**Module 5: Critical Thinking**

**Hash Table**

Ryan Thompson

Colorado State University - Global

CSC 506

Dr. Holbert

18 September 2024

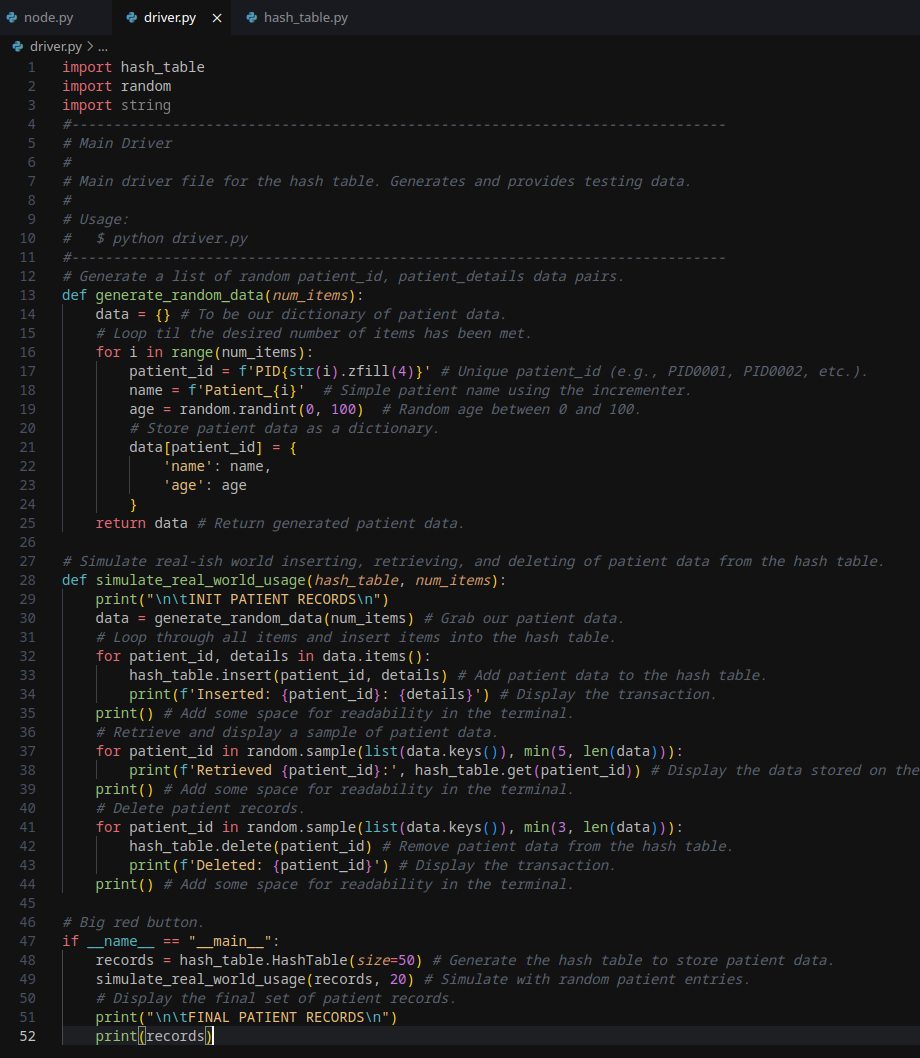
**Hash Table Exploration**

The primary purpose of the hash table implementation discussed in this project is to efficiently manage key-value pairs, specifically simulating a hospital patient record system. Hash tables are essential in computer science due to their average-case constant time complexity for operations like insertion, deletion, and retrieval. In this implementation, patient IDs serve as unique keys linked to patient details, facilitating quick access and management of records.

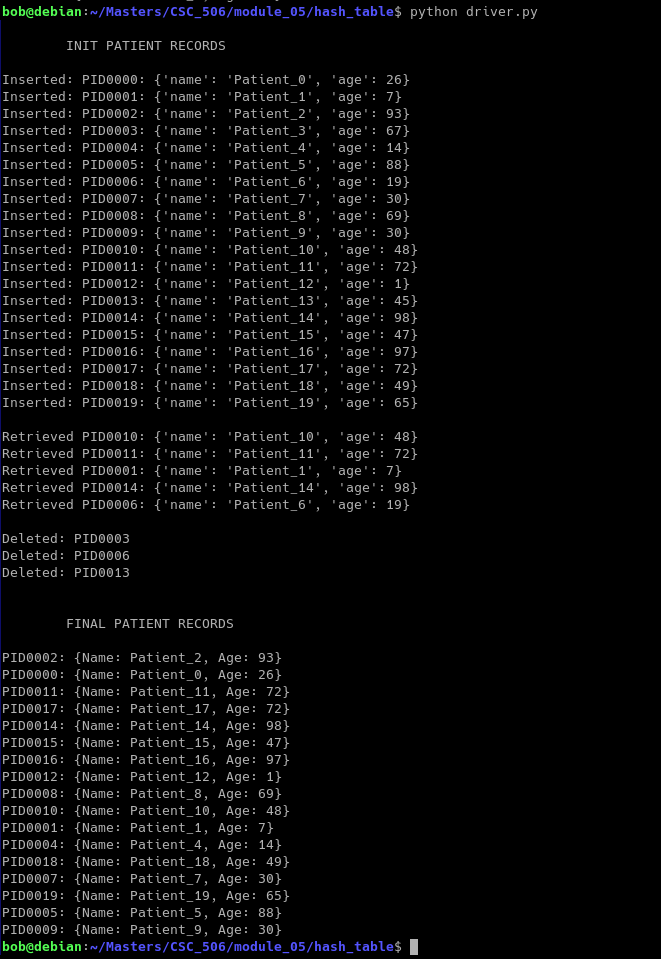
Throughout the development process, I encountered several challenges. A significant obstacle was ensuring an effective collision resolution strategy. Collisions occur when multiple keys hash to the same index, which can lead to data loss. I addressed this by implementing separate chaining, allowing linked lists to store multiple entries at the same index. Additionally, I faced difficulties in formatting the output for clarity. Initially, the hash table representation was cluttered, but by organizing entries by buckets and labeling them clearly, I improved the output's readability.

This project helped me acquire valuable skills beyond basic programming. I gained a deeper understanding of data structures, particularly hash tables and collision resolution techniques, which are critical for designing efficient algorithms. I also enhanced my problem-solving abilities, learning to debug and optimize code effectively. Moreover, my proficiency in Python and object-oriented programming improved as I designed classes to encapsulate the functionalities of the hash table and its nodes.

In summary, this hash table implementation serves as a practical example of efficient data management in a simulated hospital setting. The challenges faced during development, such as collision handling and output formatting, provided meaningful learning experiences. Ultimately, the skills acquired will be beneficial for future endeavors in computer science and software engineering.

**

*Main Driver Source Code*

*Program Output*

**References**

Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2009). Introduction to Algorithms (3rd ed.). MIT Press.

Knuth, D. E. (1998). The Art of Computer Programming, Volume 3: Sorting and Searching (2nd ed.). Addison-Wesley.

Sedgewick, R., & Wayne, K. (2011). Algorithms (4th ed.). Addison-Wesley.