**HƯỚNG DẪN ĐỌC DỮ LIỆU ĐẦU VÀO TỪ PROPERTIES FILE.**

**I. Properties file là gì ?**

Ở các bài thực hành trước, các tham số như url, email, password hay từ khóa tìm kiếm, chúng ta vẫn luôn thêm thẳng vào trong code như sau:

ví du: Class A

| @BeforeClass **public void** init() {  **URL** = **"https://automationexercise.com/products"**;  **keyword** = **"Stylish Dress"**; |
| --- |

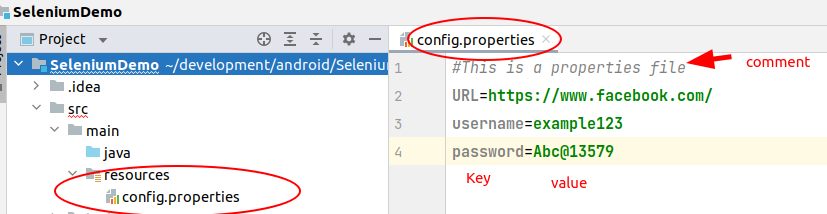
Nếu chúng ta cần sử dụng các tham số này ở trong class B khác thì chúng ta sẽ phải khao báo lại các thông tin này hoặc sẽ cần phải tạo ra 1 class chuyên biệt để lưu những giá trị này.(Utils.java ..)  
Các giá trị được khai báo phân mảnh nhiều nơi, khó quản lý, một khi giá trị này thay đổi, chúng ta sẽ phải đi cập nhật tất cả những nơi đang khai báo hoặc sẽ mất thời gian build lại source code dự án.  
Đối với dự án nhỏ thì không sao, nhưng dự án lớn thì đó là cả một vấn đề nan giải.

**Properties file được sinh ra một phần để giúp chúng ta giải quyết vấn đề này?**

Properties file trong Java được sử dụng để lưu trữ dữ liệu cấu hình dự án hoặc các thông số cài đặt (configuration).

Nội dung của file bao gồm các cặp **<key-value>**, trong đó các key không được trùng lặp.  
Properties File có định dạng : **“.properties”**

*Ví dụ về một property file đơn giản:*

Nếu có thay đổi, chúng ta chỉ cần cập nhật lại giá trị trong properties files, chương trình của chúng ta có thể đọc giá trị mới này mà không mất công phải build lại code dự án.  
  


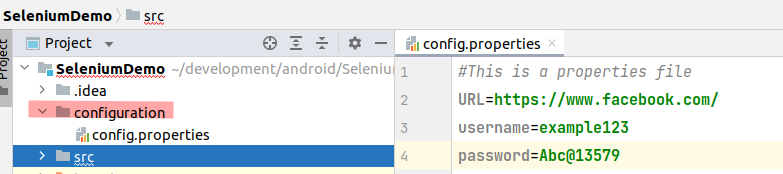
**II. Các bước để đọc/ghi nội dung file properties.**Trong tài liệu này chúng ta sẽ làm quen với việc đọc và ghi dữ liệu với properties file.

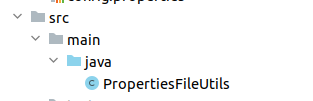
**2.1 Đọc/Ghi nội dung properties file.**

Ví dụ : Chúng ta cần thực hiện đăng nhập vào 1 URL: <https://www.facebook.com/> với thông   
 tin username: example123 và password: [Abc@13579](about:blank).  
 Sau khi đăng nhập thành công, kết quả trả về có chứa thông tin **token** (dùng để authen cho các  
 API bảo mật)

**B1**. Tạo một thư bất kì trong dự án của chúng ta. → vd : **/configuration**

Trong **/configuration**, chúng ta tạo một properties files để lưu trữ các giá trị cầu hình của  
 để bài.  
 Khao báo các cặp key-value tương ứng (key các bạn có thể đặt tùy ý, đảm bảo không trùng)  
 VD: **config.properties**

**B2**. Tạo một class java để đọc/ ghi các key-value properties files

VD: PropetiesFileUtils.java   
   
   


| **package** com.apiauto.utils;  **import** java.io.FileInputStream;  **import** java.io.FileOutputStream;  **import** java.io.IOException;  **import** java.util.Properties;  **public** **class** PropertiesFileUtils {  //Đường dẫn đến properties files trong folder configuration  **private** **static** String *CONFIG\_PATH* = "./configuration/configs.properties";  //1 Lấy ra giá trị property bất kỳ theo key.  **public** **static** String getProperty(String key) {  Properties properties= **new** Properties();  String value = **null**;  FileInputStream fileInputStream = **null**;  //bat exception  **try** {  fileInputStream = **new** FileInputStream(*CONFIG\_PATH*);  properties.load(fileInputStream);  value = properties.getProperty(key);  **return** value;  } **catch** (Exception ex) {  System.***out***.println("Xảy ra lỗi khi đọc giá trị của " + key);  ex.printStackTrace();  } **finally** {  //luôn nhảy vào đây dù có xảy ra exception hay không.  //thực hiện đóng luồng đọc  **if** (fileInputStream != **null**) {  **try** {  fileInputStream.close();  } **catch** (IOException e) {  e.printStackTrace();  }  }  }  **return** value;  }  //2 ghi property vào file  **public** **static** **void** setProperty(String key, String value) {  Properties properties= **new** Properties();  FileOutputStream fileOutputStream = **null**;  **try** {  //Khởi tạo giá trị cho đối tượng FileOutputStream  fileOutputStream = **new** FileOutputStream(*CONFIG\_PATH*);  //Load properties file hiện tại và thực hiện mapping value   //với key tương ứng  properties.setProperty(key, value);  //Lưu key và value vào properties file  properties.store(fileOutputStream, "Set new value in properties");  System.***out***.println("Set new value in file properties success.");  } **catch** (IOException ex) {  ex.printStackTrace();  } **finally** {  //luôn nhảy vào đây dù có xảy ra exception hay không.  //thực hiện đóng luồng ghi  **if** (fileOutputStream != **null**) {  **try** {  fileOutputStream.close();  } **catch** (IOException e) {  e.printStackTrace();  }  }  }  }    } |
| --- |

**Các phương thức chính:**

1. Hàm **String** **getProperty()** : Đọc giá trị property từ file theo key truyền và
2. Hàm **setProperty(String key, String value)** : Ghi một cặp key-value vào properties file.

**2.2 Cách sử dụng**

Trong testcase có thể đọc/ghi giá trị cần thiết vào ra các giá trị cần thiết.

| @Test **public void** TC\_MyOderSearchValid() **throws** InterruptedException {  *//1. đọc giá trị từ properties file*  String URL = PropertiesFileUtils.*getProperty*(**"URL"**);  System.***out***.println(**"URL = "** + URL);  String username = PropertiesFileUtils.*getProperty*(**"username"**);  System.***out***.println(**"username = "** + username);  String password = PropertiesFileUtils.*getProperty*(**"password"**);  System.***out***.println(**"password = "** + password);  ………………………..   *//2. ghi giá trị vào file*  String token =……….  PropertiesFileUtils.s*etProperty*(**"token", token**); |
| --- |

Kết quả khi chạy testcase.



**File Testcase tham khảo:**

| **import** org.openqa.selenium.chrome.ChromeDriver; **import** org.testng.annotations.AfterClass; **import** org.testng.annotations.BeforeClass; **import** org.testng.annotations.Test;  **public class** Example1Test {  **private** ChromeDriver **driver**;  @BeforeClass  **public void** init() {  System.*setProperty*(**"webdriver.chrome.driver"**, **"chromedriver"**);  **driver** = **new** ChromeDriver();  **driver**.manage().window().maximize();  }   @Test  **public void** TC\_MyOderSearchValid() {  *//1. đọc file, mở trình duyệt*  String URL = PropertiesFileUtils.*getProperty*(**"URL"**);  System.***out***.println(**"URL = "** + URL);  String username = PropertiesFileUtils.*getProperty*(**"username"**);  System.***out***.println(**"username = "** + username);  String password = PropertiesFileUtils.*getProperty*(**"password"**);  System.***out***.println(**"password = "** + password);  **driver**.get(URL);  *//2. Ghi kết quả vào file*  PropertiesFileUtils.s*etProperty*(**"result", "The testcase is passed"**);  }   @AfterClass  **public void** terminate() {  System.***out***.println(**"finish test"**);  **driver**.close();  } } |
| --- |