
SqlAlchemy Diff Documentation

Release 0.1.0

Student.com

Mar 03, 2021

Contents

1	PyTest Example	3
2	Features	5
3	Installation	7

Compare and generate a diff between two databases using SQLAlchemy's inspection API

CHAPTER 1

PyTest Example

Comparing two schemas is easy. You can verify they are the same like this:

```
>>> result = compare(uri_left, uri_right)
>>> result.is_match
True
```

When they are different, `result.is_match` will be `False`.

When two schemas don't match, you can inspect the differences between them by looking at the `errors` dict on the `result`:

```
>>> result = compare(uri_left, uri_right)
>>> result.is_match
False
>>> result.errors
{
    'tables': {
        'left_only': ['addresses'],
        'right_only': ['roles']
    },
    'tables_data': {
        'employees': {
            'columns': {
                'left_only': [
                    {
                        'default': None,
                        'name': 'favourite_meal',
                        'nullable': False,
                        'type': "ENUM('meat', 'vegan') "
                    }
                ],
                'right_only': [
                    {
                        'autoincrement': False,
```

(continues on next page)

(continued from previous page)

```

        'default': None,
        'name': 'role_id',
        'nullable': False,
        'type': 'INTEGER(11)'
    },
    {
        'autoincrement': False,
        'default': None,
        'name': 'number_of_pets',
        'nullable': False,
        'type': 'INTEGER(11)'
    },
]
},
'foreign_keys': { ... },
'primary_keys': { ... },
'indexes': { .. }
},
'phone_numbers': { ... }
},
'uris': {
    'left': "your left URI",
    'right': "your right URI",
}
}

```

You can also tell the comparer to ignore parts of the database.

For example, to ignore the users table completely, and the line3 column of the table addresses you would call compare like this:

```

>>> result = compare(
    uri_left,
    uri_right,
    ignores=['users', 'addresses.col.line3']
)

```

If you wish to persist that dict to a JSON file, you can quickly do so by calling `result.dump_errors()`.

Currently the library can detect the following differences:

- Differences in **Tables**
- Differences in **Primary Keys** for a common table
- Differences in **Foreign Keys** for a common table
- Differences in **Indexes** for a common table
- Differences in **Columns** for a common table
- Ability to ignore a **whole table**
- Ability to ignore **primary/foreign keys, indexes and columns**

CHAPTER 3

Installation

```
$ pip install sqlalchemy-diff
```