

Test_Engineer_Bukalapak

Answer sheet test engineer in bukalapak

BASIC PROGRAMMING

1. Integer, float, double

2. String, char

3. $\text{int } a = 100 + (3*5) - (10/5)$

$= 100 + 15 - 2$

$= 113$

5. $\text{int } a = 0$

$\text{for (int } i = 0 ; i < 6 ; i++)$

$i = 0 \rightarrow a = 0 + 2 = 2$

$i = 1 \rightarrow a = 2 + 2 = 4$

$i = 2 \rightarrow a = 4 + 2 = 6$

$i = 3 \rightarrow a = 6 + 2 = 8$

$i = 4 \rightarrow a = 8 + 2 = 10$

$i = 5 \rightarrow a = 10 + 2 = 12$

$i = 6 \rightarrow 6$ not smaller than 6 so loop stop

$\text{int } a = 12$

$\text{System.out.println}(a/6) \rightarrow 12/6 = 2$

7. $\text{address} = \text{"3566 Golden Street Miami"};$

$\text{System.out.println}(\text{address.replace(" ", "")});$

that means we replace every " " in variable address to "" $\rightarrow 3566\text{GoldenStreetMiami}$

Flanky Test

1. Flanky test is a test that sometimes fails and sometimes pass without any changes code.
2.
 - a. Thread.Sleep()
 - b. Implicit Wait()
 - c. Explicit Wait()
 - d. WebDriverWait
 - e. FluentWait
3. We can test the page with pass the test CAPTCHA. For example we encounter image CAPTCHA, then what we should do is input text in image with case sensitive to pass the test and if image CAPTCHA not viewable then user can reCAPTCHA to get new image.

Essay Question Topic

GET

```
1. pm.test("all data-type is correct", () => {
2.     //parse the response json and test three properties
3.     var responseJson = pm.response.json();
4.     pm.expect(responseJson[0].userId).to.be.a('number');
5.     pm.expect(responseJson[0].id).to.be.a('number');
6.     pm.expect(responseJson[0].title).to.be.a('string');
7.     pm.expect(responseJson[0].body).to.be.a('string');
8. });
```

POST

```
9. pm.test("all is equal", () => {
10.    //parse the response json and test three properties
11.    var responseJson = pm.response.json();
12.    var jsonRequest = JSON.parse(pm.request.body)
13.    pm.expect(responseJson.title).to.eql(jsonRequest.title);
14.    pm.expect(responseJson.body).to.eql(jsonRequest.body);
15.    pm.expect(responseJson.userId).to.eql(jsonRequest.userId);
16. });
```

