

## **BKM for Bug-Link Script**

The provided code allows creation and updating the test result for each test case present in a given test cycle, with options to specify the link type, status, reason, and owner. The input is a JSON file that can be easily modified to suit the user's needs and the script can be run using either Basic AUTH or Kerberos AUTH authentication APIs.

**Note:** For the testcases which have `status.reason == "complete.fail"`, the script will fetch the respective bug-ids for each testcase from the input csv file and link the BUG HSDS QUERY ID with respective Test Result.

### **JSON File Inputs:**

- a) **Path:** It is the path to csv file which contains all the TC\_IDs and their respective TC\_Title and configuration.  
**NOTE:** Order should be maintained in the csv file [TC\_ID, TC\_Title, Configuration, Bug\_id] and first row should contain the Title of each column.

**TC\_ID:** Test Case Id

**TC\_Title:** Test Case Title

**Please note that Bug\_Id column is only required in case when `status.reason == complete.fail`**

- b) **Test Cycle:** It is the name of a particular test cycle. In creation of new TR's this test cycle name will be written in the appropriate field.
- c) **Link-Type:** It should be "parent-child" in case of updating and creating the TR's.
- d) **Status:** String indicating the status of a TC for a given test cycle and configuration.
- e) **Reason:** String indicating the reason (pass, fail, optimized, warning, new, etc.)
- f) **Release\_Aff:** It will be the program release name to which TC refers, example: bios.alderlake
- g) **Owner:** User-id of person making the changes, example "namanaga"
- h) **Reason\_other:** This field is required when status\_reason is set to "blocked.other"

**Note:** For `status.reason == "blocked.other"`, please provide some input in the "reason\_other" field. It is mandatory to provide input in this case else the Test Result's won't be updated.

There are two Python Scripts included in the folder: "bug\_link\_preprod.py" for testing on the pre-prod HSDS server, and "bug\_link\_prod.py" for use on the HSDS production server. These two files fetch input from JSON file.

The new TEST RESULT created will have “title”, “parent-id”, “owner”, “link-type”, “status”, “reason” and “release” fields fetched from the JSON input file.

It should be noted that values provided in the JSON file will be applied to all the test results created or updated. Additionally, if using Basic AUTH, valid credentials can be obtained using provided link <https://hsdes.intel.com/appstore/token/>.

**Observations:**

- a) In a single API call to update or create, only 25 test results can be created or updated.
- b) For a csv containing 700-800 TC IDs, the script may take up to 20 mins to process and execute.