`GetAttemptedQuiz` takes a `loginid` and a `connectionString` as input parameters.

- It checks if the `loginid` value is null or empty. If it is, it returns an error message using the `ErrorHandler` routine.

- It establishes a connection to the PostgreSQL database using the provided `connectionString` and opens the connection.

- The SQL statement is defined to retrieve attempted quiz data. It selects the `question\_id`, `question\_text`, `option\_name`, and `option\_text` columns from the `quiz\_questions` and `quiz\_options` tables.

- The `INNER JOIN` is used to join the `quiz\_questions` and `quiz\_options` tables based on the `question\_id`.

- The `WHERE` clause filters the result set to include only the rows where the `question\_id` is present in the subquery. The subquery selects `question\_id` from the `quiz\_history` table where the `login\_id` matches the provided `loginid` using the `ILIKE` operator for case-insensitive comparison.

- The `sqlstatement` is passed to a new `NpgsqlCommand` object along with the `connection`.

- The `@loginid` parameter is added to the command and its value is set to the `loginid` parameter passed to the function.

- The `FormatQuizToJson` method is called with the command to execute the query and return the output in JSON format.

- The JSON result is returned from the function.

Overall, this code retrieves attempted quiz data from the database based on the provided `loginid` by executing the SQL statement and formatting the result into JSON format using the `FormatQuizToJson` method.