rebus Documentation

Release 0.2

Artur Barseghyan <artur.barseghyan@gmail.com>

December 20, 2013

Contents

Generate base64-encoded strings consisting of alphanumeric symbols only.

CHAPTER 1

Why did I make this app?

Recently I have been working on implementing of a Google Authenticator app for Plone (which I did for my beloved company - Goldmund, Wyldebeast & Wunderliebe).

For generating of a bar code image, I needed a base32 encoded string. While Android devices could perfectly scan all bar-code image that I would generate, Apple devices would raise errors on bar code images which were generated using strings that contain one or more "=" characters.

The solution found was to add a number of \n at the end of the string to be encoded.

If you happen to experience similar problems, you know what to do.

CHAPTER 2

Prerequisites

- Python 2.6.8+, 2.7.+, 3.3.+

CHAPTER 3

Installation

Install latest stable version from PyPI:

```
$ pip install rebus
```

...or latest stable version from GitHub:

```
$ pip install -e git+https://github.com/barseghyanartur/rebus@stable#egg=rebus
```

...or latest stable version from BitBucket:

```
$ pip install -e hg+https://bitbucket.org/barseghyanartur/rebus@stable#egg=rebus
```

CHAPTER 4

Usage and examples

Using *rebus* is damn easy. Whenever you would want to use `base64.b32encode` or `base64.b64encode`, replace *base64* with *rebus*.

4.1 Encoding

Required imports

```
>>> import rebus
```

b32encode string

```
>>> rebus.b32encode('abcdefg')
'MFRGGZDFMZTQUCQK'
```

b64encode string

```
>>> rebus.b64encode('abcdefg')
'YWJjZGVmZwoK'
```

urlsafe_b64encode

```
>>> rebus.urlsafe_b64encode('abcdefg')
'YWJjZGVmZwoK'
```

4.2 Decoding

It's possible to decode string, encoded with *rebus* to their original values.

Required imports

```
>>> import rebus
```

b32decode string

```
>>> rebus.b32decode('MFRGGZDFMZTQUCQK')
'abcdefg'
```

b64decode string

```
>>> rebus.b64decode('YWJjZGVmZwoK')
'abcdefg'
```

urlsafe_b64encode

```
>>> rebus.urlsafe_b64decode('YWJjZGVmZwoK')
'abcdefg'
```

CHAPTER 5

License

GPL 2.0/LGPL 2.1

CHAPTER 6

Support

For any issues contact me at the e-mail given in the *Author* section.

CHAPTER 7

Author

Artur Barseghyan <artur.barseghyan@gmail.com>

CHAPTER 8

Documentation!

Contents:

8.1 rebus package

8.1.1 Module contents

`rebus.encode`(*encoder*, *step*, *text*, *return_object=False*)

Parameters

- **encoder** (*callable*) –
- **step** (*int*) –
- **text** (*string*) –
- **return_object** (*bool*) –

Return string

`rebus.b32encode`(*text*, *return_object=False*)

Parameters

- **text** (*string*) –
- **return_object** (*bool*) –

Return string

`rebus.b64encode`(*text*, *return_object=False*)

Parameters

- **text** (*string*) –
- **return_object** (*bool*) –

Return string

`rebus.urlsafe_b64encode`(*text*, *return_object=False*)

Parameters

- **text** (*string*) –
- **return_object** (*bool*) –

Return string

`rebus.`**`decode`**(*decoder*, *text*, *return_object=False*)

Parameters

- **decoder** (*callable*) –
- **text** (*string*) –
- **return_object** (*bool*) –

Return string

`rebus.`**`b32decode`**(*text*, *return_object=False*)

Parameters

- **text** (*string*) –
- **return_object** (*bool*) –

Return string

`rebus.`**`b64decode`**(*text*, *return_object=False*)

Parameters

- **text** (*string*) –
- **return_object** (*bool*) –

Return string

`rebus.`**`urlsafe_b64decode`**(*text*, *return_object=False*)

Parameters

- **text** (*string*) –
- **return_object** (*bool*) –

Return string

CHAPTER 9

Indices and tables

- *genindex*
- *modindex*
- *search*

Python Module Index

r

rebus, **??**