pyquickcheck Documentation

Release 0.1-11-gf1c1f5d

Aaron Griffith

January 18, 2014

Contents

A Python port of Haskell's QuickCheck.

CHAPTER 1

Quickstart Guide

(TODO)

CHAPTER 2

Integrating with *pyquickcheck*

CHAPTER 3

API Reference

CHAPTER 4

Introduction

The *pyquickcheck* module generates random values for registered types. It comes with a bunch of random generators for built-in python types.

```
>>> import quickcheck as qc
>>> qc.arbitrary(int)
-11513
```

You can use the *quickcheck* decorator with function annotations to automatically generate and run your function with many random values.

```
>>> @qc.quickcheck(tries=3)
... def testfunc(a: int, b: str):
...     print(a, b)
...     return True
...
>>> testfunc()
0
-1 y
0 *
```

Usually, this is used to decorate test functions. If *pyquickcheck* generates a set of values that cause your function to raise an exception, it will automatically try to minimize these values before reporting them to you. For a more detailed introduction, see the *Quickstart Guide*.

CHAPTER 5

Indices and tables

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