**CustomSymbolTable Documentation**

*The `CustomSymbolTable` class is designed to create a symbol table using custom hash functions for efficient storage and retrieval of key-value pairs*

**Attributes**

- `**capacity**`: An integer representing the capacity of the symbol table, which is set to a default value of 13 if not specified during initialization.

- `**buckets**`: A list that consists of "buckets" where table elements are stored. The buckets are used to organize key-value pairs.

- `**size**`: An integer representing the current length of the table, which is the number of elements currently stored in the symbol table.

**Methods**

**`\_hash(key)`**

- Input: `key` - The key for which the hash code will be computed.

- Output: The hash code corresponding to the given `key`.

The `\_hash` method computes a custom hash code for a given key. This custom hash code is used to determine the index within the symbol table where the key-value pair will be stored.

**`add(key)`**

- Input: `key` - The key to be added to the symbol table.

- Output: A tuple containing the index and position of the added key within the symbol table.

The `add` method is used to add a key to the symbol table. If the key already exists in the table, it returns the index and position of the existing key. If the key is not found, it computes the hash code for the key and appends it to the appropriate bucket, returning the index and position where it was added.

**`exists(key)`**

- Input: `key` - The key to be checked for existence in the symbol table.

- Output: `True` if the key exists in the table, `False` otherwise.

The `exists` method checks whether a given key already exists in the symbol table. It returns `True` if the key is found and `False` if the key is not present in the table.

**`\_find(key)`**

- Input: `key` - The key to be found within the symbol table.

- Output: A tuple containing the index and position of the key if found, or (-1, -1) if the key is not in the table.

The `\_find` method searches for a given key within the symbol table. If the key is found, it returns a tuple containing the index and position of the key within the table. If the key is not found, it returns (-1, -1).

**`\_\_str\_\_()`**

- Output: A string representation of the symbol table.

The `\_\_str\_\_` method provides a string representation of the symbol table, showing the index-to-bucket mapping and the contents of each bucket in the table.