[**https://github.com/sosten6/CAPSTONE-PROJECT**](https://github.com/sosten6/CAPSTONE-PROJECT)

**Explanation of how to use the code:**

**1. Initialization:**

- The program starts by initializing bus routes and an employee account (admin) in the `main` function.

- There are two types of data saved: bus routes and reservations. The data is saved to and loaded from files ("routes.txt" and "reservations.txt") to maintain data persistence.

**2. Login:**

- The user is prompted to log in with a username and password.

- The default employee account is, "**sosten**," with the password **"62sos35ten**."Another employee account is, "admin" with the password "1234". Another employee account

- The **`\_getch**` function is used for password **input to enhance security** by **not** displaying the password on the screen.

**3. Menu:**

- After login, the program displays a menu with various options.

- Options 1-7 are available to all users, while options 8-14 are reserved for employees (admin).

- The user can choose an option by entering the corresponding number.

**4. User Options (1-7):**

- Options such as displaying routes, adding routes, deleting routes, viewing available seats, reserving seats, canceling reservations, and displaying booking history are available to all users.

**5. Employee Options (8-14):**

- These options are available only to employees who have logged in.

- Option 8 saves data to files, and option 9 loads data from files to ensure data persistence.

- Option 10 logs out the current employee.

- Option 11 exits the program.

- Option 12 deletes all data (routes, reservations, and employee accounts).

- Options 13 and 14 allow employees to add and delete employee accounts, respectively.

**6. Functions:**

- The code is organized with various functions to perform specific tasks, enhancing readability and maintainability.

- Functions include adding and deleting routes, displaying routes, handling employee login, creating employee accounts, calculating payment, validating payment secrets, securely inputting passwords, displaying available seats, reserving seats, canceling reservations, displaying booking history, saving and loading data from files, and adding/deleting employee accounts.

**7. Data Persistence**:

- The system saves and loads data from files, providing a way to maintain information between different program runs.

**8. Security Considerations:**

- Password input is masked using asterisks for security.

- M-Pesa payment secrets are validated for a specific length***.(any 6 digits or letters lenth is acceptable)***

**9. Looping:**

- The program runs in an infinite loop (`while (1)`) until the user chooses to exit.

**10. Exiting:**

- The program exits with a message when the user selects the exit option.

**11. Error Handling**:

- The code includes basic error handling, such as displaying warnings if data loading from files fails.

***To use the code:***

1. Compile the code using a C compiler (e.g., GCC).
2. Run the compiled executable.
3. Follow the prompts to interact with the Bus Reservation System.