```
import io
import re
import sys
import unittest

from test.information import *

from util import *
from ls import print_directory_title

class TestPrintDirectoryTitle(unittest.TestCase):
```

```
    def test_print_directory_title(self):
        """ Test the display of directories titles, without directory argument given to

        args = {
            "recursive": True,
            "directories": []
        }

        start_path = root

        for dir_info in directory_information:
            # Capture the standard output
            capturedOutput = io.StringIO()
            sys.stdout = capturedOutput
            print_directory_title(root, None, args)
            expectedPrint = Color.YELLOW + root + ":\n" + Color.ENDC
            self.assertEqual(capturedOutput.getvalue(), expectedPrint)

        sys.stdout = sys.__stdout__

    def test_print_directory_title_with_directories_given(self):
        """ Test the display of directories titles with directories given in arguments

        args = {
            "recursive": True,
            "directories": ["something so the directories not empty statement is true"]
        }

        start_path = root

        for dir_info in directory_information:
            # Capture the standard output
            capturedOutput = io.StringIO()
            sys.stdout = capturedOutput
            print_directory_title(dir_info["path"], start_path, args)
            expectedPrint = Color.YELLOW + "." + \
                re.sub(r'%s' % (start_path), '', root, 1) + ":" + Color.ENDC
        self.assertEqual(capturedOutput.getvalue(), expectedPrint)
```

if **name** == '**main**':

unittest.main()

sys.stdout = sys.**stdout**