Test Case

Class under test: RegisterController

# Check email:

* Input: email (String).
* Output: result (boolean).

## Black box test:

|  |  |  |
| --- | --- | --- |
| Test Case | 1 | 2 |
| String content | Contain char ‘@’ | Not Contain char ‘@’ |
| Output | true | false |

@Test  
public void checkEmailTestCase1() throws Exception {  
 *assertEquals*(true, registerAccountController.checkEmail("hungcq1996@gmail.com"));  
}  
  
@Test  
public void checkEmailTestCase2() throws Exception {  
 *assertEquals*(false, registerAccountController.checkEmail("hungcq1996.com"));  
}

## White box test:

* C0 100% Coverage test:

@Test  
public void checkEmailTestCase1() throws Exception {  
 *assertEquals*(true, registerAccountController.checkEmail("hungcq1996@gmail.com"));  
}

* C1 100% Coverage test:

@Test  
public void checkEmailTestCase1() throws Exception {  
 *assertEquals*(true, registerAccountController.checkEmail("hungcq1996@gmail.com"));  
}  
  
@Test  
public void checkEmailTestCase2() throws Exception {  
 *assertEquals*(false, registerAccountController.checkEmail("hungcq1996.com"));  
}

# Check gender:

* Input: gender (String).
* Output: result (boolean).

## Black box test:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test Case | 1 | 2 | 3 | 4 |
| String length | = 1 | = 1 | != 1 | != 1 |
| First char | ‘M’ || ‘F’ | Not ‘M’ & Not ‘F’ | ‘M’ || ‘F’ | Not ‘M’ & Not ‘F’ |
| Output | true | false | false | false |

@Test  
public void checkGenderTestCase1() throws Exception {  
 *assertEquals*(true, registerAccountController.checkGender("M"));  
}  
  
@Test  
public void checkGenderTestCase2() throws Exception {  
 *assertEquals*(false, registerAccountController.checkGender("B"));  
}  
  
@Test  
public void checkGenderTestCase3() throws Exception {  
 *assertEquals*(false, registerAccountController.checkGender("M1"));  
}  
  
@Test  
public void checkGenderTestCase4() throws Exception {  
 *assertEquals*(false, registerAccountController.checkGender("B12312"));  
}

## White box test:

* C0 100% Coverage test:

@Test  
public void checkGenderTestCase1() throws Exception {  
 *assertEquals*(true, registerAccountController.checkGender("M"));  
}

* C1 100% Coverage test:

@Test  
public void checkGenderTestCase1() throws Exception {  
 *assertEquals*(true, registerAccountController.checkGender("M"));  
}  
  
@Test  
public void checkGenderTestCase2() throws Exception {  
 *assertEquals*(false, registerAccountController.checkGender("B"));  
}  
  
@Test  
public void checkGenderTestCase3() throws Exception {  
 *assertEquals*(false, registerAccountController.checkGender("M1"));  
}  
  
@Test  
public void checkGenderTestCase4() throws Exception {  
 *assertEquals*(false, registerAccountController.checkGender("B12312"));  
}

# Check contact (phone number):

* Input: phone number (String).
* Output: result (boolean).

## Black box test:

|  |  |  |
| --- | --- | --- |
| Test Case | 1 | 2 |
| All characters | Are Digits | Not Digits |
| Output | true | false |

@Test  
public void checkContactTestCase1() throws Exception {  
 *assertEquals*(true, registerAccountController.checkContact("0123456789"));  
}  
  
@Test  
public void checkContactTestCase2() throws Exception {  
 *assertEquals*(false, registerAccountController.checkContact("0123abc789"));  
}

## White box test:

* C0 100% Coverage test:

@Test  
public void checkContactTestCase1() throws Exception {  
 *assertEquals*(true, registerAccountController.checkContact("0123456789"));  
}

* C1 100% Coverage test:

@Test  
public void checkContactTestCase1() throws Exception {  
 *assertEquals*(true, registerAccountController.checkContact("0123456789"));  
}  
  
@Test  
public void checkContactTestCase2() throws Exception {  
 *assertEquals*(false, registerAccountController.checkContact("0123abc789"));  
}

# Check username:

* Input: username (String).
* Output: result (boolean).

## Black box test:

|  |  |  |
| --- | --- | --- |
| Test Case | 1 | 2 |
| First letter | Is English letter | Not English letter |
| Output | true | false |

@Test  
public void checkUsernameTestCase1() throws Exception {  
 *assertEquals*(true, registerController.checkUsername("hungcq"));  
}  
  
@Test  
public void checkUsernameTestCase2() throws Exception {  
 *assertEquals*(false, registerController.checkUsername("\_user1"));  
}

## White box test:

* C0 100% Coverage test:

@Test  
public void checkUsernameTestCase1() throws Exception {  
 *assertEquals*(true, registerController.checkUsername("hungcq"));  
}

* C1 100% Coverage test:

@Test  
public void checkUsernameTestCase1() throws Exception {  
 *assertEquals*(true, registerController.checkUsername("hungcq"));  
}  
  
@Test  
public void checkUsernameTestCase2() throws Exception {  
 *assertEquals*(false, registerController.checkUsername("\_user1"));  
}

# Check password:

* Input: password (String).
* Output: result (boolean).

## Black box test:

|  |  |  |
| --- | --- | --- |
| Test Case | 1 | 2 |
| Password length | >= 6 | < 6 |
| Output | true | false |

@Test  
public void checkPasswordTestCase1() throws Exception {  
 *assertEquals*(true, registerController.checkPassword("password"));  
}  
  
@Test  
public void checkPasswordTestCase2() throws Exception {  
 *assertEquals*(false, registerController.checkPassword("abcd"));  
}

## White box test:

* C0 100% Coverage test:

@Test  
public void checkPasswordTestCase1() throws Exception {  
 *assertEquals*(true, registerController.checkPassword("password"));  
}

* C1 100% Coverage test:

@Test  
public void checkPasswordTestCase1() throws Exception {  
 *assertEquals*(true, registerController.checkPassword("password"));  
}  
  
@Test  
public void checkPasswordTestCase2() throws Exception {  
 *assertEquals*(false, registerController.checkPassword("abcd"));  
}

# Check name:

* Input: name (String).
* Output: result (boolean).

## Black box test:

|  |  |  |
| --- | --- | --- |
| Test Case | 1 | 2 |
| All characters | Are English or ‘ ‘ char | Not English and ‘ ‘ char |
| Output | true | false |

@Test  
public void checkNameTestCase1() throws Exception {  
 *assertEquals*(true, registerController.checkName("Chu Quoc Hung"));  
}  
  
@Test  
public void checkNameTestCase2() throws Exception {  
 *assertEquals*(false, registerController.checkName("Invalid\_name"));  
}

## White box test:

* C0 100% Coverage test:

@Test  
public void checkNameTestCase1() throws Exception {  
 *assertEquals*(true, registerController.checkName("Chu Quoc Hung"));  
}

* C1 100% Coverage test:

@Test  
public void checkNameTestCase1() throws Exception {  
 *assertEquals*(true, registerController.checkName("Chu Quoc Hung"));  
}  
  
@Test  
public void checkNameTestCase2() throws Exception {  
 *assertEquals*(false, registerController.checkName("Invalid\_name"));  
}

# Check student ID:

* Input: student ID (String).
* Output: result (boolean).

## Black box test:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test Case | 1 | 2 | 3 | 4 |
| String length | = 8 | = 8 | != 8 | != 8 |
| All characters | Are digits | Not digits | Are digits | Not digits |
| Output | true | false | false | false |

@Test  
public void checkStudentIdTestCase1() throws Exception {  
 *assertEquals*(true, registerController.checkStudentId("20142137"));  
}  
  
@Test  
public void checkStudentIdTestCase2() throws Exception {  
 *assertEquals*(false, registerController.checkStudentId("2014213721"));  
}  
  
@Test  
public void checkStudentIdTestCase3() throws Exception {  
 *assertEquals*(false, registerController.checkStudentId("20142abc"));  
}  
  
@Test  
public void checkStudentIdTestCase4() throws Exception {  
 *assertEquals*(false, registerController.checkStudentId("20142137abc"));  
}

*assertEquals*(false, borrowInfoController.checkUnreturnedBook("customer1"));  
}

## White box test:

* C0 100% Coverage test:

@Test  
public void checkStudentIdTestCase1() throws Exception {  
 *assertEquals*(true, registerController.checkStudentId("20142137"));  
}  
  
@Test  
public void checkStudentIdTestCase2() throws Exception {  
 *assertEquals*(false, registerController.checkStudentId("2014213721"));  
}  
  
@Test  
public void checkStudentIdTestCase3() throws Exception {  
 *assertEquals*(false, registerController.checkStudentId("20142abc"));  
}  
  
@Test  
public void checkStudentIdTestCase4() throws Exception {  
 *assertEquals*(false, registerController.checkStudentId("20142137abc"));  
}

* C1 100% Coverage test:

@Test  
public void checkStudentIdTestCase1() throws Exception {  
 *assertEquals*(true, registerController.checkStudentId("20142137"));  
}  
  
@Test  
public void checkStudentIdTestCase2() throws Exception {  
 *assertEquals*(false, registerController.checkStudentId("2014213721"));  
}  
  
@Test  
public void checkStudentIdTestCase3() throws Exception {  
 *assertEquals*(false, registerController.checkStudentId("20142abc"));  
}  
  
@Test  
public void checkStudentIdTestCase4() throws Exception {  
 *assertEquals*(false, registerController.checkStudentId("20142137abc"));  
}

# Check study period:

* Input: study period (String).
* Output: result (boolean).

## Black box test:

|  |  |  |
| --- | --- | --- |
| Test Case | 1 | 2 |
| Study period string | In the form “20xx-20xx” | Not in the form “20xx-20xx” |
| Output | true | false |

@Test  
public void checkStudyPeriodTestCase1() throws Exception {  
 *assertEquals*(true, registerController.checkStudyPeriod("2014-2019"));  
}  
  
@Test  
public void checkStudyPeriodTestCase2() throws Exception {  
 *assertEquals*(false, registerController.checkStudyPeriod("2014"));  
}

## White box test:

* C0 100% Coverage test:

@Test  
public void checkStudyPeriodTestCase1() throws Exception {  
 *assertEquals*(true, registerController.checkStudyPeriod("2014-2019"));  
}

* C1 100% Coverage test:

@Test  
public void checkStudyPeriodTestCase1() throws Exception {  
 *assertEquals*(true, registerController.checkStudyPeriod("2014-2019"));  
}  
  
@Test  
public void checkStudyPeriodTestCase2() throws Exception {  
 *assertEquals*(false, registerController.checkStudyPeriod("2014"));  
}