
citram*pythonapi*
Release 1.0.3

Feb 15, 2020

Contents

1	Modules description	3
2	Some tips	5
3	Example of use	7
4	Contents	9
	Python Module Index	37
	Index	39

Python wrapper of Consorcio de Transportes de Madrid (CTRM) API.

This module seeks to comprise all the CTRM endpoints, using different CTRM APIs.

With this API you can request static and live (not as a streaming) for the different public transport the CTRM manages: Madrid Metro, MetroSur Metro, Cercanías Renfe, Madrid EMT buses, intercity buses, multi-transport stations (intercambiadores), and Metro Ligero. You can request information of CTRM offices and parkings.

Any kind of transport fits in the paradigm of stops and lines.

CHAPTER 1

Modules description

- lines: Requests relative to transport lines.
- offices: Requests relative to CTRM offices.
- stops: Requests relative to transport stops.
- others: Other relevant requests that don't fit in any of the other categories.
- constants: Useful constants to use when making requests.
- utils: Useful functions to use when making requests.

CHAPTER 2

Some tips

- When writing line names, they're usually in upper case and have no whitespaces nor other characters like dashes or underscores. The functions in `utils.utils` can help you to create stop and line codes.
- With the `constants` module you can make requests about municipalities, transport modes and office types easily.
- In case your mother tongue is English and you're having trouble understanding the different transports, you might find this link helpful: <https://www.crtm.es/widgets/language.json>
- Sometimes you'll see kmz links. kmz files are used by Google Maps or Google Earth to display a set of points or geometric figures in a map.
- When the field `access` is greater than 0, it means the stop is adapted to people with special conditions.
- When the field `park` is greater than 0, it means that stop has a parking.
- When the field `nightService` or `nightLinesService` is greater than 0, it means that line has night service (sometimes this is called *Búho*, specially on bus lines).

CHAPTER 3

Example of use

```
from citram_api.constants.constants import TransportModes
from citram_api.api.lines.lines import get_line_info
from citram_api.utils.utils import create_line_cod

line_info = get_line_info(create_line_cod(TransportModes.METRO.value, 10))
```


4.1 citram_api.api.lines package

4.1.1 Submodules

4.1.2 citram_api.api.lines.lines module

`citram_api.api.lines.lines.get_incidents_affectations(mode_cod, cod_line)`

Returns incidents happening at the specified line.

Example: `get_incidents_affectations(TransportModes.METRO.value, create_line_cod(TransportModes.METRO.value, 10))`

Parameters

- **mode_cod** (*int*) – Id of a public transport. Use constants.TransportModes to easily select transport modes ids.
- **cod_line** (*str*) – Line id. Use `utils.create_line_cod` to create this id easily.

Return dict The current incidents going on in that line.

`citram_api.api.lines.lines.get_line_info(cod_line)`

Returns line information from the line id specified.

Example:

```
get_line_info(create_line_cod(TransportModes.METRO.value, 10))
```

Parameters **cod_line** (*str*) – Line id. Use `utils.create_line_cod` to create this id easily.

Returns Info of the line id specified. The returned dictionary look likes the following example:

```

{
  'lines': {
    'LineInformation': {
      'codLine': '4__10__',
      'shortDescription': '10',
      'description': 'Hospital del Norte-Puerta del Sur',
      'codMode': '4',
      'codMunicipalities': {
        'string': ['4278', '4279', '4350', '4401']
      },
      'itinerary': {
        'Itinerary': [{
          'codItinerary': '4__10____1__IT_1',
          'name': 'Hospital del Norte - Puerta del Sur',
          'direction': 1,
          'kml': 'http://www.citram.es:8080/kml/itinerarios/20160601_
↪1704/metrokmz/M4_L10a_S1_TRAMO.kmz#http://www.citram.es:8080/kml/itinerarios/
↪20160601_1704/metrokmz/M4_L10b_S1_TRAMO.kmz',
          'stops': {
            'StopInformation': [{
              'codStop': '4_284',
              'shortCodStop': '',
              'codMode': '4',
              'name': 'HOSPITAL INFANTA SOFÍA',
              'address': 'Paseo Europa SN ',
              'postCode': '',
              'codMunicipality': '4401',
              'coordinates': {
                'longitude': -3.61145,
                'latitude': 40.55977
              },
              'lines': {
                'Line': {
                  'codLine': '4__10__',
                  'shortDescription': '10',
                  'description': 'Hospital del Norte-Puerta del
↪Sur',
                  'codMode': '4',
                  'updateDate': '2019-03-22T17:21:37+01:00',
                  'updateKmlDate': '2016-06-01T17:04:00+02:00',
                  'nightService': 0,
                  'active': True,
                  'shortItinerary': {},
                  'companyCode': ''
                }
              },
              'access': 0,
              'park': 0,
              'nightLinesService': 0
            },
            ...
          ],
          'updateDate': '2019-03-22T17:21:37+01:00',
          'updateKmlDate': '2016-06-01T17:04:00+02:00',
          'nightService': 0,
          'lineTimePlanning': {
            'codLine': '4__10__',

```

(continues on next page)

(continued from previous page)

```

        'codItinerary': '',
        'type': '',
        'startService': '',
        'endService': '',
        'updateDate': '0001-01-01T00:00:00'
    }
}
}
}

```

`citram_api.api.lines.lines.get_line_location(mode_cod, cod_itinerary, cod_line, cod_stop, direction)`

It returns the location of that line at the moment the request is performed. (i.e. Location of a bus of that line right now).

Example:

```

line = get_line_info(create_line_cod(TransportModes.METRO.value, 10)) ['lines'] [
↪ 'LineInformation']
get_line_location(int(line['codMode']),
                  line['itinerary']['Itinerary'][0]['codItinerary'],
                  line['codLine'],
                  line['itinerary']['Itinerary'][0]['stops']['StopInformation
↪ ''] [0] ['codStop'],
                  line['itinerary']['Itinerary'][0]['direction'])

```

Parameters

- **mode_cod** (*int*) – Id of a public transport. Use constants.TransportModes to easily select transport modes ids.
- **cod_itinerary** (*int*) – Itinerary of that line.
- **cod_line** (*str*) – Line id. Use `utils.create_line_cod` to create this id easily.
- **cod_stop** (*str*) – Stop id. In this case this is needed as a formality. Any stop of that line gives the same result.
- **direction** (*int*) – Direction of a itinerary.

Returns

`citram_api.api.lines.lines.get_lines_by_line_code(cod_line)`

Returns the line specified. This method results in a brief description of the line.

Example:

```
get_lines_by_line_code(create_line_cod(TransportModes.METRO.value, 10))
```

Parameters **cod_line** (*str*) – Line id. Use `utils.create_line_cod` to create this id easily.

Returns The line of the line id specified. The returned dictionary look likes the following example:

```

{
  'lines': {
    'Line': {
      'codLine': '4__10__',
      'shortDescription': '10',

```

(continues on next page)

(continued from previous page)

```

        'description': 'Hospital del Norte-Puerta del Sur',
        'codMode': '4',
        'updateDate': '2019-03-22T17:21:37+01:00',
        'updateKmlDate': '2016-06-01T17:04:00+02:00',
        'nightService': 0,
        'active': True,
        'shortItinerary': {
            'Itinerary': [{
                'codItinerary': '4__10____1__IT_1',
                'name': 'Hospital del Norte - Puerta del Sur',
                'direction': 1,
                'kml': 'http://www.citram.es:8080/kml/itinerarios/20160601_
↪1704/metrokmz/M4_L10a_S1_TRAMO.kmz#http://www.citram.es:8080/kml/itinerarios/
↪20160601_1704/metrokmz/M4_L10b_S1_TRAMO.kmz',
                'stops': {}
            }, {
                'codItinerary': '4__10____2__IT_1',
                'name': 'Puerta del Sur - Hospital del Norte',
                'direction': 2,
                'kml': 'http://www.citram.es:8080/kml/itinerarios/20160601_
↪1704/metrokmz/M4_L10a_S2_TRAMO.kmz#http://www.citram.es:8080/kml/itinerarios/
↪20160601_1704/metrokmz/M4_L10b_S2_TRAMO.kmz',
                'stops': {}
            }]
        },
        'URLLine': 'http://www.crtm.es/tu-transporte-publico/metro/lineas/4__
↪10____.aspx',
        'colorLine': '#005AA9',
        'text_colorLine': '#FFFFFF',
        'companyCode': '200'
    }
}

```

`citram_api.api.lines.lines.get_lines_by_mode(mode_cod)`

Get transport lines of a transport mode given a transport mode id.

You can use constants.TransportModes to get the available transport modes ids.

Example:

```
get_lines_by_mode(TransportModes.CERCANIAS.value)
```

Parameters `mode_cod (int)` – Id of a public transport. Use constants.TransportModes to easily select transport modes ids.

Return dict Lines of the transport specified. The returned dictionary look likes the following example:

```

{
    'lines': {
        'Line': [{
            'codLine': '5__C1____',
            'shortDescription': 'C1',
            'description': 'P. Pío-Atocha-Recoletos-Chamartín-Aeropuerto T4',
            'codMode': '5',

```

(continues on next page)

(continued from previous page)

```

        'updateDate': '2019-11-19T14:30:28+01:00',
        'updateKmlDate': '2019-11-19T14:25:06+01:00',
        'nightService': 0,
        'active': True,
        'shortItinerary': {
            'Itinerary': [{
                'codItinerary': '5__C1____1__IT_1',
                'name': 'P.Pío-Atocha-Recoletos-Chamartín-Aeropuerto T4',
                'direction': 1,
                'kml': 'http://www.citram.es:8080/kml/itinerarios/20191119_
→1425/cercaniaskmz/M5_LC-1_S1_TRAMO.kmz',
                'stops': {}
            }, {
                'codItinerary': '5__C1____2__IT_1',
                'name': 'Aeropuerto T4-Chamartín-Recoletos-Atocha-P.Pío',
                'direction': 2,
                'kml': 'http://www.citram.es:8080/kml/itinerarios/20191119_
→1425/cercaniaskmz/M5_LC-1_S2_TRAMO.kmz',
                'stops': {}
            }]
        },
        'URLLine': 'http://www.crtm.es/tu-transporte-publico/cercanias-renfe/
→lineas/5__1____.aspx',
        'colorLine': '#4FB0E5',
        'text_colorLine': '#FFFFFF',
        'companyCode': '301'
    },
    ...
]
}

```

`citram_api.api.lines.lines.get_lines_by_municipality` (*cod_municipality*,
cod_mode=None)

Get transport lines of a transport mode given a municipality id.

You can use constants.Municipalities to get the available municipalities ids.

Example:

```
get_lines_by_municipality(Municipalities.FUENLABRADA.value)
```

Parameters

- **cod_municipality** (*int*) – Id of a municipality. Use constants.Municipalities to easily select transport modes ids.
- **cod_mode** (*int*) – If specified, the results will be filtered, returning only lines of that transport mode. Optional, default: None (No transport mode filtering).

Return dict Lines of the municipality specified. The returned dictionary look likes the following example:

```

{
    'lines': {
        'Line': [{
            'codLine': '4__12__',

```

(continues on next page)

(continued from previous page)

```

        'shortDescription': '12',
        'description': 'MetroSur',
        'codMode': '4',
        'updateDate': '2019-03-22T17:21:37+01:00',
        'updateKmlDate': '2016-06-01T17:04:00+02:00',
        'nightService': 0,
        'active': True,
        'shortItinerary': {
            'Itinerary': [{
                'codItinerary': '4__12____1__IT_1',
                'name': 'METROSUR (ANDEN-1)',
                'direction': 1,
                'kml': 'http://www.citram.es:8080/kml/itinerarios/20160601_
↪1704/metrokmz/M4_L12-1_S1_TRAMO.kmz',
                'stops': {}
            }, {
                'codItinerary': '4__12____2__IT_1',
                'name': 'METROSUR (ANDEN-2)',
                'direction': 2,
                'kml': 'http://www.citram.es:8080/kml/itinerarios/20160601_
↪1704/metrokmz/M4_L12-2_S1_TRAMO.kmz',
                'stops': {}
            }]
        },
        'URLLine': 'http://www.crtm.es/tu-transporte-publico/metro/lineas/4__
↪12____.aspx',
        'colorLine': '#A49800',
        'text_colorLine': '#FFFFFF',
        'companyCode': '200'
    }, {
        'codLine': '5__C5____',
        'shortDescription': 'C5',
        'description': 'Móstoles-El Soto-Atocha-Fuenlabrada-Humanes',
        'codMode': '5',
        'updateDate': '2019-11-19T14:30:28+01:00',
        'updateKmlDate': '2019-11-19T14:25:06+01:00',
        'nightService': 0,
        'active': True,
        'shortItinerary': {
            'Itinerary': [{
                'codItinerary': '5__C5____1__IT_1',
                'name': 'Móstoles El Soto-Atocha-Fuenlabrada-Humanes',
                'direction': 1,
                'kml': 'http://www.citram.es:8080/kml/itinerarios/20191119_
↪1425/cercaniaskmz/M5_LC-5_S1_TRAMO.kmz',
                'stops': {}
            }, {
                'codItinerary': '5__C5____2__IT_1',
                'name': 'Humanes-Fuenlabrada-Atocha-Móstoles El Soto',
                'direction': 2,
                'kml': 'http://www.citram.es:8080/kml/itinerarios/20191119_
↪1425/cercaniaskmz/M5_LC-5_S2_TRAMO.kmz',
                'stops': {}
            }]
        },
        'URLLine': 'http://www.crtm.es/tu-transporte-publico/cercanias-renfe/
↪lineas/5__5____.aspx',

```

(continues on next page)

(continued from previous page)

```

        'colorLine': '#F9BA13',
        'text_colorLine': '#FFFFFF',
        'companyCode': '301'
    }, {
        'codLine': '8__455__',
        'shortDescription': '455',
        'description': '455-PINTO - GETAFE',
        'codMode': '8',
        'updateDate': '2018-05-10T15:25:50+02:00',
        'updateKmlDate': '2016-06-01T17:04:00+02:00',
        'nightService': 0,
        'active': True,
        'shortItinerary': {
            'Itinerary': [{
                'codItinerary': '8__455____1_-_IT_1',
                'name': '455 GETAFE',
                'direction': 1,
                'kml': 'http://www.citram.es:8080/kml/itinerarios/20160601_
↪1704/interurbanoskmz/M8_L455_S1_obs[-]_TRAMO.kmz',
                'stops': {}
            }, {
                'codItinerary': '8__455____1_a._IT_2',
                'name': '455 GETAFE',
                'direction': 1,
                'kml': 'http://www.citram.es:8080/kml/itinerarios/20160601_
↪1704/interurbanoskmz/M8_L455_S1_obs[a.]_TRAMO.kmz',
                'stops': {}
            }, {
                'codItinerary': '8__455____1_am_IT_2',
                'name': 'Hasta ambulatorio de Getafe',
                'direction': 1,
                'kml': 'http://www.citram.es:8080/kml/itinerarios/20160601_
↪1704/interurbanoskmz/M8_L455_S1_obs[am]_TRAMO.kmz',
                'stops': {}
            }, {
                'codItinerary': '8__455____1_h._IT_2',
                'name': 'Hasta/desde hospital de Getafe',
                'direction': 1,
                'kml': 'http://www.citram.es:8080/kml/itinerarios/20160601_
↪1704/interurbanoskmz/M8_L455_S1_obs[h.]_TRAMO.kmz',
                'stops': {}
            }, {
                'codItinerary': '8__455____2_h._IT_2',
                'name': 'Hasta/desde hospital de Getafe',
                'direction': 2,
                'kml': 'http://www.citram.es:8080/kml/itinerarios/20160601_
↪1704/interurbanoskmz/M8_L455_S2_obs[h.]_TRAMO.kmz',
                'stops': {}
            }, {
                'codItinerary': '8__455____2_jc_IT_2',
                'name': 'Desde Avda. Juan de la Cierva, 28',
                'direction': 2,
                'kml': 'http://www.citram.es:8080/kml/itinerarios/20160601_
↪1704/interurbanoskmz/M8_L455_S2_obs[jc]_TRAMO.kmz',
                'stops': {}
            }
        ]
    },

```

(continues on next page)

(continued from previous page)

```
        'URLLine': 'http://www.crtm.es/tu-transporte-publico/autobuses-
↪interurbanos/lineas/8__455__.aspx',
        'colorLine': '#8EBF42',
        'text_colorLine': '#FFFFFF',
        'companyCode': '078'
    },
    ...
]
}
```

`citram_api.api.lines.lines.get_lines_timeplanning(cod_line)`
Timeplanning of the specified line id.

Example:

```
get_lines_timeplanning(create_line_cod(TransportModes.METRO.value, 10))
```

Parameters `cod_line` (*str*) – Line id. Use `utils.create_line_cod` to create this id easily.

Returns Timeplanning of the line id specified.

4.1.3 Module contents

4.2 citram_api.api.offices

4.2.1 Submodules

4.2.2 citram_api.api.offices.offices module

`citram_api.api.offices.offices.get_offices_by_municipality(cod_municipality, of-`
`ices_type=None)`

Returns all the offices from the specified municipality. If `offices_type` is specified, the results will be filtered, returning only offices of that type.

Example:

```
get_offices_by_municipality(Municipalities.FUENLABRADA.value, OfficeTypes.OFICINA.
↪value)
```

Parameters

- **cod_municipality** (*int*) – Id of a municipality. Use constants.Municipalities to easily select transport modes ids.
- **offices_type** (*str*) – Office type. If set, filters the results by the office type specified. Use constants.OfficeTypes to choose the available types easily. Optional, default: None (no filtering).

Return dict A dictionary with a list of offices in that municipality.

`citram_api.api.offices.offices.get_offices_by_postcode` (*post_code*, *of-
fices_type=None*)

Returns all the offices from the specified zip code. If *offices_type* is specified, the results will be filtered, returning only offices of that type.

Example:

```
get_offices_by_postcode(28004)
```

Parameters

- **post_code** (*int*) – Zip code of the area to look for offices.
- **offices_type** (*str*) – Office type. If set, filters the results by the office type specified. Use constants.`OfficeTypes` to choose the available types easily. Optional, default: `None` (no filtering).

Return dict A dictionary with a list of offices from the zip code specified.

`citram_api.api.offices.offices.get_offices_by_type` (*offices_type*)

Returns all the offices of the specified type.

Example:

```
get_offices_by_type(OfficeTypes.OFICINA.value)
```

Parameters **offices_type** (*str*) – Office type. Use constants.`OfficeTypes` to choose the available types easily.

Return dict A dictionary with a list of offices of the specified type.

```
{
  'offices': {
    'Office': [{
      'codOffice': '01_000015',
      'name': 'Oficina de Gestión del Consorcio Regional de Transportes de_
↳Madrid',
      'address': 'Plaza Descubridor Diego de Ordás, 3. 28003 Madrid',
      'openTime': 'Horario de atención al público: Lunes a Viernes de 8 a_
↳20 h',
      'coordinates': {
        'longitude': -3.700064,
        'latitude': 40.44086
      },
      'type': 'gestion'
    }, {
      'codOffice': '01_000020',
      'name': 'Oficina de Gestión de Moncloa',
      'address': 'Plaza de la Moncloa, 1. 28008 Madrid',
      'openTime': 'Horario de atención al público: Lunes a viernes de 7 a_
↳22 h: Sábados y domingos de 10 a 22 h',
      'coordinates': {
        'longitude': -3.719445,
        'latitude': 40.43502
      },
      'type': 'gestion'
    }, {
      'codOffice': '01_000025',
```

(continues on next page)

(continued from previous page)

```

        'name': 'Oficina de Gestión de Sol',
        'address': 'Plaza de la Puerta del Sol, 6. 28013 Madrid',
        'openTime': 'Horario de atención al público: Lunes a viernes de 7 a_
↪22 h; Sábados y domingos de 10 a 22 h',
        'coordinates': {
            'longitude': -3.703257,
            'latitude': 40.416877
        },
        'type': 'gestion'
    }, {
        'codOffice': '01_000030',
        'name': 'Oficina de Gestión de Príncipe Pío',
        'address': 'Paseo de la Florida, 2. 28008 Madrid',
        'openTime': 'Horario de atención al público: Lunes a viernes de 7 a_
↪22 h; Sábados de 10 a 22 h',
        'coordinates': {
            'longitude': -3.720322,
            'latitude': 40.421069
        },
        'type': 'gestion'
    },
    ...
]]
}

```

4.2.3 Module contents

4.3 citram_api.api.others package

4.3.1 Submodules

4.3.2 citram_api.api.others.others module

`citram_api.api.others.others.get_municipalities()`

Returns the municipalities that are in CRTM system with their ids.

Return dict A dictionary with the municipalities and ids. It looks like this:

```

{
    'municipalities': {
        'Municipality': [{
            'codMunicipality': '4273',
            'name': 'ACEBEDA, LA'
        }, {
            'codMunicipality': '4274',
            'name': 'AJALVIR'
        }, {
            'codMunicipality': '4275',
            'name': 'ALAMEDA DEL VALLE'
        }, {
            'codMunicipality': '4276',
            'name': 'ÁLAMO, EL'
        }
    ]
}

```

(continues on next page)

(continued from previous page)

```

    }, {
        'codMunicipality': '4277',
        'name': 'ALCALÁ DE HENARES'
    }, {
        'codMunicipality': '4278',
        'name': 'ALCOBENDAS'
    },
    ...
]
}

```

`citram_api.api.others.others.get_transport_modes()`

Returns the transport modes available and their id.

Return dict Transport modes and their ids. The result looks like this:

```

{
  'modes': {
    'Mode': [{
      'codMode': '4',
      'name': 'METRO'
    }, {
      'codMode': '6',
      'name': 'AUTOBUSES EMT'
    }, {
      'codMode': '5',
      'name': 'CERCANIAS'
    }, {
      'codMode': '10',
      'name': 'METRO LIGERO/TRANVÍA'
    }, {
      'codMode': '8',
      'name': 'AUTOBUSES INTERURBANOS'
    }, {
      'codMode': '9',
      'name': 'AUTOBUSES URBANOS OTROS MUNICIPIOS'
    }, {
      'codMode': '90',
      'name': 'INTERCAMBIADORES'
    }, {
      'codMode': '0',
      'name': 'LARGO RECORRIDO'
    }, {
      'codMode': '1',
      'name': 'APARCAMIENTOS'
    }
  ]
}

```

`citram_api.api.others.others.get_ttp_card_info(ttp_number)`

Get information from the specified transport card number.

The number is the concatenation of the last 3 numbers of the first row and all the numbers of the second row. See this image: <https://tarjetatransportepublico.crtm.es/CRTM-ABONOS/archivos/img/TTP.jpg>

Parameters `ttp_number` (*str*) – The number that identifies a transport card. It must be a string of the last 3 numbers

of the first row and all the numbers of the second row. :return dict: A dictionary with information of the transport card. It has information regarding the titles in that card, expiring dates, purchase dates, title types (young, normal, old, ...), among others.

4.3.3 Module contents

4.4 citram_api.api.stops package

4.4.1 Submodules

4.4.2 citram_api.api.stops.stops module

`citram_api.api.stops.stops.get_nearest_stops(latitude, longitude, distance, method=2)`

Retrieves the stops within a distance given a set of coordinates.

Example:

```
get_nearest_stops(40.453053, -3.688344, 500.0)
```

Parameters

- **latitude** (*float*) – Latitude.
- **longitude** (*float*) – Longitude.
- **distance** (*float*) – Distance from the point specified to find stops.
- **method** (*int*) – Method to find stops. Unfortunately we don't know what method each value represents, so

by default the value of 2 is provided. :return dict: Dictionary with stops within the distance of the location provided. The result looks like this:

```
{
  'stops': {
    'Stop': [{
      'codStop': '6_36',
      'shortCodStop': '36',
      'codMode': '6',
      'name': 'Castellana-San Germán',
      'address': 'Pº de la Castellana, 113',
      'postCode': '',
      'codMunicipality': '4350',
      'coordinates': {
        'longitude': -3.6904863648497,
        'latitude': 40.456610013049
      },
      'lines': {
        'Line': [{
          'codLine': '6__5__',
          'shortDescription': '5',
          'description': '5-SOL/SEVILLA-CHAMARTIN',
          'codMode': '6',
          'updateDate': '2017-11-08T12:43:20+01:00',
          'updateKmlDate': '2017-11-08T12:37:23+01:00',

```

(continues on next page)

(continued from previous page)

```

        'nightService': 0,
        'active': True,
        'shortItinerary': {},
        'companyCode': ''
    }, {
        'codLine': '6__27__',
        'shortDescription': '27',
        'description': '27-EMBAJADORES-PLAZA CASTILLA',
        'codMode': '6',
        'updateDate': '2017-11-08T12:43:20+01:00',
        'updateKmlDate': '2017-11-08T12:37:23+01:00',
        'nightService': 0,
        'active': True,
        'shortItinerary': {},
        'companyCode': ''
    }, {
        'codLine': '6__40__',
        'shortDescription': '40',
        'description': '40-TRIBUNAL-ALFONSO XIII',
        'codMode': '6',
        'updateDate': '2016-06-02T09:31:47+02:00',
        'updateKmlDate': '2016-06-01T17:04:00+02:00',
        'nightService': 0,
        'active': True,
        'shortItinerary': {},
        'companyCode': ''
    }, {
        'codLine': '6__147__',
        'shortDescription': '147',
        'description': '147-CALLAO-BARRIO DEL PILAR',
        'codMode': '6',
        'updateDate': '2016-06-02T09:31:47+02:00',
        'updateKmlDate': '2016-06-01T17:04:00+02:00',
        'nightService': 0,
        'active': True,
        'shortItinerary': {},
        'companyCode': ''
    }, {
        'codLine': '6_N_22__',
        'shortDescription': 'N22',
        'description': 'N22-CIBELES-BARRIO DEL PILAR',
        'codMode': '6',
        'updateDate': '2017-06-27T11:21:02+02:00',
        'updateKmlDate': '2017-06-27T10:08:00+02:00',
        'nightService': 1,
        'active': True,
        'shortItinerary': {},
        'companyCode': ''
    }, {
        'codLine': '6_N_24__',
        'shortDescription': 'N24',
        'description': 'N24-CIBELES-LAS TABLAS',
        'codMode': '6',
        'updateDate': '2016-06-02T09:31:47+02:00',
        'updateKmlDate': '2016-06-01T17:04:00+02:00',
        'nightService': 1,
        'active': True,

```

(continues on next page)

(continued from previous page)

```

        'shortItinerary': {},
        'companyCode': ''
    }
},
'access': 2,
'park': 0,
'nightLinesService': 1
}, {
    'codStop': '6_4828',
    'shortCodStop': '4828',
    'codMode': '6',
    'name': 'Castellana-San Germán',
    'address': 'Pº de la Castellana, 152',
    'postCode': '',
    'codMunicipality': '4350',
    'coordinates': {
        'longitude': -3.6899180284389,
        'latitude': 40.456032321623
    },
    'lines': {
        'Line': {
            'codLine': '6__5__',
            'shortDescription': '5',
            'description': '5-SOL/SEVILLA-CHAMARTIN',
            'codMode': '6',
            'updateDate': '2017-11-08T12:43:20+01:00',
            'updateKmlDate': '2017-11-08T12:37:23+01:00',
            'nightService': 0,
            'active': True,
            'shortItinerary': {},
            'companyCode': ''
        }
    },
    'access': 2,
    'park': 0,
    'nightLinesService': 0
}, {
    'codStop': '4_192',
    'shortCodStop': '192',
    'codMode': '4',
    'name': 'SANTIAGO BERNABEU',
    'address': 'Paseo de la Castellana 97 ',
    'postCode': '',
    'codMunicipality': '4350',
    'coordinates': {
        'longitude': -3.69038,
        'latitude': 40.45159
    },
    'lines': {
        'Line': {
            'codLine': '4__10__',
            'shortDescription': '10',
            'description': 'Hospital del Norte-Puerta del Sur',
            'codMode': '4',
            'updateDate': '2019-03-22T17:21:37+01:00',
            'updateKmlDate': '2016-06-01T17:04:00+02:00',
            'nightService': 0,

```

(continues on next page)

(continued from previous page)

```

        'active': True,
        'shortItinerary': {},
        'companyCode': ''
    }
},
'access': 0,
'park': 0,
'nightLinesService': 0
},
...
]
}

```

`citram_api.api.stops.stops.get_stop_info(cod_stop)`

Returns detailed information about the specified stop.

You can create a `cod_stop` easily by using `utils.create_cod_stop`.

Example:

```
get_stop_info(create_stop_cod(TransportModes.METRO.value, 276))
```

Parameters `cod_stop` (*str*) – Stop id. You can create a `cod_stop` easily by using `utils.create_cod_stop`.

Return dict Detailed information about the specified stop. The result looks like this:

```

{
  'stops': {
    'Stop': {
      'codStop': '4_276',
      'shortCodStop': '276',
      'codMode': '4',
      'name': 'LAS TABLAS',
      'address': 'Calle de Palas de Rey 48 ',
      'postCode': '28050',
      'codMunicipality': '4350',
      'coordinates': {
        'longitude': -3.66944,
        'latitude': 40.50833
      },
      'codLines': {
        'Line': ['4__10__', '10__ML1__']
      },
      'access': 2,
      'park': 0,
      'nightLinesService': 0,
      'stopType': 0
    }
  }
}

```

`citram_api.api.stops.stops.get_stop_times(cod_stop, stop_type, stop_times_by_iti, order_by=2)`

Get the stop times for the stop and line itinerary specified.

You can create a `cod_stop` easily by using `utils.create_cod_stop`.

```

stop = get_stop_info(create_stop_cod(TransportModes.METRO.value, 276))
line = get_line_info(stop['stops']['Stop']['codLines']['Line'][0])['lines'][
    ↪ 'LineInformation']
get_stop_times(stop['stops']['Stop']['codStop'],
               stop['stops']['Stop']['stopType'],
               stop['stops']['Stop']['codStop'],
               line['itinerary']['Itinerary'][0]['codItinerary'])

```

Parameters

- **cod_stop** (*str*) – Stop id. You can create a cod_stop easily by using `utils.create_cod_stop`.
- **stop_type** (*str*) – Stop type number. Unfortunately, we don't have information of what type of stop each number

represents, so the easiest thing is to get it from the stop info. :param str stop_times_by_iti: Itinerary name. :param int order_by: Rule to sort results. Optional, 2 by default. Unfortunately, we don't know what numbers correspond to what order criteria, so the default value is provided by default. :return dict: A dictionary with the times of the itinerary for that stop.

```

{
  'stopTimes': {
    'actualDate': '2020-01-02T01:35:27+01:00',
    'stop': {
      'codStop': '4_276',
      'shortCodStop': '276',
      'name': 'LAS TABLAS',
      'park': 0,
      'nightLinesService': 0
    },
    'times': {
      'Time': [{
        'line': {
          'codLine': '4__10__',
          'shortDescription': '10',
          'description': '10-Hospital del Norte-Puerta del Sur',
          'codMode': '4',
          'updateDate': '2019-03-22T17:21:37+01:00',
          'updateKmlDate': '2016-06-01T17:04:00+02:00',
          'nightService': 0,
          'active': True,
          'shortItinerary': {},
          'companyCode': ''
        },
        'direction': 1,
        'destination': 'PUERTA DEL SUR',
        'destinationStop': {
          'codStop': '4_205',
          'shortCodStop': '205',
          'name': 'PUERTA DEL SUR',
          'park': 0,
          'nightLinesService': 0
        },
        'time': '2020-01-02T01:48:39+01:00',
        'codVehicle': '',
        'codIssue': ''
      }]
    }
  }
}

```

(continues on next page)

(continued from previous page)

```

    }, {
      'line': {
        'codLine': '4__10__',
        'shortDescription': '10',
        'description': '10-Hospital del Norte-Puerta del Sur',
        'codMode': '4',
        'updateDate': '2019-03-22T17:21:37+01:00',
        'updateKmlDate': '2016-06-01T17:04:00+02:00',
        'nightService': 0,
        'active': True,
        'shortItinerary': {},
        'companyCode': ''
      },
      'direction': 1,
      'destination': 'PUERTA DEL SUR',
      'destinationStop': {
        'codStop': '4_205',
        'shortCodStop': '205',
        'name': 'PUERTA DEL SUR',
        'park': 0,
        'nightLinesService': 0
      },
      'time': '2020-01-02T01:48:39+01:00',
      'codVehicle': '',
      'codIssue': ''
    }, {
      'line': {
        'codLine': '4__10__',
        'shortDescription': '10',
        'description': '10-Hospital del Norte-Puerta del Sur',
        'codMode': '4',
        'updateDate': '2019-03-22T17:21:37+01:00',
        'updateKmlDate': '2016-06-01T17:04:00+02:00',
        'nightService': 0,
        'active': True,
        'shortItinerary': {},
        'companyCode': ''
      },
      'direction': 1,
      'destination': 'PUERTA DEL SUR',
      'destinationStop': {
        'codStop': '4_205',
        'shortCodStop': '205',
        'name': 'PUERTA DEL SUR',
        'park': 0,
        'nightLinesService': 0
      },
      'time': '2020-01-02T01:49:39+01:00',
      'codVehicle': '',
      'codIssue': ''
    }
  ],
  'linesStatus': {
    'LineStatus': {
      'line': {
        'codLine': '4__10__',
        'shortDescription': '10'

```

(continues on next page)

(continued from previous page)

```

        },
        'SAEStatus': True
    }
}
}
}

```

`citram_api.api.stops.stops.get_stops_by_cod_stop(cod_stop)`

It returns the stop specified by a stop id. This method results in a brief description of the stop.

You can create a `cod_stop` easily by using `utils.create_cod_stop`.

```
get_stops_by_cod_stop(create_stop_cod(TransportModes.METRO.value, 276))
```

Parameters `cod_stop` (*str*) – Stop id. You can create a `cod_stop` easily by using `utils.create_cod_stop`.

Return dict Short description of stop. It looks like this:

```

{
  'stops': {
    'Stop': {
      'codStop': '4_276',
      'shortCodStop': '276',
      'codMode': '4',
      'name': 'LAS TABLAS',
      'address': 'Calle de Palas de Rey 48 ',
      'postCode': '28050',
      'codMunicipality': '4350',
      'coordinates': {
        'longitude': -3.66944,
        'latitude': 40.50833
      },
      'codLines': {
        'Line': ['4__10__', '10__ML1__']
      },
      'access': 2,
      'park': 0,
      'nightLinesService': 0,
      'stopType': 0
    }
  }
}

```

`citram_api.api.stops.stops.get_stops_by_custom_search(custom_search)`

It returns stops based on the custom search string specified.

Example:

```
get_stops_by_custom_search('tres aguas')
```

Parameters `custom_search` (*str*) – Custom search string.

Return dict A dictionary of the stops that match your search. The result looks like this:

```

{
  'stops': {
    'Stop': [{
      'codStop': '8_08469',
      'shortCodStop': '08469',
      'codMode': '8',
      'name': 'AV.S.MARTÍN VALDEIGLESIAS-C.C.TRES AGUAS',
      'address': 'AV.S.MARTÍN VALDEIGLESIAS-C.C.TRES AGUAS',
      'postCode': '28925',
      'codMunicipality': '4279',
      'coordinates': {
        'longitude': -3.8321874141693,
        'latitude': 40.357341766357
      },
      'codLines': {
        'Line': ['8__510__', '8__510_A__', '8__518__', '8__551__', '8__
↪581__', '8_N_504__', '9__3__007_']
      },
      'access': 2,
      'park': 0,
      'nightLinesService': 1,
      'stopType': 0
    }, {
      'codStop': '8_09364',
      'shortCodStop': '09364',
      'codMode': '8',
      'name': 'ARGENTINA-C.C.TRES AGUAS',
      'address': 'ARGENTINA-C.C.TRES AGUAS',
      'postCode': '28922',
      'codMunicipality': '4279',
      'coordinates': {
        'longitude': -3.8295669555664,
        'latitude': 40.355888366699
      },
      'codLines': {
        'Line': ['8__510__', '8__510_A__', '8__518__', '8__551__', '8__
↪581__', '8_N_504__', '9__3__007_']
      },
      'access': 2,
      'park': 0,
      'nightLinesService': 1,
      'stopType': 0
    }
  ]
}
}

```

`citram_api.api.stops.stops.get_stops_by_municipality(cod_municipality)`

It returns the stops that belong to the specified municipality.

You can use constants.Municipalities to get the available municipalities ids.

Example:

```
get_stops_by_municipality(Municipalities.FUENLABRADA.value)
```

Parameters `cod_municipality` (*int*) – Id of a municipality. Use constants.Municipalities to easily select transport modes ids.

Return dict Stops that belong to the municipality specified. The result looks like this:

```
{
  'stops': {
    'Stop': [{
      'codStop': '4_218',
      'shortCodStop': '218',
      'codMode': '4',
      'name': 'LORANCA',
      'address': 'Calle de la Alegría SN ',
      'postCode': '28942',
      'codMunicipality': '4330',
      'coordinates': {
        'longitude': -3.83768,
        'latitude': 40.29681
      },
      'codLines': {
        'Line': '4__12__'
      },
      'access': 2,
      'park': 0,
      'nightLinesService': 0,
      'stopType': 0
    }, {
      'codStop': '4_219',
      'shortCodStop': '219',
      'codMode': '4',
      'name': 'HOSPITAL DE FUENLABRADA',
      'address': 'Cmno del Molino SN ',
      'postCode': '28942',
      'codMunicipality': '4330',
      'coordinates': {
        'longitude': -3.81642,
        'latitude': 40.28576
      },
      'codLines': {
        'Line': '4__12__'
      },
      'access': 2,
      'park': 0,
      'nightLinesService': 0,
      'stopType': 0
    },
    ...
  ]
}
```

`citram_api.api.stops.stops.get_stops_by_zip_code(postcode)`

It returns the stops that belong to a zip code.

Example:

```
get_stops_by_zip_code(28922)
```

Parameters `postcode` (*int*) – Zip code from where the stops are going to be retrieved.

Return dict Stops that belong to the zip code specified. The result looks like this:

```

{
  'stops': {
    'Stop': [{
      'codStop': '4_211',
      'shortCodStop': '211',
      'codMode': '4',
      'name': 'ALCORCON CENTRAL',
      'address': 'Avda de Móstoles 12 ',
      'postCode': '28922',
      'codMunicipality': '4279',
      'coordinates': {
        'longitude': -3.83178,
        'latitude': 40.35008
      },
      'codLines': {
        'Line': ['4__12__', '5__C5__']
      },
      'access': 2,
      'park': 0,
      'nightLinesService': 0,
      'stopType': 0
    }, {
      'codStop': '4_212',
      'shortCodStop': '212',
      'codMode': '4',
      'name': 'PARQUE OESTE',
      'address': 'Calle de Estambul 5 ',
      'postCode': '28922',
      'codMunicipality': '4279',
      'coordinates': {
        'longitude': -3.84934,
        'latitude': 40.34589
      },
      'codLines': {
        'Line': '4__12__'
      },
      'access': 2,
      'park': 0,
      'nightLinesService': 0,
      'stopType': 0
    }, {
      'codStop': '5_5',
      'shortCodStop': '5',
      'codMode': '5',
      'name': 'ALCORCON',
      'address': 'ALCORCON',
      'postCode': '28922',
      'codMunicipality': '4279',
      'coordinates': {
        'longitude': -3.8316340093152,
        'latitude': 40.35008820867
      },
      'codLines': {
        'Line': ['4__12__', '5__C5__']
      },
      'access': 1,
      'park': 0,
    }
  ]
}

```

(continues on next page)

(continued from previous page)

```

        'nightLinesService': 0,
        'stopType': 0
    }, {
        'codStop': '5_66',
        'shortCodStop': '66',
        'codMode': '5',
        'name': 'RETAMAS, LAS',
        'address': 'RETAMAS, LAS',
        'postCode': '28922',
        'codMunicipality': '4279',
        'coordinates': {
            'longitude': -3.8423299815775,
            'latitude': 40.341902701431
        },
        'codLines': {
            'Line': '5__C5__'
        },
        'access': 1,
        'park': 0,
        'nightLinesService': 0,
        'stopType': 0
    }, {
        'codStop': '8_08434',
        'shortCodStop': '08434',
        'codMode': '8',
        'name': 'AV.MÓSTOLES-VIVERO',
        'address': 'AV.MÓSTOLES-VIVERO',
        'postCode': '28922',
        'codMunicipality': '4279',
        'coordinates': {
            'longitude': -3.8468658924103,
            'latitude': 40.337970733643
        },
        'codLines': {
            'Line': ['8__520__', '8_N_501__']
        },
        'access': 2,
        'park': 0,
        'nightLinesService': 1,
        'stopType': 0
    },
    ...
]
}

```

4.4.3 Module contents

4.5 citram_api.utils package

4.5.1 Submodules

4.5.2 citram_api.utils.utils module

```
citram_api.utils.utils.common_request(url)
citram_api.utils.utils.create_line_cod(mode_cod, line)
citram_api.utils.utils.create_stop_cod(mode_cod, stop_cod)
```

4.5.3 Module contents

4.6 citram_api.constants package

4.6.1 Submodules

4.6.2 citram_api.constants.constants module

```
class citram_api.constants.constants.Municipalities
    Bases: enum.Enum

    An enumeration.

    ACEBEDA_LA = 4273
    AJALVIR = 4274
    ALAMEDA_DEL_VALLE = 4275
    ALAMO_EL = 4276
    ALCALA_DE_HENARES = 4277
    ALCOBENDAS = 4278
    ALCORCON = 4279
    ALDEA_DEL_FRESNO = 4280
    ALGETE = 4281
    ALPEDRETE = 4282
    AMBITE = 4283
    ANCHUELO = 4284
    ARANJUEZ = 4285
    ARGANDA_DEL_REY = 4286
    ARROYOMOLINOS = 4287
    ATAZAR_EL = 4288
    BATRES = 4289
```

BECERRIL_DE_LA_SIERRA = 4290
BELMONTE_DE_TAJO = 4291
BERRUECO_EL = 4293
BERZOSA_DE_LOZOYA = 4292
BOADILLA_DEL_MONTE = 4294
BOALO_EL = 4295
BRAOJOS = 4296
BREA_DE_TAJO = 4297
BRUNETE = 4298
BUITRAGO_DEL_LOZOYA = 4299
BUSTARVIEJO = 4300
CABANILLAS_DE_LA_SIERRA = 4301
CABRERA_LA = 4302
CADALSO_DE_LOS_VIDRIOS = 4303
CAMARMA_DE_ETERUELAS = 4304
CAMPO_REAL = 4305
CANENCIA = 4306
CARABANA = 4307
CASARRUBUELOS = 4308
CENICIENTOS = 4309
CERCEDILLA = 4310
CERVERA_DE_BUITRAGO = 4311
CHAPINERIA = 4323
CHINCHON = 4324
CIEMPOZUELOS = 4312
COBENA = 4313
COLLADO_MEDIANO = 4318
COLLADO_VILLALBA = 4319
COLMENAREJO = 4316
COLMENAR_DEL_ARROYO = 4314
COLMENAR_DE_OREJA = 4315
COLMENAR_VIEJO = 4317
CORPA = 4320
COSLADA = 4321
CUBAS_DE_LA_SAGRA = 4322
DAGANZO_DE_ARRIBA = 4325

ESCORIAL_EL = 4326
ESTREMER = 4327
FRESNEDILLAS_DE_LA_OLIVA = 4328
FRESNO_DE_TOROTE = 4329
FUENLABRADA = 4330
FUENTE_EL_SAZ_DE_JARAMA = 4331
FUENTIDUENA_DE_TAJO = 4332
GALAPAGAR = 4333
GARGANTA_DE_LOS_MONTES = 4334
GARGANTILLA_DEL_LOZOYA = 4335
GASCONES = 4336
GETAFE = 4337
GRINON = 4338
GUADALIX_DE_LA_SIERRA = 4339
GUADARRAMA = 4340
HORCAJO_DE_LA_SIERRA = 4342
HORCAJUELO_DE_LA_SIERRA = 4343
HOYO_DE_MANZANARES = 4344
HUMANES_DE_MADRID = 4345
LEGANES = 4346
LOECHES = 4347
LOZOYA = 4348
LOZOYUELA_NAVAS_SIETEIGLESIAS = 4450
MADARCOS = 4349
MADRID = 4350
MAJADAHONDA = 4351
MANZANARES_EL_REAL = 4352
MECO = 4353
MEJORADA_DEL_CAMPO = 4354
MIRAFLORES_DE_LA_SIERRA = 4355
MOLAR_EL = 4356
MOLINOS_LOS = 4357
MONTEJO_DE_LA_SIERRA = 4358
MORALEJA_DE_ENMEDIO = 4359
MORALZARZAL = 4360
MORATA_DE_TAJUNA = 4361

MOSTOLES = 4362
NAVACERRADA = 4363
NAVALAFUENTE = 4364
NAVALAGAMELLA = 4365
NAVALCARNERO = 4366
NAVARREDONDA = 4367
NAVAS_DEL_REY = 4368
NUEVO_BAZTAN = 4369
OLMEDA_DE_LAS_FUENTES = 4370
ORUSCO_DE_TAJUNA = 4371
PARACUELLOS_DE_JARAMA = 4372
PARLA = 4373
PATONES = 4374
PEDREZUELA = 4375
PELAYOS_DE_LA_PRESA = 4376
PERALES_DE_TAJUNA = 4377
PEZUELA_DE_LAS_TORRES = 4378
PINILLA_DEL_VALLE = 4379
PINTO = 4380
PINUECAR = 4381
POZUELO_DEL_REY = 4383
POZUELO_DE_ALARCON = 4382
PRADENA_DEL_RINCON = 4384
PUENTES_VIEJAS = 4451
QUIJORNA = 4386
RASCAFRIA = 4387
REDUENA = 4388
RIBATEJADA = 4389
RIVAS_VACIAMADRID = 4390
ROBLEDILLO_DE_LA_JARA = 4391
ROBLEDO_DE_CHAVELA = 4392
ROBREGORDO = 4393
ROZAS_DE_MADRID_LAS = 4394
ROZAS_DE_PUERTO_REAL = 4395
SANTA_MARIA_DE_LA_ALAMEDA = 4402
SANTORCAZ = 4403

SANTOS_DE_LA_HUMOSA_LOS = 4404
SAN_AGUSTIN_DE_GUADALIX = 4396
SAN_FERNANDO_DE_HENARES = 4397
SAN_LORENZO_DE_EL_ESCORIAL = 4398
SAN_MARTIN_DE_LA_VEGA = 4399
SAN_MARTIN_DE_VALDEIGLESIAS = 4400
SAN_SEBASTIAN_DE_LOS_REYES = 4401
SERRA_DEL_MONTE_LA = 4405
SERRANILLOS_DEL_VALLE = 4406
SEVILLA_LA_NUEVA = 4407
SOMOSIERRA = 4408
SOTO_DEL_REAL = 4409
TALAMANCA_DEL_JARAMA = 4410
TIELMES = 4411
TITULCIA = 4412
TORREJON_DE_ARDOZ = 4413
TORREJON_DE_LA_CALZADA = 4414
TORREJON_DE_VELASCO = 4415
TORRELAGUNA = 4416
TORRELODONES = 4417
TORREMOCHA_DE_JARAMA = 4418
TORRES_DE_LA_ALAMEDA = 4419
TRES_CANTOS = 4452
VALDARACETE = 4420
VALDEAVERO = 4421
VALDELAGUNA = 4422
VALDEMANCO = 4423
VALDEMAQUEDA = 4424
VALDEMORILLO = 4425
VALDEMORO = 4426
VALDEOLMOS_ALALPARDO = 4427
VALDEPIELAGOS = 4428
VALDETORRES_DE_JARAMA = 4429
VALDILECHA = 4430
VALVERDE_DE_ALCALA = 4431
VELILLA_DE_SAN_ANTONIO = 4432

```
VELLON_EL = 4433
VENTURADA = 4434
VILLACONEJOS = 4435
VILLALBILLA = 4437
VILLAMANRIQUE_DE_TAJO = 4438
VILLAMANTA = 4439
VILLAMANTILLA = 4440
VILLANUEVA_DEL_PARDILLO = 4442
VILLANUEVA_DE_LA_CANADA = 4441
VILLANUEVA_DE_PERALES = 4443
VILLAREJO_DE_SALVANES = 4445
VILLAR_DEL_OLMO = 4444
VILLAVICIOSA_DE_ODON = 4446
VILLAVIEJA_DEL_LOZOYA = 4447
VILLA_DEL_PRADO = 4436
ZARZALEJO = 4448
```

```
class citram_api.constants.constants.OfficeTypes
```

```
    Bases: enum.Enum
```

```
    An enumeration.
```

```
    OFICINA = 'OFICINA'
```

```
    RECARGA = 'RECARGA'
```

```
class citram_api.constants.constants.TransportModes
```

```
    Bases: enum.Enum
```

```
    An enumeration.
```

```
    APARCAMIENTOS = 1
```

```
    AUTOBUSES_EMT = 6
```

```
    AUTOBUSES_INTERURBANOS = 8
```

```
    AUTOBUSES_URBANOS_OTROS_MUNICIPIOS = 9
```

```
    CERCANIAS = 5
```

```
    INTERCAMBIADORES = 90
```

```
    LARGO_RECORRIDO = 0
```

```
    METRO = 4
```

```
    METRO_LIGERO_TRANVIA = 10
```

4.6.3 Module contents

C

- `citram_api.api.lines`, [16](#)
- `citram_api.api.lines.lines`, [9](#)
- `citram_api.api.offices`, [18](#)
- `citram_api.api.offices.offices`, [16](#)
- `citram_api.api.others`, [20](#)
- `citram_api.api.others.others`, [18](#)
- `citram_api.api.stops`, [31](#)
- `citram_api.api.stops.stops`, [20](#)
- `citram_api.constants`, [36](#)
- `citram_api.constants.constants`, [31](#)
- `citram_api.utils`, [31](#)
- `citram_api.utils.utils`, [31](#)

A

- ACEBEDA_LA (*citram_api.constants.constants.Municipalities*
attribute), 31
- AJALVIR (*citram_api.constants.constants.Municipalities*
attribute), 31
- ALAMEDA_DEL_VALLE (*citram_api.constants.constants.Municipalities*
attribute), 31
- ALAMO_EL (*citram_api.constants.constants.Municipalities*
attribute), 31
- ALCALA_DE_HENARES (*citram_api.constants.constants.Municipalities*
attribute), 31
- ALCOBENDAS (*citram_api.constants.constants.Municipalities*
attribute), 31
- ALCORCON (*citram_api.constants.constants.Municipalities*
attribute), 31
- ALDEA_DEL_FRESNO (*citram_api.constants.constants.Municipalities*
attribute), 31
- ALGETE (*citram_api.constants.constants.Municipalities*
attribute), 31
- ALPEDRETE (*citram_api.constants.constants.Municipalities*
attribute), 31
- AMBITE (*citram_api.constants.constants.Municipalities*
attribute), 31
- ANCHUELO (*citram_api.constants.constants.Municipalities*
attribute), 31
- APARCAMIENTOS (*citram_api.constants.constants.TransportModes*
attribute), 36
- ARANJUEZ (*citram_api.constants.constants.Municipalities*
attribute), 31
- ARGANDA_DEL_REY (*citram_api.constants.constants.Municipalities*
attribute), 31
- ARROYOMOLINOS (*citram_api.constants.constants.Municipalities*
attribute), 31
- ATAZAR_EL (*citram_api.constants.constants.Municipalities*
attribute), 31
- AUTOBUSES_EMT (*citram_api.constants.constants.TransportModes*
attribute), 36
- AUTOBUSES_INTERURBANOS (*citram_api.constants.constants.TransportModes*
attribute), 36
- AUTOBUSES_URBANOS_OTROS_MUNICIPIOS (*citram_api.constants.constants.TransportModes*
attribute), 36

B

- BATRES (*citram_api.constants.constants.Municipalities*
attribute), 31
- BECERRIL_DE_LA_SIERRA (*citram_api.constants.constants.Municipalities*
attribute), 31
- BELMONTE_DE_TAJO (*citram_api.constants.constants.Municipalities*
attribute), 32
- BERRUECO_EL (*citram_api.constants.constants.Municipalities*
attribute), 32
- BERZOSA_DE_LOZOYA (*citram_api.constants.constants.Municipalities*
attribute), 32
- BOADILLA_DEL_MONTE (*citram_api.constants.constants.Municipalities*
attribute), 32
- BOALO_EL (*citram_api.constants.constants.Municipalities*
attribute), 32
- BRAOJOS (*citram_api.constants.constants.Municipalities*
attribute), 32
- BREA_DE_TAJO (*citram_api.constants.constants.Municipalities*
attribute), 32
- BRUNETE (*citram_api.constants.constants.Municipalities*
attribute), 32
- BUITRAGO_DEL_LOZOYA (*citram_api.constants.constants.Municipalities*
attribute), 32

- BUSTARVIEJO (*citram_api.constants.constants.Municipalities* attribute), 32
- ## C
- CABANILLAS_DE_LA_SIERRA (*citram_api.constants.constants.Municipalities* attribute), 32
- CABRERA_LA (*citram_api.constants.constants.Municipalities* attribute), 32
- CADALSO_DE_LOS_VIDRIOS (*citram_api.constants.constants.Municipalities* attribute), 32
- CAMARMA_DE_ESTERUELAS (*citram_api.constants.constants.Municipalities* attribute), 32
- CAMPO_REAL (*citram_api.constants.constants.Municipalities* attribute), 32
- CANENCIA (*citram_api.constants.constants.Municipalities* attribute), 32
- CARABANA (*citram_api.constants.constants.Municipalities* attribute), 32
- CASARRUBUELOS (*citram_api.constants.constants.Municipalities* attribute), 32
- CENICIENTOS (*citram_api.constants.constants.Municipalities* attribute), 32
- CERCANIAS (*citram_api.constants.constants.TransportModes* attribute), 36
- CERCEDILLA (*citram_api.constants.constants.Municipalities* attribute), 32
- CERVERA_DE_BUITRAGO (*citram_api.constants.constants.Municipalities* attribute), 32
- CHAPINERIA (*citram_api.constants.constants.Municipalities* attribute), 32
- CHINCHON (*citram_api.constants.constants.Municipalities* attribute), 32
- CIEMPOZUELOS (*citram_api.constants.constants.Municipalities* attribute), 32
- citram_api.api.lines* (module), 16
- citram_api.api.lines.lines* (module), 9
- citram_api.api.offices* (module), 18
- citram_api.api.offices.offices* (module), 16
- citram_api.api.others* (module), 20
- citram_api.api.others.others* (module), 18
- citram_api.api.stops* (module), 31
- citram_api.api.stops.stops* (module), 20
- citram_api.constants* (module), 36
- citram_api.constants.constants* (module), 31
- citram_api.utils* (module), 31
- citram_api.utils.utils* (module), 31
- COBENA (*citram_api.constants.constants.Municipalities* attribute), 32
- COLLADO_MEDIANO (*citram_api.constants.constants.Municipalities* attribute), 32
- COLLADO_VILLALBA (*citram_api.constants.constants.Municipalities* attribute), 32
- COLMENAR_DE_OREJA (*citram_api.constants.constants.Municipalities* attribute), 32
- COLMENAR_DEL_ARROYO (*citram_api.constants.constants.Municipalities* attribute), 32
- COLMENAR_VIEJO (*citram_api.constants.constants.Municipalities* attribute), 32
- COLMENAREJO (*citram_api.constants.constants.Municipalities* attribute), 32
- common_request* () (in module *citram_api.utils.utils*), 31
- CORPA (*citram_api.constants.constants.Municipalities* attribute), 32
- COSLADA (*citram_api.constants.constants.Municipalities* attribute), 32
- create_line_cod* () (in module *citram_api.utils.utils*), 31
- create_stop_cod* () (in module *citram_api.utils.utils*), 31
- CUBAS_DE_LA_SAGRA (*citram_api.constants.constants.Municipalities* attribute), 32
- ## D
- DAGANZO_DE_ARRIBA (*citram_api.constants.constants.Municipalities* attribute), 32
- ## E
- ESCORIAL_EL (*citram_api.constants.constants.Municipalities* attribute), 32
- ESTREMERIA (*citram_api.constants.constants.Municipalities* attribute), 33
- ## F
- FRESNEDILLAS_DE_LA_OLIVA (*citram_api.constants.constants.Municipalities* attribute), 33
- FRESNO_DE_TOROTE (*citram_api.constants.constants.Municipalities* attribute), 33
- FUENLABRADA (*citram_api.constants.constants.Municipalities* attribute), 33

FUENTE_EL_SAZ_DE_JARAMA (citram_{api}.constants.constants.Municipalities attribute), 33

FUENTIDUENA_DE_TAJO (citram_{api}.constants.constants.Municipalities attribute), 33

G

GALAPAGAR (citram_{api}.constants.constants.Municipalities attribute), 33

GARGANTA_DE_LOS_MONTES (citram_{api}.constants.constants.Municipalities attribute), 33

GARGANTILLA_DEL_LOZOYA (citram_{api}.constants.constants.Municipalities attribute), 33

GASCONES (citram_{api}.constants.constants.Municipalities attribute), 33

get_incidents_affectations() (in module citram_{api}.api.lines.lines), 9

get_line_info() (in module citram_{api}.api.lines.lines), 9

get_line_location() (in module citram_{api}.api.lines.lines), 11

get_lines_by_line_code() (in module citram_{api}.api.lines.lines), 11

get_lines_by_mode() (in module citram_{api}.api.lines.lines), 12

get_lines_by_municipality() (in module citram_{api}.api.lines.lines), 13

get_lines_timeplanning() (in module citram_{api}.api.lines.lines), 16

get_municipalities() (in module citram_{api}.api.others.others), 18

get_nearest_stops() (in module citram_{api}.api.stops.stops), 20

get_offices_by_municipality() (in module citram_{api}.api.offices.offices), 16

get_offices_by_postcode() (in module citram_{api}.api.offices.offices), 16

get_offices_by_type() (in module citram_{api}.api.offices.offices), 17

get_stop_info() (in module citram_{api}.api.stops.stops), 23

get_stop_times() (in module citram_{api}.api.stops.stops), 23

get_stops_by_cod_stop() (in module citram_{api}.api.stops.stops), 26

get_stops_by_custom_search() (in module citram_{api}.api.stops.stops), 26

get_stops_by_municipality() (in module citram_{api}.api.stops.stops), 27

get_stops_by_zip_code() (in module citram_{api}.api.stops.stops), 28

get_transport_modes() (in module citram_{api}.api.others.others), 19

get_ttp_card_info() (in module citram_{api}.api.others.others), 19

GETAFE (citram_{api}.constants.constants.Municipalities attribute), 33

GRINON (citram_{api}.constants.constants.Municipalities attribute), 33

GUADALIX_DE_LA_SIERRA (citram_{api}.constants.constants.Municipalities attribute), 33

GUADARRAMA (citram_{api}.constants.constants.Municipalities attribute), 33

H

HORCAJO_DE_LA_SIERRA (citram_{api}.constants.constants.Municipalities attribute), 33

HORCAJUELO_DE_LA_SIERRA (citram_{api}.constants.constants.Municipalities attribute), 33

HOYO_DE_MANZANARES (citram_{api}.constants.constants.Municipalities attribute), 33

HUMANES_DE_MADRID (citram_{api}.constants.constants.Municipalities attribute), 33

I

INTERCAMBIADORES (citram_{api}.constants.constants.TransportModes attribute), 36

L

LARGO_RECORRIDO (citram_{api}.constants.constants.TransportModes attribute), 36

LEGANES (citram_{api}.constants.constants.Municipalities attribute), 33

LOECHES (citram_{api}.constants.constants.Municipalities attribute), 33

LOZOYA (citram_{api}.constants.constants.Municipalities attribute), 33

LOZOYUELA_NAVAS_SIETEIGLESIAS (citram_{api}.constants.constants.Municipalities attribute), 33

M

MADARCOS (citram_{api}.constants.constants.Municipalities attribute), 33

MADRID (citram_{api}.constants.constants.Municipalities attribute), 33

MAJADAHONDA (citram_{api}.constants.constants.Municipalities attribute), 33

MANZANARES_EL_REAL	(citram_api.constants.constants.Municipalities attribute), 33		
MECO	(citram_api.constants.constants.Municipalities attribute), 33		
MEJORADA_DEL_CAMPO	(citram_api.constants.constants.Municipalities attribute), 33		
METRO	(citram_api.constants.constants.TransportModes attribute), 36		
METRO_LIGERO_TRANVIA	(citram_api.constants.constants.TransportModes attribute), 36		
MIRAFLORES_DE_LA_SIERRA	(citram_api.constants.constants.Municipalities attribute), 33		
MOLAR_EL	(citram_api.constants.constants.Municipalities attribute), 33		
MOLINOS_LOS	(citram_api.constants.constants.Municipalities attribute), 33		
MONTEJO_DE_LA_SIERRA	(citram_api.constants.constants.Municipalities attribute), 33		
MORALEJA_DE_ENMEDIO	(citram_api.constants.constants.Municipalities attribute), 33		
MORALZARZAL	(citram_api.constants.constants.Municipalities attribute), 33		
MORATA_DE_TAJUNA	(citram_api.constants.constants.Municipalities attribute), 33		
MOSTOLES	(citram_api.constants.constants.Municipalities attribute), 33		
Municipalities	(class in citram_api.constants.constants), 31		
N			
NAVACERRADA	(citram_api.constants.constants.Municipalities attribute), 34		
NAVALAFUENTE	(citram_api.constants.constants.Municipalities attribute), 34		
NAVALAGAMELLA	(citram_api.constants.constants.Municipalities attribute), 34		
NAVALCARNERO	(citram_api.constants.constants.Municipalities attribute), 34		
NAVARREDONDA	(citram_api.constants.constants.Municipalities attribute), 34		
NAVAS_DEL_REY	(citram_api.constants.constants.Municipalities attribute), 34		
NUEVO_BAZTAN	(citram_api.constants.constants.Municipalities attribute), 34		
		OfficeTypes	(class in citram_api.constants.constants), 36
		OFICINA	(citram_api.constants.constants.OfficeTypes attribute), 36
		OLMEDA_DE_LAS_FUENTES	(citram_api.constants.constants.Municipalities attribute), 34
		ORUSCO_DE_TAJUNA	(citram_api.constants.constants.Municipalities attribute), 34
P			
		PARACUELLOS_DE_JARAMA	(citram_api.constants.constants.Municipalities attribute), 34
		PARLA	(citram_api.constants.constants.Municipalities attribute), 34
		PAONES	(citram_api.constants.constants.Municipalities attribute), 34
		PEDREZUELA	(citram_api.constants.constants.Municipalities attribute), 34
		PELAYOS_DE_LA_PRESA	(citram_api.constants.constants.Municipalities attribute), 34
		PERALES_DE_TAJUNA	(citram_api.constants.constants.Municipalities attribute), 34
		PEZUELA_DE_LAS_TORRES	(citram_api.constants.constants.Municipalities attribute), 34
		PINILLA_DEL_VALLE	(citram_api.constants.constants.Municipalities attribute), 34
		PINTO	(citram_api.constants.constants.Municipalities attribute), 34
		PINUECAR	(citram_api.constants.constants.Municipalities attribute), 34
		POZUELO_DE_ALARCON	(citram_api.constants.constants.Municipalities attribute), 34
		POZUELO_DEL_REY	(citram_api.constants.constants.Municipalities attribute), 34
		PUEDENA_DEL_RINCON	(citram_api.constants.constants.Municipalities attribute), 34
		PUENTES_VIEJAS	(citram_api.constants.constants.Municipalities attribute), 34
		QUIJORNA	(citram_api.constants.constants.Municipalities attribute), 34

R

RASCAFRIA (*citram_api.constants.constants.Municipalities*
attribute), 34

RECARGA (*citram_api.constants.constants.OfficeTypes*
attribute), 36

REDUENA (*citram_api.constants.constants.Municipalities*
attribute), 34

RIBATEJADA (*citram_api.constants.constants.Municipalities*
attribute), 34

RIVAS_VACIAMADRID (*citram_api.constants.constants.Municipalities*
attribute), 34

ROBLEDILLO_DE_LA_JARA (*citram_api.constants.constants.Municipalities*
attribute), 34

ROBLEDO_DE_CHAVELA (*citram_api.constants.constants.Municipalities*
attribute), 34

ROBREGORDO (*citram_api.constants.constants.Municipalities*
attribute), 34

ROZAS_DE_MADRID_LAS (*citram_api.constants.constants.Municipalities*
attribute), 34

ROZAS_DE_PUERTO_REAL (*citram_api.constants.constants.Municipalities*
attribute), 34

S

SAN_AGUSTIN_DE_GUADALIX (*citram_api.constants.constants.Municipalities*
attribute), 35

SAN_FERNANDO_DE_HENARES (*citram_api.constants.constants.Municipalities*
attribute), 35

SAN_LORENZO_DE_EL_ESCORIAL (*citram_api.constants.constants.Municipalities*
attribute), 35

SAN_MARTIN_DE_LA_VEGA (*citram_api.constants.constants.Municipalities*
attribute), 35

SAN_MARTIN_DE_VALDEIGLESIAS (*citram_api.constants.constants.Municipalities*
attribute), 35

SAN_SEBASTIAN_DE_LOS_REYES (*citram_api.constants.constants.Municipalities*
attribute), 35

SANTA_MARIA_DE_LA_ALAMEDA (*citram_api.constants.constants.Municipalities*
attribute), 34

SANTORCAZ (*citram_api.constants.constants.Municipalities*
attribute), 34

SANTOS_DE_LA_HUMOSA_LOS (*citram_api.constants.constants.Municipalities*
attribute), 34

SERNA_DEL_MONTE_LA (*citram_api.constants.constants.Municipalities*
attribute), 35

SERRANILLOS_DEL_VALLE (*citram_api.constants.constants.Municipalities*
attribute), 35

SEVILLA_LA_NUEVA (*citram_api.constants.constants.Municipalities*
attribute), 35

SOMOSIERRA (*citram_api.constants.constants.Municipalities*
attribute), 35

SOTO_DEL_REAL (*citram_api.constants.constants.Municipalities*
attribute), 35

T

TALAMANCA_DEL_JARAMA (*citram_api.constants.constants.Municipalities*
attribute), 35

TIELMES (*citram_api.constants.constants.Municipalities*
attribute), 35

TITULCIA (*citram_api.constants.constants.Municipalities*
attribute), 35

TORREJON_DE_ARDOZ (*citram_api.constants.constants.Municipalities*
attribute), 35

TORREJON_DE_LA_CALZADA (*citram_api.constants.constants.Municipalities*
attribute), 35

TORREJON_DE_VELASCO (*citram_api.constants.constants.Municipalities*
attribute), 35

TORRELAGUNA (*citram_api.constants.constants.Municipalities*
attribute), 35

TORRELODONES (*citram_api.constants.constants.Municipalities*
attribute), 35

TORREMOCHA_DE_JARAMA (*citram_api.constants.constants.Municipalities*
attribute), 35

TORRES_DE_LA_ALAMEDA (*citram_api.constants.constants.Municipalities*
attribute), 35

TransportModes (*class in citram_api.constants.constants*), 36

TRES_CANTOS (*citram_api.constants.constants.Municipalities*
attribute), 35

V

VALDARACETE (*citram_api.constants.constants.Municipalities*
attribute), 35

VALDEAVERO (*citram_api.constants.constants.Municipalities*
attribute), 35

VALDELAGUNA (*citram_api.constants.constants.Municipalities*
attribute), 35

VALDEMANCO (*citram_api.constants.constants.Municipalities*
attribute), 35

VALDEMAQUEDA (*citram_api.constants.constants.Municipalities*
attribute), 35

VALDEMORILLO (*citram_api.constants.constants.Municipalities*
attribute), 35

VALDEMORO (*citram_api.constants.constants.Municipalities*
attribute), 35

VALDEOLMOS_ALALPARDO (*citram_api.constants.constants.Municipalities*
attribute), 35

VALDEPIELAGOS (*citram_api.constants.constants.Municipalities*
attribute), 35

VALDETORRES_DE_JARAMA (*citram_api.constants.constants.Municipalities*
attribute), 35

VALDILECHA (*citram_api.constants.constants.Municipalities*
attribute), 35

VALVERDE_DE_ALCALA (*citram_api.constants.constants.Municipalities*
attribute), 35

VELILLA_DE_SAN_ANTONIO (*citram_api.constants.constants.Municipalities*
attribute), 35

VELLON_EL (*citram_api.constants.constants.Municipalities*
attribute), 35

VENTURADA (*citram_api.constants.constants.Municipalities*
attribute), 36

VILLA_DEL_PRADO (*citram_api.constants.constants.Municipalities*
attribute), 36

VILLACONEJOS (*citram_api.constants.constants.Municipalities*
attribute), 36

VILLALBILLA (*citram_api.constants.constants.Municipalities*
attribute), 36

VILLAMANRIQUE_DE_TAJO (*citram_api.constants.constants.Municipalities*
attribute), 36

VILLAMANTA (*citram_api.constants.constants.Municipalities*
attribute), 36

VILLAMANTILLA (*citram_api.constants.constants.Municipalities*
attribute), 36

VILLANUEVA_DE_LA_CANADA (*citram_api.constants.constants.Municipalities*
attribute), 36

VILLANUEVA_DE_PERALES (*citram_api.constants.constants.Municipalities*
attribute), 36

VILLANUEVA_DEL_PARDILLO (*citram_api.constants.constants.Municipalities*
attribute), 36

VILLAR_DEL_OLMO (*citram_api.constants.constants.Municipalities*
attribute), 36

VILLAVICIOSA_DE_ODON (*citram_api.constants.constants.Municipalities*
attribute), 36

VILLAVIEJA_DEL_LOZOYA (*citram_api.constants.constants.Municipalities*
attribute), 36

ZARZALEJO (*citram_api.constants.constants.Municipalities*
attribute), 36