main.py

def check(number):

return number % 2 == 0

test.py

import unittest

from main import check

class TestCheck(unittest.TestCase):

def test\_check(self):

self.assertTrue(check(2))

self.assertTrue(check(6))

self.assertTrue(check(220))

self.assertFalse(check(1))

self.assertFalse(check(3))

self.assertFalse(check(57))

if \_\_name\_\_ == '\_\_main\_\_':

unittest.main()

main.py

def divide(a, b):

if b == 0:

raise ValueError('На ноль делить нельзя')

return a/b

test.py

import unittest

from main import divide

class TestDivide(unittest.TestCase):

def test\_divide\_success(self):

self.assertEqual(divide(10, 2), 5)

self.assertEqual(divide(6, 3), 2)

self.assertEqual(divide(70, 2), 35)

def test\_divide\_by\_zero(self):

self.assertRaises(ValueError, divide, 6, 0)

if \_\_name\_\_ == '\_\_main\_\_':

unittest.main()