iWAL-intuitive Web Automation Language

(your wish is my command)

February 26, 2014

iWAL (intuitive Web Automation Language) is a simple programming language specifically designed to assist amateur programmers in writing programs that make web-browsing automatic, time-saving and a fun experience. It supports functional programming in a top-down fashion and equips the user with a set of basic but highly useful functions that simplify and expedite some of the monotonous browser tasks like sending bulk mails, regularly downloading material from a website, paying monthly bills, filling online forms etc . Though we have a number of web-automation tools available to us, iWAL's appeal lies in its easy usability and wide applicability. To sum it up, iWAL's intuitive and easy to learn syntax lets a web-browser user (even with little programming experience) explore the automation power this language lends to some of his mundane browsing tasks.

1 Team Members

Ashima Arora: System Integrator
Nithin Reddy: Project Manager
Pranita Dewan: System Architect
Priyanka Singh: Verification and Validation Person
Rafica AR: Language Guru

2 Introduction

With the increasing usage of web based applications, the number of monotonous tasks that need to be performed on the web have gone up exponentially. iWAL aims to automate these monotonous manual tasks. Some of these tasks could be data entry, sending bulk emails on a regular basis, trying to find information on certain topics from a variety of sources or downloading documents (For example, downloading all assignments or course notes from CourseWorks) etc.

In addition to automating monotonous tasks, iWAL could also be used to perform tasks for which the user does not know the exact time at which the task should be performed. For example, if a student wants to register for a course that is currently full, he/she would have to keep checking the registration website until a spot opens up. Instead, he/she could write a program in our language that would automate this process.

Thus, we aim to develop a practical and easy to use automation language that can be used to simplify numerous tasks on the web.

3 Buzzwords

Intuitive, simple yet powerful, easy-peasy, scraping, automation, readable

4 Current Languages

Existing solutions for web-browser automation include tools like Selenium and Mechanize. They provide a host of modules compatible with Java and Python. The drawback being that the user has to have the knowledge of Java or Python in order to be able to use their features. iWAL leverages the rich feature-set of Selenium and provides clean and simple commands, intuitive for amateur programmers.

iWAL is born out of our desire to make web browser automation simpler and more productive. We attempt to devise a language that can incorporate the automation power of Selenium, portability of Java bytecode and familiarity of English language structure to enrich the web browsing experience of users, encouraging them to program some of their monotonous tasks.

Although presently available tools and IDEs cater to the web automation needs of programmers, our language will appeal to a wider spectrum of users who want to reap the benefits of automated browsing, but do not have a knack for programming.

5 Targeted Users

Our language can be used by people with little programming experience and a sound knowledge of navigating web browsers.

- Website administrators - It can hold a great appeal to website administrators who have to ensure that the web servers, hardware and software are operating correctly, designing the website, generating and revising web pages, A/B testing, replying to user comments, and examining traffic through the site.
- Testers who want to create robust, browser-based regression automation.
- Any user who wants to automate boring monotonous tasks on the web.
- Non-CS students who want to register online for courses. This seemingly simple task can be incredibly difficult when done manually during the rush hours of

course selection. The student might have to keep tracking about when particular course has free seats. This problem can be solved using our language with simple commands.

6 Properties

6.1 Operators

Table 1: Operators

Operator	Purpose
+	Addition of integer or double values, or concatenation of strings
-	Subtraction of integer or double values
*	Multiplication of of integer or double values
=	Assignment operator
==	Equality comparison
>	Greater than comparison
<	Less than comparison
>=	Greater than or equal to comparison
<=	Less than or equal to comparison
!	Not(negation)
!=	Not equal to
||	Logical OR
&&	Logical AND

6.2 Primary data types

Table 2: Primary Data Types

Data Type	Purpose
int	stores a 32-bit signed integer
double	stores a double precision number
char	stores a single 16-bit unicode character
boolean	can take two possible values, true and false
key	can take value of any key on a standard keyboard(eg. Enter, Tab, a, z, etc.)

6.3 Keywords and Comments

- Keywords : true, false, break, if, else, elif(like else if in java), repeat(like for loop in java), until(like while loop in java)

- Comments in iWAL will have the same format as that of Java.
- Single line comments can be given as :
  // All text on this line is a comment
- Multi-line comments can be given as :
  /* This comment can go on for multiple lines.. */

6.4 Derived Data Types

Table 3: Derived Data Types

Data Type	**Purpose**
String	stores series of characters (eg. “This is a string”, “1234”)
array	a container object that holds a fixed number of values of any one of the single primitive data types defined above.(eg. int [], double[])

6.5 Functions Supported

1. Opening a browser
2. Opening a website
3. Input from user
4. Input from file
5. Fill in a text box
6. Simulate a key tap/ Combination of keys
7. Click a link
8. Click a button
9. Check a checkbox
10. Select a radio button
11. Select option is a dropdown menu
12. Proceed to next element or skip a specified number of elements
13. Download an media file using its element ID
14. Give a delay before proceeding, etc

7 Sample Applications and programs

Some programs that can be written using iWAL:

1. Open and login to Student Services Online(ssol) website.

   start();
   String url = “https://ssol.columbia.edu/”;
   open(url);
   tab(1);
   input(“username”);
   tab(1);
   input(“password”);
   tap(Enter);

2. Download all the images from a google image search

   start();
   String url = “www.google.com/imphp”;
   String element = “q”;
   String query = “This is a search query for google”;
   inputE(query, element);
   tap(Enter);
   String path = “/users/XYZ/Desktop/Folder”;
   download(“images”, “all”, path);

3. Automatically filling a form on a website taking the input from user for only one field that varies

   start();
   int numOfRepeats = 20
   repeat(numOfRepeats) {
   String url = “www.fillformrepeatedly.com”;
   String name = inputU(“Enter the Name : ”);
   String element = “element1”;
   input(name);
   tab(2);
   input(“DepartmentX”);
   tab(2);
   input(“SchoolY”);
   tab(3);
   input(“UniversityZ”);
   tap(Enter);
   }