# Python scripts documentation

## GLDFile

Functions:

* extract\_data (input\_file\_name, create\_output\_file, output\_folder)

Parses a GL5 file and extracts raw data and event information.

### Parameters

* **input\_file\_name**: str; GL5 file name;
* **create\_output\_file**: bool; True = creates a file containing the output object; False = doesn’t create a file containing the output object;
* **output\_folder**: str; file path for the output file; if create\_output\_file is False than this parameter isn't used.

### Return value

**T object**:

A class used to represent the extracted data

**Attributes**:

* **segments**: list[Segment];o f the events data;
* **file\_info**: FileInfo; information about the file;
* **device**: Device; information about the GL5 device.

**Segment**

A class used to represent the event data

**Attributes**:

* **drive** : Dict[int,Drive]; key: hemisphere(1 = left, 2 = right); value: Drive object;
* **offset\_ac\_mer** : Dict[int, np.uint32]; key: channel, value: AC offset calibration constant;
* **offset\_dc\_mer** : Dict[int, np.uint32]; key: channel, value: DC offset calibration constant
* **start\_timestamp\_mer**: np.uint32; MER start timestamp
* **start\_timestamp\_lf**: np.uint32; LF start timestamp
* **sampling\_rate\_mer**: list[np.single]; MER sampling rate
* **sampling\_rate\_lf**: list[np.single]; LF sampling rate;
* **v\_cal\_mer**: Dict[int, np.single]; key: channel; value: Scale raw MER data to microvolts
* **v\_cal\_lf**: Dict[int, np.single]; key: channel; value: Scale raw LF data to microvolts
* **channels**: Dict[int, Channel]; key: channel; value: Channel object
* **stim\_on**: np.uint32; stimulation on timestamp
* **stim\_off**: np.uint32; stimulation off timestamp
* **sync**: Sync;
* **motion**: motion
* **offset\_dc\_lfs**: Dict[int, np.uint32]; key: channel ,value: DC offset calibration constant
* **aux**: Aux
* **stim\_params**: List[StimParams]

**Drive**

**Attributes**:

* **timestamps**: List[np.uint32];
* **drive\_depths**: List[np.uint32];

**Channel**

A class used to represent the signal data

**Attributes**:

* **continuous** : List[np.single];extracted MER data for the segment, in microvolts;
* **lf**: List[np.single]; extracted LFP data for the segment, in microvolts.

**Sync**

**Attributes**:

* **rt\_timestamps** : List[np.uint32]; timestamps of the real time digital input on the Sync unit;
* **timestamps**: List[np.uint32];
* **port1**: List[np.uint32];
* **port2**: List[np.uint32];
* **digin**: List[np.uint32];

**Motion**

**Attributes**:

* **sampling\_rate**: int;
* **start\_timestamp**: int;
* **data**: MotionData;

**MotionData**

**Attributes**:

* **timestamps**: List[np.int64]; array of 64-bit integers that match the binary representation of a System.DateTime .NET Framework object;
* **x**: list[np.single]; Accelerometer X axis values;
* **y**: list[np.single]; Accelerometer Y axis values;
* **z**: list[np.single]; Accelerometer Z axis values;

**Aux**

**Attributes**:

* timestamps\_aux: List[np.uint32];
* channels: Dict[int, Channel]; key: channel; value: Channel object;
* sampling\_rate: np.uint32;

**StimParams**

A class used to represent the stimulation parameters

**Attributes**:

* **stim\_type**: np.uint32; 1 - macro, 0 – micro;
* **stim\_mode**: np.uint32; 1 - constant voltage, 0 - constant current;
* **output\_channel\_map**: np.uint32; output channel bitmap;
* **return\_channel\_map**: np.uint32; return channel bitmap;
* **pulse**: Dict[int,Pulse]; key:;value:pulse parameters object

**Pulse**

**Attributes**:

* **freq**: np.uint32;
* **channel**: np.uint32;
* **amplitude**: np.uint32;
* **duration**: np.uint32;
* **phase**: np.uint32;
* **polarity**: np.uint32;

**FileInfo**

**Attributes**:

* **version** : np.uint32; file version;
* **deviceId**: np.uint32; device identifier (0 - GL5 port 1 or 1 - GL5 port 2);

## GLDAlign

Functions:

* gld\_align(merFile, eogFile)

This script performs data alignment given two Guideline 5 .gld file, where one is configured as uE Interface and the other as Lf Interface.

### Parameters

* **merFile** - Full path to the uE Interface data file;
* **eogFile** - Full path to the Lf Interface data file.

### Return values

**t1** – T object holding the uE Interface aligned data;

**t2** – T object holding the Lf Interface aligned data;

## GLDViewAligned

Functions:

* gld\_view\_aligned(t1, t2)

Call this function after running gld\_align to visualize the aligned data.

### Parameters

* **t1** – T object holding the uE Interface aligned data;
* **t2** – T object holding the Lf Interface aligned data;

### Assumptions

* uE Interface : Channel 1, microelectrode recording, sampled at 32 kHz.
* Lf Interface : Channel 1, EOG channel, sampled at 1 kHz.
* Sync Interface: Digin 1, TTL input, sampled at 32 kHZ.