

baiwang.input.invoice.queryinvoicepool

(此文档内容适用于SDK版本为3.4.354及以上版本)

接口简介

根据条件查询进项发票池的发票全票面信息。当请求参数里输入分页序号时，每页返回100条，可查询到全部数据。如未传入分页序号，则最多返回500条，超出则报错，需缩小查询时间范围后再次调用。


一、公共参数

名称	类型	是否必须	描述
method	String	是	接口名称。
appKey	String	是	给开发者颁发的 key。
sign	String	是	请求签名，使用MD5，用于安全控制。
token	String	是	token 用于调用接口前授权检查。
timestamp	String	是	Unix时间戳，例如：1506304524。
format	String	否	响应格式。默认为json格式，可选值：xml，json。
version	String	是	API协议版本，当前版本号6.0。
type	String	否	请求类型，默认 sync，可选值asyn，sync。
requestId	String	否	请求唯一标识，标识唯一请求，默认UUID。

二、请求业务参数

ID	类型	长度	是否必须	描述
taxNo	String	20	是	购方税号
pageNum	Long	10	否	分页序号(传入page_num则按分页查询,每页最大100条否则全量查询)
salesName	String	100	否	销方名称

ID	类型	长度	是否必须	描述
state	String	1	否	发票状态[0-正常, 1-失控, 2-作废, 3-红冲, 4-异常, 7-部分红冲, 8-全额红冲, 80-红冲发票待确认]
invoiceType	String	4	否	01:增值税专用发票 03:机动车销售统一发票 04:增值税普通发票 08:增值税电子专用发票 10:增值税电子普通发票 11:增值税普通发票(卷式) 14:通行费电子发票 15:二手车销售统一发票 31:数电发票（增值税专用发票） 32:数电发票（普通发票） 51:数电发票（铁路电子客票） 61:数电发票（航空运输电子客票行程单） 83:数电发票（机动车销售统一发票） 84:数电发票（二手车销售统一发票） 85:数电纸质发票（增值税专用发票） 86:数电纸质发票（普通发票） 87:数电纸质发票（机动车销售统一发票） 88:数电纸质发票（二手车销售统一发票）
billingDateStart	String	10	否	开票日期开始时间(yyyy-MM-dd)
billingDateEnd	String	10	否	开票日期结束时间(yyyy-MM-dd)
isComplianceStatus	String	1	否	是否合规查验1-已校验 0-未校验
isCheckStatus	String	1	否	是否查验 1-已查验 0-未查验
isCollectStatus	String	1	否	是否采集 1-已采集 0-未采集
pold	String	20	否	业务单号
collector	String	20	否	采集人
invInputUser	String	20	否	认证人, 认证申请提交人
invDeduResult	String	2	否	认证结果 0-未认证 1-认证通过 2-认证不通过
startTime	String	12	是	发票更新开始时间(yyyy-MM-dd HH:mm:ss)
endTime	String	12	是	发票更新结束时间(yyyy-MM-dd HH:mm:ss)
collectStartTime	String	12	否	采集开始时间(yyyy-MM-dd HH:mm:ss 必须与采集结束时间同时传递才生效)
collectEndTime	String	12	否	采集结束时间(yyyy-MM-dd HH:mm:ss 必须与采集开始时间同时传递才生效)
isCompliancedCollectults	String	2	否	发票合规结果 0-合规, 1-不合规, 2-未校验

备注：用  标识层级关系

三、响应公共参数

名称	类型	描述
success	Boolean	接口调用成功或失败的标志 true：成功，false：失败
method	String	接口的唯一标识
requestId	String	请求唯一标识， 与调用接口时传入的request_id一致
response		业务接口调用成功后返回的具体响应信息，不同的接口返回的model不同
errorResponse		错误响应信息
└ code	String	调用失败后，返回的错误码（开放平台错误码）
└ message	String	调用失败后，返回的错误信息（开放平台错误信息）
└ subCode	String	调用失败后，返回的错误码（业务端返回错误码——详见【五、错误码】部分）
└ subMessage	String	调用失败后，返回的错误信息（业务端返回错误信息——详见【五、错误码】部分）

四、响应业务参数

ID	名称	类型	长度	描述
total	发票数量	Integer	20	发票数量
invoiceList	发票列表	List	500	List<InputInvoiceQueryinvoicepoolInvoice>, 发票列表
└ invoiceType	发票类型	String	3	01:增值税专用发票 03:机动车销售统一发票 04:增值税普通发票 08:增值税电子专用发票 10:增值税电子普通发票 11:增值税普通发票(卷式) 14:通行费电子发票 15:二手车销售统一发票 31:数电发票（增值税专用发票） 32:数电发票（普通发票） 51:数电发票（铁路电子客票） 61:数电发票（航空运输电子客票行程单） 83:数电发票（机动车销售统一发票） 84:数电发票（二手车销售统一发票） 85:数电纸质发票（增值税专用发票） 86:数电纸质发票（普通发票） 87:数电纸质发票（机动车销售统一发票） 88:数电纸质发票（二手车销售统一发票）
└ collector	采集人	String	50	采集人
└ collectType	采集方式	Integer	3	采集方式，1：拍照,2：批量,3：手工,4：扫码,5：扫描,6：文件,7：微信,8：支付宝,9：接口,10：邮箱,99：其它
└ taskId	(邮箱归集)导入任务	String	35	(邮箱归集)导入任务批次号

ID	名称	类型	长度	描述
	批次号			
└ administrativeDivisionNo	所属行政区编码	String	32	所属行政区编码
└ administrativeDivisionName	所属行政区名称	String	150	所属行政区名称
└ invoiceCode	发票代码	String	10-12	发票代码
└ invoiceNumber	发票号码	String	20	发票号码
└ billingDate	开票日期	String	14	开票日期: yyyy-MM-dd
└ purchaserName	购方名称	String	100	购方名称
└ purchaserTaxNo	购方税号	String	20	购方税号
└ purchaserBank	购方开户行账户	String	150	购方开户行账户
└ purchaserAddressPhone	购方地址电话	String	150	购方地址电话
└ salesName	销方名称	String	100	销方名称
└ salesTaxNo	销方税号	String	20	销方税号
└ salesAddressPhone	销方地址电话	String	150	销方地址电话
└ salesBank	销方开户行账户	String	150	销方开户行账户

ID	名称	类型	长度	描述
└ totalAmount	合计金额	BigDecimal	30	合计金额（元）[两位小数]
└ totalTax	合计税额	BigDecimal	30	合计税额（元）[两位小数]
└ amountTax	价税合计	String	30	价税合计（元）[两位小数]
└ amountTaxCn	价税合计_中文	String	64	价税合计_中文
└ checkCode	校验码	String	64	校验码
└ state	发票状态	String	2	发票状态：0-正常，1-失控，2-作废，3-红冲，4-异常，7-部分红冲，8-全额红冲，80-红冲发票待确认
└ carrierName	承运人名称	String	150	承运人名称
└ carrierTaxNo	承运人识别号	String	32	承运人识别号
└ draweeName	受票方名称	String	100	受票方名称
└ draweeTaxNo	受票方识别号	String	20	受票方识别号
└ cargoInformation	运输货物信息	String	150	运输货物信息
└ transportRoute	起运地、经由、到达地	String	100	起运地、经由、到达地
└ machineCode	税控盘号	String	32	税控盘号
└ tonnage	车船吨位	String	10	车船吨位

ID	名称	类型	长度	描述
└ taxAuthorityNo	主管 税务 机关	String	32	主管税务机关
└ taxAuthorityName	主管 税务 名称	String	100	主管税务名称
└ remarks	备注	String	184	备注
└ iDCardNo	身 份 证 号 码 / 组 织 机 构 代 码	String	32	身份证号码/组织机构代码
└ originPlace	产地	String	100	产地
└ certificateNo	合 格 证 号	String	32	合格证号【铁路(航空)电子客票证件号】
└ inspectionListNo	商 检 单 号	String	64	商检单号
└ engineNo	发 动 机 号	String	32	发动机号
└ vehicleNo	车 辆 识 别 代 号 / 车 架 号 码	String	32	车辆识别代号/车架号码
└ importCertificateNo	进 口 证 明 书 号	String	64	进口证明书号
└ salesPhone	销 方 地 址 电 话	String	150	销方地址电话
└ salesBankNo	销 方 银 行 账 号	String	64	销方银行账号
└ salesAddress	销 方 地 址	String	150	销方地址

ID	名称	类型	长度	描述
└ paymentVoucherNo	完税凭证号码	String	32	完税凭证号码
└ passengersLimited	限乘人数	String	10	限乘人数
└ tollSign	通行费标志	String	2	通行费标志：06-可抵扣通行费，07-不可抵扣通行费
└ brandModel	厂牌型号	String	100	厂牌型号
└ totalPrice	合计税额	Double	30	合计税额（元）[两位小数]
└ carNumber	车牌照号	String	50	车牌照号
└ vehicleType	车辆类型	String	50	车辆类型
└ auctionTaxNo	经营，拍卖单位纳税人识别号	String	30	经营，拍卖单位纳税人识别号
└ auctionName	经营，拍卖单位	String	100	经营，拍卖单位
└ auctionPhone	经营，拍卖单位电话	String	15	经营，拍卖单位电话
└ auctionAddress	经营，拍卖单位地址	String	200	经营，拍卖单位地址
└ auctionBank	经营，开户银行及	String	200	经营，开户银行及账号

ID	名称	类型	长度	描述
	账号			
└ usedCarTaxNo	二手车市场纳税人识别号	String	30	二手车市场纳税人识别号
└ usedCarName	二手车市场	String	100	二手车市场
└ usedCarPhone	二手车市场电话	String	15	二手车市场电话
└ usedCarAddress	二手车市场地址	String	200	二手车市场地址
└ usedCarBank	二手车市场开户银行及账号	String	200	二手车市场开户银行及账号
└ registrationNumber	登记证号	String	100	登记证号
└ vehiclePlaceName	转入地车辆车管所名称	String	36	转入地车辆车管所名称
└ specialTicketMark	特殊票种标志	String	3	特殊票种标志 01-成品油增值税专用发票 02-稀土增值税专用发票 03-机动车增值税专用发票(增值税专票及增值税电专票涉及特殊票种)
└ versionInfo	版本信息	String	2	版本信息 0-旧版机动车发票 1-新版机动车发票
└ officialfileUrl	电票源文件下载地址	String	200	电票源文件下载地址

ID	名称	类型	长度	描述
└ eInvoiceNumber	全电 纸发 票号 码	String	30	全电纸发票号码
└ businessType	业务 类型	String	4	铁路电子客票专属
└ airConditionFeature	空调 特征	String	20	铁路电子客票专属
└ ticketType	票种	String	20	铁路电子客票专属
└ seatNumber	席位	String	20	铁路电子客票专属
└ carriageNumber	车厢	String	20	铁路电子客票专属
└ seatLevel	席别	String	20	铁路电子客票专属
└ travelDate	出发 时间	String	20	铁路电子客票专属
└ carriagDate	乘车 日期	String	20	铁路电子客票专属
└ trainNumber	车次	String	20	铁路电子客票专属
└ toStation	目的 地	String	100	铁路电子客票专属
└ fromStation	出发 地	String	100	铁路电子客票专属
└ name	姓名	String	100	铁路(航空)电子客票专属
└ eTicketNumber	电子 客票 号	String	100	铁路(航空)电子客票专属
└ domesticOrInternational	国内 国际 标识	String	20	航空电子行程单专属
└ gpnumber	GP 单号	String	64	航空电子行程单专属
└ detailList	发票 详细 信息	List	~	List<InputInvoiceQueryinvoicepoolDetail>, 发票详细信息列表

ID	名称	类型	长度	描述
	列表			
└ └ rowNo	行号： 从1开始	String	10	行号：从1开始
└ └ commodityCode	商品编码	String	40	商品编码
└ └ commodityName	货物或应 税劳务名称	String	150	货物或应税劳务名称
└ └ specificationModel	规格型号	String	32	规格型号
└ └ unit	单位	String	32	单位
└ └ quantity	数量	String	20	数量
└ └ unitPrice	单价	String	30	单价（元）
└ └ amount	金额	String	30	金额（元）
└ └ taxRate	税率	String	10	税率（小数形式）
└ └ tax	税额	String	30	税额（元）
└ └ currentDateEnd	通行日期起	String	20	通行日期起(yyyyMMdd)
└ └ currentDateStart	通行日期止	String	20	通行日期止(yyyyMMdd)
└ └ licensePlateNum	车牌号	String	20	车牌号
└ └ type	类型	String	10	类型
└ └ flightSegment	航段	String	2	航空电子行程单专属明细
└ └ fromCity	始发站	String	64	航空电子行程单专属明细

ID	名称	类型	长度	描述
└└ toCity	目的站	String	64	航空电子行程单专属明细
└└ carrier	承运人	String	64	航空电子行程单专属明细
└└ flightNumber	航班号	String	64	航空电子行程单专属明细
└└ scatLevel	座位等级	String	10	航空电子行程单专属明细
└└ carriageDate	承运日期	String	32	航空电子行程单专属明细
└└ flightDate	起飞时间	String	32	航空电子行程单专属明细
└└ ticketLevel	客票级别	String	8	航空电子行程单专属明细
└ checkErrMsg	不合规原因	List	90	List<Object>, 不合规原因
└ isCompliancedCollectults	发票合规结果	String	1	发票合规结果 0-合规未校验, 1-合规结果已通过, 2-合规结果不通过
└ invDeduResult	认证结果	String	2	认证结果: 0-未认证 1-认证通过 2-认证不通过
└ isComplianceCollect	是否完成合规查验	String	2	是否完成合规查验 1-已校验 0-未校验
└ checkErrCode	异常原因	String	20	异常原因
└ Id	Id	String	1	Id

五、错误码

名称	错误描述
-1	系统异常，请稍后再试

名称	错误描述
3005	不能通过凭证号查询
005	该请求不是内网地址!
4014	每次查询发票数量不超过500条
704	请求入参为空
409	所查发票不存在

六、请求示例

json格式示例

POST

```
http://ip:port/router/rest?method=baiwang.input.invoice.queryinvoicepool&version=6.0&appKey=1234567&form=2674-45d9-81a8-be7c6f865009&type=sync&requestId=6f0a06db-e419-4836-8a83-25dff6a33743&sign=16FEA5FE4E50222FF78BD7397878932F
```

```
{
  "invDeduResult": "1",
  "billingDateEnd": "2020-02-03",
  "invInputUser": "admin",
  "pageNum": 1,
  "isCheckedStatus": "1",
  "poId": "Hz1234112",
  "collector": "admin",
  "isCollectStatus": "1",
  "salesName": "销方名称",
  "billingDateStart": "2020-02-02",
  "collectEndTime": "2020-01-01 11:12:30",
  "invoiceType": "01",
  "isComplianceStatus": "1",
  "startTime": "2020-01-01 12:12:20",
  "state": "0",
  "endTime": "2020-01-01 12:12:30",
  "taxNo": "91110108094560534R",
  "collectStartTime": "2020-01-01 11:12:20",
  "isCompliancedCollectults": "0"
}
```

sdk调用示例

```
String url = "请求地址";
String appKey = "appKey";
String appSecret = "appSecret";
String username = "用户名";
String password = "用户密码";
String userSalt = "用户盐值";

try {
    String requestId = UUID.randomUUID().toString(); //自定义请求唯一标识
    PasswordLoginConfig loginConfig = new PasswordLoginConfig();
    loginConfig.setUrl(url);
```

```

loginConfig.setClientId(appKey);
loginConfig.setClientSecret(appSecret);
loginConfig.setUsername(username);
loginConfig.setPassword(password);
loginConfig.setUserSalt(userSalt);
BopLoginClient loginClient = new PasswordLoginClient(loginConfig);

// 获取token, token相关说明请参考 文档中心 -> 用户授权说明
BopLoginResponse loginResponse = loginClient.login();
// 重载方法, 可以传自定义请求唯一标识
// BopLoginResponse loginResponse = loginClient.login(requestId);
String token = loginResponse.getResponse().getAccessToken();

InputInvoiceQueryinvoicepoolRequest request = new InputInvoiceQueryinvoicepoolRequest();
request.setInvDeduResult("1");
request.setBillingDateEnd("2020-02-03");
request.setInvInputUser("admin");
request.setPageNum(1L);
request.setIsCheckStatus("1");
request.setPoId("Hz1234112");
request.setCollector("admin");
request.setIsCollectStatus("1");
request.setSalesName("\u9500\u65B9\u540D\u79F0");
request.setBillingDateStart("2020-02-02");
request.setCollectEndTime("2020-01-01 11:12:30");
request.setInvoiceType("01");
request.setIsComplianceStatus("1");
request.setStartTime("2020-01-01 12:12:20");
request.setState("0");
request.setEndTime("2020-01-01 12:12:30");
request.setTaxNo("91110108094560534R");
request.setCollectStartTime("2020-01-01 11:12:20");
request.setIsCompliancedCollectults("0");

IBWClient client = new BWRestClient(url, appKey, appSecret); // 初始化一个客户端
InputInvoiceQueryinvoicepoolResponse response = client.inputInvoice().queryinvoicepool(request, token);
// 重载方法, 可以传自定义请求唯一标识
// InputInvoiceQueryinvoicepoolResponse response =
client.inputInvoice().queryinvoicepool(request, token, requestId);
System.out.println(response);
} catch (BWOpenException e) {
    System.out.println(e.getCode());
    System.out.println(e.getMessage());
    System.out.println(e.getSubCode());
    System.out.println(e.getSubMessage());
}
}

```

七、响应示例

成功返回示例

```

{
  "method": "baiwang.input.invoice.queryinvoicepool",
  "success": true,
  "requestId": "aad36643-665d-4848-838a-95822dc41b82",
  "response": {
    "invoiceList": [

```

```
{
  "invDeduResult": "0",
  "draweeName": "测试",
  "brandModel": "大众",
  "transportRoute": "测试",
  "isComplianceCollect": "",
  "totalTax": 2,
  "administrativeDivisionNo": "0020",
  "carrierName": "测试",
  "salesBankNo": "6855636434534534",
  "usedCarBank": "测试",
  "salesBank": "888777666555111",
  "usedCarTaxNo": "888777666555111",
  "checkErrMsg": [],
  "invoiceType": "01",
  "purchaserAddressPhone": "176768776878",
  "state": "0",
  "id": "",
  "cargoInformation": "测试",
  "vehicleType": "小车",
  "officialFileUrl": "",
  "engineNo": "34254325",
  "auctionBank": "888777666555111",
  "machineCode": "445435243523",
  "idCardNo": "410502978868089676978",
  "usedCarName": "测试市场",
  "versionInfo": "1",
  "auctionName": "测试",
  "invoiceCode": "1000012349",
  "certificateNo": "132412334",
  "auctionAddress": "测试",
  "purchaserName": "测试公司",
  "fromStation": "",
  "checkCode": "112233",
  "salesAddress": "测试地址",
  "carNumber": "京A-12345",
  "billingDate": "2020-02-02",
  "totalAmount": 20,
  "vehicleNo": "6745143",
  "salesAddressPhone": "176768776878",
  "salesName": "测试公司",
  "purchaserTaxNo": "888777666555111",
  "name": "",
  "detailList": [
    {
      "unitPrice": "20",
      "rowNo": "1",
      "amount": "40",
      "specificationModel": "测试规格",
      "quantity": "2",
      "ticketLevel": "",
      "flightDate": "",
      "toCity": "",
      "tax": "5",
      "type": "小车",
      "flightNumber": "",
      "taxRate": "0.07",
      "flightSegment": "",
      "carriageDate": "",
      "unit": "件",
      "licensePlateNum": "京A-12345",
      "carrier": ""
    }
  ]
}
```

```

        "currentDateEnd": "20200202",
        "commodityCode": "3421312",
        "currentDateStart": "20200202",
        "fromCity": "",
        "seatLevel": "",
        "commodityName": "测试商品"
    }
},
"auctionTaxNo": "888777666555111",
    "usedCarPhone": "15366666666",
    "carrierTaxNo": "112313213",
    "trainNumber": "",
    "taxAuthorityName": "测试",
"inspectionListNo": "4634254532",
    "amountTax": "22",
    "originPlace": "北京",
"totalPrice": 2,
    "specialTicketMark": "01",
    "eTicketNumber": "",
    "collector": "admin",
    "seatNumber": "",
    "paymentVoucherNo": "65454656",
    "amountTaxCn": "贰拾贰",
"auctionPhone": "15366666666",
    "salesPhone": "15222222222",
    "domesticOrInternational": "",
    "invoiceNumber": "87692434",
    "tollSign": "06",
    "gpnumber": "",
    "tonnage": "13213",
    "taxAuthorityNo": "7544643543",
    "eInvoiceNumber": "",
    "seatLevel": "",
    "isCompliancedCollectults": "0",
    "purchaserBank": "888777666555111",
    "administrativeDivisionName": "北京",
"passengersLimited": "2",
    "draweeTaxNo": "112313213",
    "vehiclePlaceName": "测试",
"ticketType": "",
    "carriageNumber": "",
    "salesTaxNo": "888777666555111",
    "importCertificateNo": "436546765476",
    "checkErrCode": "",
    "toStation": "",
    "collectType": 0,
    "travelDate": "",
    "registrationNumber": "76475362345425",
    "carriagDate": "",
    "usedCarAddress": "测试地址",
"businessType": "",
    "taskId": "",
    "remarks": "备注",
"airConditionFeature": ""
    }
},
    "total": 1
}
}

```

失败返回示例

```
{
  "success": false,
  "method": "baiwang.input.invoice.queryinvoicepool",
  "requestId": "6f0a06db-e419-4836-8a83-25dff6a33743",
  "errorResponse": {
    "code": 100002,
    "message": "token不正确",
  "subCode": null,
    "subMessage": null
  }
}
```