**3 Type of Variables**

* Scalar
* List
  + \*\*\* Variables \*\*\*
  + **@{search\_text}**  books travel
  + \*\*\* Keywords \*\*\*
  + Input Text Xpath://\*[@id=”gh-ac”] **${search\_textabc}[1]**
* Dictionary
  + \*\*\* Variables \*\*\*
  + **${env}** qa
  + **&{url}**  **qa**=http://qademocom uat=http://uatdemocom dev=http://devdemocom
  + \*\*\* Keywords \*\*\*
  + Open Browser **${url${env}}** Chrome
  + **&{DICTIONARY}** username=testuser Password=demo
  + Log &{DICTIONARY}[username] // WARN: & deprecated
  + Log **${DICTIONARY}[username]**
* ENVIRONMENT
  + 별도 선언 없이 사용 가능

**Demo Scenario**

Step 1 : Open RIDE

Step 2 : Create a Test Case for login [http://opensourcedemoorangehrmlive](https://www.youtube.com/redirect?event=video_description&redir_token=QUFFLUhqbUJVWDZTakVpYTRmTTU5a1A5WFJCMk9OM25yZ3xBQ3Jtc0trVjBobkxEeVNHSElSZnpFQ2xyR1o3bENTakV5UUY5b1NKcnFGeGNJXzJSUEVCTHRyUlE3WDNQSkE5MVZKSDhxWV9yNlhndHpxNGZhcXRwZUxRMDN2Ri1FRkJKUk1ncEE2QUw0Z1duRmowR01yX056Yw&q=http%3A%2F%2Fopensource.demo.orangehrmlive.com%2Flogin&v=KlyOQrn8HjU)

Step 3 : Create a SCALAR variable for url and refer in TestCase

syntax : ${VariableName}

Step 4 : Create a LIST variable for username and password and refer in TestCase

syntax : @{VariableName}

Step 5 : Create a DICTIONARY variable for username and password and refer inTestCase

syntax : &{VariableName}

**${GLOBAL\_DICT}**= Create Dictionary 1000=[1,2,3,4,5,6]

Log To Console \n----1.0 **${GLOBAL\_DICT}**[1000]

Environment

%{variable name}

**Useful TIPS =========**

1 If a non-existent variable is used in test-data, the keyword using it will fail

2 We can set variables from command line

3 Variables are case-insensitive

4 As best practice, use uppercase variable names for global variables And lowercase variable names for local variables

5 Built in variables - [http://robotframeworkorg/robotframew](https://www.youtube.com/redirect?event=video_description&redir_token=QUFFLUhqbDRIdWF3MWxpdC1SV0VibWRYRWpzZEx4R2R2QXxBQ3Jtc0tsSWdYWlJkeGppdWFGckVYeEJXbnlsWU42VENsOWxHcDB6c2UtYnNWbzZKUnYyaDhUUnZrNzZDUGpXVXN6Z3VJRHRBZGNmU2xjWkRydTdoanVfdGxyN2lhMFRvS2c0dTRQcFByLTFzaUFIWHVhOEY4dw&q=http%3A%2F%2Frobotframework.org%2Frobotframework%2Flatest%2FRobotFrameworkUserGuide.html%23built-in-variables&v=KlyOQrn8HjU)

Robot Framework Playlist - [https://wwwyoutubecom/playlist?list](https://www.youtube.com/playlist?list=PLhW3qG5bs-L9l2I8K8dEhw6HXy-Z-33w3)

All Playlists - [https://wwwyoutubecom/channel/UCTt7](https://www.youtube.com/channel/UCTt7pyY-o0eltq14glaG5dg/playlists/playlists)

Like, Share and Subscribe

Thanks, Raghav

**ONLINE COURSES TO LEARN \_\_\_\_\_\_**

Visit - [http://automationstepbystepcom/](https://www.youtube.com/redirect?event=video_description&redir_token=QUFFLUhqbnktaDFFUTAtdlRYOUk0bXo3QU4zU2ZkVERjZ3xBQ3Jtc0ttVmE0M1JsZmFFUWRlS2psMGhxZXlJbGYyZU9QZ2c0UFNZR1R4OTYwQjRKSW1RejZSS3VvX1VDVzlLM2prS1lEOWhKelpldXNBTWpNX3IteG9KOEN0THVNMHhncjVCMXBtTnNubnpqR2VpaGs2d1BESQ&q=http%3A%2F%2Fautomationstepbystep.com%2F&v=KlyOQrn8HjU)

------------ UI TESTING ------------

Selenium Beginners - [https://bitly/2MGRS8K](https://www.youtube.com/redirect?event=video_description&redir_token=QUFFLUhqbFA1U1FrZGNuWWVaNHJoTjJHMVFxVVF1STdBZ3xBQ3Jtc0tubEhwczdIT1RBaTJJRTVJWElhZlJjQ3VBSElTbWtxY1N0REV6eXlINGpFNnlTV2dOdV9wVDB6N0FXTTE4cDczUm54aHp2enJpM2pNZHBiMzcyTjI5U0ljLTA2RVdnXzN6Z3VpNExzT2xQT1pTZ1U3bw&q=https%3A%2F%2Fbit.ly%2F2MGRS8K&v=KlyOQrn8HjU)

Selenium Java Framework from Scratch - [https://bitly/2N9xvR6](https://www.youtube.com/redirect?event=video_description&redir_token=QUFFLUhqbjhsckc1Y2JxVHFwM0pPQXE2dC1CUTh2Tmo4QXxBQ3Jtc0tsYVR0S0hPZlVKa09Lc3FobGpCX05aVW9tbS0wZ3I3YnF3Zjc1NEJSYzhObXJmZExWNDltUzZsVzlNdml5d3FNaFY2SDl5aDJPTWhqMDZWY25rM1ZMM0Q4NzBmdEd3UFFMRTJKM3ZmdzV4SmlvQnlPQQ&q=https%3A%2F%2Fbit.ly%2F2N9xvR6&v=KlyOQrn8HjU)

Selenium Python - [https://bitly/2oyMp5x](https://www.youtube.com/redirect?event=video_description&redir_token=QUFFLUhqa1ZMbzA5VVNzMHpVMVhoSnBiS1FwMWdXc0tsZ3xBQ3Jtc0tsRVVPM055VU5mOUVyVnQzaTFzMmpfdllhMHFoMVNocllCMGREbG5QT09zVHRmdVBDck80cEJWM2NidTRUQlpnSFlRZWRhbU9yVXB3Zm5DTmdUWkxXVWppZFZiNENmMlNaVnVxYlFudmQwcDVlOXRRSQ&q=https%3A%2F%2Fbit.ly%2F2oyMp5x&v=KlyOQrn8HjU)

Selenium Tips - [https://bitly/2owxc50](https://www.youtube.com/redirect?event=video_description&redir_token=QUFFLUhqa1NfSUlQOTVmQnhHMERNTEZ0LVlKdXdRX3ZlZ3xBQ3Jtc0ttQV9fLWhPQW1lWXJrSFdUeHJkcE1kM0xxOXZyZzFfdEdod0p6bGNueVppLW9RYldtREZkclJRRW9HQmRSU3VIdG45N0ZzRzQwcVJRNXk3UjlSZGhEdzdIQmVqaTRrUUduTXFwZFkwdGloU28td0YyYw&q=https%3A%2F%2Fbit.ly%2F2owxc50&v=KlyOQrn8HjU)

Selenium Builder - [https://bitly/2MKNtlq](https://www.youtube.com/redirect?event=video_description&redir_token=QUFFLUhqbk92RXJzNkhaVnNOZjcyNUQtbmF4aXVqV0tnUXxBQ3Jtc0tuVDJSMXlMNmxPS29LOXdYU1VuRElqdjRiOVpzUm5YWUg3bkpBV3ltZUp2VkFvYk5RUkI5OHBia2NCUy1LdzJrdkp2bExfRjV6YkY0eUtienN1dkpzbUl3UDFoczJrMjBHcXliZjBad1pXNVZGaE14WQ&q=https%3A%2F%2Fbit.ly%2F2MKNtlq&v=KlyOQrn8HjU)

Katalon Studio - [https://bitly/2wARFdi](https://www.youtube.com/redirect?event=video_description&redir_token=QUFFLUhqbU5pUkRINUZVc1MxQUVGSkZzeXpRT2tIQU1RUXxBQ3Jtc0tsWHZDUm9aMXRRcm5qUk55Y25PdnhOZml0UkZ6d0RnRkFJWHd5UGYwa19sOTRuQnZROVY4bmFwMVFUT1lFc0tubF95Qk53RUVyT2xZeGpjRUlNRlBnc2w4ZzZUYS1XWTQxOENwaTdZRHhHRHcyU0hqMA&q=https%3A%2F%2Fbit.ly%2F2wARFdi&v=KlyOQrn8HjU)

Robot Framework - [https://bitly/2Px6Ue9](https://www.youtube.com/redirect?event=video_description&redir_token=QUFFLUhqbUl0S2dkMDdwUmE5dXBicVJOME1aSzFoZHdZZ3xBQ3Jtc0tsU0toQjdRcU83M0hoVUxYaklPSFMzSThIMWtaUUZ1NmQ3Q3VwQ282d3p4NERkeGZfb0pIVWFCd3dtUko4SG5Nb3RkUmNudF95U2FXcVNuNE9oN05lWUtYTnhwejBwMGd3M0tFRC1VaG01VDVjZDRjMA&q=https%3A%2F%2Fbit.ly%2F2Px6Ue9&v=KlyOQrn8HjU)

------------ API TESTING ------------

Web Services (API) - [https://bitly/2MGafL7](https://www.youtube.com/redirect?event=video_description&redir_token=QUFFLUhqbldCWlhiN3p5dy1jN2xQTnF0cVNvSXdJQXMtQXxBQ3Jtc0ttQnVMX3lmSHdobkljeFUyR2dsN0JDRXdvUXZ6X1RfZHBQaWptUW1ndUx1c1NiNXk4aXEyOFRkN2E4Zy0wQWE4MEZubGVvZ01fVFNRNmY5T2doUUItYlRJWDYxYkNMVi1YN20yeTBKOHFEQWxaVzdjNA&q=https%3A%2F%2Fbit.ly%2F2MGafL7&v=KlyOQrn8HjU)

SoapUI - [https://bitly/2MGahmd](https://www.youtube.com/redirect?event=video_description&redir_token=QUFFLUhqa0hVakhqYVgycmEtb1A3dTJBWEdmSEVtMUZCQXxBQ3Jtc0tuV2hnT2FVLUJlVVVtWlVVS0s4UE42bjlvLTNMb0szT09EUC1BVnZ1a2FNYWt0Q1c2dVJZLXFuNERtYTQ4WklXbFh6WjNjc05VVmo0VncyV25tMGFHT1RGem1NRkdzNDVsNlRIUEExcVdLazFWRmdVNA&q=https%3A%2F%2Fbit.ly%2F2MGahmd&v=KlyOQrn8HjU)

Postman - [https://bitly/2wz8LrW](https://www.youtube.com/redirect?event=video_description&redir_token=QUFFLUhqbDFtQ3h4b2JRdS02OXNmQllodUIzUm9BSjdFUXxBQ3Jtc0trT2diRHRjbWYydUJzUlBacmRoaUJ5QURQaXp5NDlTeFVLRnVURkYyRzBLNXkxZXZheERxOWNmNGUzQmtxVG1hNFF3cGdTNVVaZndjZl9SMElZOVFvRTM3SVVoMXpWNGRjczhBYVI1aEZoNC1rRFdTOA&q=https%3A%2F%2Fbit.ly%2F2wz8LrW&v=KlyOQrn8HjU)

General - [https://bitly/2PYdwmV](https://www.youtube.com/redirect?event=video_description&redir_token=QUFFLUhqa2p5d1R4OWRBeFl5Z0s4NEpBUTRqa3VFMHBXUXxBQ3Jtc0tsSTdqdFc1TXJTY0pURWNuejhZTXA1UjhjRWYwbF9iQnh3TkNPdFVBeFVOb2tQSEs4WWxmV3BxeFg3dnZfbUNJd1VuNXRUWVVYNnhMT1E2clFNZ3E3VWhSazgtS205SU4yMW1xVHViNmVWX3p2Ty1nUQ&q=https%3A%2F%2Fbit.ly%2F2PYdwmV&v=KlyOQrn8HjU)

------------ MOBILE TESTING ------------

Mobile Playlist - [https://bitly/2PxpeUv](https://www.youtube.com/redirect?event=video_description&redir_token=QUFFLUhqblhlWUlGajF5U3dZVU8tZ3JvSHNxM1MwZHhpd3xBQ3Jtc0ttUi1SR2l4NGNnV25MUXNHOFdDcUJSanRyWmpGaU1TNGxZZ1lvT3NwR3NZczA3cW95bTVIYVBVTkFTTmRwVTJjNXlmSlpyeFlNTVBvLUdzZEhTc2dIbzVVVUtsTWdOQzl3ZlBINGRQWGxQM2wyTmhRWQ&q=https%3A%2F%2Fbit.ly%2F2PxpeUv&v=KlyOQrn8HjU)

------------ CI | CD | DEVOPS ------------

Jenkins Beginner - [https://bitly/2MIn8EC](https://www.youtube.com/redirect?event=video_description&redir_token=QUFFLUhqbG9lYUpDNWozMDRNenlCbmVUZ01KbkVFdG95QXxBQ3Jtc0ttZVRxRnQ5c3QtU3NuTHIwVnE5U1ZEOGJaOVB2bzZSTEdKWExDbzBQOGZveENFS2lnNTVVcHFuQlo1aE9BNzFiZTlnRkJBWVkxMklaMUNYdDVKMk96dHVKcVVfWjByeDhETEJEMFZKWjNlSDRxRjNobw&q=https%3A%2F%2Fbit.ly%2F2MIn8EC&v=KlyOQrn8HjU)

Jenkins Tips & Trick - [https://bitly/2LRt6xC](https://www.youtube.com/redirect?event=video_description&redir_token=QUFFLUhqbG5sc2t2UkdpWjlFSTJTUWtvSG9NWVBHa1Zyd3xBQ3Jtc0tsM1FZTG5ySTdJVXpaZU9TazI0TWdNYjBBWEdKZ0oxeG5BallnaVpPbk1pWmNwbWV3WXYxM3U2NnVycER3YzdEUE1ZNktCMjRnemJOdGpUdTJYTVNHRWpQYlg0aEc3Y2d2OVprQ1BDZ0x5NC1zTVRxRQ&q=https%3A%2F%2Fbit.ly%2F2LRt6xC&v=KlyOQrn8HjU)

Docker - [https://bitly/2MInnzx](https://www.youtube.com/redirect?event=video_description&redir_token=QUFFLUhqbW5FeGxiZXF0Tm1nWERhTEZwaTJxMWhhT1JZZ3xBQ3Jtc0tsV3NRQ0Faa3QwWF8xVm9DT3ljNEFxQU9Ta2xRQ3VUWm9pTVhhdkZ5OGp2Q0o2RE9SY2QtWkdHTXZQNS1LdkRwc21Jc1Z2Y19iYjkzVFNHd3Uwc0NoaFZVV0VzaE11QU85OC03bzdZUHpsdzlOdWRTSQ&q=https%3A%2F%2Fbit.ly%2F2MInnzx&v=KlyOQrn8HjU)

------------ VERSION CONTROL SYSTEM ------------

Git & GitHub - [https://bitly/2Q1pagY](https://www.youtube.com/redirect?event=video_description&redir_token=QUFFLUhqa1hBcF9DWU5rbmpValV3SFlRYTZ6djgzOEItZ3xBQ3Jtc0tudWo1eUh4ZktVbGczMENzbDlIOUNocUd0c1A2R2lLZVBIZ2Fjb1JNOEFXLVZVQXZ2a3htZXBmazhoNFk1Q1dQZGU1Um5iSmZtT2Z0dk90by1ZSVVKY0UtWU9CU0NSSkdsXzc0eVh5NXBtUEw2S243aw&q=https%3A%2F%2Fbit.ly%2F2Q1pagY&v=KlyOQrn8HjU)

------------ PERFORMANCE TESTING ------------

JMeter Beginner - [https://bitly/2oBbtIU](https://www.youtube.com/redirect?event=video_description&redir_token=QUFFLUhqa1hxWUd4UzQ4anRORGRkRk15V3RuaG5CVnR1QXxBQ3Jtc0trQk5xVE5QTzI4anY5eHNUNDlyem1od2FnbDdpcm50cXVwODBIeGFaUmJNSnEzaC04eXZFQ3Jka3Ytb1A2NDFXcWVXX2tBWXhiV2wtVzV3QnNyUkdVMnRQck5Zc3QyY1Y3T0VTanRCZUlYMmxSaFFINA&q=https%3A%2F%2Fbit.ly%2F2oBbtIU&v=KlyOQrn8HjU)

JMeter Intermediate - [https://bitly/2oziNVB](https://www.youtube.com/redirect?event=video_description&redir_token=QUFFLUhqa0JudEtRMjJ0OGFTVmJEbGVlZXJnSmhQdlhCUXxBQ3Jtc0ttVEJTc2dkaVZoZkZqUEF0cU5XZnZpcV9hMWpna2lrdUI2TkNTZVcxbFZBa2VhR09DUzNXT1ZPd3hoNktiZ0Rxd19OczZxZjIzMVVtVnBidGMteGo3SEZqWDRlTktUZzlTeXdxekpmTkw1MGRRS2gyVQ&q=https%3A%2F%2Fbit.ly%2F2oziNVB&v=KlyOQrn8HjU)

JMeter Advanced - [https://bitly/2Q22Y6a](https://www.youtube.com/redirect?event=video_description&redir_token=QUFFLUhqazhzNnN0bDJSTE1Fc0JsajlkcnB1NnlnenpBQXxBQ3Jtc0tuY3J5MG1mZnY0SkJBVUMyb3FRN3g0ZWJjV3hBV0l0Ymg1LWFPV0NaRGhPVWs2RVUzaFdVbjB2bW9DanJvNnhwb3NWMmlINzAyUzlyQnZ3d0dnYVJNcVpnOEJlMkk5bjBMTzZpa042QmF1aVVyOTNNWQ&q=https%3A%2F%2Fbit.ly%2F2Q22Y6a&v=KlyOQrn8HjU)

JMeter Tips & Tricks - [https://bitly/2NOfWD2](https://www.youtube.com/redirect?event=video_description&redir_token=QUFFLUhqbFJLN2phWlpmQmp2SVU3X2NmZkhOUkZUdzlPQXxBQ3Jtc0tseDE0UjVLU2hkanBQb2JSUUo3bkJIOWdjSUE5alhqSFVUSDltOUMtbG9ET25CN0M4MnJmMG9vVlFOSk9mcHd0VDd5dW5CMmY1WFZDaDV0OTh6WFFxeDZqRURNZnJXQUdxMlJVemZFRV9LaVJta24yWQ&q=https%3A%2F%2Fbit.ly%2F2NOfWD2&v=KlyOQrn8HjU)

Performance Testing - [https://bitly/2wEXbLS](https://www.youtube.com/redirect?event=video_description&redir_token=QUFFLUhqa0FhY1ZwNFVWVnlDdWdYOFBhX2RlN2d4MmUtUXxBQ3Jtc0tscDNaT3R4MmYwWXA3Ulp0eXZ2ZXRrckFzVm4yLS1mNUY4QVo1ekNkSUNuMnkyV1dpYWQ1UEdJUUpJNnZrSkJ0dTRYaXdiWEhZMUNseXd6aWpua1hUTjhmTWpLTzdEdTJiRWNtZFdrVm5vbS1XZ2I4TQ&q=https%3A%2F%2Fbit.ly%2F2wEXbLS&v=KlyOQrn8HjU)

------------ JAVA ------------

Java Beginners - [https://bitly/2PVUcXs](https://www.youtube.com/redirect?event=video_description&redir_token=QUFFLUhqbjlnVnI4dmUzNE1JRFd6Z3Q2XzZDelVPVzgxZ3xBQ3Jtc0trbTZGbUZZLVBRRHBHNXp3bzA3aFkyREtuSG5sMTNiYm5rNks3TWhBYlZHZzU0MVpESU9Md2V6c2RFUnZqVi12OVhxRUlmV1gwNmJKWFRLYUJNN3AwZ2REX0doalpvWFZGc1p5RVg3UUZmNjVwazhIOA&q=https%3A%2F%2Fbit.ly%2F2PVUcXs&v=KlyOQrn8HjU)

Java Tips & Tricks - [https://bitly/2CdcDnJ](https://www.youtube.com/redirect?event=video_description&redir_token=QUFFLUhqbmpPYUhibVJVbmhMUzMzMjM0SER6LWVqeEZOZ3xBQ3Jtc0tuOERHOGRTUG04UGN6bUhXcWZDeVZRWlhkdWp0YzlZQ2JDVXlJLUF5NFc3bWRUbThEMUw0SEV3QkVEZktqZV9meW9iQXBDaVhSazNRODBMSTJSQ011RFZJcko5V3JTWUtVaVZ3YWx6QlEtS05CVno4QQ&q=https%3A%2F%2Fbit.ly%2F2CdcDnJ&v=KlyOQrn8HjU)

------------ MAVEN ------------

Maven - [https://bitly/2NJdDRS](https://www.youtube.com/redirect?event=video_description&redir_token=QUFFLUhqbUxiTFJ6ejJGTlB2SjQzWkdlRzNPek01RXJRUXxBQ3Jtc0tuNEg2QjgtaWprZTJXVnhtck5KZG9sNVk1d3IxbFA3ZkI3OUlKTWt4YzNJRlRVOWdNaXBpaUxXa2dJV2pKWVE3Zk82ejRUOGxvUnVNbkc0X2EtMllSLVlIRUczYmk0M0NkNGdZUlhyZ3UyVHFHZFUyOA&q=https%3A%2F%2Fbit.ly%2F2NJdDRS&v=KlyOQrn8HjU)

------------ OTHERS ------------

Redis- [https://bitly/2N9jyCG](https://www.youtube.com/redirect?event=video_description&redir_token=QUFFLUhqazl6TTVURGZmSTI0WVpKVzVwWFpuYW1WZk9oUXxBQ3Jtc0tuTjh2ZUNCSjAyTTQtNGt0VXRjR2dGVDUxRUFjQ091SmRuN0JPWGFRRGd4NmhhTW9TWk5vd3FqenR0TklHQ3JQWjlrQzVOMGtRTVhVbV8wQkw5YVlCbGcxRnVVVW9jWGZ6R3loMG5ka3VSRWJYZUlicw&q=https%3A%2F%2Fbit.ly%2F2N9jyCG&v=KlyOQrn8HjU)

Misc - [https://bitly/2Q2q5xQ](https://www.youtube.com/redirect?event=video_description&redir_token=QUFFLUhqbGFNN1RPLWtBMlJQYnlhbU56SU9EenBGYk1Ld3xBQ3Jtc0trM21mSjdhSnRjVFJQNk44aVJVY2pXbzBMNXkxU0NuRVo3STNPSU9NOUZXUU9MckRVVjY0cTFRVzJpd0RDUTgyZWpVMzE5ZEhyeHlIRXlkVlNja2MzYmF5M3V2T3Y3NUppQ0w5RG1Tb2tSUGRBUTZHQQ&q=https%3A%2F%2Fbit.ly%2F2Q2q5xQ&v=KlyOQrn8HjU)

Tools & Tips - [https://bitly/2oBfwoR](https://www.youtube.com/redirect?event=video_description&redir_token=QUFFLUhqa0ZDOEpCS25iUEVvdjRtWWdqYlJqMVhjSlByQXxBQ3Jtc0treFVLS1BITV8tNVluRW9tZlFWaENvRTVyU01MTzlvbndkQzN4dWpaU2R0cEtOaUNCQk1rdk96RFoxVFlERDFyakZFYVR6NzNlMlZxMzlIMU9IOWZ0QlM5cm5MS2xReFVUTGtNMHRSMzVqOUtWeUxJRQ&q=https%3A%2F%2Fbit.ly%2F2oBfwoR&v=KlyOQrn8HjU)

QnA Friday- [https://bitly/2NgwGpw](https://www.youtube.com/redirect?event=video_description&redir_token=QUFFLUhqa2tmWFFYZUlIeXhTSEY4ejF2eFE3Y3dhc1dRd3xBQ3Jtc0tuWTU4M0I4QzljZ1ZfOG9vVnNZR1NTb1JyaXlEZFFTcURpNUpDU2JGOUhNZkdEcjRWc2Q3WS1uM1VaTWpkNmJxYVRVdG9TcnBmSXZGMmVDVFpDc2dxWXM0UWFEMThHam1Ocnh4b1lycXZybWdoOGd2Yw&q=https%3A%2F%2Fbit.ly%2F2NgwGpw&v=KlyOQrn8HjU)

Sunday Special - [https://bitly/2wB23BO](https://www.youtube.com/redirect?event=video_description&redir_token=QUFFLUhqa1QtZVE4VXp5dzUxWTlmY3BDM2FsVDVOdEhxZ3xBQ3Jtc0tsSzZhclpjSWU4RGN3NklIY2ZTOVl0QVhpbEYtN3BrWUN0SngwTkktNkRfTG9tak9saTJxNVIxSU5ZYUhPd1k1ZDNfMnRTOVZHdng4R19fXzRaWUVLUUlGU3dnTXJoYlZvMV9xUmlCb2Y4TzhSRFc0MA&q=https%3A%2F%2Fbit.ly%2F2wB23BO&v=KlyOQrn8HjU)

Ask Raghav- [https://bitly/2CoJGWf](https://www.youtube.com/redirect?event=video_description&redir_token=QUFFLUhqbG43S2RIRFhUM3ZUcWZKaW9HUkdHQlc4SWlTUXxBQ3Jtc0trajk0aFpDVWdwbloyR0tGUGJtSmhIRDY2SHU5ZGVqZnpib2x2T29aejNmbklxSjJOeE1vN3hKcUQyejh6WWhUdExaWENQX3hVcTBoZzRYVGNFemJLdXNEMWhzLWg4Qk10VkdBUWpRWi1VS3M1RFdHUQ&q=https%3A%2F%2Fbit.ly%2F2CoJGWf&v=KlyOQrn8HjU)

Interviews - [https://bitly/2NIPPxk](https://www.youtube.com/redirect?event=video_description&redir_token=QUFFLUhqbEg2bHlQNkF6elBCY2ZXal90cnRLQ3Z0UE5SUXxBQ3Jtc0tsVmliaFhXVEVqVm9MTU80cmNFUUJ3bXl5MC1nWk1tcDU0RU9zTC1GZG9ZdVFyV0ZCa0o4SFctM2dHSTNpTEZ4UjFobzJLTHBGdU1md0dOWU5WRVZpdGs5Zm8wUEE5ZEtiNDRFSGhKSEk4WS1QZGc1UQ&q=https%3A%2F%2Fbit.ly%2F2NIPPxk&v=KlyOQrn8HjU)

All Playlists - [https://bitly/2LSiezA](https://www.youtube.com/redirect?event=video_description&redir_token=QUFFLUhqbEtZLTBzb1ZxUHlzT2cyX0h6dmJnaEFqb1ZOZ3xBQ3Jtc0tudG45UzBpN3Z4VHhmSTZyYlpiTTlya2MzdmRyY0QtMnZ1MGc2NFIzT0hHeC1ISVpxNzRiSTNya2tZT2lBWWN3N3BqLWF5NGNkVEs2SGhJSWVxN3drZ0lVTk92cjVyQkpxVlFHMkdaV1UwY1p4azRqVQ&q=https%3A%2F%2Fbit.ly%2F2LSiezA&v=KlyOQrn8HjU)

------------ Follow ------------

Facebook - [https://wwwfacebookcom/automationst](https://www.youtube.com/redirect?event=video_description&redir_token=QUFFLUhqbTFqU3Z0MVlCWGtTMG1ZR3M2WDNUVXB3ZVNyUXxBQ3Jtc0trY20tUWFhbnBlRlJNWmhIOGtUMTY5MGg3STRtQlpjS0ExT05ZUzZHanNQbjkxOGZiSDI1NHBld0l0OGU5bUJBQVV5dG9JbWNXcGdaeE1QVUI4T01LOXV4V1Z5QTZmS3luY3pka19hWWVNYXpCa3ZWSQ&q=https%3A%2F%2Fwww.facebook.com%2Fautomationstepbystep&v=KlyOQrn8HjU)

Twitter - [https://twittercom/automationsbs](https://www.youtube.com/redirect?event=video_description&redir_token=QUFFLUhqbS1QR0lvMC01bC1yWnpRWldqT0hDQzdEYng0QXxBQ3Jtc0tsZjFEdXByaXNCcllQekNLRzlQaTBwVEJ2MWVFRXJuZVlYd19xSWl0SXMweXY4ZGNxb3ZDRUNuS2lHaDljNERvWl83MlF2TU9iY2UwYjJmV3gwNXluR1lGR0NiRWwzNFZfdnJJOWpJRjYxcGNBT2JmZw&q=https%3A%2F%2Ftwitter.com%2Fautomationsbs&v=KlyOQrn8HjU)

Youtube - [http://youtubecom/automationstepbystep](http://youtube.com/automationstepbystep)

------------ Variables ------------

List variable - [https://wwwyoutubecom/watch?v=ZSenYS\_sb8c&ab\_channel=AutomationLearning-SDET](https://www.youtube.com/watch?v=ZSenYS_sb8c&ab_channel=AutomationLearning-SDET)

Dictionary - [https://wwwyoutubecom/watch?v=P8lpTs7j3Vw&ab\_channel=SoftwareTestingMentor](https://www.youtube.com/watch?v=P8lpTs7j3Vw&ab_channel=SoftwareTestingMentor)

**(여기까지는 Robot Framework 관련 내용으로 RPA Framework과 동일한지 ?)**

**RPA Framework Forum에서 발췌 #1**

The list (@{variable}) and dictionary (&{variable}) syntax is optional when assigning variables. You *can* use the list syntax when assigning a Table to variable, but it is not necessary. I can try to explain the difference between them.

First off, when doing variable assignment, it’s entirely valid to assign lists and dictionaries to scalar variables like this:

${my\_list}= Create list 1 2 3

${my\_dict}= Create dictionary one=1 two=2

The reason to use the special syntax is that Robot Framework does some extra validation for you:

# This will validate that the keyword returns a list (or something list-like):

@{my\_list}= Create list 1 2 3

# This will validate that the keyword returns a dictionary (or something dictionary-like):

&{my\_dict}= Create dictionary one=1 two=2

* **Hint: 테스트 결과, Global 변수를 위해 Variable section에 선언시에는 @{}, &{}등이 필요하나, Create List, Create Dictionary 등 keyword 사용시에는 ${} 사용해도 무방. @{}, &{} 사용시에는 framework에서 추가적인 validation 과정을 수행함**

With dictionary variables, the assignment does another thing, which is it turns it into a Robot Framework DotDict. A DotDict is otherwise identical to a normal Python dictionary, but allows accessing values through dot-notation: ${my\_dict.two}.

**Note**: If you assigned to @{my\_list}, you can always access it through ${my\_list}, and vice-versa. They refer to the same variable.

While keys of a DotDict can be accessed the same as keys in a standard dict, they can also be accessed as attributes.

* **RPA Framework에서는 DotDict가 동작하지 않는 듯**

**${GLOBAL\_DICT}**= Create Dictionary 1000=[1,2,3,4,5,6]

Log To Console \n----1.0 **${GLOBAL\_DICT}**[1000]

Log To Console \n----1.0 **${GLOBAL\_DICT.1000}** # failed: SyntaxError: unexpected EOF while parsing (<string>, line 1)

The second place where this syntax is used is when calling keywords. With list or dictionary variables, they are ***unpacked*** as arguments automatically. If you are familiar with the Python syntax \*args or \*\*kwargs, this is similar to that.

For example, here we call the first keyword in three different ways: with individual arguments, by unpacking a list, and by unpacking a dictionary:

**Example keyword**

[Arguments] ${first\_arg} ${second\_arg}

Log ${first\_arg} and ${second\_arg}

${the type}= Evaluate type(${first\_arg})

Log The type of 1st argument: ${the type}

${size}= Get Length ${first\_arg}

Log Size: ${size}

A second keyword

${as\_list}= Create list 1 2

${as\_dict}= Create dictionary first\_arg=1 first\_arg=2

# All three of the following will log "Got arguments 1 and 2"

**Example keyword** 1 2 // 1 and 2, <class ‘int’>, Size:1

**Example keyword** **@{as\_list}** // 1 and 2, <class ‘int’>, Size: 1

**Example keyword** **${as\_list}** dummy // 2개 param 필요, ['1', '2'] and dummy, <class ‘list’>, Size: 2

**Example keyword** **&{as\_dict}** // 1 and 2, <class ‘int’>, Size: 1

**Example keyword** **${as\_dict}** dummy // 2개 param 필요, {'first\_arg': '1', 'second\_arg': '2'} and dummy, <class ‘dict’>, Size: 2

* **Hint: 타 keyword 호출 시 list, dictionary 타입 변수도 ${} 형식으로 전달해야 한다 (parameter 개수, 원래 타입 유지, 각 멤버 데이터 활용 등에서)**

This is also why it is used when FOR looping. These two examples are identical:

Loop through scalars

${name\_one}= Set variable John

${name\_two}= Set variable Robert

FOR ${name} IN ${name\_one} ${name\_two} Mark

Log Name is ${name} // John, Robert, Mark 각각 수행, 총 3회 수행

**END**

Loop through list1

${names}= Create list John Robert Mark

FOR ${name} IN @{names} // John, Robert, Mark 각각 수행, 총 3회 수행

Log Name is ${name}

**END**

Loop through list2

${names}= Create list John Robert Mark

FOR ${name} IN ${names} // [‘John’, ‘Robert, ‘Mark’] 없다고 오류, 1회 수행

Log Name is ${name}

**END**

If the second FOR loop used **${names}** instead of @{names}, it would not loop the names *inside* the list, but would instead only loop once and return the list itself.

* **Hint: list 타입 local 변수는 ${} (즉, scalar 변수) 사용시 scalar 변수로 인식되므로, @{}의 원래 타입으로 사용**

${as\_dict}= Create dictionary first\_arg=1 first\_arg=2

${key\_list}= Create List first\_arg second\_arg

**FOR** ${val} IN @{key\_list}

TRY

${value}= Get From Dictionary &{as\_dict} ${val} // 오류: got positional argument after named arguments. Try없이 종료됨

${value}= Get From Dictionary ${as\_dict} ${val} // 정상 동작

Log To Console ----13-1. ${value} found

EXCEPT

Log To Console ----13-2. ${val} not in keys

END

END

With dictionary variables, the assignment does another thing, which is it turns it into a Robot Framework DotDict. A DotDict is otherwise identical to a normal Python dictionary, but allows accessing values through dot-notation: ${my\_dict.two}.

* **Hint: Dictionary 타입 local 변수는 &{}의 원래 타입으로 사용 시 Try-Except 에 인식되지 않는 오류 발생하므로, ${} (즉, scalar 변수) 사용**

So in conclusion: We don’t use the @-syntax when assigning a Table to a variable because it is optional, but it would work because a Table is list-like. We also have to use the @-syntax when looping over a Table, because we want to loop over the rows inside of it.

* **Hint: list 타입 변수는 Table을 assign할 때 ${} 사용, For loop 등에서는 @{}를 사용**

**RPA Framework Forum에서 발췌 #2**

In general Robot Framework tries to make the type handling as simple as possible, but this is one case where it is a bit “odd”.

I believe that the value in integer is in fact a str. But Evaluate type(${integer}) does a direct string replacement and replaces the variable name with it’s value. Thus it becomes Evaluate type(100), which ofc returns type int. If you want to get the type of the value actually stored in the variable you need to use different (less known) variation of variable reference: Evaluate type($integer), which will return the real type str.

Here is a screen shot of similar case:

User dir

${value}= Set Variable 100

${type1}= Evaluate type($value)

Log ${type1} // <class ‘str’>

${type2}= Evaluate type(${value})

Log ${type2} // <class ‘int’>

**RPA global variable 사용법 #1**

(나의 테스트를 분석한 내용으로 좀 더 확인 필요)

\*\*\* Variables \*\*\* section에 선언된 변수는 일종의 global임

\*\*\* Variables \*\*\*

&{DICK\_DATA}

* **${DICK\_DATA} 를 이용하여 타 keyword 내에서 사용 가능**

Sub-keyword에서 dictionary type 변수를 return 또는 global 변수를 직접 access하던 OK

단,

임의의 keyword 내에서 “Create Dictionary” keyword를 사용하면 생성된 dictionary 변수는 global variable과 이름이 동일해도 local variable.

따라서 업무 수행 후 별도 처리없이 keyword를 종료하면 global variable의 dictionary 변수에는 영향이 없다

**RPA global variable 사용법 #2**

* **Global variable과 local variable 사용법에 대한 추가 검토 필요**

Global Variable을 선언하고, 각 keyword 내에서 global variable의 데이터 존재 여부를 확인하면 도중에 데이터가 없어지고 초기화됨.

따라서 아래와 같이 각 keyword에서 RETURN하고 다음 keyword에서 Argument로 받으면 이상 없음

(터무니 없는 얘기로 차근차근 검토 필요)

Prepare resources

# --- 삭제 Get Global Dict

**${ALL\_PICKS}**= Load data

Open Page

**${ALL\_PICKS}**= Gather Numbers **${ALL\_PICKS}**

Save to File **${ALL\_PICKS}**

Recommend Number