1. **Database Class:**
    - **addRecords**: Allows the user to add student records to the file.
    - **readRecords**: Reads and displays all records from the file.
    - **searchRecords**: Searches for a student record based on the student ID.
    - **deleteRecords**: Deletes a student record based on the student name.
    - **updateRecords**: Updates the marks of a student based on the student name.
    - **clear**: Clears all records from the file.
2. **Filehandling Class (main class):**
    - Creates an instance of the **Database** class (**f**).
    - Presents a menu to the user with options to perform different operations on student records.
    - Reads the user's choice and invokes the corresponding method from the **Database** class.
3. **Menu Options:**
    - **1: Add Records:** Adds student records to the file.
    - **2: Display Records:** Displays all student records from the file.
    - **3: Clear All Records:** Clears all records from the file.
    - **4: Search Records:** Searches for a student record based on the student ID.
    - **5: Delete Records:** Deletes a student record based on the student name.
    - **6: Update Records:** Updates the marks of a student based on the student name.
    - **7: Exit:** Exits the program.
4. **Infinite Loop:**
    - The program runs in an infinite loop until the user chooses to exit (option 7).
5. **Exception Handling:**
    - The program handles **FileNotFoundException** in case the file is not found.

Note: The program uses a simple text file (**sample.txt**) to store student records. Make sure the file is present in the correct location before running the program. Additionally, there's a potential issue with the way the program updates records, as it involves file deletion and renaming. It's recommended to use a more straightforward approach, such as reading all records, updating in memory, and then writing back to the file.