

PoliCal

Release 0.1.6

Luis Andrade

Jul 14, 2020

1	PoliCal main	1
2	PoliCal Get_Trello_MoodleEPN_Keys	3
3	PoliCal configuration	5
4	PoliCal SendTaskToTrello	7
5	PoliCal SimpleIcsToCSV	9
6	PoliCal TareasCSVToBD	11
7	PoliCal MateriasLoaderToDB	13
8	PoliCal connectSQLite	15
	Python Module Index	19
	Index	21

PoliCal Get_Trello_MoodleEPN_Keys

class polical.Get_Trello_MoodleEPN_Keys.DevNullRedirect

Temporarily eat stdout/stderr to allow no output. This is used to suppress browser messages in webbrowser.open

polical.Get_Trello_MoodleEPN_Keys.**check_file_existence** (*output_file*)

This functions checks if the file already exists on filesystem

Args:

output_file (str): File location

Returns:

bool. The return code:

<p>False -- If the file not exists or is blank True -- If already exists</p>

polical.Get_Trello_MoodleEPN_Keys.**check_user_on_file** (*output_file, username*)

This functions checks if an user already exists on yaml file.

Args:

output_file (str): File location username (str): The username that would be checked.

Returns:

Stop onboard function if the username already exists and the user answers N to overwrite it.

polical.Get_Trello_MoodleEPN_Keys.**get_working_board_id** (*api_key, api_secret, oauth_token, oauth_token_secret*)

This function is for getting the TareasPoli Board ID that is created in the onboard process

Args:

api_key (str): The api key to acceso Trello api_secret (str): The api secret to acceso Trello oauth_token (str): The oauth token to acceso Trello oauth_token_secret (str): The oauth token secret to acceso Trello

Returns:

board_id (str): This is the board id of TareasPoli board owner_id (str): This is the owner id of Trello user

polical.Get_Trello_MoodleEPN_Keys.**onboard** (*no_open, output_path='polical.yaml'*)

Obtain Trello API credentials and put them into your config file. This is invoked automatically the first time you attempt to do an operation which requires authentication. The configuration file is put in an appropriate place for your operating system.

PoliCal configuration

`polical.configuration.Add_Subject_To_Trello_List (subjectsBoard, subject_name, subjCod)`

This function adds a list to trello board with subject name.

Args:

subjectsBoard (Trello.Board): Tareas Poli's Board object from Trello library subject_name (str): Subject name to create new list. subjCode (str): Subject code to add it to name list.

`polical.configuration.check_for_url (url)`

This function is for checking moodle calendar url

Args:

url (str): Moodle Calendar Address to be checked

Returns:

bool. The return code:

```
False -- If the url does not start with https and ends with recentupcoming
True  -- If the url starts with https and ends with recentupcoming
```

`polical.configuration.create_subject (subjCod, task_title, user_dict)`

This function creates a subject in Trello and adds it to local database.

Args:

subjCod (str): Subject Code to check with local database and trello. task_title (str): Subject title for showing to the user if subject is not founded in local database. user_dict (dict): User dictionary with keys to connect to trello.

`polical.configuration.get_file_location (filename)`

This function is for getting full path location of a file from its filename

Args:

filename (str): Filename that needs the full path location

Returns:

full_path_file_location (str): Full path location of the file

`polical.configuration.get_working_directory ()`

This functions gets the working directory path.

Returns:

workingDirectory (str): The directory where database and yaml are located.

`polical.configuration.load_config_file (config_file_path)`

This function is for loading yaml config file

Args:

config_file_path (str): File path for the config file to be loaded

Returns:

file_config (dict): Dictionary with config keys.

Raises:

IOError

PoliCal SendTaskToTrello

`polical.SendTaskToTrello.SendTaskToTrello (username, user_dict)`

This function sends tasks from database that are stored as not send to trello.

Args:

username (str): The username for the current task. user_dict (dict): User dictionary with keys to access to trello.

PoliCal SimpleIcsToCSV

`polical.SimpleIcsToCSV.addEvent (header, filename)`

This function adds a event as a new line in csv file.

Args:

header (list): Header for the csv. filename (str): The filename where file would be written.

Returns:

None. If not has headers

`polical.SimpleIcsToCSV.convertICStoCSV (url)`

This function downloads the moodle calendar and addEvents to a CSV file.

Args:

url (str): URL to download moodle calendar

`polical.SimpleIcsToCSV.findHeader (icsCal)`

This function looks for all the file to get the most longest header.

Args:

icsCal (str): The ics file location.

Returns:

(list): List containing the largest header list.

PoliCal TareasCSVToBD

`polical.TareasCSVToBD.Get_Subject_Name_From_CSV (full_subject_name)`

This function gets subject name from csv

Args:

full_subject_name (str): Full subject name with format XXX_YYY_ZZZ

Return:

subject_name (str): Subject name

`polical.TareasCSVToBD.LoadCSVTasktoDB (username, user_dict)`

This function loads csv tasks to the database

Args:

username (str): The username for the current task. user_dict (dict): User dictionary with keys to acces to trello.

Raises:

FileNotFoundError

PoliCal MateriasLoaderToDB

`polical.MateriasLoaderToDB.loadSubjectToDB ()`

This function loads any subject located on materias.csv to the database.

PoliCal connectSQLite

`polical.connectSQLite.addTarTID (TarUID, TarTID, username)`

This function adds the task trello ID into the database

Args:

TarUID (str): Task UID from ICS file. TarTID (str): New Task Trello ID from trello. username (str): Username from the user that owns the task

Returns:

cur (Cursor): Database cursor that access to tasks and subjects.

`polical.connectSQLite.check_no_subjectID (subjCod)`

This function checks if the subject has an ID in the database.

Args:

subjCod (str): Subject code from the database to check if it has ID or not.

Returns:

result (str): Returns '0' if does not has the ID and '1' if it has it.

`polical.connectSQLite.check_user_existence (username)`

This function checks if the username has an ID in the database.

Args:

username (str): username from the database to check if it has ID or not.

Returns:

result (str): Returns '0' if does not has the ID and '1' if it has it.

`polical.connectSQLite.exec (command)`

This function executes a command in the database.

Args:

command (str): Query that needs to be executed on the database.

Returns:

cur (Cursor): Database cursor that access to tasks and subjects.

`polical.connectSQLite.getCardsdb (db)`

This function get all cards from Tareas table.

Args:

db (Connection): Database connection.

Returns:

cards (list): List containing all cards from Tareas table.

`polical.connectSQLite.getCur ()`

This function returns the database cursor

Returns:

cur (Cursor): Database cursor that access to tasks and subjects.

`polical.connectSQLite.getSubjectID (subjCod)`

This function gets the subject ID from the database

Args:

subjCod (str): Subject code from the subject to get the ID.

Returns:

sbjID (str): Subject ID from the subject.

`polical.connectSQLite.getSubjectName (subjCod)`

This function gets the subject Name from the database

Args:

subjCod (str): Subject code for get the subject name.

Returns:

sbjName (str): The subject name from the subject code.

`polical.connectSQLite.getTaskID (TarUID)`

This function gets the task ID from the database

Args:

TarUID (str): Task UID from the task to get the ID.

Returns:

idTareas (str): Task ID from the task.

`polical.connectSQLite.getTasks (username)`

This function gets all unsended tasks from the user.

Args:

username (str): Username from the user that owns the task

Returns:

tasks (list): Database cursor that access to tasks and subjects.

`polical.connectSQLite.getUserID (username)`

This function gets the User ID from the database

Args:

username (str): Username to get his ID.

Returns:

idUsuarios (str): The user id from the database.

`polical.connectSQLite.getdb ()`

This function returns the sqlite3 database connection that storages all tasks and subjects.

Returns:

db (Connection): Database connection that access to tasks and subjects.

`polical.connectSQLite.saveSubjectID (subject)`

This function saves the trello list ID into the database

Args:

subject (Materia): Subject that owns the ID that would be added to the database.

Returns:

cur (Cursor): Database cursor that access to tasks and subjects.

`polical.connectSQLite.saveSubjects (subject)`

This function saves a subject into the database

Args:

subject (Materia): Subject that would be added to the database.

Returns:

cur (Cursor): Database cursor that access to tasks and subjects.

`polical.connectSQLite.saveTask (task, username)`

This function saves a task from a user into the database

Args:

task (Tarea): Tasks that would be added to the database. username(str): User owner of the task.

Returns:

cur (Cursor): Database cursor that access to tasks and subjects.

`polical.connectSQLite.saveUser (username)`

This function saves a user into the database

Args:

username (str): User to be added into the database

Returns:

cur (Cursor): Database cursor that access to tasks and subjects.

c

configuration, 5

g

Get_Trello_MoodleEPN_Keys, 3

p

polical

- polical.__main__, 1
- polical.configuration, 5
- polical.connectSQLite, 15
- polical.Get_Trello_MoodleEPN_Keys,
3
- polical.MateriasLoaderToDB, 13
- polical.SendTaskToTrello, 7
- polical.SimpleIcsToCSV, 9
- polical.TareasCSVtoBD, 11

s

SendTaskToTrello, 7

SimpleIcsToCSV, 9

t

TareasCSVtoBD, 11

A

Add_Subject_To_Trello_List() (in module polical.configuration), 5
addEvent() (in module polical.SimpleIcsToCSV), 9
addTarTID() (in module polical.connectSQLite), 15

C

check_file_existence() (in module polical.Get_Trello_MoodleEPN_Keys), 3
check_for_url() (in module polical.configuration), 5
check_no_subjectID() (in module polical.connectSQLite), 15
check_user_existence() (in module polical.connectSQLite), 15
check_user_on_file() (in module polical.Get_Trello_MoodleEPN_Keys), 3
configuration module, 5
convertICStoCSV() (in module polical.SimpleIcsToCSV), 9
create_subject() (in module polical.configuration), 5

D

DevNullRedirect (class in polical.Get_Trello_MoodleEPN_Keys), 3

E

exec() (in module polical.connectSQLite), 15

F

findHeader() (in module polical.SimpleIcsToCSV), 9

G

get_file_location() (in module polical.configuration), 5
Get_Subject_Name_From_CSV() (in module polical.TareasCSVToBD), 11
Get_Trello_MoodleEPN_Keys module, 3
get_working_board_id() (in module polical.Get_Trello_MoodleEPN_Keys), 3
get_working_directory() (in module polical.configuration), 5
getCardsdb() (in module polical.connectSQLite), 15
getCur() (in module polical.connectSQLite), 16
getdb() (in module polical.connectSQLite), 16
getSubjectID() (in module polical.connectSQLite), 16
getSubjectName() (in module polical.connectSQLite), 16
getTaskID() (in module polical.connectSQLite), 16
getTasks() (in module polical.connectSQLite), 16
getUserID() (in module polical.connectSQLite), 16

L

load_config_file() (in module polical.configuration), 6
LoadCSVTasktoDB() (in module polical.TareasCSVToBD), 11
loadSubjectToDB() (in module polical.MaterialsLoaderToDB), 13

M

module
configuration, 5
Get_Trello_MoodleEPN_Keys, 3
polical.__main__, 1

- polical.configuration, 5
- polical.connectSQLite, 15
- polical.Get_Trello_MoodleEPN_Keys, 3
- polical.MateriasLoaderToDB, 13
- polical.SendTaskToTrello, 7
- polical.SimpleIcsToCSV, 9
- polical.TareasCSVtoBD, 11
- SendTaskToTrello, 7
- SimpleIcsToCSV, 9
- TareasCSVtoBD, 11

O

- onboard() (in module polical.Get_Trello_MoodleEPN_Keys), 3

P

- polical.__main__
 - module, 1
- polical.configuration
 - module, 5
- polical.connectSQLite
 - module, 15
- polical.Get_Trello_MoodleEPN_Keys
 - module, 3
- polical.MateriasLoaderToDB
 - module, 13
- polical.SendTaskToTrello
 - module, 7
- polical.SimpleIcsToCSV
 - module, 9
- polical.TareasCSVtoBD
 - module, 11

S

- saveSubjectID() (in module polical.connectSQLite), 16
- saveSubjects() (in module polical.connectSQLite), 17
- saveTask() (in module polical.connectSQLite), 17
- saveUser() (in module polical.connectSQLite), 17
- SendTaskToTrello
 - module, 7
- SendTaskToTrello() (in module polical.SendTaskToTrello), 7
- SimpleIcsToCSV
 - module, 9

T

- TareasCSVtoBD
 - module, 11