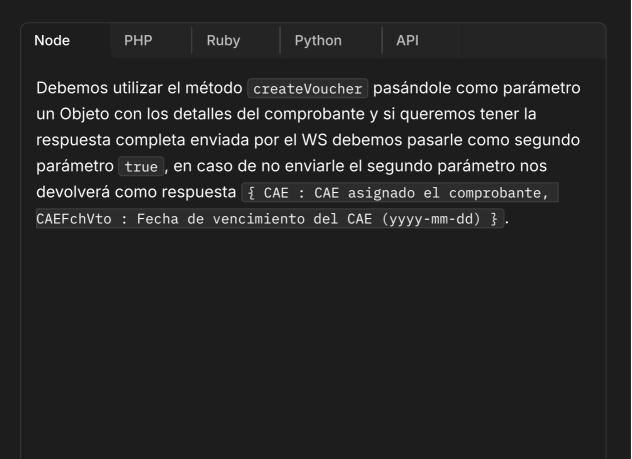
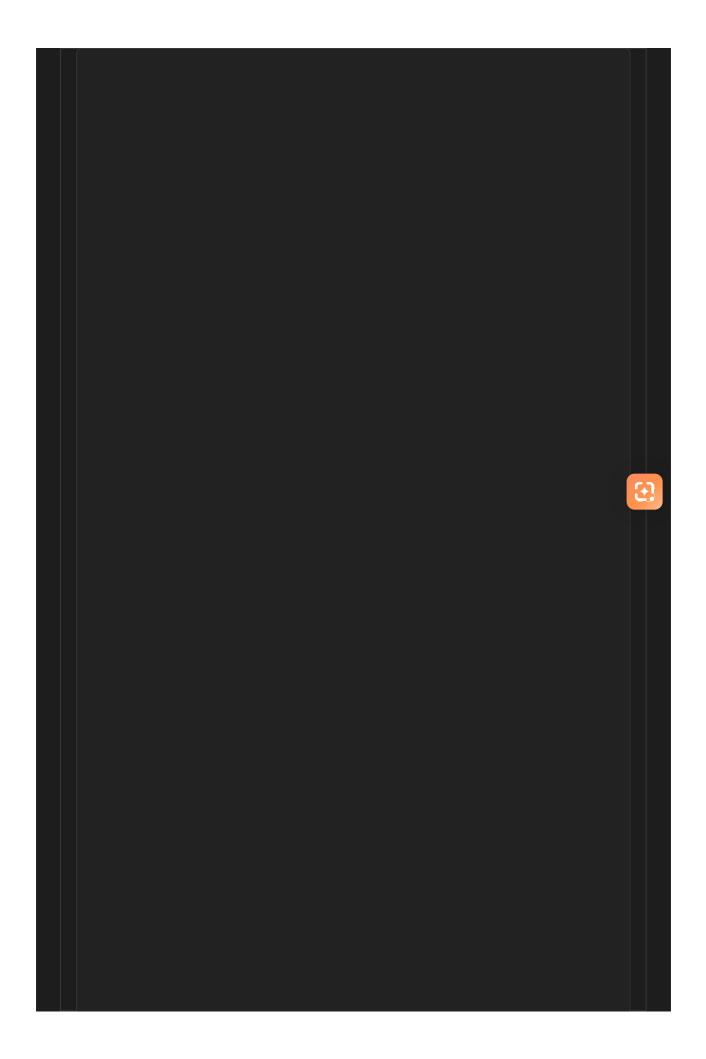


Crear y asignar CAE a un comprobante





```
const returnFullResponse = false;
const date = new Date(Date.now() - ((new
Date()).getTimezoneOffset() * 60000)).toISOString().split('T')
let data = {
    'CantReg' : 1, // Cantidad de comprobantes a registrar
    'PtoVta'
    'CbteTipo' : 6, // Tipo de comprobante (ver tipos
    'Concepto' : 1, // Concepto del Comprobante:
    'DocTipo' : 99, // Tipo de documento del comprador (99
    'DocNro' : 0, // Número de documento del comprador (0
consumidor final)
    'CbteDesde' : 1, // Número de comprobante o numero
    'CbteHasta' : 1, // Número de comprobante o numero
del último comprobante en caso de ser mas de uno
    'CbteFch' : parseInt(date.replace(/-/g, '')), //
    'ImpTotal' : 121, // Importe total del comprobante
    'ImpTotConc' : 0, // Importe neto no gravado
    'ImpNeto' : 100, // Importe neto gravado
    'ImpOpEx' : 0, // Importe exento de IVA
    'ImpIVA' : 21, //Importe total de IVA
    'ImpTrib'
              : 0, //Importe total de tributos
               : 'PES', //Tipo de moneda usada en el
comprobante (ver tipos disponibles)('PES' para pesos
   'MonCotiz' : 1, // Cotización de la moneda usada (1
    'Iva'
              : [ // (Opcional) Alícuotas asociadas al
       {
           'Id'
                      : 5, // Id del tipo de IVA (5 para
21%) (ver tipos disponibles)
           'BaseImp' : 100, // Base imponible
           'Importe' : 21 // Importe
       }
   ],
};
```

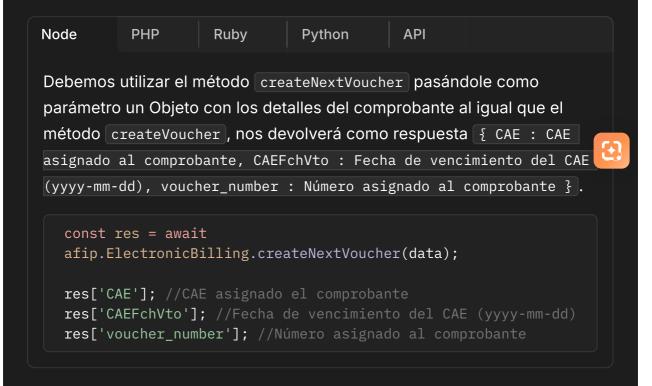
```
const res = await afip.ElectronicBilling.createVoucher(data, returnFullResponse);

res['CAE']; //CAE asignado el comprobante res['CAEFchVto']; //Fecha de vencimiento del CAE (yyyy-mm-dd)

② Este método acepta mas parámetros, podes encontrar todos los parámetros disponibles En este ejemplo 

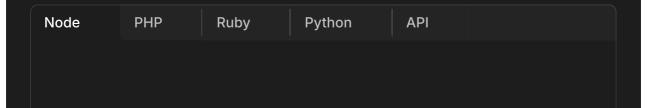
→
```

Crear y asignar CAE a siguiente comprobante



Obtener información de un comprobante ya emitido

Con este método podemos obtener toda la información relacionada a un comprobante o simplemente saber si el comprobante existe.



```
// Numero de comprobante
const numeroDeComprobante = 1;

// Numero de punto de venta
const puntoDeVenta = 1;

// Tipo de comprobante
const tipoDeComprobante = 6; // 6 = Factura B

const voucherInfo = await
afip.ElectronicBilling.getVoucherInfo(numeroDeComprobante,
puntoDeVenta, tipoDeComprobante);

if(voucherInfo === null){
    console.log('El comprobante no existe');
}
else{
    console.log('Esta es la información del comprobante:');
    console.log(voucherInfo);
}
```

Crear CAEA

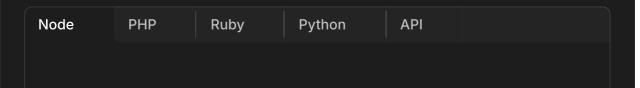
```
Node PHP Ruby Python API

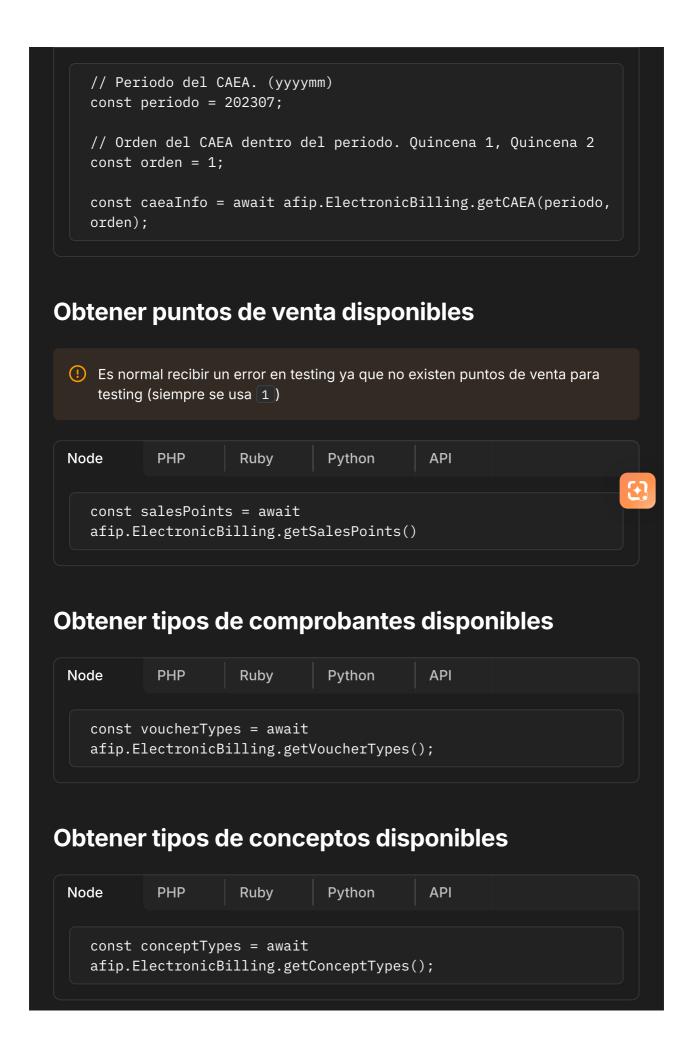
// Periodo del CAEA. (yyyymm)
const periodo = 202307;

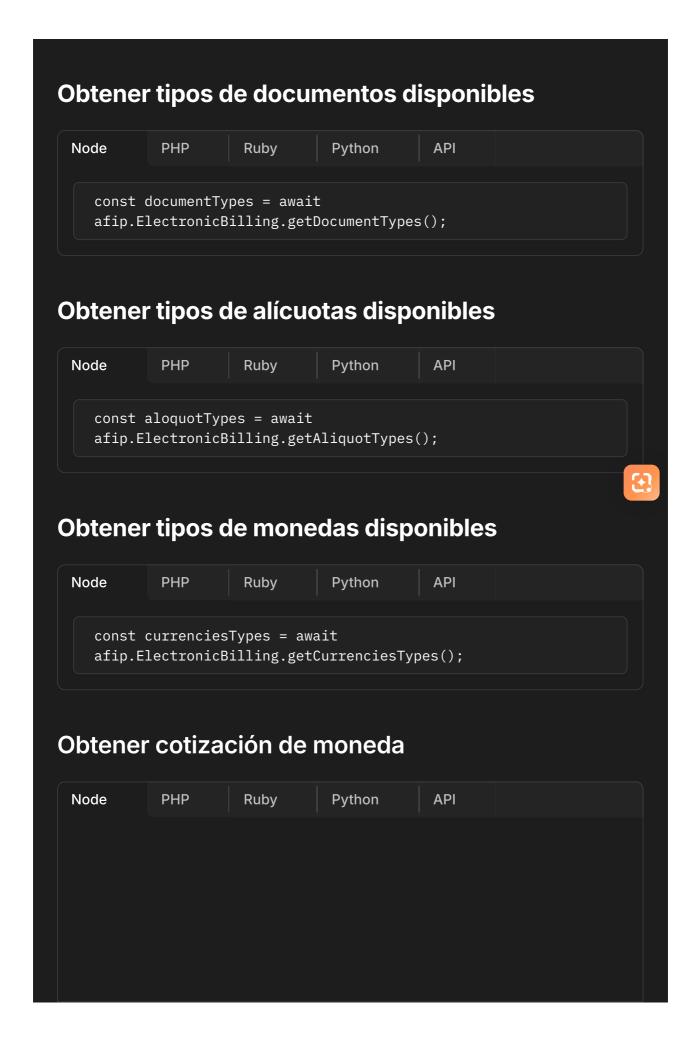
// Orden del CAEA dentro del periodo. Quincena 1, Quincena 2
const orden = 1;

const caeaInfo = await
afip.ElectronicBilling.createCAEA(periodo, orden);
```

Obtener información de CAEA ya emitido





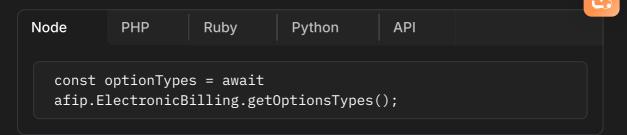


```
// ID de la moneda a consultar (El ID se obtiene del método
para obtener los tipos de monedas)
const currencyID = 'DOL'

// Fecha de la cotización en formato aaaammdd | Por ej.
20250221 = 21 de febrero de 2025
const rateDate = '20250221';

const exchangeRate = await
afip.ElectronicBilling.executeRequest('FEParamGetCotizacion',
{
    MonId: currencyID,
    FchCotiz: rateDate
})
```

Obtener tipos de opciones disponibles para el comprobante



Obtener tipos de tributos disponibles



Obtener tipos de condiciones frente al IVA disponibles

Node PHP Ruby Python API



