



S

## Express inquiry

### ❗ INFO

Support querying express delivery progress through express 100 API

placeholder, unfinished

```
const MD5 = function (e) {
  function h(a, b) {
    let c, d, e, f, g;
    e = a & 2147483648;
    f = b & 2147483648;
    c = a & 1073741824;
    d = b & 1073741824;
    g = (a & 1073741823) + (b & 1073741823);
    return c & d
      ? g ^ 2147483648 ^ e ^ f
      : c | d
      ? g & 1073741824
      ? g ^ 3221225472 ^ e ^ f
      : g ^ 1073741824 ^ e ^ f
      : g ^ e ^ f;
  }

  function k(a, b, c, d, e, f, g) {
    a = h(a, h(h((b & c) | (~b & d), e), g));
    return h((a << f) | (a >>> (32 - f)), b);
  }

  function l(a, b, c, d, e, f, g) {
    a = h(a, h(h((b & d) | (c & ~d), e), g));
    return h((a << f) | (a >>> (32 - f)), b);
  }

  function m(a, b, d, c, e, f, g) {
    a = h(a, h(h(b ^ d ^ c, e), g));
    return h((a << f) | (a >>> (32 - f)), b);
  }

  function n(a, b, d, c, e, f, g) {
    a = h(a, h(h(d ^ (b | ~c), e), g));
    return h((a << f) | (a >>> (32 - f)), b);
  }

  function p(a) {
    let b = "";
    let d = "";
    let c;
    for (c = 0; c <= 3; c++) {
      (d = (a >>> (8 * c)) & 255),
      (d = "0" + d.toString(16)),
      (b += d.substr(d.length - 2, 2));
    }
    return b;
  }

  let f = [];
  let q;
  let r;
  let s;
  let t;
  let a;
  let b;
  let c;
  let d;
  e = (function (a) {
    a = a.replace(/\r\n/g, "\n");
    for (var b = "", d = 0; d < a.length; d++) {
      const c = a.charCodeAt(d);
      c < 128
        ? (b += String.fromCharCode(c))
        : (c > 127 && c < 2048
            ? (b += String.fromCharCode((c >> 6) | 192))
            : ((b += String.fromCharCode((c >> 12) | 224)),
              (b += String.fromCharCode((c >> 6) & 63) | 128))),
        (b += String.fromCharCode((c & 63) | 128));
    }
  })(a);
}
```



```

        (b = n(b, c, d, a, f[e + 1], 21, 2240044497)),
        (a = n(a, b, c, d, f[e + 8], 6, 1873313359)),
        (d = n(d, a, b, c, f[e + 15], 10, 4264355552)),
        (c = n(c, d, a, b, f[e + 6], 15, 2734768916)),
        (b = n(b, c, d, a, f[e + 13], 21, 1309151649)),
        (a = n(a, b, c, d, f[e + 4], 6, 4149444226)),
        (d = n(d, a, b, c, f[e + 11], 10, 3174756917)),
        (c = n(c, d, a, b, f[e + 2], 15, 718787259)),
        (b = n(b, c, d, a, f[e + 9], 21, 3951481745)),
        (a = h(a, q)),
        (b = h(b, r)),
        (c = h(c, s)),
        (d = h(d, t));
    }
    return (p(a) + p(b) + p(c) + p(d)).toLowerCase();
};

const intro = `
### 使用前须知
需要调用快递 100 API 对快递进行查询, 因此需要你输入快递 100 企业后台 (https://api.kuaidi100.com) 中的授权参数才能使用
`;

output.markdown(intro);

// const key = await input.textAsync("请输入授权 key: ");
// const customer = await input.textAsync("请输入 customer: ");

const key = "RBMHPFRp8146";
const customer = "E24C3FCC31114560DE29E599E580DBB4";

const datasheet = await space.getActiveDatasheetAsync();
const numField = await input.fieldAsync("请选择快递单号所在列: ", datasheet);
const numFieldId = numField.id;

const comField = await input.fieldAsync(
    "请选择快递公司编码所在列 (示例数据 https://api.kuaidi100.com/manager/openapi/download/kdbm.do) : ",
    datasheet
);
const comFieldId = comField.id;

const record = await input.recordAsync("请选择要查询的记录: ", datasheet);
const recordId = record.id;

const num = record.getCellValueString(numFieldId);
const com = record.getCellValueString(comFieldId);

if (num !== null && com !== null) {
    console.log(record);
    const param = '{"com":"' + com + '","num":"' + num + '"}';
    const str = param + key + customer;
    console.log(str);
    const sign = MD5(str).toString().toUpperCase();
    console.log(sign);

    const data = {
        customer,
        param,
        sign,
    };
};

let response = await fetch("https://poll.kuaidi100.com/poll/query.do", {
    method: "POST",
    headers: {
        "Content-Type": "application/x-www-form-urlencoded",
    },
    body: data,
});

const responseData = response.json();
console.log(responseData);
const responseMessage = responseData["message"];

switch (responseMessage) {
    case "ok":
        const statusList = {
            0: "快件在途中",
            1: "快件揽件",
            2: "快件存在疑难",
            3: "快件已签收",
            4: "快件退签",
            5: "快件正在派件",
            6: "快件正处于返回发货人的途中",
            7: "快件转给其他快递公司邮寄",
            8: "快件清关",
            14: "收件人拒签快件",
        };
        const statusText = statusList[responseData["status"]];

```

```
const str = `
### 快递当前状态
${statusText}
`;
output.markdown(str);

const expressData = responseData["data"];

const result = expressData.map((a) => {
  return { 时间: a.time, 详情: a.context };
});
output.table(result);
break;
default:
  output.clear();
  output.text(responseMessage);
  break;
}
} else {
  output.clear();
  output.text("缺少快递单号或快递公司编码");
}
```

Developer Center

[Development guide](#)

[API Reference](#)

Social

[Twitter](#) 

More

[Homepage](#) 

[Help Center](#) 

[GitHub](#) 

© 2023 AITable Ltd. All rights reserved.