

# SPEC Utils

September 2, 2020

## 1 SPEC Utils

sdks for SPEC, SA and third applications

### 1.1 Install

Can use:

```
pip install spec-utils
```

Or:

```
python -m pip install spec-utils
```

### 1.2 Net-Time

#### 1.2.1 Import module

```
[2]: from spec_utils import nettime6 as nt6
```

#### 1.2.2 Client settings

```
[5]: URL = 'http://localhost:6091'  
     USERNAME = 'admin'  
     PWD = 'Spec.2020'
```

#### 1.2.3 Create a client

```
[6]: client = nt6.Client(url=URL, username=USERNAME, pwd=PWD)  
     client.is_connected
```

```
[6]: True
```

#### 1.2.4 Get employees with summary method

```
[45]: client.get_employees()
```

```
[45]: {'total': 2,  
      'items': [{'id': 1, 'nif': '123789456'}, {'id': 2, 'nif': '987321654'}]}
```

### 1.2.5 Filter in response (frontend)

```
[46]: client.get_employees(search="1237")
```

```
[46]: {'total': 1, 'items': [{'id': 1, 'nif': '123789456'}]}
```

### 1.2.6 Specify fields

```
[47]: query = nt6.Query(fields=["nif", "Apellidos_Nombre", "Province", "birthdate"])
client.get_employees(query=query)
```

```
[47]: {'total': 2,
      'items': [{'nif': '123789456',
                  'Apellidos_Nombre': 'Lucyk, Lucas',
                  'birthdate': '0001-01-01T00:00:00.0000000'},
                {'nif': '987321654',
                  'Apellidos_Nombre': 'Doe, John',
                  'Province': 'Ciudad Autónoma de Buenos Aires',
                  'birthdate': '1987-08-05T00:00:00.0000000'}]}
```

### 1.2.7 Filter in backend with nettime filter

```
[9]: query = nt6.Query(
      fields=["id", "nif", "Apellidos_Nombre", "Province", "birthdate"],
      filterExp='Contains(this.nif, "987321654")'
    )
client.get_employees(query=query)
```

```
[9]: {'total': 1,
      'items': [{'id': 2,
                  'nif': '987321654',
                  'Apellidos_Nombre': 'Doe, John',
                  'Province': 'Ciudad Autónoma de Buenos Aires',
                  'birthdate': '1987-08-05T00:00:00.0000000'}]}
```

```
[10]: query = nt6.Query(
      fields=["id", "nif", "Apellidos_Nombre", "birthdate", "Persona.
      ↪CalendarioBase"],
    )
client.get_employees(query=query)
```

```
[10]: {'total': 2,
      'items': [{'id': 1,
                  'nif': '123789456',
                  'Apellidos_Nombre': 'Lucyk, Lucas',
                  'birthdate': '0001-01-01T00:00:00.0000000',
                  'Persona.CalendarioBase': 'Flexible 8h'},
                {'id': 2,
                  'nif': '987321654',
                  'Apellidos_Nombre': 'Doe, John',
                  'Province': 'Ciudad Autónoma de Buenos Aires',
                  'birthdate': '1987-08-05T00:00:00.0000000'}]}
```

```
{'id': 2,
  'nif': '987321654',
  'Apellidos_Nombre': 'Doe, John',
  'birthdate': '1987-08-05T00:00:00.0000000',
  'Persona.CalendarioBase': 'LaV 09 a 18'}}}
```

### 1.2.8 Fields definition

```
[49]: fields = client.get_fields("Persona")
print('Total:', fields.get('total'))
fields.get('items')[:2]
```

Total: 323

```
[49]: [{ 'id': 30,
        'name': 'id',
        'displayName': 'Id',
        'expr': 'this.id',
        'type': 'int',
        'align': 2,
        'sortable': False,
        'width': 0,
        'group': '',
        'numTempOperators': 0},
       { 'id': 31,
        'name': 'name',
        'displayName': 'Clave',
        'expr': 'this.name',
        'type': 'String',
        'align': 2,
        'sortable': True,
        'width': 20,
        'group': 'Datos personales',
        'numTempOperators': 0}]
```

### 1.2.9 Fields definition filtering Properties only

```
[50]: fields_filter = client.get_fields("Persona", filterFields=True)
print('Total:', fields_filter.get('total'))
fields_filter.get('items')[:2]
```

Total: 316

```
[50]: [{ 'id': 30,
        'name': 'id',
        'displayName': 'Id',
        'expr': 'this.id',
```

```

        'type': 'int',
        'align': 2,
        'sortable': False,
        'width': 0,
        'group': '',
        'numTempOperators': 0},
    {'id': 31,
     'name': 'name',
     'displayName': 'Clave',
     'expr': 'this.name',
     'type': 'String',
     'align': 2,
     'sortable': True,
     'width': 20,
     'group': 'Datos personales',
     'numTempOperators': 0}]

```

#### 1.2.10 Get incidencias, and update employee

```

[51]: nt_incidencias = client.get_elements("Incidencia").get('items')
      nt_incidencias[:10]

```

```

[51]: [{ 'id': 0, 'name': 'Sin incidencia'},
      { 'id': 1, 'name': 'Inc. horas extra'},
      { 'id': 2, 'name': 'Asuntos propios'},
      { 'id': 3, 'name': 'Vacaciones'},
      { 'id': 4, 'name': 'Lactancia 30 minutos'},
      { 'id': 5, 'name': 'Lactancia 1 hora'},
      { 'id': 6, 'name': 'Visita al médico'},
      { 'id': 7, 'name': 'Horas sindicales'},
      { 'id': 8, 'name': 'Accidente laboral'},
      { 'id': 9, 'name': 'Enfermedad'}]

```

```

[52]: incidencias = []
      for incidencia in nt_incidencias:
          incidencias.append({"id": incidencia.get("id")})

      data = {
          "container": "Persona",
          "elements": [2],
          "dataObj": {
              "TimeTypesEmployee": incidencias
          }
      }

      client.save_element(**data)

```

```
[52]: [{ 'type': 6,
  'dataObject': { '_c_': 'Persona',
    'id': 2,
    'name': '987654',
    'created': '2020-08-03T10:51:59.1430000',
    'changePassword': False,
    'firstWONumDays': 0,
    'firstWO': -1,
    'firstWANumDays': 0,
    'firstWA': -1,
    'firstTTNumDays': 0,
    'firstTT': -1,
    'termVoice': False,
    'timeoutEnrollMinutes': 0,
    'timeoutEnrollDate': '0001-01-01T00:00:00.0000000',
    'enrollActive': False,
    'PersonalPhone': '5238-8000',
    'sex': 0,
    'Visitas.RecibirVisitas': False,
    'securityLevel': 0,
    'ticketEmail': False,
    'address': 'Reconquista 968',
    'timeoutEnroll': '0001-01-01T00:00:00.0000000',
    'createdBy': 'Admin',
    'modifiedBy': 'Admin',
    'useTasksWE': False,
    'useTasksWO': False,
    'useTasksWA': False,
    'useTasks': False,
    'ProfaceAdmin': False,
    'MobileClocking': False,
    'mobileId': '',
    'RemoteClocking': False,
    'Portal.DisablePasswordChange': False,
    'Portal.DisableCalendar': False,
    'Portal.DisablePlannings': False,
    'Portal.DisableVistaResumen': False,
    'Portal.DisableMovimientos': False,
    'Portal.NoPuedeEditar': False,
    'Portal.NoRequiereValidacionEnCorreccion': False,
    'Portal.UsaPortal': False,
    'virtualCard': False,
    'geolocalize': 'geoNever',
    'offline': False,
    'totalDocs': 0,
    'Portal.ChangeLanguage': True,
    'rev': 56,
```

```

'modified': '2020-08-05T19:15:49.1907439-03:00',
'birthdate': '1987-08-05T00:00:00.0000000',
'Visitas.ProgramarVisitas': False,
'acceptAllReaders': False,
'acceptAllTT': False,
'NoAttendance': False,
'RegisterSystemDate': '2019-02-11T00:00:00.0000000',
'htmlPortal': False,
'inactive': True,
'pwdCantChange': False,
'exboss': True,
'pwdNextLogin': False,
'pwdExpires': True,
'pwdRetries': 0,
'lastPwdChange': '0001-01-01T00:00:00.0000000',
'NumFingers': 0,
'nif': '987321654',
'Town': 'CABA',
'PIN': 0,
'PostalCode': '1003',
'FirstDayNotValid': '2020-08-05T00:00:00.0000000-03:00',
'Province': 'Ciudad Autónoma de Buenos Aires',
'FingerIEV02EnrollDate': '0001-01-01T00:00:00.0000000',
'nameEmployee': 'John',
'companyCode': 'SPEC AR',
'Finger2EnrollDate': '0001-01-01T00:00:00.0000000',
'Finger1EnrollDate': '0001-01-01T00:00:00.0000000',
'LastName': 'Doe',
'employeeCode': '5048',
'FingerIEV01EnrollDate': '0001-01-01T00:00:00.0000000',
'ActiveDays': {'validity': [{'start': '2011-01-01T00:00:00.0000000',
    'end': '2040-12-31T00:00:00.0000000'}]},
'Readers': [1, 2],
'Cards': [],
'Calendar': {'id': 2,
    '_c_': '',
    'created': '0001-01-01T00:00:00.0000000',
    'modified': '0001-01-01T00:00:00.0000000',
    'name': '',
    'rev': 0,
    'years': [{'Year': 2012, 'days': {}},
        {'Year': 2016, 'days': {}},
        {'Year': 2017, 'days': {}},
        {'Year': 2018, 'days': {}},
        {'Year': 2019, 'days': {}},
        {'Year': 2020,
            'days': {'0': {'shifts': [4]}},

```

```

    '1': {'shifts': [4]},
    '33': {'shifts': [3]},
    '34': {'shifts': [3]},
    '35': {'shifts': [3]},
    '36': {'shifts': [3]},
    '37': {'shifts': [3]},
    '54': {'shifts': [3]},
    '55': {'shifts': [3]},
    '56': {'shifts': [3]},
    '57': {'shifts': [3]},
    '58': {'shifts': [3]}}},
    {'Year': 2021, 'days': {}}],
    'Cycles': [],
    'Calendars': [{'id': 3, 'name': 'LaV 09 a 18'}],
    'nodesSource': [],
    'multiName': {'es-ES': ''}},
    'node': 1,
    'doAccess': {'total': True, 'offsetIn': 0, 'offsetOut': 0},
    'TimeTypesEmployee': [{'id': 0},
    {'id': 1},
    {'id': 2},
    {'id': 3},
    {'id': 4},
    {'id': 5},
    {'id': 6},
    {'id': 7},
    {'id': 8},
    {'id': 9},
    {'id': 10},
    {'id': 11},
    {'id': 12},
    {'id': 13},
    {'id': 14},
    {'id': 15},
    {'id': 16},
    {'id': 17},
    {'id': 18},
    {'id': 19},
    {'id': 20},
    {'id': 21},
    {'id': 22},
    {'id': 23},
    {'id': 24}],
    'Departments': [{'id': 1}],
    'enrollDevices': [],
    'geolocSource': [{'data': 'geoAlways', 'label': 'Siempre'},
    {'data': 'geoIfPossible', 'label': 'Si es posible'},

```

```

{'data': 'geoNever', 'label': 'Nunca'}],
'source': [{ 'id': 1,
  'name': 'Proximity',
  'displayName': 'Proximity',
  'description': '',
  'type': 'terminal',
  'icon': 'Terminal',
  'children': [{ 'id': 2,
    'displayName': 'Proximity_1',
    'type': 'reader',
    'description': 'Lector 2 del terminal Proximity',
    'icon': 'Lector',
    'children': [{ 'type': 'device',
      'name': 'CD1',
      'displayName': 'Lector de proximidad interno conectado en CD1',
      'description': 'CD1',
      'default': 1,
      'deviceType': 'Tarjeta',
      'hasElement': True,
      'expr': '',
      'allow': False,
      'id': 'Proximity_Proximity_1_CD1' } ] } ] },
{ 'id': 3,
  'name': 'Fingerprint',
  'displayName': 'Fingerprint',
  'description': '',
  'type': 'terminal',
  'icon': 'Terminal',
  'children': [{ 'id': 1,
    'displayName': 'Fingerprint_1',
    'type': 'reader',
    'description': 'Lector 1 del terminal Fingerprint',
    'icon': 'Lector',
    'children': [{ 'type': 'device',
      'name': 'COM1',
      'displayName': 'Lector de huella interno conectado en COM1',
      'description': 'COM1',
      'default': 1,
      'deviceType': 'Huella',
      'hasElement': True,
      'expr': '',
      'allow': False,
      'id': 'Fingerprint_Fingerprint_1_COM1' },
    { 'type': 'device',
      'name': 'BI1',
      'displayName': 'Teclado interno conectado en BI1',
      'description': 'BI1',

```



```

        'default': 1,
        'deviceType': 'Teclado',
        'hasElement': False,
        'expr': '',
        'allow': False,
        'id': 'Fingerprint_Fingerprint_1_BI1'}}]]]],
    'initialValuesList': [],
    'nodesSource': [{'data': '1', 'label': 'SPEC SA'},
                     {'data': '2', 'label': 'SPEC SA · Argentina'},
                     {'data': '3', 'label': 'SPEC SA · Argentina · Information Technology'}],
    'Accesos.Zona': 1,
    'languages': [{'data': 'ca-ES', 'label': 'Català'},
                  {'data': 'en-GB', 'label': 'English'},
                  {'data': 'es-ES', 'label': 'Español'},
                  {'data': 'eu-ES', 'label': 'Euskara'},
                  {'data': 'fr-FR', 'label': 'Français'},
                  {'data': 'pt-PT', 'label': 'Português'}]],
    'message': 'El elemento se ha modificado correctamente.',
    'showActions': True}]

```

To add periods, use the key “validity” with a list of elements like:

```

{
    "id": incidencia.get("id"),
    "validity": [{
        "end": "2040-12-31T00:00:00-03:00",
        "start": "2004-01-01T00:00:00-03:00",
    }]
}

```

### 1.2.11 Get elements definition

```

[53]: employee = client.get_element_def(container="Persona", elements=[2])
      employee = employee[0]

# show calendars
employee.get('Calendar')

```

```

[53]: {'id': 2,
      '_c_': '',
      'created': '0001-01-01T00:00:00.0000000',
      'modified': '0001-01-01T00:00:00.0000000',
      'name': '',
      'rev': 0,
      'years': [{'Year': 2012, 'days': {}},
                 {'Year': 2016, 'days': {}},
                 {'Year': 2017, 'days': {}},

```

```

{'Year': 2018, 'days': {}},
{'Year': 2019, 'days': {}},
{'Year': 2020,
 'days': {'0': {'shifts': [4]},
 '1': {'shifts': [4]},
 '33': {'shifts': [3]},
 '34': {'shifts': [3]},
 '35': {'shifts': [3]},
 '36': {'shifts': [3]},
 '37': {'shifts': [3]},
 '54': {'shifts': [3]},
 '55': {'shifts': [3]},
 '56': {'shifts': [3]},
 '57': {'shifts': [3]},
 '58': {'shifts': [3]}}},
{'Year': 2021, 'days': {}},
'Cycles': [],
'Calendars': [{'id': 3, 'name': 'LaV 09 a 18'}],
'nodesSource': [],
'multiName': {'es-ES': ''}}

```

### 1.2.12 Get default values in create form

```
[54]: client.get_create_form("Jornada")
```

```

[54]: {'_c_': 'Jornada',
 'id': -1,
 'modified': '0001-01-01T00:00:00.0000000',
 'created': '0001-01-01T00:00:00.0000000',
 'color': '808080',
 'rev': 0,
 'minutosCortesia': 0,
 'minutosPenalizacion': 0,
 'totalTeorico': 480,
 'minutoFinal': 2880,
 'minutosRetraso': 0,
 'resultados': [],
 'multiName': {},
 'nodesSource': [],
 'incidencias': [],
 'baObligada': [],
 'baFlexible': [],
 'pausas': [],
 'intervalSource': [{'data': 'flex', 'label': 'Bloque flexible'},
 {'data': 'oblig', 'label': 'Bloque obligatorio'},
 {'data': 'all', 'label': 'Toda la jornada'},
 {'data': 'bloque', 'label': 'Bloque del grupo de incidencia'}],

```

```
{'data': 'relative', 'label': 'Relativo a...', 'state': 'relative'},
{'data': 'delete', 'label': 'Sin validez'}]],
'ShifttimeTypesMassIdinci': [],
'relativeSource': [{'data': 'inishift', 'label': 'Inicio de la jornada'},
{'data': 'firstflexmin', 'label': 'Inicio bloque flexible'},
{'data': 'firstoblmin', 'label': 'Inicio bloque obligatorio'},
{'data': 'endshift', 'label': 'Final de la jornada'},
{'data': 'endflex', 'label': 'Final bloque flexible'},
{'data': 'endobli', 'label': 'Final bloque obligatorio'}]]}
```

### 1.2.13 Get dataObj for duplicate element

```
[55]: client.get_for_duplicate(container="Arbol", element=3)
# edit and then use client.save_element() method
```

```
[55]: {'_c_': 'Arbol',
'id': -1,
'name': 'Information Technology',
'rev': 0,
'createdBy': 'Admin',
'created': '2020-08-03T11:08:27.4050000',
'modified': '0001-01-01T00:00:00.0000000',
'color': '7D7D7D',
'allowedContainerNames': 'Empleados',
'order': 0,
'internalName': 'Information_Technology',
'idNodeParent': 2,
'baAllowedContainers': [14],
'nodesSource': [],
'nodes': [],
'multiName': {'es-ES': 'Information Technology'}}
```

### 1.2.14 Delete element

```
[56]: client.get_elements("Jornada", search="TEST")
```

```
[56]: {'total': 1, 'items': [{'id': 6, 'name': 'TEST'}]}
```

```
[58]: client.delete_element(container="Jornada", elements=[6])
```

```
[58]: [{'type': 8,
'id': 'BwAAAAEAAgeAAA==',
'rev': 60,
'message': 'Los elementos se han eliminado correctamente.'}]
```

```
[59]: client.get_elements("Jornada", search="TEST")
```

```
[59]: {'total': 0, 'items': []}
```

### 1.2.15 Getting day results info

```
[60]: client.get_day_info(employee=2, _from="2020-07-03", to="2020-07-03")
```

```
[60]: {'idEmp': 2,
      'days': [{'date': '2020-07-03T00:00:00.0000000',
                  'shift': {'date': '2020-07-03T00:00:00.0000000',
                             'idEmp': 2,
                             'shift': 5,
                             'minFin': 2880,
                             'minFinForced': False,
                             'shiftPetition': {'actions': ['Change']}},
                  'clockings': [{'id': 1,
                                  'date': '2020-07-03T08:55:00.0000000',
                                  'idElem': 0,
                                  'type': 'timetypes',
                                  'idReader': 0,
                                  'user': 'Admin',
                                  'ip': '127.0.0.1',
                                  'status': {'effective': True,
                                              'desc': 'Entrando',
                                              'state': '',
                                              'entering': True,
                                              'actions': ['Delete', 'Edit', 'Comment']}},
                                  'app': True,
                                  'numDocuments': 0}],
                  {'id': 2,
                   'date': '2020-07-03T18:30:00.0000000',
                   'idElem': 0,
                   'type': 'timetypes',
                   'idReader': 0,
                   'user': 'Admin',
                   'ip': '127.0.0.1',
                   'status': {'effective': True,
                               'desc': 'Saliendo',
                               'state': '',
                               'entering': False,
                               'actions': ['Delete', 'Edit', 'Comment']}},
                   'app': True,
                   'numDocuments': 0}],
      'info': {'Change': 'Cambiar',
                'Delete': 'Eliminar',
                'Edit': 'Editar',
                'Comment': 'Comentar'}},
      'results': {'date': '2020-07-03T00:00:00.0000000',
```

```

    'hasComments': False,
    'hasPending': False,
    'shift': {'id': 5, 'minutes': {'start': 1440, 'end': 2880}},
    'minutesTypes': [{'name': 'Incidencia',
        'results': [{'id': 0,
            'minutes': [{'start': 1975, 'end': 2160, 'endApr': True},
                {'start': 2220, 'end': 2550, 'startApr': True}]}]}],
    {'name': 'Sistema',
        'results': [{'id': 9, 'minutes': [{'start': 2160, 'end': 2220}]}]}],
    'taskConfig': 71,
    'info': [{'id': 0,
        'type': 'Incidencia',
        'name': 'Sin incidencia',
        'color': '009933'},
        {'id': 5, 'type': 'Jornada', 'name': '09:00 a 18:00', 'color': '808080'},
        {'id': 9, 'type': 'Sistema', 'name': 'Pausas', 'color': 'F0F0F0'}]]

```

### 1.2.16 Get day results

```
[61]: client.get_results(employee=2, _from="2020-07-03", to="2020-07-03")
```

```

[61]: {'results': [{'date': '2020-07-03T00:00:00.0000000',
    'hasComments': False,
    'hasPending': False,
    'shift': {'id': 5, 'minutes': {'start': 1440, 'end': 2880}},
    'minutesTypes': [{'name': 'Lector',
        'results': [{'id': -1,
            'values': [{'name': 'Min', 'value': 575},
                {'name': 'MinDes', 'value': 0},
                {'name': 'Evt', 'value': 1},
                {'name': 'EvtDes', 'value': 0}]}]}],
    {'name': 'Incidencia',
        'results': [{'id': 0,
            'values': [{'name': 'Min', 'value': 515},
                {'name': 'MinDes', 'value': 60},
                {'name': 'Evt', 'value': 1},
                {'name': 'EvtDes', 'value': 1}]}]}],
    {'name': 'Sistema',
        'results': [{'id': 0,
            'values': [{'name': 'Min', 'value': 515},
                {'name': 'MinDes', 'value': 60},
                {'name': 'Evt', 'value': 1},
                {'name': 'EvtDes', 'value': 1}]}],
        {'id': 5,
            'values': [{'name': 'Min', 'value': 0},
                {'name': 'MinDes', 'value': 0},
                {'name': 'Evt', 'value': 0},

```

```

    {'name': 'EvtDes', 'value': 0}}],
{'id': 6,
 'values': [{'name': 'Min', 'value': 515},
             {'name': 'MinDes', 'value': 60},
             {'name': 'Evt', 'value': 1},
             {'name': 'EvtDes', 'value': 1}]],
{'id': 8,
 'values': [{'name': 'Min', 'value': 480},
             {'name': 'MinDes', 'value': 35},
             {'name': 'Evt', 'value': 1},
             {'name': 'EvtDes', 'value': 1}]],
{'id': 9,
 'values': [{'name': 'Min', 'value': 60},
             {'name': 'MinDes', 'value': 0},
             {'name': 'Evt', 'value': 1},
             {'name': 'EvtDes', 'value': 0}]],
{'id': 10,
 'values': [{'name': 'Min', 'value': 0},
             {'name': 'MinDes', 'value': 0},
             {'name': 'Evt', 'value': 0},
             {'name': 'EvtDes', 'value': 0}]],
{'id': 11,
 'values': [{'name': 'Min', 'value': 480},
             {'name': 'MinDes', 'value': 0},
             {'name': 'Evt', 'value': 1},
             {'name': 'EvtDes', 'value': 0}]],
{'id': 12,
 'values': [{'name': 'Min', 'value': 1380},
             {'name': 'MinDes', 'value': 0},
             {'name': 'Evt', 'value': 1},
             {'name': 'EvtDes', 'value': 0}]]}],
{'name': 'Anomalia',
 'results': [{'id': 0,
               'values': [{'name': 'Min', 'value': 0},
                           {'name': 'MinDes', 'value': 0},
                           {'name': 'Evt', 'value': 1},
                           {'name': 'EvtDes', 'value': 0}]]}],
{'name': 'Calculo',
 'results': [{'id': 1,
               'values': [{'name': 'Min', 'value': 515},
                           {'name': 'MinDes', 'value': 60},
                           {'name': 'Evt', 'value': 1},
                           {'name': 'EvtDes', 'value': 1}]],
              {'id': 2,
               'values': [{'name': 'Min', 'value': 480},
                           {'name': 'MinDes', 'value': 0},
                           {'name': 'Evt', 'value': 1},

```

```

    {'name': 'EvtDes', 'value': 0}}]]},
{'name': 'Jornada',
 'results': [{ 'id': 5,
  'values': [{ 'name': 'Min', 'value': 480},
   {'name': 'MinDes', 'value': 0},
   {'name': 'Evt', 'value': 1},
   {'name': 'EvtDes', 'value': 0}}]]},
{'name': 'Aritmetico',
 'results': [{ 'id': 1,
  'values': [{ 'name': 'Value', 'value': 35},
   {'name': 'Accumulated', 'value': 35},
   {'name': 'Available', 'value': -35},
   {'name': 'Discard', 'value': 0}}],
 'id': 2,
  'values': [{ 'name': 'Value', 'value': 35},
   {'name': 'Accumulated', 'value': -925},
   {'name': 'Available', 'value': 925},
   {'name': 'Discard', 'value': 0}}],
 'id': 3,
  'values': [{ 'name': 'Value', 'value': 35},
   {'name': 'Accumulated', 'value': -63325},
   {'name': 'Available', 'value': 63325},
   {'name': 'Discard', 'value': 0}}],
 'id': 4,
  'values': [{ 'name': 'Value', 'value': 0},
   {'name': 'Accumulated', 'value': 0},
   {'name': 'Available', 'value': 22},
   {'name': 'Discard', 'value': 0}}]]]]},
'info': [{ 'id': -1, 'type': 'Lector', 'name': 'UNKNOWN', 'color': 'FDFDFD'},
 { 'id': 0, 'type': 'Incidencia', 'name': 'Sin incidencia', 'color': '009933'},
 { 'id': 0,
  'type': 'Sistema',
  'name': 'Productivas en el centro',
  'color': '009933'},
 { 'id': 0, 'type': 'Anomalia', 'name': 'Sin anomalía', 'color': 'EEEEEE'},
 { 'id': 1, 'type': 'Calculo', 'name': 'PROD', 'color': 'CCCCCC'},
 { 'id': 2, 'type': 'Calculo', 'name': 'NORMALES', 'color': 'ffffff'},
 { 'id': 5,
  'type': 'Sistema',
  'name': 'Ausencias no justificadas',
  'color': 'C83327'},
 { 'id': 5, 'type': 'Jornada', 'name': '09:00 a 18:00', 'color': '808080'},
 { 'id': 6, 'type': 'Sistema', 'name': 'Productivas', 'color': '3399FF'},
 { 'id': 8, 'type': 'Sistema', 'name': 'Trabajadas', 'color': '009933'},
 { 'id': 9, 'type': 'Sistema', 'name': 'Pausas', 'color': 'F0F0F0'},
 { 'id': 10, 'type': 'Sistema', 'name': 'Retrasos', 'color': 'CC6600'},
 { 'id': 11, 'type': 'Sistema', 'name': 'Jornada teórica', 'color': 'FFFFFF'},

```

```
{'id': 12, 'type': 'Sistema', 'name': 'Filtro día', 'color': 'FFFFFF'},
{'id': 1, 'type': 'Aritmetico', 'name': 'Saldo diario', 'color': '0066CC'},
{'id': 2, 'type': 'Aritmetico', 'name': 'Saldo mensual', 'color': 'CC9900'},
{'id': 3, 'type': 'Aritmetico', 'name': 'Saldo anual', 'color': 'FF6600'},
{'id': 4,
  'type': 'Aritmetico',
  'name': 'Saldo Días Vacaciones',
  'color': '007373',
  'numeric': True}}}]}
```

### 1.2.17 Add clocking

```
[62]: client.add_clocking(employee=2, date="2020-07-02", time="18:30")
```

```
[62]: {'ok': True}
```

### 1.2.18 Get day clockings

```
[63]: client.get_day_clockings(employee=2, date="2020-07-02")
```

```
[63]: {'date': '2020-07-02T00:00:00.0000000',
  'idEmp': 2,
  'shift': 5,
  'minFin': 0,
  'minFinForced': False,
  'shiftPetition': {'actions': ['Change']},
  'clockings': [{'id': 3,
    'date': '2020-07-02T09:00:00.0000000',
    'idElem': 0,
    'type': 'timetypes',
    'idReader': 0,
    'user': 'Admin',
    'ip': '127.0.0.1',
    'status': {'effective': True,
      'desc': 'Entrando',
      'state': '',
      'entering': True,
      'actions': ['Delete', 'Edit', 'Comment']},
    'app': True,
    'numDocuments': 0},
  {'id': 7,
    'date': '2020-07-02T18:30:00.0000000',
    'idElem': 0,
    'type': 'timetypes',
    'idReader': 0,
    'user': 'Admin',
```



```

'ip': '::1',
'status': {'effective': True,
'desc': 'Saliendo',
'state': '',
'entering': False,
'actions': ['Delete', 'Edit', 'Comment']},
'app': True,
'numDocuments': 0}],
'info': {'Change': 'Cambiar',
'Delete': 'Eliminar',
'Edit': 'Editar',
'Comment': 'Comentar'}}

```

### 1.2.19 Edit clocking

```
[64]: client.edit_clocking(employee=2, clocking_id=7, date="2020-07-02", time="20:30")
```

```
[64]: {'ok': True}
```

### 1.2.20 Delete clocking

```
[65]: client.delete_clocking(employee=2, clocking_id=7, date="2020-07-02", time="20:
↪30")
```

```
[65]: {'ok': True}
```

### 1.2.21 Plannings

```
[66]: planning = client.get_create_form("Persona", action="NewPlanificacion")
planning.update({
    "name": "Testing Planning",
    "allDay": True,
    "allDayId": 9, #Timetype ID
    "employee": [2], #Employee ID
    "dateInterval": client.get_days_offset(["2020-10-10", "2020-10-11"]),
})
# dateInterval is a list of int with differences between setting firstDate
# can use get_days_offset() method
```

```
[67]: data = {
    "container": "IncidenciaFutura",
    "dataObj": planning,
}
client.save_element(**data)
```

```

[67]: [{ 'type': 6,
        'dataObject': { 'id': 3,
                        '_c_': 'IncidenciaFutura',
                        'firstDay': '2020-10-10T00:00:00.0000000',
                        'rev': 47,
                        'modifiedBy': 'Admin',
                        'modified': '2020-08-05T19:19:48.5323882-03:00',
                        'createdBy': 'Admin',
                        'created': '2020-08-05T19:19:48.5023432-03:00',
                        'allDayId': 9,
                        'allDay': True,
                        'selfOwner': False,
                        'state': '0',
                        'name': 'Testing Planning',
                        'comments': '',
                        'intState': 0,
                        'isAccepted': True,
                        'lastDay': '2020-10-11T00:00:00.0000000',
                        'numDays': 2,
                        'confirmBy': 'Admin',
                        'confirm': '2020-08-05T19:19:48.5173865-03:00',
                        'describe': '2 días, del 10/10/2020 al 11/10/2020 Todo el día con incidencia
Enfermedad',
                        'hasComment': False,
                        'error': '',
                        'stateDescription': 'Aceptada',
                        'describeTT': 'Enfermedad',
                        'isPending': False,
                        'isValidationPending': False,
                        'isDenied': False,
                        'totalDocs': 0,
                        'validatedDays': [],
                        'dateInterval': [6127, 6128],
                        'baValidElems': [1,
2,
3,
4,
5,
6,
7,
8,
9,
10,
11,
12,
13,
14,

```

```

15,
16,
17,
18,
19,
20,
21,
22,
23,
24],
'timeTypes': [9],
'nodesSource': [],
'employee': 2},
'message': 'El elemento se ha creado correctamente.',
'showActions': True}]

```

### 1.2.22 Add planning with summary method

```

[68]: # Delete last added element
client.delete_element(container="IncidenciaFutura", elements=[3])

```

```

[68]: [{ 'type': 8,
'id': 'BAAAAAEAEaAAA==',
'rev': 50,
'message': 'Realizada la solicitud de eliminación.'}]

```

```

[69]: client.add_planning(
    employee=2,
    name="Testing summary planning",
    timetype=9,
    days=["2020-10-10", "2020-10-11"],
)

```

```

[69]: [{ 'type': 6,
'dataObject': { 'id': 4,
'_c_': 'IncidenciaFutura',
'firstDay': '2020-10-10T00:00:00.0000000',
'rev': 53,
'modifiedBy': 'Admin',
'modified': '2020-08-05T19:20:18.5712279-03:00',
'createdBy': 'Admin',
'created': '2020-08-05T19:20:18.5457647-03:00',
'allDayId': 9,
'allDay': True,
'selfOwner': False,
'state': '0',
'name': 'Testing summary planning',

```

```

'comments': '',
'intState': 0,
'isAccepted': True,
'lastDay': '2020-10-11T00:00:00.0000000',
'numDays': 2,
'confirmBy': 'Admin',
'confirm': '2020-08-05T19:20:18.5586427-03:00',
'describe': '2 días, del 10/10/2020 al 11/10/2020 Todo el día con incidencia
Enfermedad',
'hasComment': False,
'error': '',
'stateDescription': 'Aceptada',
'describeTT': 'Enfermedad',
'isPending': False,
'isValidationPending': False,
'isDenied': False,
'totalDocs': 0,
'validatedDays': [],
'dateInterval': [6127, 6128],
'baValidElems': [1,
2,
3,
4,
5,
6,
7,
8,
9,
10,
11,
12,
13,
14,
15,
16,
17,
18,
19,
20,
21,
22,
23,
24],
'timeTypes': [9],
'nodesSource': [],
'employee': 2},
'message': 'El elemento se ha creado correctamente.',

```

```
'showActions': True}]
```

### 1.2.23 Edit planning

```
[70]: # search planning
planning_query = nt6.Query(
    fields=["id", "name", "allDayId", ],
    filterExp="((this.employee.nif = '987321654') && (this.firstDay =_
↪ '2020-10-10'))"
)
client.get_elements(container="IncidenciaFutura", query=planning_query)
```

```
[70]: {'total': 1,
      'items': [{'id': 4, 'name': 'Testing summary planning', 'allDayId': 9}]}
```

```
[71]: edit_planning = client.get_element_def(container="IncidenciaFutura",_
↪ elements=[4])
edit_planning = edit_planning[0]
edit_planning["dateInterval"]
```

```
[71]: [6127, 6128]
```

```
[72]: # ever fix employee; must be a list
edit_planning["employee"] = [edit_planning.get("employee")]

# apply your changes
day_to_add = client.get_days_offset(["2020-10-12"])
edit_planning["dateInterval"].extend(day_to_add)
edit_planning["dateInterval"]
```

```
[72]: [6127, 6128, 6129]
```

```
[73]: response = client.save_element(
    container="IncidenciaFutura",
    elements=[4],
    dataObj=edit_planning
)
response[0].get('message')
```

```
[73]: 'El elemento se ha modificado correctamente.'
```

### 1.2.24 Add Activator

```
[74]: client.get_elements("Activadores")
```

```
[74]: {'total': 1, 'items': [{'id': 1, 'name': 'APROBACION HS'}]}
```

```
[75]: client.add_activator(
    name="Testing summary activator",
    employees=[2],           #list of id's
    days=["2020-07-03"],    #list of str with isoformat date
    activator=1,            #activator id
    value=1,                #required depending activator config
    comment="testing"       #optional
)
```

```
[75]: [{ 'type': 6,
  'dataObject': { '_c_': 'UsoActivadores',
  'id': 3,
  'name': 'Testing summary activator',
  'modified': '0001-01-01T00:00:00.0000000',
  'createdBy': 'Admin',
  'rev': 40,
  'created': '2020-08-05T19:21:16.8877146-03:00',
  'numDays': 1,
  'lastDay': '2020-07-03T00:00:00.0000000',
  'firstDay': '2020-07-03T00:00:00.0000000',
  'comment': 'testing',
  'employees': [2],
  'activators': [{ 'value': 1, 'activator': 1}],
  'days': [6028],
  'nodesSource': [],
  'multiName': { 'es-ES': 'Testing summary activator' },
  'validActvs': [1] },
  'message': 'El elemento se ha creado correctamente.',
  'showActions': True}]
```

### 1.2.25 Get activity monitor

```
[76]: client.get_activity_monitor(employees=[2], _from="2020-07-02", to="2020-07-03")
```

```
[76]: [{ 'id': 2,
  'days': [{ 'id': 6027,
  'idAno': 2,
  'mars': [{ 'm': 540,
  'i': 0,
  't': 18,
  'e': 268435457,
  'o': 8,
  'c': '009933' } ] },
  'horari': [{ 'mi': 540,
  'mf': 720,
  'ia': False,
  'fa': True,
```

```

        'tv': 2,
        'idv': 5},
    {'mi': 780, 'mf': 1080, 'ia': True, 'fa': True, 'tv': 2, 'idv': 5},
    {'mi': 720, 'mf': 780, 'ia': True, 'fa': True, 'tv': 2, 'idv': 9}],
    'idJor': 5,
    'bJor': [{'startTime': 0, 'endTime': 1440}]],
    'planInfos': {}},
    {'info': {}}]

```

Data structured like:

```

[{'id': employee_ID,
  'days': [{
    'id': offset_day_number #{settings.firstDate},
    'idAno': anomaly_id,
    'mars': [{ # Clockings
      'm': minute_of_day,
      'i': timtype_id,
      't': type,
      'e': state,
      'o': origin,
      'c': color}],
    'horari': [{ # Jornada
      'mi': start,
      'mf': end,
      'ia': False,
      'fa': True,
      'tv': 2,
      'idv': 5},
    {'mi': 780, 'mf': 1080, 'ia': True, 'fa': True, 'tv': 2, 'idv': 5},
    {'mi': 720, 'mf': 780, 'ia': True, 'fa': True, 'tv': 2, 'idv': 9}],
    'idJor': shift_id,
    'bJor': [{'startTime': shift_start, 'endTime': shift_end}]]],
  'planInfos': {}},
  {'info': {}}]

```

### 1.2.26 Get cube results (like results query window)

```

[5]: client.get_cube_results(
    dateIni="2020-07-02",
    dateEnd="2020-07-04",
    dimensions=[
        ["id", "Apellidos_Nombre"],
        ["date"],
        ["ResultValue_C.Min.NORMALES", "ResultValue_S.Min.Jornada_teorica"]
    ]
)

```

```
[5]: [{'dimKey': [1, 'Lucyk, Lucas'],
      'values': [0, 1440],
      'children': [{'dimKey': ['2020-07-02'], 'values': [0, 480]},
                    {'dimKey': ['2020-07-03'], 'values': [0, 480]},
                    {'dimKey': ['2020-07-04'], 'values': [0, 480]}]},
      {'dimKey': [2, 'Doe, John'],
      'values': [480, 960],
      'children': [{'dimKey': ['2020-07-02'], 'values': [0, 480]},
                    {'dimKey': ['2020-07-03'], 'values': [480, 480]},
                    {'dimKey': ['2020-07-04'], 'values': [0, 0]}]}]
```

**For more information see the documentation**

```
[15]: help(client.get_cube_results)
```

Help on method get\_cube\_results in module nettime6:

```
get_cube_results(dimensions: list, dateIni: str, dateEnd: str, interfilters:
list = [], filters: list = [], ids: list = []) method of nettime6.Client
instance
```

Gets nettime results using the "Results Query" window engine.

```
:param dimensions: List of list where each one contains the desired
                    fields or results. The order of the values does matter.
:param dateIni: Start day where you want to calculate.
:param dateEnd: End day where you want to calculate.
:param interfilters: (Optional) If you use the dimension "system",
                    specify the ids of the results in this parameter.
:param filters: (Optional) Nettime compatible filter expression.
:param ids: (Optional) List of employee ids in case you want to filter.

:return: :class:`list` object
:rtype: json
```

**Another example...**

```
[17]: # get system result ids
client.get_elements("SystemResult")
```

```
[17]: {'total': 13,
      'items': [{'id': 0, 'name': 'Productivas en el centro'},
                 {'id': 1, 'name': 'Productivas fuera del centro'},
                 {'id': 2, 'name': 'No productivas en el centro'},
                 {'id': 3, 'name': 'No productivas fuera del centro'},
                 {'id': 4, 'name': 'Ausencias justificadas'},
                 {'id': 5, 'name': 'Ausencias no justificadas'},
                 {'id': 6, 'name': 'Productivas'}],
```



```
{'id': 7, 'name': 'No productivas'},
{'id': 8, 'name': 'Trabajadas'},
{'id': 9, 'name': 'Pausas'},
{'id': 10, 'name': 'Retrasos'},
{'id': 11, 'name': 'Jornada teórica'},
{'id': 12, 'name': 'Filtro día']}]}
```

```
[26]: # get results with system engine
client.get_cube_results(
    dateIni="2020-07-02",
    dateEnd="2020-07-04",
    dimensions=[
        ["id", "Apellidos_Nombre"],
        ["date"],
        ["sistema"],
        ["interTotal"]
    ],
    interFilters=[{
        "dimension": "sistema",
        "elements": [6, 11] #Productivas and Jornada teórica
    }]
)
```

```
[26]: [{'dimKey': [1, 'Lucyk, Lucas'],
'values': [1440],
'children': [{'dimKey': ['2020-07-02'],
'values': [480],
'children': [{'dimKey': ['Jornada teórica'], 'values': [480]}]}],
{'dimKey': ['2020-07-03'],
'values': [480],
'children': [{'dimKey': ['Jornada teórica'], 'values': [480]}]}],
{'dimKey': ['2020-07-04'],
'values': [480],
'children': [{'dimKey': ['Jornada teórica'], 'values': [480]}]}]}],
{'dimKey': [2, 'Doe, John'],
'values': [1475],
'children': [{'dimKey': ['2020-07-02'],
'values': [480],
'children': [{'dimKey': ['Jornada teórica'], 'values': [480]}]}],
{'dimKey': ['2020-07-03'],
'values': [995],
'children': [{'dimKey': ['Productivas'], 'values': [515]},
{'dimKey': ['Jornada teórica'], 'values': [480]}]}]}]}
```

### 1.2.27 Assign calendar

```
[139]: edit_employee = client.set_employee_calendar(employee=2, calendar="Flexible 8h")
edit_employee[0].get('dataObject').get('Calendar')
```

```
[139]: {'id': 2,
       '_c_': '',
       'created': '0001-01-01T00:00:00.0000000',
       'modified': '0001-01-01T00:00:00.0000000',
       'name': '',
       'rev': 0,
       'years': [{'Year': 2012, 'days': {}},
                  {'Year': 2016, 'days': {}},
                  {'Year': 2017, 'days': {}},
                  {'Year': 2018, 'days': {}},
                  {'Year': 2019, 'days': {}},
                  {'Year': 2020,
                   'days': {'0': {'shifts': [4]},
                             '1': {'shifts': [4]},
                             '33': {'shifts': [3]},
                             '34': {'shifts': [3]},
                             '35': {'shifts': [3]},
                             '36': {'shifts': [3]},
                             '37': {'shifts': [3]},
                             '54': {'shifts': [3]},
                             '55': {'shifts': [3]},
                             '56': {'shifts': [3]},
                             '57': {'shifts': [3]},
                             '58': {'shifts': [3]},
                             '217': {'shifts': [2]}}},
                  {'Year': 2021, 'days': {}}],
       'Cycles': [],
       'Calendars': [{'id': 2, 'name': 'Flexible 8h'}],
       'nodesSource': [],
       'multiName': {'es-ES': ''}}
```

### 1.2.28 Assign department

```
[16]: path = ["SPEC SA", "Argentina", "CABA", "Information Technology", "Help Desk"]
edit_employee = client.set_employee_department(employee=2, node_path=path)
edit_employee[0].get('dataObject').get('Departments')
```

```
[16]: [{'id': 17}]
```

### 1.2.29 Get departments

```
[ ]: # to dev...
```

### 1.2.30 Disconnect

```
[51]: client.disconnect()  
      client.is_connected
```

```
[51]: False
```

```
[7]: client.get_element_def(container="Persona", elements=[4])
```

```
[7]: [{'_c_': 'Persona',  
      'id': -1,  
      'created': '0001-01-01T00:00:00.0000000',  
      'changePassword': False,  
      'firstWONumDays': 0,  
      'firstWO': -1,  
      'firstWANumDays': 0,  
      'firstWA': -1,  
      'firstTTNumDays': 0,  
      'firstTT': -1,  
      'termVoice': False,  
      'timeoutEnrollMinutes': 0,  
      'timeoutEnrollDate': '0001-01-01T00:00:00.0000000',  
      'enrollActive': False,  
      'sex': 0,  
      'Visitas.RecibirVisitas': False,  
      'securityLevel': 0,  
      'ticketEmail': False,  
      'timeoutEnroll': '0001-01-01T00:00:00.0000000',  
      'useTasksWE': False,  
      'useTasksWO': False,  
      'useTasksWA': False,  
      'useTasks': False,  
      'ProfaceAdmin': False,  
      'MobileClocking': False,  
      'RemoteClocking': False,  
      'Portal.DisablePasswordChange': False,  
      'Portal.DisableCalendar': False,  
      'Portal.DisablePlannings': False,  
      'Portal.DisableVistaResumen': False,  
      'Portal.DisableMovimientos': False,  
      'Portal.NoPuedeEditar': False,  
      'Portal.NoRequiereValidacionEnCorreccion': False,  
      'Portal.UsaPortal': False,
```

```

'virtualCard': False,
'geolocalize': 'geoNever',
'offline': False,
'totalDocs': 0,
'Portal.ChangeLanguage': True,
'rev': 0,
'modified': '0001-01-01T00:00:00.0000000',
'birthdate': '0001-01-01T00:00:00.0000000',
'Visitas.ProgramarVisitas': False,
'acceptAllReaders': False,
'acceptAllTT': False,
'NoAttendance': False,
'RegisterSystemDate': '2020-08-27T00:00:00.0000000-03:00',
'htmlPortal': False,
'inactive': True,
'pwdCantChange': False,
'exboss': False,
'pwdNextLogin': False,
'pwdExpires': True,
'pwdRetries': 0,
'lastPwdChange': '0001-01-01T00:00:00.0000000',
'NumFingers': 0,
'PIN': 0,
'FirstDayNotValid': '2004-01-01T00:00:00.0000000',
'FingerIEV02EnrollDate': '0001-01-01T00:00:00.0000000',
'Finger2EnrollDate': '0001-01-01T00:00:00.0000000',
'Finger1EnrollDate': '0001-01-01T00:00:00.0000000',
'LastName': '',
'FingerIEV01EnrollDate': '0001-01-01T00:00:00.0000000',
'ActiveDays': {'validity': [{'start': '2020-08-27T00:00:00.0000000-03:00',
    'end': '2040-12-31T00:00:00.0000000'}]}},
'Readers': [],
'Cards': [],
'Calendar': {'id': -1,
    '_c_': '',
    'created': '0001-01-01T00:00:00.0000000',
    'modified': '0001-01-01T00:00:00.0000000',
    'name': '-1',
    'rev': 0,
    'Cycles': [],
    'Calendars': [],
    'nodesSource': [],
    'multiName': {'es-ES': '-1'}},
'doAccess': {'total': True, 'offsetIn': 0, 'offsetOut': 0},
'TimeTypesEmployee': [],
'Departments': [],
'enrollDevices': [],

```

```

'geolocSource': [{'data': 'geoAlways', 'label': 'Siempre'},
{'data': 'geoIfPossible', 'label': 'Si es posible'},
{'data': 'geoNever', 'label': 'Nunca'}],
'source': [{'id': 1,
  'name': 'Proximity',
  'displayName': 'Proximity',
  'description': '',
  'type': 'terminal',
  'icon': 'Terminal',
  'children': [{'id': 2,
    'displayName': 'Proximity_1',
    'type': 'reader',
    'description': 'Lector 2 del terminal Proximity',
    'icon': 'Lector',
    'children': [{'type': 'device',
      'name': 'CD1',
      'displayName': 'Lector de proximidad interno conectado en CD1',
      'description': 'CD1',
      'default': 1,
      'deviceType': 'Tarjeta',
      'hasElement': True,
      'expr': '',
      'allow': False,
      'id': 'Proximity_Proximity_1_CD1'}]}]},
{'id': 3,
  'name': 'Fingerprint',
  'displayName': 'Fingerprint',
  'description': '',
  'type': 'terminal',
  'icon': 'Terminal',
  'children': [{'id': 1,
    'displayName': 'Fingerprint_1',
    'type': 'reader',
    'description': 'Lector 1 del terminal Fingerprint',
    'icon': 'Lector',
    'children': [{'type': 'device',
      'name': 'COM1',
      'displayName': 'Lector de huella interno conectado en COM1',
      'description': 'COM1',
      'default': 1,
      'deviceType': 'Huella',
      'hasElement': True,
      'expr': '',
      'allow': False,
      'id': 'Fingerprint_Fingerprint_1_COM1'},
{'type': 'device',
  'name': 'BI1',

```

```

        'displayName': 'Teclado interno conectado en BI1',
        'description': 'BI1',
        'default': 1,
        'deviceType': 'Teclado',
        'hasElement': False,
        'expr': '',
        'allow': False,
        'id': 'Fingerprint_Fingerprint_1_BI1'}}]]]],
'initialValuesList': [],
'nodesSource': [{ 'data': '1', 'label': 'SPEC SA'},
{ 'data': '2', 'label': 'SPEC SA · Argentina'},
{ 'data': '15', 'label': 'SPEC SA · Argentina · CABA'},
{ 'data': '16',
  'label': 'SPEC SA · Argentina · CABA · Information Technology'},
{ 'data': '17',
  'label': 'SPEC SA · Argentina · CABA · Information Technology · Help
Desk'}],
'languages': [{ 'data': 'ca-ES', 'label': 'Català'},
{ 'data': 'en-GB', 'label': 'English'},
{ 'data': 'es-ES', 'label': 'Español'},
{ 'data': 'eu-ES', 'label': 'Euskara'},
{ 'data': 'fr-FR', 'label': 'Français'},
{ 'data': 'pt-PT', 'label': 'Português'}]]]

```

```

[11]: query = nt6.Query(
        fields=["nif", "name", "LastName", "nameEmployee", "modified"],
        filterExp='this.modified >= "2020-08-15"'
    )
    client.get_employees(query=query)

```

```

[11]: {'total': 2,
      'items': [{ 'nif': '9876451',
                    'name': 'GGDOE',
                    'LastName': 'Doe',
                    'nameEmployee': 'John',
                    'modified': '2020-08-27T13:43:23.9364016-03:00'},
                  { 'nif': '12345678',
                    'name': 'N08915465',
                    'LastName': 'Spec',
                    'nameEmployee': 'Argentina',
                    'modified': '2020-08-27T14:17:29.5623089-03:00'}]}

```

```
[ ]:
```