```
import unittest

class TestLoops(unittest.TestCase):
    def test_loops(self):
        # Using ``range``
        # Write variable ``nests`` that holds 0 to 5 (use the ``range`` function)
        # ================================
        nests = range(6)
        self.assertEquals(nests, [0,1,2,3,4,5])

        # ``range`` 2
        # Create variable ``sample_id`` that holds values from 3 to 11
        # ================================
        sample_id = range(3,12)
        self.assertEquals(sample_id, [3,4,5,6,7,8,9,10,11])

        # Write a function ``even`` that takes a list of
        # number and returns a list of even numbers
        # ================================
        def even(some_list):
            return_list = []
            for item in some_list:
                if item%2==0:
                    return_list.append(item)
            return return_list

        self.assertEquals(even(nests), [0, 2, 4])
        self.assertEquals(even(sample_id), [4, 6, 8, 10])

        # Write a function ``even_index`` that takes a list
        # of numbers and returns those that are in an even
        # index position (hint: enumerate)
        # ================================
        def even_index(some_list):
            return_list = []
            for index,value in enumerate(some_list):
                if index%2==0:
                    return_list.append(value)
            return return_list
        self.assertEquals(even_index(nests), [0, 2, 4])
        self.assertEquals(even_index(sample_id), [3, 5, 7, 9, 11])


if __name__ == '__main__':
    unittest.main()
```