Лабораторна робота №3

Тема: Створення макросів за допомогою VBA з використанням розгалужень, циклів, масивів у середовищі Excel.

Мета роботи: Створення макроси в додатках MS Office та вивчення мови програмування VBA (Visual Basic for Applications). Навчити створювати прості макроси для книг Excel та змінювати їх за допомогою редактора VBA.

Хід роботи:

Завдання: припустимо, що книга Excel містить списки на багатьох аркушах, і потрібно змінити ім'я кожного аркуша у відповідності із заголовком списку на цьому аркуші. Не кожен аркуш містить список, але якщо список ϵ , його заголовок знаходиться в комірці B1, а якщо списку нема ϵ , комірка B1 ϵ порожньою. Імена аркушів без списків не повинні змінюватися.

Створила макрос, який виконує це завдання, але за допомогою Google Apps Script, тому що я працювала у онлайн версії Google Таблині:

```
/** @OnlyCurrentDoc */
function renameSheetsByHeader() {
  var spreadsheet = SpreadsheetApp.getActiveSpreadsheet();
  var sheets = spreadsheet.getSheets();

  // Перебір кожного аркуша
  sheets.forEach(function(sheet) {
    var headerCell = sheet.getRange('B1').getValue(); // Отримання значення з
  комірки В1

  // Перевірка, чи комірка В1 не порожня
  if (headerCell) {
    var newName = headerCell;
    var i = 1;
```

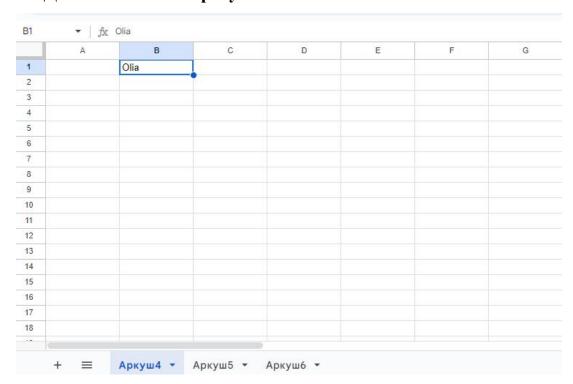
```
// Перевіряємо, чи існує аркуш із такою назвою
     while (sheetExists(newName)) {
       newName = headerCell + "_" + i; // Додаємо індекс до назви
     }
     try {
       sheet.setName(newName); // Зміна назви аркуша
     } catch (error) {
       Logger.log("Помилка при зміні назви аркуша " + sheet.getName() + ": " +
error.message);
     }
   }
 });
 // Додатково можна активувати комірку А1, якщо це необхідно
 var firstSheet = spreadsheet.getSheets()[0];
 firstSheet.getRange('A1').activate();
}
// Функція для перевірки, чи існує аркуш з вказаною назвою
function sheetExists(name) {
 var sheets = SpreadsheetApp.getActiveSpreadsheet().getSheets();
 for (var i = 0; i < sheets.length; i++) {</pre>
   if (sheets[i].getName() == name) {
     return true;
 }
 return false;
```

Журнал виконання

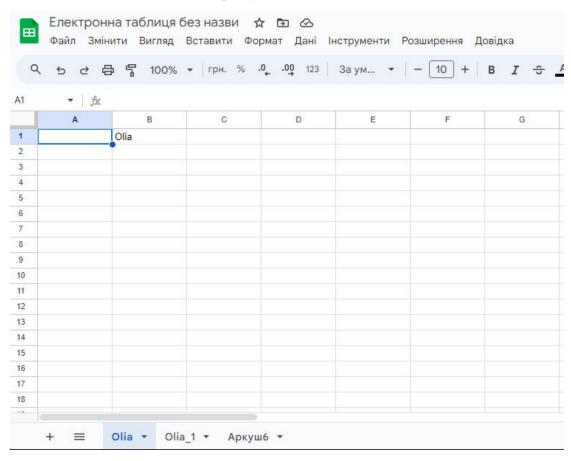
14:08:39	Примітка	Виконання почато
14:08:41	Примітка	Виконання завершено

Так як у двох аркушах в комірці В1 однаково було записано "Olia", то перейменувався 1 аркуш на Olia, а другий, відповідно, на Olia_1.

До виконання макросу:



Після виконання макросу:



Ця лабораторна робота була присвячена створенню макросів у середовищі Excel за допомогою мови програмування VBA (Visual Basic for Applications). Основна мета роботи полягає в тому, щоб навчити студентів автоматизувати процеси в Excel, зокрема перейменування аркушів на основі вмісту комірок. Робота знайомить із базовими принципами програмування в VBA, такими як використання циклів, розгалужень і обробки умов. Але так як у мене немає Excel на комп'ютері, я використала Google Apps Script.

Щодо вдосконалення роботи:

- 1. Оптимізація коду для уникнення помилок з іменами: поточний код працює, але можна його вдосконалити, додаючи обробку можливих конфліктів у назвах більш ефективно. Наприклад, можна використовувати функцію перевірки наявності імені до його присвоєння, щоб уникнути циклічного створення унікальних імен.
- 2. Валідація вмісту комірок: замість простого перевірки на порожнечу, можна впровадити додаткові умови, які перевірятимуть, чи вміст у комірці В1 є коректною назвою. Це дозволить уникнути ситуацій, коли в заголовках можуть бути недійсні символи, неприпустимі для назв аркушів.
- 3. Додаткове повідомлення про помилки: можна вдосконалити макрос, додавши обробку помилок, яка б відображала користувачу, які аркуші не вдалося перейменувати, і чому (наприклад, через вже існуюче ім'я чи неправильний формат).
- 4. *Можливість вибору діапазону для заголовка:* можна зробити макрос більш гнучким, дозволивши користувачеві обирати, з якого діапазону (комірок) брати заголовки для перейменування

аркушів. Це дозволить використовувати макрос у різноманітних сценаріях.

Загалом, ця лабораторна робота пропонує базову, але важливу задачу автоматизації в Excel, але її можна вдосконалити шляхом додавання більш складних перевірок і гнучкості у використанні макросів, що дозволить зробити процес більш універсальним і надійним.