SCC Module/insert_yard_conf.py

Main Program

Start

if log.logger is NONE — **NO** →

YES

my_log = log()

Read scc.conf file using read configuration method of SccDimConfRead class.

connect database with scc API using 'connect.database' method of Scc_API class.

Read Section Configuration by passing 'section.conf' file into read configuration method of section configuration class.

get number of yards using length 'yards' column from yard_config.conf file.

i+=1 →

for yard_idx in range(number of yard)

dpu_idx+=1 →

for dpu_idx in range(number of **DPU**)

section_idx+=1 →

for section_idx in range(number of **sections**)

Save (yard id, yard name, DPU ID, DPU name, Section ID, Section type, DP ID) into yard configuration table through 'YardConfigInfo' class.

Class SectionConfig

Constructor with no attributes.

Create a empty string variable to store json_data file.

method to read configuration file with attribute 'file name' of json data.

if file exist which is taken as attribute — **NO** → log critical message "Program Terminated file not found"

YES

open file using "with " clause

Try Clause: convert file from JSON to python object using JSON.load() and load the file in variable created in constructor

if file loaded successfully — **NO** → Except Clause: log critical message "program terminated due to invalid json configuration file"

YES

return python object file.

method to print configured file

Try clause: to print the file it will convert python object file to JSON file using JSON.dumps()

if file converted successfully — **NO** → Except Clause: log critical message with error info.

YES

return JSON file.

---

if file exist which is taken as attribute — **NO** → log critical message "Program Terminated file not found"

YES

CONNECTORS

← ← → →

FLOW CHART SHAPES

Top Level

Second Level

Third Level

Fourth Level

i +=1 →

NO

j+=1