

تست برنامه ماشین حساب

امین ملایی

درس آزمون نرم افزار

در این برنامه قصد داریم تا ماشین حساب قرار داده شده در آدرس <https://github.com/ahfarmer/calculator> را تست کنیم.

این برنامه با دامنه <https://ahfarmer.github.io/calculator> قابل اجراست و ابتدا آنرا با webdriver موجود در selenium باز میکنیم.

```
def setUp(self):
    # open Firefox browser
    self.driver = webdriver.Firefox()
    # set window size
    self.driver.set_window_rect(0, 0, 1540, 550)
    # delete the cookies
    self.driver.delete_all_cookies()
    # navigate to the url
    self.driver.get('https://ahfarmer.github.io/calculator/')
```

برنامه ذکر شده شامل اعداد 0-9 و عملگر های ریاضی و جواب نهایی است.

ابتدا دکمه مربوط به اعداد و عملگر ها و فیلد جواب را ذخیره میکنیم.

لیست numbers به ترتیب اعداد 0-9 را شامل میشود.

```
self.result = self.driver.find_element(By.XPATH, '/html/body/div/div/div[1]/div')
self.numbers = [
    self.driver.find_element(By.XPATH,
        '/html/body/div/div/div[2]/div[5]/div[1]/button'),
    self.driver.find_element(By.XPATH,
        '/html/body/div/div/div[2]/div[4]/div[1]/button'),
    self.driver.find_element(By.XPATH,
        '/html/body/div/div/div[2]/div[4]/div[2]/button'),
    self.driver.find_element(By.XPATH,
        '/html/body/div/div/div[2]/div[4]/div[3]/button'),
    self.driver.find_element(By.XPATH,
        '/html/body/div/div/div[2]/div[3]/div[1]/button'),
    self.driver.find_element(By.XPATH,
        '/html/body/div/div/div[2]/div[3]/div[2]/button'),
    self.driver.find_element(By.XPATH,
        '/html/body/div/div/div[2]/div[3]/div[3]/button'),
    self.driver.find_element(By.XPATH,
        '/html/body/div/div/div[2]/div[2]/div[1]/button'),
    self.driver.find_element(By.XPATH,
        '/html/body/div/div/div[2]/div[2]/div[2]/button'),
    self.driver.find_element(By.XPATH,
        '/html/body/div/div/div[2]/div[2]/div[3]/button')
```

```

]
self.divide = self.driver.find_element(By.XPATH,
'/html/body/div/div/div[2]/div[1]/div[4]/button')
self.multiply = self.driver.find_element(By.XPATH,
'/html/body/div/div/div[2]/div[2]/div[4]/button')
self.subtract = self.driver.find_element(By.XPATH,
'/html/body/div/div/div[2]/div[3]/div[4]/button')
self.add = self.driver.find_element(By.XPATH,
'/html/body/div/div/div[2]/div[4]/div[4]/button')
self.equal = self.driver.find_element(By.XPATH,
'/html/body/div/div/div[2]/div[5]/div[3]/button')
self.decimal = self.driver.find_element(By.XPATH,
'/html/body/div/div/div[2]/div[5]/div[2]/button')

```

در این برنامه 5 تست طراحی شده که درستی عملگر جمع، تفریق، ضرب، تقسیم و اعداد اعشاری را اعتبار سنجی میکند.

1. تست عملگر جمع

```
def test_add(self):
```

با استفاده از تابع `click()` بر روی دکمه های 9, 2, 5 کلیک میکنیم.

```

# 925
self.numbers[9].click()
self.numbers[2].click()
self.numbers[5].click()

```

سپس دکمه جمع را میزنیم

```
self.add.click()
```

در ادامه عدد 186 را وارد میکنیم و دکمه مساوری را میزنیم

```

# 186
self.numbers[1].click()
self.numbers[8].click()
self.numbers[6].click()

self.equal.click()

```

در صورتی که مقدار فیلد جواب برابر 1111 بود تست پاس شده است

```

self.assertTrue(self.result.text == '1111')
print('Test_Add passed successfully')

```

2. تست عملگر تفریق

در این تست قصد داریم عملگر تفریق را تست کنیم.

```
def test_subtract(self)
```

عدد 10 را از 25 کم میکنیم

```
# 10
self.numbers[1].click()
self.numbers[0].click()

self.subtract.click()

# 25
self.numbers[2].click()
self.numbers[5].click()

self.equal.click()
```

در صورتی که جواب برابر -15 بود تست پاس شده است

```
self.assertTrue(self.result.text == '-15')
print('Test_Subtract passed successfully')
```

3. تست عملگر ضرب

```
def test_multiply(self)
```

این تست عدد 14 را در 2 ضرب میکند

```
# 14
self.numbers[1].click()
self.numbers[4].click()

self.multiply.click()

# 2
self.numbers[2].click()

self.equal.click()
```

جواب باید برابر 28 باشد تا تست پاس شود

```
self.assertTrue(self.result.text == '28')
print('Test_Multiply passed successfully')
```

4. تست عملگر تقسیم

```
def test_divide(self):
```

200 را بر 4 تقسیم میکنیم

```
# 200
self.numbers[2].click()
self.numbers[0].click()
self.numbers[0].click()

self.divide.click()

# 4
self.numbers[4].click()

self.equal.click()
```

اگر جواب برابر 50 بود تست پاس میشود

```
self.assertTrue(self.result.text == '50')
print('Test_Divide passed successfully')
```

5. تست اعشاری

قصد داریم صحت عملگر تقسیم و جمع را بر روی اعداد اعشاری بررسی کنیم

```
def test_decimal(self):
```

ابتدا 6 را بر 10 تقسیم میکنیم

```
# 6
self.numbers[6].click()

self.divide.click()

self.numbers[1].click()
self.numbers[0].click()
```

سپس عدد بدست آمده (0.6) را با 6.5 جمع میکنیم

```
self.add.click()

# 6.5
self.numbers[6].click()
self.decimal.click()
self.numbers[5].click()

self.equal.click()
```

در صورتی که جواب برابر 7.1 باشد تست پاس شده است

```
self.assertTrue(self.result.text == '7.1')
print('Test_Decimal passed successfully')
```

جواب تست ها

برنامه ذکر شده تمام تست های طراحی شده را پاس میکند

```
Running Test_Add
Test_Add passed successfully
Running Test_Subtract
Test_Subtract passed successfully
Running Test_Multiply
Test_Multiply passed successfully
Running Test_Divide
Test_Divide passed successfully
Running Test_Decimal
Test_Decimal passed successfully
```