

版本V2.0

企业发票库存查询接口： 都传入的时候，各参数优先级：分机号>机器编号>分机号+机器编号的联合查询>部门ID

请求地址

环境	请求地址
正式环境	https://sdk.nuonuo.com/open/v1/services
沙箱环境	https://sandbox.nuonuocs.cn/open/v1/services

注：请 [下载SDK](#) 并完成报文组装后发送接口调用请求，accessToken获取方式请参考 [自用型应用创建](#) 和 [第三方应用创建](#) 。

公共请求参数

消息头

名称	类型	是否必须	描述
Content-type	String	Y	MIME类型【消息头】
X-Nuonuo-Sign	String	Y	签名（编码格式请使用UTF-8）【消息头】
accessToken	String	Y	授权码【消息头】
userTax	String	N	授权商户的税号（自用型应用非必填，第三方应用必填）【消息头】
method	String	Y	请求api对应的方法名称【消息头】

消息体

名称	类型	是否必须	描述
senid	String	Y	唯一标识，由企业自己生成32位随机码【消息体】
nonce	String	Y	8位随机正整数【消息体】
timestamp	String	Y	时间戳(当前时间的秒数)【消息体】
appkey	String	Y	平台分配给应用的appKey【消息体】

私有请求参数

名称	类型	是否必须	示例值	最大长度	描述
departmentId	String	N			部门id（不传分机号、机器编号时使用）；部门和分机、机器编号等都不传时，返回税号下全部报税信息
<div><div></div>extensionNums</div>	Array	N	1	100	分机号列表（只传分机号时使用）
machineCode	String	N	123456789012	12	机器编号单个查询（只传机器编号时使用）
<div><div></div>extMachineCodePairs</div>	Array	N		1	分机号+机器编号联合查询（只能传入一对分机号和机器编号），精确查询某设备时建议使用此种方式

名称	类型	是否必须	示例值	最大长度	描述
extensionNum	String	N	1	5	分机号
machineCode	String	N	123456789012	12	机器编号

响应参数

参数	类型	描述	示例值
code	String	状态码	E0000
describe	String	返回信息	调用成功
<div><div></div>result</div>	Array	具体库存信息	
extensionNumber	Number	分机号	1
machineCode	String	机器编号	
terminalNumber	String	终端号	
invoiceLine	String	发票种类：p:电子增值税普通发票；c:增值税普通发票(纸票)；s:增值税专用发票；e:收购发票(电子)；f:收购发票(纸质)；r:增值税普通发票(卷式)；b:增值税电子专用发票；j:机动车销售统一发票；u:二手车销售统一发票	p
remainNum	Number	剩余数量	200
typeCode	String	发票代码	
invoiceNumStart	String	起始发票号码	
invoiceNumEnd	String	终止发票号码	
updateTime	String	更新时间	2019-08-16 15:39:28

请求示例

JAVA

```
NNOpenSDK sdk = NNOpenSDK.getIntance();
String taxnum = "23***789"; // 授权企业税号
String appKey = "Hn***XL ";
String appSecret = "F65***65F";
String method = "nuonuo.ElectronInvoice.getInvoiceStock"; // API方法名
String token = "2d484e*****pdui"; // 访问令牌
String content = "{
  \"extensionNums\": [
    \"1\"
  ],
  \"machineCode\": \"123456789012\",
  \"departmentId\": \"\",
  \"extMachineCodePairs\": [
    {
      \"machineCode\": \"123456789012\",
      \"extensionNum\": \"1\"
    }
  ]
}
```

```
]
}";
String url = "https://sdk.nuonuo.com/open/v1/services"; // SDK请求地址
String senid = UUID.randomUUID().toString().replace("-", ""); // 唯一标识，32位随机码，无需修改，保持默认即可
String result = sdk.sendPostSyncRequest(url, senid, appKey, appSecret, token, taxnum, method, content);
System.out.println(result);
```

响应示例

JSON格式

```
{
  "code": "E0000",
  "describe": "调用成功",
  "result": [
    {
      "extensionNumber": 0,
      "terminalNumber": "",
      "invoiceLine": "s",
      "machineCode": "661513330644",
      "remainNum": 88,
      "typeCode": "",
      "invoiceNumStart": "20941908",
      "invoiceNumEnd": "20942406",
      "updateTime": "2021-09-09 12:05:49"
    },
    {
      "extensionNumber": 0,
      "terminalNumber": "",
      "invoiceLine": "c",
      "machineCode": "661513330644",
      "remainNum": 11,
      "typeCode": "",
      "invoiceNumStart": "10405090",
      "invoiceNumEnd": "10405109",
      "updateTime": "2021-09-09 12:05:49"
    },
    {
      "extensionNumber": 0,
      "terminalNumber": "",
      "invoiceLine": "p",
      "machineCode": "661513330644",
      "remainNum": 140,
      "typeCode": "",
      "invoiceNumStart": "69989672",
      "invoiceNumEnd": "69990060",
      "updateTime": "2021-09-09 12:05:49"
    }
  ]
}
```

返回码说明

公共异常码

返回码	返回描述	解决方案
E9999	分机号数量超过100，不 持查询	
E0000	调 成功	
E1001	企业资质没有配置，请联系诺诺 ！	