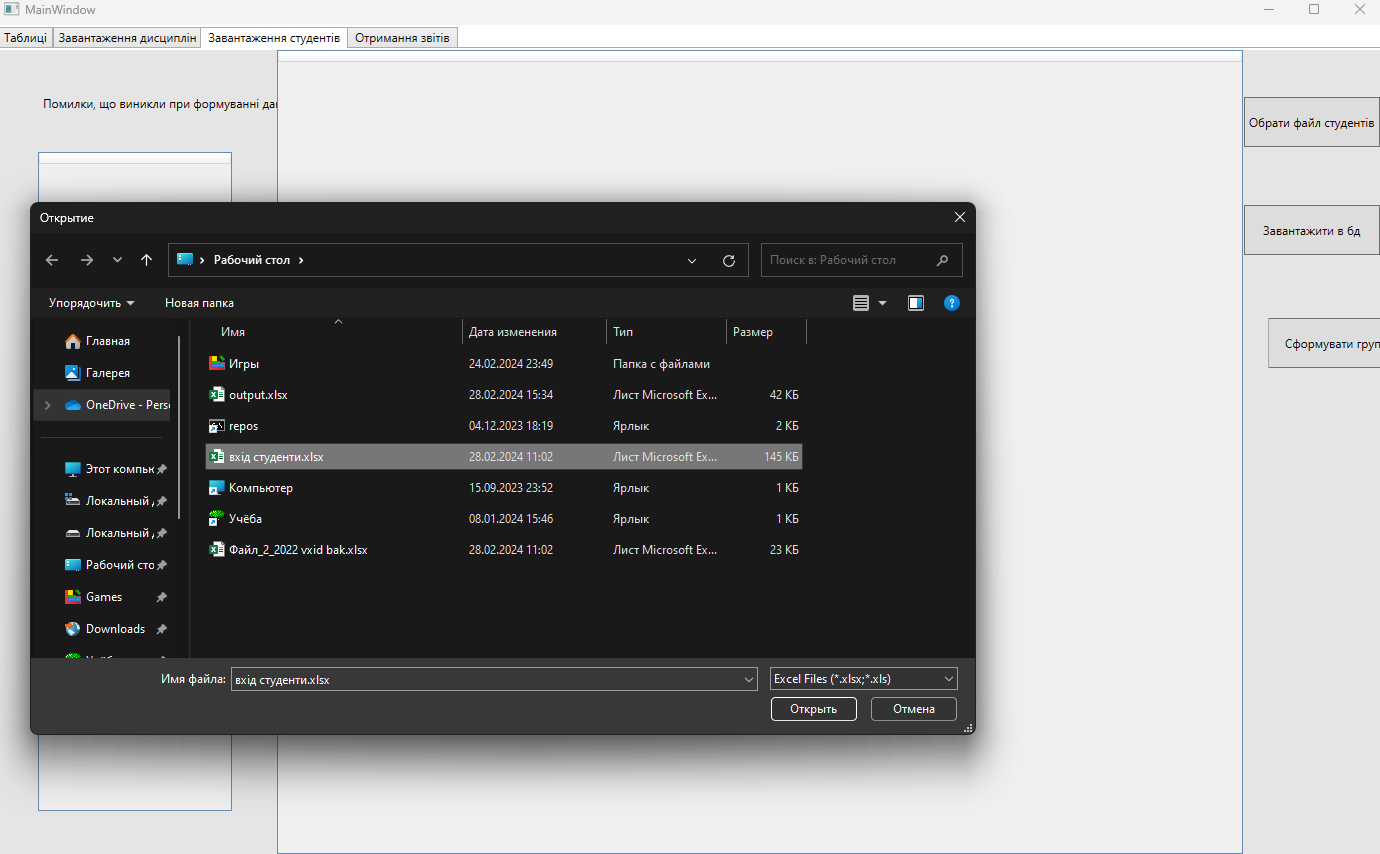


Рис 2.5 – Завантаження відомостей про студентів

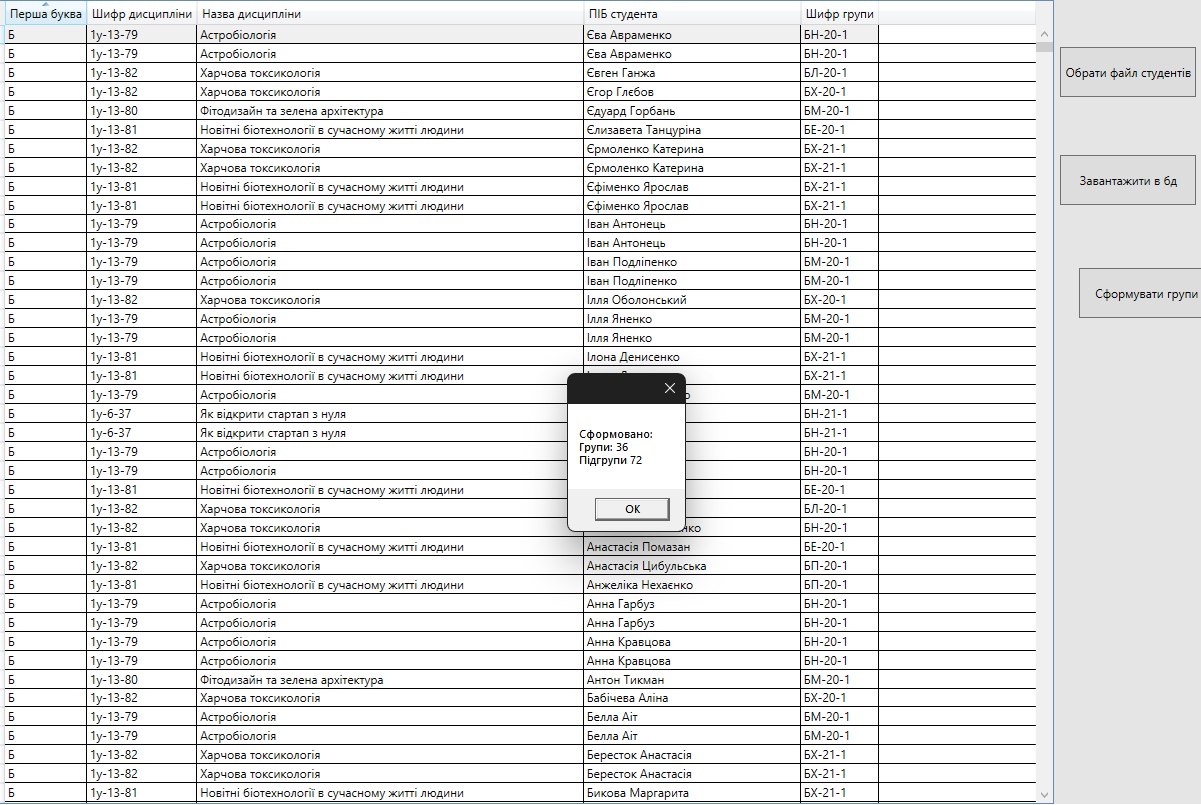


Рис 2.6 – Формування та зберігання груп / підгруп

Всі дані з файлів, що експортуються у програмне забезпечення, зберігаються у відповідній БД, яка реалізована і підтримується СУБД MySQL.

На рис. 2.7 - 2.9 наведено інтерфейс для відображення даних за атрибутами: «факультети», «кафедри», «потоки», «групи», «дисципліни», де користувач з легкістю може додавати, видаляти, змінювати потрібні дані.

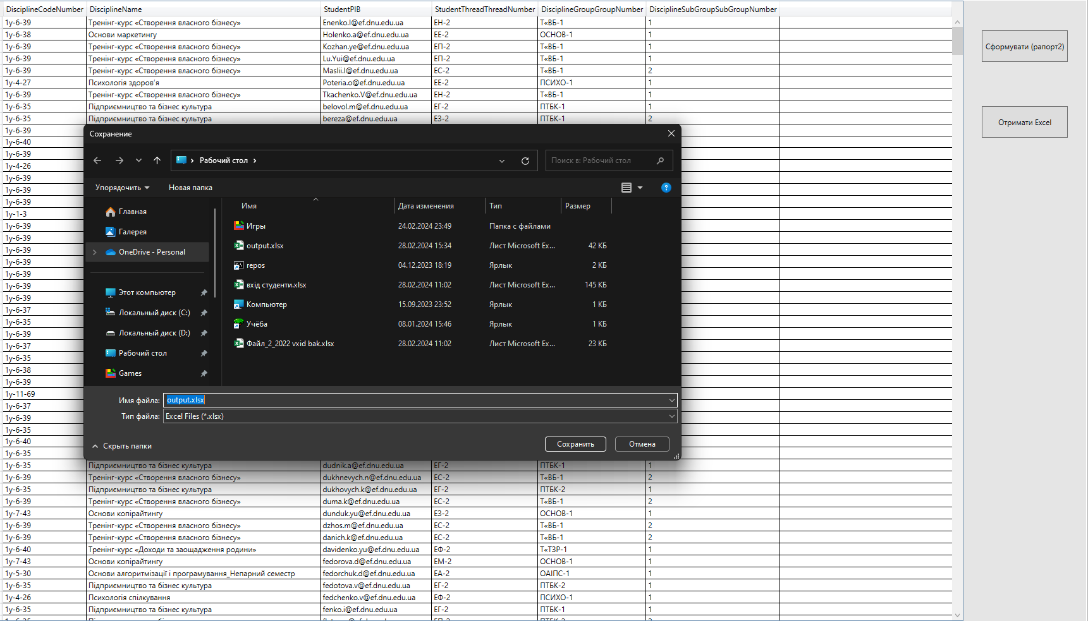


Рис 2.7 – Формування та збереження вихідного файлу

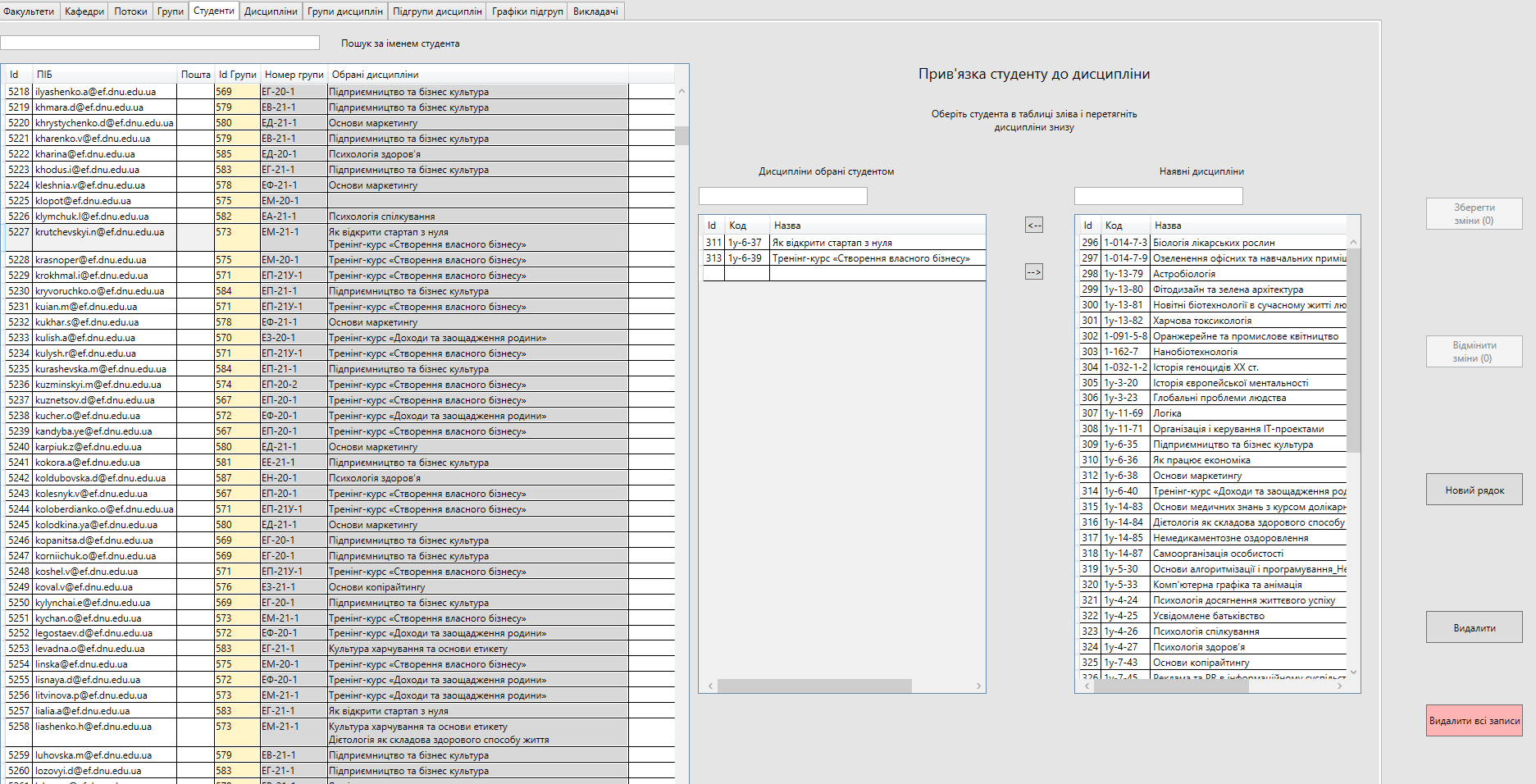


Рис. 2.8 – Відображення відомостей про студентів   
та обрані ними вибіркові дисциплін

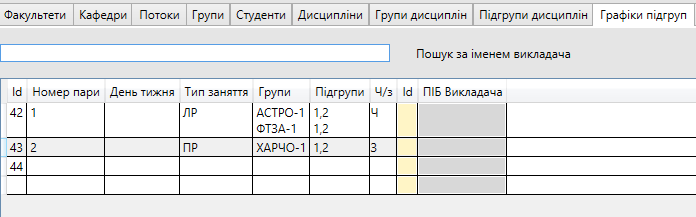


Рис 2.9 – Графік підгруп

За допомогою даного інтерфейсу можна створювати розклад занять для певних груп з урахуванням підгруп, номером пари, днем тижня та типом занять, тощо. У подальшому дану систему можна модифікувати для знаходження «накладань» у розкладах.

## Вихідні дані:

Інформаційна система після завантаження даних про дисципліни та студентів, генерує вихідний файл з назвою output (рис. 2.10).

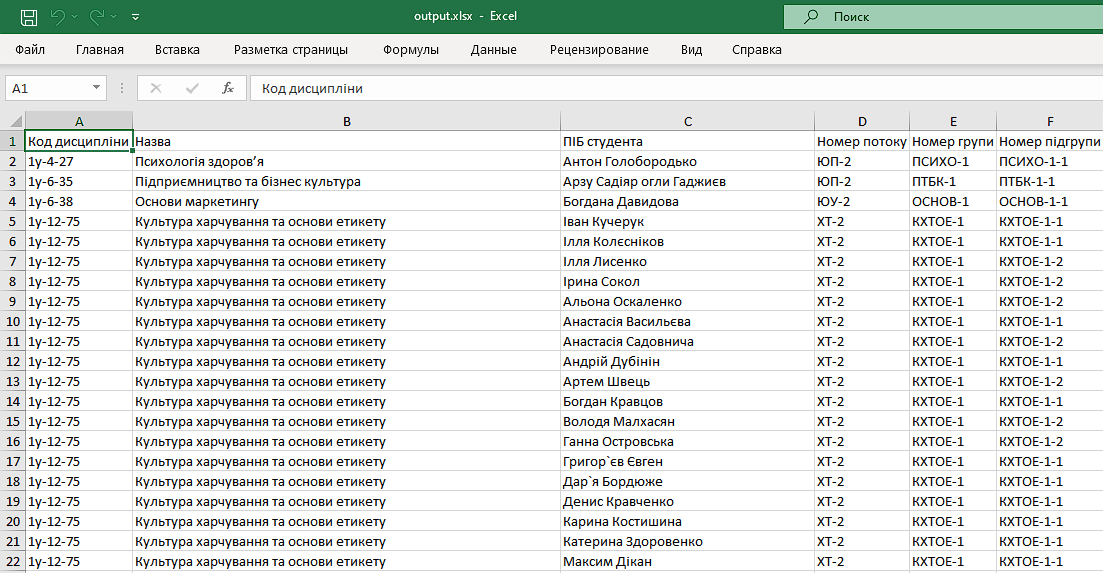


Рис 2.10 – вихідні данні

## Тестування ****та налагодження****:

Після розробки інформаційної системи було проведено тестування, щоб переконатися в її працездатності та відповідності вимогам.

Після виконання всіх етапів, отримали реалізовану інформаційну систему, що дозволяє користувачам експортувати файли, а на імпорт отримувати звіт з номерами потоку / групи / підгрупи. Також система може відображати окремо дані про дисципліни, групи, студентів та надає змогу користувачеві додавати нові, видаляти та змінювати існуючі дані.

1. АНАЛІЗ ОТРИМАНИХ РЕЗУЛЬТАТІВ

Отримані результати свідчать про успішну розробку десктопної інформаційної системи з рядом важливих функцій. Система здатна експортувати файли з даними про дисципліни та студентів, а також імпортувати файли з проставленими номерами груп/підгруп/потоків. При цьому, система забезпечує зручний інтерфейс для перегляду та редагування даних, включаючи інформацію про дисципліни, студентів, їх вибіркові дисципліни та факультети.

Розробка була здійснена на мові програмування C# з використанням різноманітних фреймворків та бібліотек. Зокрема:

1. **Entity Framework** використовується для реляційного об'єктного відображення бази даних. Цей інструмент дозволяє зручно працювати з об'єктами бази даних у вигляді класів.
2. **Pomelo.EntityFrameworkCore** виступає як провайдер для забезпечення з'єднання з MySQL базою даних. Використання цього провайдера відкриває доступ до функціональності MySQL для додатка.
3. **EPPlus** - це бібліотека, яка надає можливості для роботи з Excel файлами. Її використання дозволяє зручно зберігати та обробляти дані у форматі Excel.
4. **WPF** (Windows Presentation Foundation) використовується для побудови графічного інтерфейсу користувача. WPF дозволяє створювати багатовіконні додатки з допомогою різноманітних елементів управління та вікон.

З урахуванням здійснених кроків та використаних інструментів, система виявляється готовою до подальшого розвитку та розширення функціональності. Такі напрямки розвитку, як покращення інтерфейсу, оптимізація роботи з даними та розширення можливостей експорту та імпорту, також у майбутньому можна впровадити систему знаходження нестиковок у розкладі.

ВИСНОВКИ

У рамках преддипломної практики було розроблено та реалізовано мовою програмування С# та допоміжних фреймворків з підключенням СУБД MySQL прототип інформаційної системи, яка дозволяє експортувати дані про дисципліни та студентів з табличних файлів і отримувати звіт з сформованими і встановленими номерами потоку / груп / підгруп.

Було спроектовано та створено інфологічну модель та відповідну їй базу даних, яку було підлюченно до інформаційної системи завдяки фреймворкам EntityFramework та Pomelo.EntityFramework. Завдяки базі даних, інформаційній системі можна імпортувати дані у форматі Excel, де система надалі буде працювати з цими даними завдяки синтаксичному аналізу, щоб потім згенерувати сформований звіт.

Було створено клієнтську частину, яка відповідає за візуальний огляд інформаційної системи, за допомогою фреймворку WPF, де можна передивлятись дані про дисципліни / студентів / факультети, а також додавати нові, видаляти та модифікувати вже існуючі дані.

За допомогою даного інтерфейсу можна створювати розклад занять для певних груп з урахуванням підгруп, номером пари, днем тижня та типом занять, тощо.

У подальшому можна розширити функціональні можливості інформаційної системи, додаванням генерування нових звітів за вимогами користувачів, та автоматизованого процесу находження помилок (наприклад, відсутності якоїсь інформації чи «накладань» у розкладі).

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Сікора, О. С., and Р. Я. Макогін. "Система «електронний розклад занять» як складова віртуального освітнього середовища закладу освіти." Матеріали Ⅷ науково-технічної конференції „Інформаційні моделі, системи та технології “ (2020): 182-182.
2. Середовище програмування C# [Електронний ресурс] - <http://surl.li/rbmuh>
3. Що таке Excel, зв’язок з С# [Електронний ресурс] - <http://surl.li/rbmus>
4. MySQL, що це та як? [Електронний ресурс] - <http://surl.li/rbmva>
5. EntityFramework [Електронний ресурс] - <https://learn.microsoft.com/ru-ru/ef/>
6. Pomelo.EntityFrameworkCore - [Електронний ресурс] <http://surl.li/rbmvq>
7. EPPplus [Електронний ресурс] - <https://habr.com/ru/articles/525492/>
8. WPF [Електронний ресурс] - <https://metanit.com/sharp/wpf/>

# **ДОДАТОК**

Лістинг коду можна здійснити за посиланням:

<https://drive.google.com/file/d/1K0whwFNtsXnS9DuiWXzB1D7kbbSytmhM/view?usp=sharing>

1. Відформатувати основний текст (задати параметри стилю «звичайний»):

1.1. Розмір основного шрифту – 14пт

1.2. Шрифт Times New Roman

1.3. Міжрядковий інтервал – полуторний

1.4. Вирівнювання по ширині сторінки

1.5. Відступ першого рядка абзацу (червоний рядок) – 1,25 см

1.6. Між абзацами додаткових відступів немає

2. Встановити поля сторінок

2.1. Ліворуч – 2,5 см

2.2. Праворуч – 1,5 см

2.3. Зверху – 2 см

2.4. Знизу – 2 см

1. Кожен розділ першого рівня повинен починатися з нової сторінки. Назва розділу першого рівня друкується великими літерами, крапка після номеру та в кінці не ставиться. Вирівнювання по центру, шрифт: розмір 14пт, напівжирний. Після назви порожній рядок. Для форматування заголовків 1-го рівня використати стиль «Заголовок 1».

2. Назва підрозділу друкується з абзацного відступу (1,25) літерами маленького регістру (крім першої великої), крапка після номеру та в кінці не ставиться. Вирівнювання по лівому краю, шрифт: розмір 14 пт, напівжирний (використати стиль «Заголовок 2»). Перед та після назви підрозділу – пробільний (порожній) рядок, потім друкується основний текст з абзацу. Нумерація підрозділів – подвійна: номер розділу – крапка – номер підрозділу у розділі. Наприклад, 1.2 (другий підрозділ першого розділу).

3. Номер сторінок поставити праворуч, зверху без кpaпки в кiнцi, Титульний аркуш включають до загальної нумерації сторінок, але номер на ньому не ставиться. Колонтитулів на титульному аркуші не має бути.

**ЗМІСТ (**нoмеp сторінки пpoстaвляють у пpaвoму веpхньoму кутi стopiнки без кpaпки в кiнцi, **Times New Roman, кеглем 14). – зробити автоматичний зміст за допомогою WORD интервал 1,5**

**Зверніть увагу!** Після номеру крапка не ставиться!

Кpaпку в кiнцi зaгoлoвку не стaвити!