ESC: Software Testing Mini Campaign

**Equivalence Class Parititoning & Boundary Value Analysis**

Scenario 1: Valid/Invalid arguments passed to system

* Input Space: Arguments passed into system
* **Equivalence Class Paritions**:
  + Category 1: Number of arguements
    - **Invalid Partion 1: No argument passed to system**
      * Boundary Value Analysis
        + Minimum**:** No argument passed
    - **Invalid Partion 2: 1 argument passed to system**
      * Boundary Value Analysis
        + Minimum**:** 1 argument passed
    - **Valid Partion 1: 2 arguments passed to system**
      * Boundary Value Analysis
        + Minimum**:** 2 argument passed
    - **Invalid Partion 3: >2 arguments passed to system**
      * Boundary Value Analysis
        + Just Above Minimum:3 arguments passed; 4 arguments passed, 5 arguments passed..
        + Middle: 10 Arguments
        + Just below maximum: 1000 arguments passed
  + Category 2: Validity of argument
    - **Invalid Partion 1: Argument contains illegal characters for a file path**
      * Boundary Value Analysis
        + Just Above Minimum:#%&{}/$!`”:@
        + Middle: my%other#secret{file
        + Just Below Maximum: my#file
    - **Invalid Partion 2: Argument does not contain illegal character but is not a file path (i.e. incorrect file path syntax, has no extension)**
      * Boundary Value Analysis
        + Just Above Minimum:file\c:;
        + Middle: C:\\\\; C:\myfile, myfile;
        + Just Below Maximum: C:\app\myfile
    - **Invalid Partion 3: Argument is a file path, but the file extension is not “.csv”**
      * Boundary Value Analysis
        + Just Above Minimum:C:\app\myfile.txt
        + Middle: C:\app\myfile.c
        + Just Below Maximum: C:\app\myfile.cs
    - **Invalid Partion 4: Argument is a file path with “.csv” file extension, but file path length exceeds 256 characters**
      * Boundary Value Analysis
        + Just Above Minimum:app0123456789101112131415161718192021222324252627282930313233343536373839404142434445464748495051525354555657585960616263646566676869707172737475767778798081828384858687888990919293949596979899100101102103104105106107108109110111112113114115116117118119.csv
    - **Invalid Partion 5: Argument is a file path with “.csv” file extension, with <= 256** character but file doesn’t exist
      * Boundary Value Analysis
        + Just Above Minimum:myfile.csv
        + Middle: app123456789101112131415161718192021222324252627282930313233343536373839404142434445464748495051525354555657585960616263646566676869707172737475767778798081828384858687888990919293949596979899100101102103104105106107108109110111112113114115116117118119.csv
        + Just Below Maximum: C:\app\app\23456789\0\2\3\4\5\6\7\8\9202\2223242526272829303\3233343536373839404\4243444546474849505\5253545556575859606\6263646566676869707\7273747576777879808\8283848586878889909\9293949596979899\00\0\02\03\04\05\06\07\08\09\0\2\3\4\5\6\7\8\9\101112.csv
    - **Invalid Partion 6: Argument is a file path to an existing csv file but system is not able to access/read file**
      * **Subclass 1: system has no read permission to file**
        + Boundary Value Analysis

Just Above Minimum:myfile.csv

* + - * **Subclass 2: other processes are writing to or locking file**
        + Boundary Value Analysis

Just Above Minimum:myfile.csv

* + - * **Subclass 3: file is an online-only cloud file**
        + Boundary Value Analysis

Just Above Minimum:myfile.csv

* + - **Valid Partion 1: Argument is a file path to an existing csv file and system is able to read file**
      * Boundary Value Analysis
        + Just Above Minimum:myfile.csv
        + Middle: app123456789101112131415161718192021222324252627282930313233343536373839404142434445464748495051525354555657585960616263646566676869707172737475767778798081828384858687888990919293949596979899100101102103104105106107108109110111112113114115116117118119.csv
        + Just Below Maximum: C:\app\app\23456789\0\2\3\4\5\6\7\8\9202\2223242526272829303\3233343536373839404\4243444546474849505\5253545556575859606\6263646566676869707\7273747576777879808\8283848586878889909\9293949596979899\00\0\02\03\04\05\06\07\08\09\0\2\3\4\5\6\7\8\9\101112.csv

Scenario 2: Valid/Invalid file format passed

* Input Space: File’s data content
* **Equivalence Class Paritions:**
  + Category 1: Comma seperated file content format
    - **Invalid Partion 1: File is empty and has no content**
      * Boundary Value Analysis
        + Minimum:File is empty
    - **Invalid Partion 2: File is not empty but file is not encoded in UTF-8 format**
      * Boundary Value Analysis
        + Just Above Minimum:File is UTF-32, UTF-16, ASCII, ansi encoding
    - **Invalid Partion 3: File contents are in UTF-8 format, but file only contains whitespaces and/or only newlines**
      * Boundary Value Analysis
        + Just Above Minimum:File contains: “ “; “\n”
        + Middle: File contains: “ \n \n”
        + Just Below Maximum: File CONTAINS: “ \n\n\n \t\t\t “
    - **Valid Parition 1: File contents are in UTF-8 encoding and contains 1 line of non-whitespace/newlines content**
      * Boundary Value Analysis
        + Just Above Minimum: “mycol1”
        + Middle: “mycol1,mycol2”
        + Just Below Maximum: “mycol1,mycol2,mycol3,mycol4”
    - **Invalid Parition 4**: File contents are in UTF-8 encoding and contains more than 1 line of non-whitespace/newlines content but the number of commas (which are not wrapped within double quotes) between lines are inconsistent
    - Valid Partion 2: File contents are in UTF-8 encoding and contains more than 1 line of non-whitespace/newlines content and the number of commas (which are not wrapped within double quotes) between lines are consistent for all lines

Scenario3: File comparasions

* Input Space: 2 Files
* **Equivalence Class Paritions:**
  + Category 1: CRLF vs LF inconsistency (Assuming both uses UTF-8) between files
  + Category 2: Commas inconsistencies between files
  + Category 3: Column Headers Provided
  + Category 4: Column Headers Not Provided
  + Category 5: Exact identical file is referenced for both comparasions

Scenario 3: Output CSV file

* Input Space: System reconciliation output
* **Equivalence Class Paritions:**
  + Category 1: File is in csv format
    - is of extension is “.csv”
    - output file is not empty even if csv input files are empty or identical
    - data is UTF-8 CLRF (windows) encoded
    - data is comma seperated
    - File overwrites existing output file if it exists
    - Out put file name is appended

Scenario 4: Atomicity of CSV output file

* Input Space: System reconciliation output
* **Equivalence Class Paritions:**
  + Category 1: Atomicity when interrupted or fails midway (Either complete output on succesfull completion or no output)