
reprit

Release 0.7.0

Azat Ibrakov

Apr 18, 2022

CONTENTS

Python Module Index	3
Index	5

Note: If object is not listed in documentation it should be considered as implementation detail that can change and should not be relied upon.

```
replit.base.generate_repr(method: typing.Union[typing.Callable[[...], replit.core.hints.Domain],
                          typing.Callable[[...], None]], *, argument_serializer:
                          typing.Callable[[typing.Any], str] = <built-in function repr>, field_seeker:
                          typing.Callable[[replit.core.hints.Domain, str], typing.Any] = <built-in function
                          getattr>, prefer_keyword: bool = False, skip_defaults: bool = False,
                          with_module_name: bool = False) → Callable[[replit.core.hints.Domain], str]
```

Generates `__repr__` method based on constructor/initializer parameters.

We are assuming that no parameters data get thrown away during instance creation, so we can re-create it after.

Parameters

- **method** – constructor/initializer method which parameters will be used in resulting representation.
- **argument_serializer** – function that serializes argument to string.
- **field_seeker** – function that re-creates parameter value based on class instance and name.
- **prefer_keyword** – flag that specifies if positional-or-keyword parameters should be outputted as keyword ones when possible.
- **skip_defaults** – flag that specifies if optional parameters with default arguments should be skipped.
- **with_module_name** – flag that specifies if module name should be added.

```
>>> from replit.base import generate_repr
>>> class Person:
...     def __init__(self, name, *, address=None):
...         self.name = name
...         self.address = address
...         __repr__ = generate_repr(__init__,
...                                   skip_defaults=True)
>>> Person('Adrian')
Person('Adrian')
>>> Person('Mary', address='Somewhere on Earth')
Person('Mary', address='Somewhere on Earth')
>>> class ScoreBoard:
...     def __init__(self, first, *rest):
...         self.first = first
...         self.rest = rest
...         __repr__ = generate_repr(__init__,
...                                   prefer_keyword=True)
>>> ScoreBoard(1)
ScoreBoard(first=1)
>>> ScoreBoard(1, 40)
ScoreBoard(1, 40)
>>> class Student:
...     def __init__(self, name, group):
...         self.name = name
...         self.group = group
```

(continues on next page)

(continued from previous page)

```
...     __repr__ = generate_repr(__init__,
...                               with_module_name=True)
>>> Student('Kira', 132)
replit.base.Student('Kira', 132)
>>> Student('Naomi', 248)
replit.base.Student('Naomi', 248)
>>> from replit import seekers
>>> class Account:
...     def __init__(self, id_, *, balance=0):
...         self.id = id_
...         self.balance = balance
...     __repr__ = generate_repr(__init__,
...                               field_seeker=seekers.complex_)
>>> Account(1)
Account(1, balance=0)
>>> Account(100, balance=-10)
Account(100, balance=-10)
>>> import json
>>> class Object:
...     def __init__(self, value):
...         self.value = value
...     def serialized(self):
...         return json.dumps(self.value)
...     @classmethod
...     def from_serialized(cls, serialized):
...         return cls(json.loads(serialized))
...     __repr__ = generate_repr(from_serialized)
>>> Object.from_serialized('0')
Object.from_serialized('0')
>>> Object.from_serialized({'key': 'value'})
Object.from_serialized({'key': 'value'})
```

PYTHON MODULE INDEX

r

`reprit.base`, [1](#)

INDEX

G

`generate_repr()` (*in module `reprit.base`*), 1

M

module

`reprit.base`, 1

R

`reprit.base`

module, 1