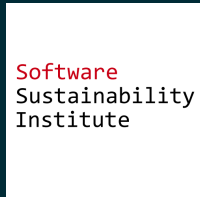
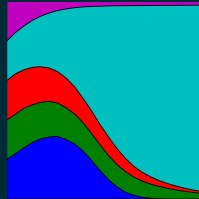


Writing tests for research software

@NikoletaGlyn





IMPORTANCE OF TESTING

USE
97%

DEVELOP
56%

TRAINING
69%

IMPOSSIBLE
79%

<https://www.software.ac.uk/blog/2016-09-12-its-impossible-conduct-research-without-software-say-7-out-10-uk-researchers>

'main.py'

```
def absolute_value(x):  
    if x < 0:  
        return -x  
    else:  
        return x
```

'test_main.py'

```
import unittest  
import main  
  
class TestExample(unittest.TestCase):  
  
    def test_absolute(self):  
        self.assertEqual(main.absolute_value(-2), 2)  
        self.assertTrue(main.absolute_value(-2), 2)  
        self.assertEqual(main.absolute_value(-3),  
                           main.absolute_value(3))  
        self.assertTrue(main.absolute_value(-2) >= 0)
```

```
python -m unittest test_main.py
```



```
python -m unittest test_main.py
```

```
-----
```

```
Ran 1 test in 0.000s
```

```
OK
```


'main.py'

```
import unittest

def absolute_value(x):
    """
    Return the absolute value of an argument.

    For example:

        >>> absolute_value(-2)
        2
        >>> absolute_value(3)
        3
    """
    if x < 0:
        return -x
    else:
        return x
```

'main.py'

```
import unittest

def absolute_value(x):
    """
    Return the absolute value of an argument.

    For example:

        >>> absolute_value(-2)
        2
        >>> absolute_value(3)
        3
    """
    if x < 0:
        return -x
    else:
        return x
```

```
python -m doctest main.py
```

