Java Coding Exercise: File Reading and Writing

Problem Description:

You are required to create a Java program to manage a simple student database using a text file (`students.txt`). The program should perform the following tasks:

1. **Write student data to a file**: The program should write a list of students with their details (Name, Enrollment Year, and Major) to a file named `students.txt`.

2. **Read student data from the file**: The program should read the student data from the `students.txt` file and display it on the console.

3. **Extract and manipulate data from the file**: After reading the data, the program should extract the details of each student, calculate their study duration based on the current year (assume the current year is 2024), and display the extracted information and the calculated duration.

Tasks:

1. **Writing to File**:

   - Create a `students.txt` file.

   - Write the following data into the file using a `FileWriter`:

```yaml
Emre, 2003, Computer Science
John, 2004, Physics
Alice, 2005, Mathematics
Mary, 2006, Biology
George, 2002, Chemistry
```

   - Ensure the file is properly closed after writing to avoid any resource leaks.

2. **Reading from File**:

   - Read the content of the `students.txt` file using a `Scanner`.

   - Print each line of the file to the console.

3. **Extracting Data from File**:

   - After reading the data, split each line by commas to extract the student's name, enrollment year, and major.

   - Calculate the number of years each student has been studying up to the current year (2024).

   - Display the extracted information in the following format:

```yaml
Name: Emre, Year: 2003, Major: Computer Science, Duration: 21 years
Name: John, Year: 2004, Major: Physics, Duration: 20 years
Name: Alice, Year: 2005, Major: Mathematics, Duration: 19 years
Name: Mary, Year: 2006, Major: Biology, Duration: 18 years
Name: George, Year: 2002, Major: Chemistry, Duration: 22 years
```

4. **Bonus Task** (Optional):

   - Create a method to add new students to the `students.txt` file without overwriting the existing data. Prompt the user to input new student details (Name, Enrollment Year, and Major) and append this data to the file.

   - Ensure the program can read the updated file and display the new content with the calculated duration.

Implementation Guidelines:

- Use `File`, `FileWriter`, and `Scanner` classes for file operations.

- Handle exceptions appropriately using `try-catch` blocks.

- Ensure all resources (like files and scanners) are properly closed after their operations are completed.

- Use String manipulation methods such as `split()` to process each line of data.

Expected Output:

After completing the tasks, the program should display:

1. Confirmation of data written to the file.

2. The content of the file read and displayed line-by-line.

3. Extracted and formatted student details with calculated durations.

4. If the bonus task is implemented, prompt for new student details, append to the file, and display the updated data.

Hints:

- To write data to the file, use `FileWriter` with `append` mode set to `true` if implementing the bonus task.

- Use `split(", ")` to extract parts of the student data line.

- To calculate the study duration, subtract the enrollment year from 2024.

- Use the `nextLine()` method of `Scanner` to read each line of the file.