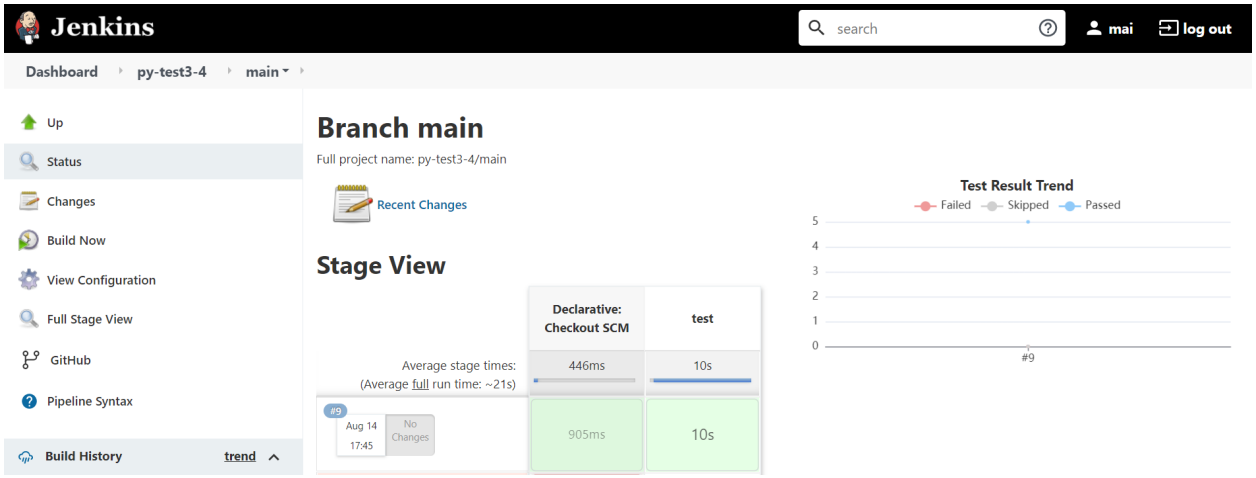
Maisha Ahmed
Deployment #3

First successful test build: It took me a long time to get my first successful build because I did not update my Jenkins plug-ins.



Added Feature: I added a subtraction function that takes in three arguments.

```
29
30  #The subtraction function
31  def subtract3(arg1, arg2, arg3):
32      arg1conv = conv(arg1)
33      arg2conv = conv(arg2)
34      arg3conv = conv(arg3)
35
36      if isinstance(arg1conv,str) or isinstance(arg2conv, str) or isinstance(arg3conv, str):
37          arg1conv = str(arg1conv)
38          arg2conv = str(arg2conv)
39          arg3conv = str(arg3conv)
40      return arg1conv - arg2conv - arg3conv
```

```
10
11  if argnumbers == 3 :
12      print("")
13      print("The result is " + str(calc.subtract3(str(sys.argv[1]), str(sys.argv[2]), str(sys.argv[3]))))
14      print("")
15      sys.exit(0)
16
17  elif argnumbers == 2:
18      print("")
19      print("The result is " + str(calc.add2(str(sys.argv[1]), str(sys.argv[2]))))
20      print("")
21      sys.exit(0)
22
23  elif argnumbers != 3 and argnumbers != 2:
24      print("")
25      print("You entered " + str(argnumbers) + " value/s.")
26      print("")
27      print("Usage: 'add2vals X Y' where X and Y are individual values. or 'subtract3vals' XYZ where xyz are individual values.")
28
29      print("       If add2vals is not in your path, usage is './add2vals X Y'.")
30      print("       If unbundled, usage is 'python add2vals.py X Y'.")
31      print("")
32      sys.exit(1)
```

Failed Tests

1. Subtract three floats: I passed a string (‘Maisha’) as one of the arguments and predictably Jenkins returned an error.

```
arg1 = '14.5', arg2 = '4.2', arg3 = 'Maisha'

    def subtract3(arg1, arg2, arg3):
        arg1conv = conv(arg1)
        arg2conv = conv(arg2)
        arg3conv = conv(arg3)

        if isinstance(arg1conv,str) or isinstance(arg2conv, str) or isinstance(arg3conv, str):
            arg1conv = str(arg1conv)
            arg2conv = str(arg2conv)
            arg3conv = str(arg3conv)
>       return arg1conv - arg2conv - arg3conv
E       TypeError: unsupported operand type(s) for -: 'str' and 'str'

sources/calc.py:40: TypeError
```

2. Subtract three integers: I passed two arguments instead of three and I received an error message letting me know that an argument is missing.

```
_____________________ TestCalc.test_subtract_integers _____________________

self = <test_calc.TestCalc testMethod=test_subtract_integers>

    def test_subtract_integers(self):
        """
        Test that the subtraction of three integers returns the correct total
        """
>       result = calc.subtract3(10, 2)
E       TypeError: subtract3() missing 1 required positional argument: 'arg3'
```

3. Unexpected error: I fixed the subtract_floats test by replacing the string argument with a float but I got a strange error. I calculated the result to be 10.1 but the program added a lot of zeroes. I admit that I was taken by surprise.

```
=================================== FAILURES ===================================
_____________________ TestCalc.test_subtract_floats ____________________

self = <test_calc.TestCalc testMethod=test_subtract_floats>

    def test_subtract_floats(self):
        """
        Test that the subtraction of three floats returns the correct result
        """
        result = calc.subtract3('14.5', '4.2', '0.2')
>       self.assertEqual(result, 10.1)
E       AssertionError: 10.100000000000001 != 10.1

sources/test_calc.py:59: AssertionError
- generated xml file: /var/lib/jenkins/workspace/py_test2_main/test-reports/results.xml -
============================ short test summary info ============================
FAILED sources/test_calc.py::TestCalc::test_subtract_floats - AssertionError:...
======================== 1 failed, 6 passed in 0.06s ========================
```

Test Success: I replaced one of the arguments so the result would be a whole number.

```
    def test_subtract_floats(self):
        """
        Test that the subtraction of three floats returns the correct result
        """
        result = calc.subtract3('14.5', '4.2', '0.3')
        self.assertEqual(result, 10.0)
```

```
============================= test session starts ==============================
platform linux -- Python 3.7.10, pytest-6.2.5, py-1.10.0, pluggy-1.0.0 --
/var/lib/jenkins/workspace/py_test2_main/test3/bin/python3
cachedir: .pytest_cache
rootdir: /var/lib/jenkins/workspace/py_test2_main
collecting ... collected 7 items

sources/test_calc.py::TestCalc::test_add_floats PASSED                   [ 14%]
sources/test_calc.py::TestCalc::test_add_integers PASSED                 [ 28%]
sources/test_calc.py::TestCalc::test_add_string_and_integer PASSED       [ 42%]
sources/test_calc.py::TestCalc::test_add_string_and_number PASSED        [ 57%]
sources/test_calc.py::TestCalc::test_add_strings PASSED                  [ 71%]
sources/test_calc.py::TestCalc::test_subtract_floats PASSED              [ 85%]
sources/test_calc.py::TestCalc::test_subtract_integers PASSED            [100%]

- generated xml file: /var/lib/jenkins/workspace/py_test2_main/test-reports/results.xml -
============================== 7 passed in 0.04s ===============================
```