

```

const moment = require('moment');
const fs = require('fs');

var aApiSettings = {
  //Echo Gerät für Sprachausgabe
  'echodevice': 'alexa2.0.Echo-Devices.G090XG12211700GD',
  'url': getState("0_userdata.0.cubinode.apiurl").val,
  'action': {
    'print': 'printpaper',
    'devicestatus': 'getdeviceinfo',
    'printstatus': 'getprintstat',
    'requestbind': 'requestbind',
    'requestbindstatus': 'getbindrequeststatus'
  },
  'appid': '?appID=' +
getState("0_userdata.0.cubinode.appid").val,
  'accesskey': '&ak=' +
getState("0_userdata.0.cubinode.accesskey").val,
  'deviceid': '&deviceID=' +
getState("0_userdata.0.cubinode.deviceid").val,
  'bindid': '&bindID=' +
getState("0_userdata.0.cubinode.bindid").val,
  'userid': '&useridentifying' +
getState("0_userdata.0.cubinode.userid").val,
  'liststates': {
    '0_userdata.0.cubinode.actionStates.print_shoppinglist_state':
    'alexa2.0.Lists.SHOPPING_LIST.json',
    '0_userdata.0.cubinode.actionStates.print_todolist_state':
    'alexa2.0.Lists.TO_DO.json'
  }
}

/**Print States */
for (const [state, list] of
Object.entries(aApiSettings.liststates)) {
  on({ id: state, val: true }, function (obj) {
    var value = obj.state.val;
    var oldValue = obj.oldState.val;

    moment.locale('de'); // de
    var Timestamp = moment().format('YYYY-MM-DD
hh:mm:ss');
    var sTimestamp = '&timestamp=' + Timestamp;
    var apiUrl = aApiSettings.url +
aApiSettings.action.print + aApiSettings.appid +
aApiSettings.accesskey + sTimestamp + aApiSettings.deviceid +

```

```

aApiSettings.bindid;
    apiUrl += '&printcontent=T: ';

    var aLists = JSON.parse(getState(list).val);
    var printList = '';
    var TimeStampList = moment().format('DD.MM.YYYY
HH:mm');
    if(list == 'alexa2.0.Lists.SHOPPING_LIST.json'){
        printList = 'Einkaufsliste ' + TimeStampList + '\n\r';
        printList +=
'=====\n\n\r';
    }
    if(list == 'alexa2.0.Lists.TO_DO.json'){
        printList = 'TODO Liste ' + TimeStampList + '\n\r';
        printList +=
'=====\n\n\r';
    }
    printList += aLists.map(function (val) {
        var item = val.value
        item = item[0].toUpperCase() + item.substring(1);
        return (val.completed ? '[X] ' : '[ ] ') + item;
    }).join('\n\n\r');

    //es sind keine Umlaute zulässig somit müssen diese
ersetzt werden
    printList = printList.replace(/Ä/g, 'Ae');
    printList = printList.replace(/ä/g, 'ae');
    printList = printList.replace(/Ö/g, 'Oe');
    printList = printList.replace(/ö/g, 'oe');
    printList = printList.replace(/Ü/g, 'Ue');
    printList = printList.replace(/ü/g, 'ue');
    printList = printList.replace(/ß/g, 'ss');

    var base64 = new Buffer(printList).toString('base64');

    apiUrl += base64;

    //console.log(apiUrl);
    const request = require('request');
    request({ 'uri': apiUrl }, function (error, response,
json) {
        if (!error && response.statusCode === 200) {

            var aReturn = JSON.parse(json);
            //
{"showapi_res_code":1,"showapi_res_error":"ok","result":2,"pri
ntcontentid":52476}
            if (aReturn.result == 2) {
                var statusUrl = aApiSettings.url +

```

```
aApiSettings.action.printstatus + aApiSettings.appid +
aApiSettings.accesskey + sTimestamp;
        statusUrl += '&printcontentid=' +
aReturn.printcontentid;
        request({ 'uri': statusUrl }, function
(err, resp, jsonStatus) {
        //
{"showapi_res_code":1,"showapi_res_error":"ok","printflag":1,"
printcontentid":52476}
        var outputDevice =
aApiSettings.echodevice;
        setState(outputDevice +
".Commands.speak", '50;bitteschön!');
    });
}
}
});
        setStateDelayed(state, false, 2000, false);
    });
```