The function that we will be using for the survey

1. From dataset compat.py from line 89 to 112 (as\_text)

A computer code on a black background

AI-generated content may be incorrect.

Human-generated: Converts any string-like python input types to unicode.

Returns the input as a unicode string. Uses utf-8 encoding for text

by default.

Args: bytes\_or\_text: A `bytes`, `str`, or `unicode` object.

encoding: A string indicating the charset for decoding unicode.

Returns: A `unicode` (Python 2) or `str` (Python 3) object.

Raises: TypeError: If `bytes\_or\_text` is not a binary or unicode string.

LLM-generated: \*\*as\_text\*\*: The function of as\_text is to convert any string-like Python input types to a unicode string.

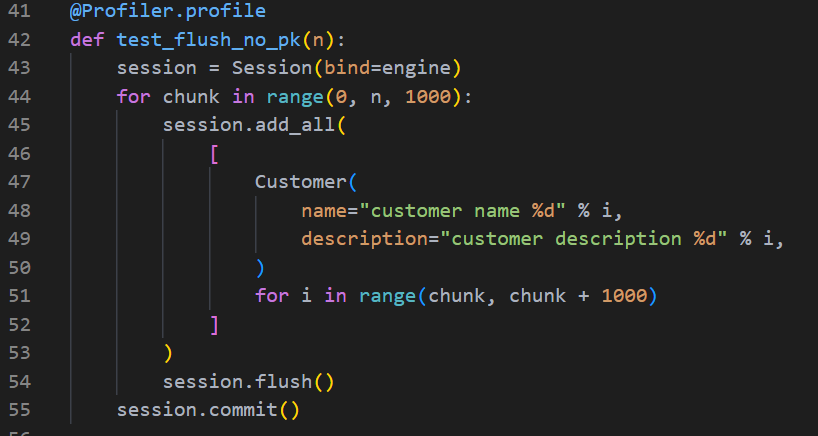
\*\*parameters\*\*: The parameters of this Function.

· bytes\_or\_text: A `bytes`, `str`, or `unicode` object that needs to be converted to a unicode string.

· encoding: A string indicating the charset for decoding unicode. The default value is 'utf-8'.

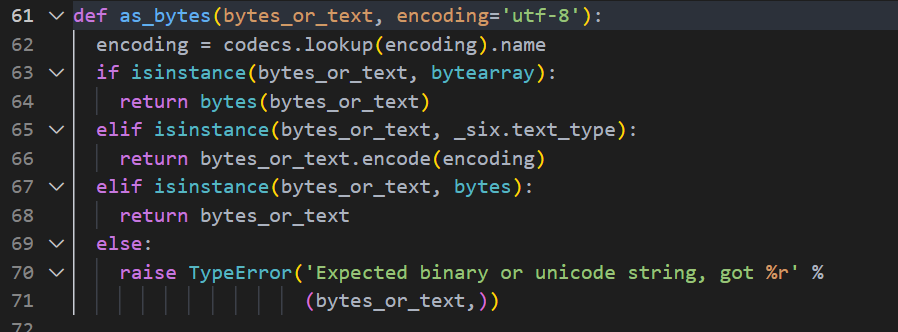
\*\*Note\*\*:

- Ensure that the input `bytes\_or\_text` is either a `bytes`, `str`, or `unicode` object to avoid raising a `TypeError`.

1. From dataset bulk\_inserts.py from line 42 to 57 (test\_flush\_no\_pk)

Human-generated: Insert statements via the ORM (batched with RETURNING if available), fetching generated row id

LLM-generated: The function of test\_flush\_no\_pk is to demonstrate the insertion of multiple customer records into a database using the ORM (Object-Relational Mapping) in batches, without specifying primary keys, and fetching generated row IDs if available.

1.  From dataset compat.py from line 61 to 86 (as\_bytes)

Human-generated: Converts `bytearray`, `bytes`, or unicode python input types to `bytes`.

Uses utf-8 encoding for text by default.

Args: bytes\_or\_text: A `bytearray`, `bytes`, `str`, or `unicode` object.

encoding: A string indicating the charset for encoding unicode.

Returns: A `bytes` object.

Raises: TypeError: If `bytes\_or\_text` is not a binary or unicode string.

LLM-generated: \*\*as\_bytes\*\*: The function of as\_bytes is to convert various Python input types, such as `bytearray`, `bytes`, or unicode strings, into a `bytes` object.

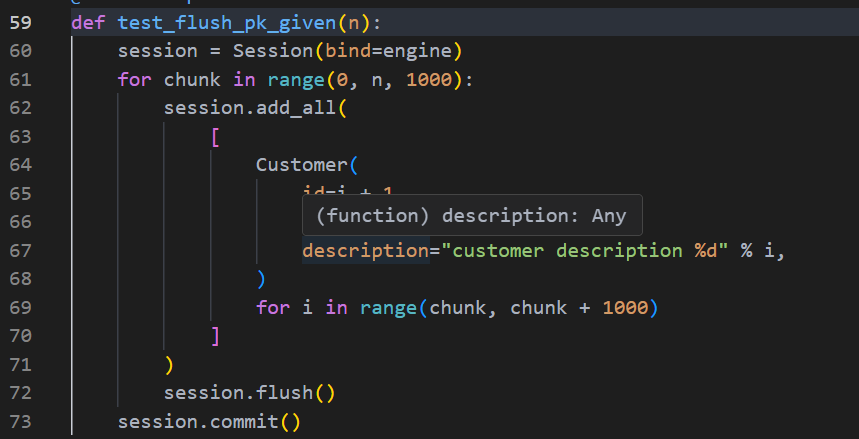
\*\*parameters\*\*: The parameters of this Function.

· bytes\_or\_text: A `bytearray`, `bytes`, `str`, or `unicode` object that needs to be converted to `bytes`.

· encoding: A string indicating the charset for encoding unicode. The default value is 'utf-8'.

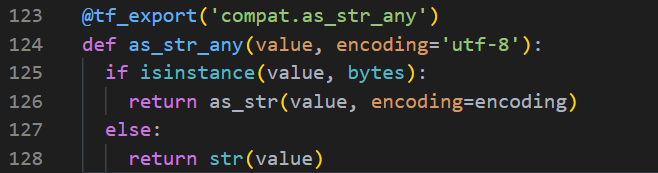
\*\*Note\*\*:

- Ensure that the input to `as\_bytes` is either a `bytearray`, `bytes`, or a unicode string. Providing an unsupported type will result in a `TypeError`.

1. From dataset bulk\_inserts.py from line 59 to 47 (test\_flush\_pk\_given)

Human-generated: Batched INSERT statements via the ORM, PKs already defined

LLM-generated: The function of test\_flush\_pk\_given is to demonstrate batched INSERT operations using the ORM (Object-Relational Mapping) with predefined primary keys (PKs) for the `Customer` records.

1. From dataset compat.py from line 124 to 140 (as\_str\_any)

Human-generated: Converts input to `str` type.

Uses `str(value)`, except for `bytes` typed inputs, which are converted

using `as\_str`.

Args: value: A object that can be converted to `str`.

encoding: Encoding for `bytes` typed inputs.

Returns: A `str` object.

LLM-generated: \*\*as\_str\_any\*\*: The function of as\_str\_any is to convert any input object to a `str` type, handling `bytes` inputs with a specific encoding.

\*\*parameters\*\*: The parameters of this Function.

· value: An object that can be converted to `str`. This can be of any type, including `bytes`, `str`, or other objects that support the `str()` conversion.

· encoding: A string indicating the charset for decoding `bytes` typed inputs. The default value is 'utf-8'.