Table of Contents

[INTRODUCTION 2](#_Toc200013399)

[ENDPOINTS 3](#_Toc200013400)

[API SPECIFICATION 4](#_Toc200013401)

[SERVER TO SERVER (S2S) 4](#_Toc200013402)

[Request: 4](#_Toc200013403)

[paymentMethod 4](#_Toc200013404)

[paymentFilter 4](#_Toc200013405)

[**Response** 6](#_Toc200013406)

[**Data will be included when calling the callbackUrl** 7](#_Toc200013407)

[TRANSACTION QUERY 8](#_Toc200013408)

[Request 8](#_Toc200013409)

[Response 8](#_Toc200013410)

[NOTIFICATION 9](#_Toc200013411)

[Payload 9](#_Toc200013412)

[Response 10](#_Toc200013413)

[Test Cards with merchant ID start with 11xxxxxxxx 11](#_Toc200013414)

[Test Cards with merchant ID start with 14xxxxxxxx 11](#_Toc200013415)

[APPENDIX 11](#_Toc200013416)

[Payment Methods 11](#_Toc200013417)

[Status Codes 12](#_Toc200013418)

[Country Codes 13](#_Toc200013419)

[Card Type 20](#_Toc200013420)

[Code example 21](#_Toc200013421)

[PHP 21](#_Toc200013422)

[NodeJS 22](#_Toc200013423)

# INTRODUCTION

This document involves description of:

* GLODIPAY Server to Server (S2S): supports merchants to accept credit/debit cards

# ENDPOINTS

|  |  |
| --- | --- |
| **Environment** | **Base Endpoint** |
| **Test** | **https://payment-sandbox.gpayprocessing.com** |
| **Production** | **Get it from the** [**API Keys**](https://portal.glodipay.com/api-config) **page of the GLODIPAY Portal** |

# API SPECIFICATION

## SERVER TO SERVER (S2S)

### Request:

* Endpoint: /v1/card/api
* Method: POST
* Content Type: application/json

|  |  |  |  |
| --- | --- | --- | --- |
| **Parameter** | **Field Type** | **Required** | **Description** |
| merchantId | String L (1,50) | M | Merchant’s id |
| orderRef | String L (1,250) | M | Merchant **unique Identify** transaction |
| amount | Float | M | Invoice amount |
| currency | String L (3) | M | Invoice [currency](#page26), ISO 4217 standard specifies three-letter ("Alpha-3") codes. e.g. USD |
| cancelUrl | String L (1,300) | M | (Frontend) The URL the customer will be directed to if they decide to cancel payment and return to your website.  Please make sure that the URL is under https! |
| callbackUrl | String L (1,300) | M | (Frontend) URL where the customer is directed to after a successful payment. Please make sure that the URL is under https! |
| notificationUrl | String L (1,300) | M | Endpoint on your server for GLODIPAY to send webhooks.  Please make sure that the URL is under https! |
| errorUrl | String L (1,300) | M | (Frontend) URL where the customer is redirected in case of an error in the payment.  Please make sure that the URL is under https! |
| orderDescription | String | M | A short description of the underlying transaction or trade to be displayed on the redirection screen |
| metadata | JSON | O | Set of key-value pairs that can be attached to the checkout session |
| transactionDocuments | JSON | O | Supporting documents for the transaction |
| paymentMethod | String | M | Specify the [payment methods](#_Payment_Methods) you want displayed for the session |
| paymentFilter | JSON | O | Specify the payment methods you want removed for the session |
| feeBySeller | Number min(0), max(100) | O | An integer between 0 and 100. Percentage of processing fee to be paid by the merchant. GLODIPAY will charge the merchant an amount. |
| billingFirstName | String | M | Billing Details - Address, Recipient Name |
| billingLastName | String | M |
| billingStreet1 | String | M |
| billingStreet2 | String | M |
| billingCity | String | M |
| billingEmail | String | M |
| billingState | String | M |
| billingCountry | String | M | Address [country](#_Country_Codes) (Country (ISO 3166-1 alpha\_2 country code) |
| billingPostalCode | String | M | Postal Code |
| billingPhoneCountryCode | String | M | Calling country code (for example, '1' for United States, '91' for India) |
| billingPhoneNumber | String | M | Phone Number |
| brandName | String L (1,255) | O | Override the Brand Name to be displayed on the GLODIPAY hosted checkout screen |
| colorMode | String L (1,255) | O | You can pass up to 3 color codes, separated by ---.  Examples:  Color names:  darkgreen---lightgreen---mediumseagreen  HEX codes:  #2e7d32---#e8f5e9---#81c784  RGBA values:  rgba(46,125,50,1)---rgba(232,245,233,1)--- rgba(129,199,132,1) |
| logoSource | String L (1,255) | O | Override the Logo to be displayed on the GLODIPAY hosted checkout screen |
| customerIp | String | M | IP address of the customer |
| signature | String | M | [Signature](#_Code_example) is calculated in the following steps:   1. Concatenate all parameters values except the signature (arranged in ascending order of parameter’s name) to a string 2. RSA sign the hash by merchant’s private key |

M = Mandatory

O = Optional

C = Condition

**Response**

* Method: Form POST
* Content type: application/json

|  |  |  |  |
| --- | --- | --- | --- |
| **Parameter** | **Field Type** | **Required** | **Description** |
| status | String | M | **success, redirect, error** |
| message | String | M |  |
| errors | JSON | O | Response with validation failed {  "status": "error",  "message": "Invalid request data.",  "errors": [  {  "field": "billingEmail",  "message": [  "The billing email field is required."  ]  },  {  "field": "customerIp",  "message": [  "The customer ip field is required."  ]  }  ]  } |
| data | JSON | O | Response with Authenticate {  "status": "redirect",  "message": "Please redirect the user to complete the payment.",  "data": {  "transactionId": "01jwz0ty1640apxvmyzqpvc18a",  "url": "https://payment-sandbox.gpayprocessing.com/card/3ds/01jwz0ty1640apxvmyzqpvc18a"  }  }  Response bypasses 3DS authentication  {  "status": "success",  "message": "The transaction has been successfully completed.",  "data": {  "transactionId": "01jwz13qfcx4z61ded3jcj0tf2"  }  } |

**Data will be included when calling the callbackUrl**

* Method: GET

GLODIPAY will sending payment result to merchant server’s endpoint as specified in **callbackUrl**.

|  |  |  |  |
| --- | --- | --- | --- |
| **Parameter** | **Field Type** | **Required** | **Description** |
| [payload](#_Payload_Base_64) | Base 64 String | M | Base 64 string contains JSON. |

**Payload Base 64 decode JSON fields**

|  |  |  |  |
| --- | --- | --- | --- |
| **Parameter** | **Field Type** | **Required** | **Description** |
| status | String | M | General status of transaction. |
| transactionId | String | O | GLODIPAY Identify transaction |
| ref | String | O | Merchant Identify transaction |
| amount | Float | O |  |
| currency | String | O | Invoice [currency](#page26), ISO 4217 standard specifies three-letter ("Alpha-3") codes. e.g. USD |
| signature | String | O | [Signature](#_Code_example) is calculated in the following steps:   1. Concatenate all parameters values except the signature (arranged in ascending order of parameter’s name) to a string 2. RSA sign the hash by merchant’s private key |

## TRANSACTION QUERY

### Request

* Function: query the info of a transaction.
* Endpoint: /v1/checkout/query
* Method: POST
* Data format: JSON

|  |  |  |  |
| --- | --- | --- | --- |
| **Parameter** | **Field Type** | **Required** | **Description** |
| merchantId | String | M | Merchant’s id |
| transactionId | String | M | GLODIPAY Identify transaction |
| signature | String | M | [Signature](#_Code_example) is calculated in the following steps:   1. Concatenate all parameters values except the signature (arranged in ascending order of parameter’s name) to a string 2. RSA sign the hash by merchant’s private key |

### Response

The same with [NOTIFICATION](#_NOTIFICATION) [payload](#_Payload)

## NOTIFICATION

* Function: send instant payment notification to merchant over API
* Method: POST
* Data format: JSON
* [Return JSON payload {“returnCode”:”100”} in 30 seconds to let GLODIPAY recognize that you have received the Webhook. If not, GLODIPAY will send the Webhook 10 times. The first time is 30 minutes after the transaction is initiated, and then every 5 minutes thereafter if transaction is completely succeeded.](#_Response)

### Payload

|  |  |  |  |
| --- | --- | --- | --- |
| **Parameter** | **Field Type** | **Required** | **Description** |
| merchantId | String | M | Merchant’s id |
| transactionId | String | M | GLODIPAY Identify transaction |
| transactionNumber | String | M | GLODIPAY Transaction Number |
| ref | String | M | Merchant Identify transaction |
| currency | String | M | Invoice [currency](#_Currency_Codes), ISO 4217 standard specifies three-letter ("Alpha-3") codes. e.g. USD |
| amount | Float | M |  |
| paidAmount | Float | O | Paid amount |
| settlementAmount | Float | O | Settlement amount |
| estimationSettlementAt | ISO 8601 datetime format | O | Estimation Settlement Datetime E.g: 2023-12-16T02:13:37+00:00 |
| [fees](#_fees_JSON_fields) | JSON | O | Fees JSON |
| [paymentMethodDetails](#_paymentMethodDetails_JSON_fields) | JSON | O |  |
| status | String | M | Status of transaction |
| [statusCode](#_Status_Codes) | Number | M | Status code of transaction |
| metadata | JSON | O | Set of key-value pairs of checkout session |
| transactionDocuments | JSON | O | Documents of the transaction |
| message | String | O | Transaction message |
| transactionCreatedAt | ISO 8601 datetime format | M | Datetime from GLODIPAY system E.g: 2023-12-16T02:13:37+00:00 |
| originalTransactionCreatedAt | ISO 8601 datetime format | M | Original Datetime E.g: 2023-12-16T02:13:37+00:00 |
| signature | String | M | [Signature](#_Code_example) is calculated in the following steps:   1. Concatenate all parameters values except the signature (arranged in ascending order of parameter’s name) to a string 2. RSA sign the hash by merchant’s private key |

#### fees JSON fields

|  |  |  |  |
| --- | --- | --- | --- |
| **Parameter** | **Field Type** | **Required** | **Description** |
| operate | Float | O | Total fees |
| buyer | Float | O | Buyer fees amount |
| seller | Float | O | Seller fees amount |
| rolling | Float | O | Rolling fees amount |
| estimationRollingReleaseAt | ISO 8601 datetime format | O | Estimation Rolling Release Datetime  E.g: 2025-04-27T02:47:02+00:00 |

#### paymentMethodDetails JSON fields

|  |  |  |  |
| --- | --- | --- | --- |
| **Parameter** | **Field Type** | **Required** | **Description** |
| displayName | String | O | Payment method label |
| group | String | O | Payment method group type |
| family | String | O | Payment method family type |
| type | String | O | Name of payment method type  You can use this value to specify Payment method in [POST PAYMENT](#_Request:) of field [paymentMethod](#_paymentMethod) or [paymentFilter](#_paymentFilter) |
| Value of [*type*](#_type) | JSON | O |  |

### Response

* Data format: JSON

|  |  |  |  |
| --- | --- | --- | --- |
| **Parameter** | **Field Type** | **Required** | **Description** |
| returnCode | String | R | Return code “100” to let GLODIPAY recognize that you have received the webhook. |
| description | String L (1,1500) | O | Description for the response |

# Test Cards with merchant ID start with 11xxxxxxxx

| **Card Number** | **Expire** | **CVV** | **Password** |  |
| --- | --- | --- | --- | --- |
| 4242424242424242 | 12/30 | 123 |  | Success |
| 4000000000001018 | 12/30 | 123 |  | Fail |
| 4012888888881881 | 01/30 | 029 | Success: 123456  Fail: 111111 | 3DS Payment |
| 5111111111111118 | 01/30 | 029 | Success: 123456  Fail: 111111 | 3DS Payment |
| 4141414141414141 | 12/30 | 123 | Success: 123456  Fail: 111111 | 3DS Payment |

# Test Cards with merchant ID start with 14xxxxxxxx

| **Card Number** | **Expire** | **CVV** | **Password** |  |
| --- | --- | --- | --- | --- |
| 4111111111111111 | 01/30 | 029 |  | Success |
| 5555555555554444 | 01/30 | 029 |  | Success |
| 4012888888881881 | 01/30 | 029 | Success: 123456  Fail: 111111 | 3DS Payment |
| 5111111111111118 | 01/30 | 029 | Success: 123456  Fail: 111111 | 3DS Payment |
| 4141414141414141 | 12/30 | 123 | Success: 123456  Fail: 111111 | 3DS Payment |

# APPENDIX

## Payment Methods

|  |  |
| --- | --- |
| **NAME** | **DESCRIPTION** |
| card | Credit or Debit cards |

## Status Codes

|  |  |
| --- | --- |
| **CODE** | **DESCRIPTION** |
| 1 | incomplete |
| 2 | pending |
| 3 | error |
| 4 | failed |
| 5 | under\_review |
| 6 | successful |
| 7 | released |
| 8 | refund\_initiated |
| 9 | refund\_failed |
| 10 | refund\_under\_review |
| 11 | refund\_successful |
| 12 | refund\_partially\_failed |
| 13 | refund\_partially\_successful |
| 14 | canceled |
| 15 | rejected |
| 16 | expired |
| 17 | documents\_uploaded |
| 18 | void\_initiated |
| 19 | void\_under\_review |
| 20 | void\_successful |
| 21 | void\_failed |
| 22 | void\_partially\_successful |
| 23 | void\_partially\_failed |

## Country Codes

| **Code** | **Country name (using**[**title case**](https://en.wikipedia.org/wiki/Title_case)**)** |
| --- | --- |
| [AD](https://en.wikipedia.org/wiki/ISO_3166-2:AD) | [Andorra](https://en.wikipedia.org/wiki/Andorra) |
| [AE](https://en.wikipedia.org/wiki/ISO_3166-2:AE) | [United Arab Emirates](https://en.wikipedia.org/wiki/United_Arab_Emirates) |
| [AF](https://en.wikipedia.org/wiki/ISO_3166-2:AF) | [Afghanistan](https://en.wikipedia.org/wiki/Afghanistan) |
| [AG](https://en.wikipedia.org/wiki/ISO_3166-2:AG) | [Antigua and Barbuda](https://en.wikipedia.org/wiki/Antigua_and_Barbuda) |
| [AI](https://en.wikipedia.org/wiki/ISO_3166-2:AI) | [Anguilla](https://en.wikipedia.org/wiki/Anguilla) |
| [AL](https://en.wikipedia.org/wiki/ISO_3166-2:AL) | [Albania](https://en.wikipedia.org/wiki/Albania) |
| [AM](https://en.wikipedia.org/wiki/ISO_3166-2:AM) | [Armenia](https://en.wikipedia.org/wiki/Armenia) |
| [AO](https://en.wikipedia.org/wiki/ISO_3166-2:AO) | [Angola](https://en.wikipedia.org/wiki/Angola) |
| [AQ](https://en.wikipedia.org/wiki/ISO_3166-2:AQ) | [Antarctica](https://en.wikipedia.org/wiki/Antarctica) |
| [AR](https://en.wikipedia.org/wiki/ISO_3166-2:AR) | [Argentina](https://en.wikipedia.org/wiki/Argentina) |
| [AS](https://en.wikipedia.org/wiki/ISO_3166-2:AS) | [American Samoa](https://en.wikipedia.org/wiki/American_Samoa) |
| [AT](https://en.wikipedia.org/wiki/ISO_3166-2:AT) | [Austria](https://en.wikipedia.org/wiki/Austria) |
| [AU](https://en.wikipedia.org/wiki/ISO_3166-2:AU) | [Australia](https://en.wikipedia.org/wiki/Australia) |
| [AW](https://en.wikipedia.org/wiki/ISO_3166-2:AW) | [Aruba](https://en.wikipedia.org/wiki/Aruba) |
| [AX](https://en.wikipedia.org/wiki/ISO_3166-2:AX) | [Åland Islands](https://en.wikipedia.org/wiki/%C3%85land) |
| [AZ](https://en.wikipedia.org/wiki/ISO_3166-2:AZ) | [Azerbaijan](https://en.wikipedia.org/wiki/Azerbaijan) |
| [BA](https://en.wikipedia.org/wiki/ISO_3166-2:BA) | [Bosnia and Herzegovina](https://en.wikipedia.org/wiki/Bosnia_and_Herzegovina) |
| [BB](https://en.wikipedia.org/wiki/ISO_3166-2:BB) | [Barbados](https://en.wikipedia.org/wiki/Barbados) |
| [BD](https://en.wikipedia.org/wiki/ISO_3166-2:BD) | [Bangladesh](https://en.wikipedia.org/wiki/Bangladesh) |
| [BE](https://en.wikipedia.org/wiki/ISO_3166-2:BE) | [Belgium](https://en.wikipedia.org/wiki/Belgium) |
| [BF](https://en.wikipedia.org/wiki/ISO_3166-2:BF) | [Burkina Faso](https://en.wikipedia.org/wiki/Burkina_Faso) |
| [BG](https://en.wikipedia.org/wiki/ISO_3166-2:BG) | [Bulgaria](https://en.wikipedia.org/wiki/Bulgaria) |
| [BH](https://en.wikipedia.org/wiki/ISO_3166-2:BH) | [Bahrain](https://en.wikipedia.org/wiki/Bahrain) |
| [BI](https://en.wikipedia.org/wiki/ISO_3166-2:BI) | [Burundi](https://en.wikipedia.org/wiki/Burundi) |
| [BJ](https://en.wikipedia.org/wiki/ISO_3166-2:BJ) | [Benin](https://en.wikipedia.org/wiki/Benin) |
| [BL](https://en.wikipedia.org/wiki/ISO_3166-2:BL) | [Saint Barthélemy](https://en.wikipedia.org/wiki/Saint_Barth%C3%A9lemy) |
| [BM](https://en.wikipedia.org/wiki/ISO_3166-2:BM) | [Bermuda](https://en.wikipedia.org/wiki/Bermuda) |
| [BN](https://en.wikipedia.org/wiki/ISO_3166-2:BN) | [Brunei Darussalam](https://en.wikipedia.org/wiki/Brunei) |
| [BO](https://en.wikipedia.org/wiki/ISO_3166-2:BO) | [Bolivia, Plurinational State of](https://en.wikipedia.org/wiki/Bolivia) |
| [BQ](https://en.wikipedia.org/wiki/ISO_3166-2:BQ) | [Bonaire, Sint Eustatius and Saba](https://en.wikipedia.org/wiki/Caribbean_Netherlands) |
| [BR](https://en.wikipedia.org/wiki/ISO_3166-2:BR) | [Brazil](https://en.wikipedia.org/wiki/Brazil) |
| [BS](https://en.wikipedia.org/wiki/ISO_3166-2:BS) | [Bahamas](https://en.wikipedia.org/wiki/The_Bahamas) |
| [BT](https://en.wikipedia.org/wiki/ISO_3166-2:BT) | [Bhutan](https://en.wikipedia.org/wiki/Bhutan) |
| [BV](https://en.wikipedia.org/wiki/ISO_3166-2:BV) | [Bouvet Island](https://en.wikipedia.org/wiki/Bouvet_Island) |
| [BW](https://en.wikipedia.org/wiki/ISO_3166-2:BW) | [Botswana](https://en.wikipedia.org/wiki/Botswana) |
| [BY](https://en.wikipedia.org/wiki/ISO_3166-2:BY) | [Belarus](https://en.wikipedia.org/wiki/Belarus) |
| [BZ](https://en.wikipedia.org/wiki/ISO_3166-2:BZ) | [Belize](https://en.wikipedia.org/wiki/Belize) |
| [CA](https://en.wikipedia.org/wiki/ISO_3166-2:CA) | [Canada](https://en.wikipedia.org/wiki/Canada) |
| [CC](https://en.wikipedia.org/wiki/ISO_3166-2:CC) | [Cocos (Keeling) Islands](https://en.wikipedia.org/wiki/Cocos_(Keeling)_Islands) |
| [CD](https://en.wikipedia.org/wiki/ISO_3166-2:CD) | [Congo, Democratic Republic of the](https://en.wikipedia.org/wiki/Democratic_Republic_of_the_Congo) |
| [CF](https://en.wikipedia.org/wiki/ISO_3166-2:CF) | [Central African Republic](https://en.wikipedia.org/wiki/Central_African_Republic) |
| [CG](https://en.wikipedia.org/wiki/ISO_3166-2:CG) | [Congo](https://en.wikipedia.org/wiki/Republic_of_the_Congo) |
| [CH](https://en.wikipedia.org/wiki/ISO_3166-2:CH) | [Switzerland](https://en.wikipedia.org/wiki/Switzerland) |
| [CI](https://en.wikipedia.org/wiki/ISO_3166-2:CI) | [Côte d'Ivoire](https://en.wikipedia.org/wiki/Ivory_Coast) |
| [CK](https://en.wikipedia.org/wiki/ISO_3166-2:CK) | [Cook Islands](https://en.wikipedia.org/wiki/Cook_Islands) |
| [CL](https://en.wikipedia.org/wiki/ISO_3166-2:CL) | [Chile](https://en.wikipedia.org/wiki/Chile) |
| [CM](https://en.wikipedia.org/wiki/ISO_3166-2:CM) | [Cameroon](https://en.wikipedia.org/wiki/Cameroon) |
| [CN](https://en.wikipedia.org/wiki/ISO_3166-2:CN) | [China](https://en.wikipedia.org/wiki/China) |
| [CO](https://en.wikipedia.org/wiki/ISO_3166-2:CO) | [Colombia](https://en.wikipedia.org/wiki/Colombia) |
| [CR](https://en.wikipedia.org/wiki/ISO_3166-2:CR) | [Costa Rica](https://en.wikipedia.org/wiki/Costa_Rica) |
| [CU](https://en.wikipedia.org/wiki/ISO_3166-2:CU) | [Cuba](https://en.wikipedia.org/wiki/Cuba) |
| [CV](https://en.wikipedia.org/wiki/ISO_3166-2:CV) | [Cabo Verde](https://en.wikipedia.org/wiki/Cape_Verde) |
| [CW](https://en.wikipedia.org/wiki/ISO_3166-2:CW) | [Curaçao](https://en.wikipedia.org/wiki/Cura%C3%A7ao) |
| [CX](https://en.wikipedia.org/wiki/ISO_3166-2:CX) | [Christmas Island](https://en.wikipedia.org/wiki/Christmas_Island) |
| [CY](https://en.wikipedia.org/wiki/ISO_3166-2:CY) | [Cyprus](https://en.wikipedia.org/wiki/Cyprus) |
| [CZ](https://en.wikipedia.org/wiki/ISO_3166-2:CZ) | [Czechia](https://en.wikipedia.org/wiki/Czech_Republic) |
| [DE](https://en.wikipedia.org/wiki/ISO_3166-2:DE) | [Germany](https://en.wikipedia.org/wiki/Germany) |
| [DJ](https://en.wikipedia.org/wiki/ISO_3166-2:DJ) | [Djibouti](https://en.wikipedia.org/wiki/Djibouti) |
| [DK](https://en.wikipedia.org/wiki/ISO_3166-2:DK) | [Denmark](https://en.wikipedia.org/wiki/Denmark) |
| [DM](https://en.wikipedia.org/wiki/ISO_3166-2:DM) | [Dominica](https://en.wikipedia.org/wiki/Dominica) |
| [DO](https://en.wikipedia.org/wiki/ISO_3166-2:DO) | [Dominican Republic](https://en.wikipedia.org/wiki/Dominican_Republic) |
| [DZ](https://en.wikipedia.org/wiki/ISO_3166-2:DZ) | [Algeria](https://en.wikipedia.org/wiki/Algeria) |
| [EC](https://en.wikipedia.org/wiki/ISO_3166-2:EC) | [Ecuador](https://en.wikipedia.org/wiki/Ecuador) |
| [EE](https://en.wikipedia.org/wiki/ISO_3166-2:EE) | [Estonia](https://en.wikipedia.org/wiki/Estonia) |
| [EG](https://en.wikipedia.org/wiki/ISO_3166-2:EG) | [Egypt](https://en.wikipedia.org/wiki/Egypt) |
| [EH](https://en.wikipedia.org/wiki/ISO_3166-2:EH) | [Western Sahara](https://en.wikipedia.org/wiki/Western_Sahara) |
| [ER](https://en.wikipedia.org/wiki/ISO_3166-2:ER) | [Eritrea](https://en.wikipedia.org/wiki/Eritrea) |
| [ES](https://en.wikipedia.org/wiki/ISO_3166-2:ES) | [Spain](https://en.wikipedia.org/wiki/Spain) |
| [ET](https://en.wikipedia.org/wiki/ISO_3166-2:ET) | [Ethiopia](https://en.wikipedia.org/wiki/Ethiopia) |
| [FI](https://en.wikipedia.org/wiki/ISO_3166-2:FI) | [Finland](https://en.wikipedia.org/wiki/Finland) |
| [FJ](https://en.wikipedia.org/wiki/ISO_3166-2:FJ) | [Fiji](https://en.wikipedia.org/wiki/Fiji) |
| [FK](https://en.wikipedia.org/wiki/ISO_3166-2:FK) | [Falkland Islands (Malvinas)](https://en.wikipedia.org/wiki/Falkland_Islands) |
| [FM](https://en.wikipedia.org/wiki/ISO_3166-2:FM) | [Micronesia, Federated States of](https://en.wikipedia.org/wiki/Federated_States_of_Micronesia) |
| [FO](https://en.wikipedia.org/wiki/ISO_3166-2:FO) | [Faroe Islands](https://en.wikipedia.org/wiki/Faroe_Islands) |
| [FR](https://en.wikipedia.org/wiki/ISO_3166-2:FR) | [France](https://en.wikipedia.org/wiki/France) |
| [GA](https://en.wikipedia.org/wiki/ISO_3166-2:GA) | [Gabon](https://en.wikipedia.org/wiki/Gabon) |
| [GB](https://en.wikipedia.org/wiki/ISO_3166-2:GB) | [United Kingdom of Great Britain and Northern Ireland](https://en.wikipedia.org/wiki/United_Kingdom) |
| [GD](https://en.wikipedia.org/wiki/ISO_3166-2:GD) | [Grenada](https://en.wikipedia.org/wiki/Grenada) |
| [GE](https://en.wikipedia.org/wiki/ISO_3166-2:GE) | [Georgia](https://en.wikipedia.org/wiki/Georgia_(country)) |
| [GF](https://en.wikipedia.org/wiki/ISO_3166-2:GF) | [French Guiana](https://en.wikipedia.org/wiki/French_Guiana) |
| [GG](https://en.wikipedia.org/wiki/ISO_3166-2:GG) | [Guernsey](https://en.wikipedia.org/wiki/Bailiwick_of_Guernsey) |
| [GH](https://en.wikipedia.org/wiki/ISO_3166-2:GH) | [Ghana](https://en.wikipedia.org/wiki/Ghana) |
| [GI](https://en.wikipedia.org/wiki/ISO_3166-2:GI) | [Gibraltar](https://en.wikipedia.org/wiki/Gibraltar) |
| [GL](https://en.wikipedia.org/wiki/ISO_3166-2:GL) | [Greenland](https://en.wikipedia.org/wiki/Greenland) |
| [GM](https://en.wikipedia.org/wiki/ISO_3166-2:GM) | [Gambia](https://en.wikipedia.org/wiki/The_Gambia) |
| [GN](https://en.wikipedia.org/wiki/ISO_3166-2:GN) | [Guinea](https://en.wikipedia.org/wiki/Guinea) |
| [GP](https://en.wikipedia.org/wiki/ISO_3166-2:GP) | [Guadeloupe](https://en.wikipedia.org/wiki/Guadeloupe) |
| [GQ](https://en.wikipedia.org/wiki/ISO_3166-2:GQ) | [Equatorial Guinea](https://en.wikipedia.org/wiki/Equatorial_Guinea) |
| [GR](https://en.wikipedia.org/wiki/ISO_3166-2:GR) | [Greece](https://en.wikipedia.org/wiki/Greece) |
| [GS](https://en.wikipedia.org/wiki/ISO_3166-2:GS) | [South Georgia and the South Sandwich Islands](https://en.wikipedia.org/wiki/South_Georgia_and_the_South_Sandwich_Islands) |
| [GT](https://en.wikipedia.org/wiki/ISO_3166-2:GT) | [Guatemala](https://en.wikipedia.org/wiki/Guatemala) |
| [GU](https://en.wikipedia.org/wiki/ISO_3166-2:GU) | [Guam](https://en.wikipedia.org/wiki/Guam) |
| [GW](https://en.wikipedia.org/wiki/ISO_3166-2:GW) | [Guinea-Bissau](https://en.wikipedia.org/wiki/Guinea-Bissau) |
| [GY](https://en.wikipedia.org/wiki/ISO_3166-2:GY) | [Guyana](https://en.wikipedia.org/wiki/Guyana) |
| [HK](https://en.wikipedia.org/wiki/ISO_3166-2:HK) | [Hong Kong](https://en.wikipedia.org/wiki/Hong_Kong) |
| [HM](https://en.wikipedia.org/wiki/ISO_3166-2:HM) | [Heard Island and McDonald Islands](https://en.wikipedia.org/wiki/Heard_Island_and_McDonald_Islands) |
| [HN](https://en.wikipedia.org/wiki/ISO_3166-2:HN) | [Honduras](https://en.wikipedia.org/wiki/Honduras) |
| [HR](https://en.wikipedia.org/wiki/ISO_3166-2:HR) | [Croatia](https://en.wikipedia.org/wiki/Croatia) |
| [HT](https://en.wikipedia.org/wiki/ISO_3166-2:HT) | [Haiti](https://en.wikipedia.org/wiki/Haiti) |
| [HU](https://en.wikipedia.org/wiki/ISO_3166-2:HU) | [Hungary](https://en.wikipedia.org/wiki/Hungary) |
| [ID](https://en.wikipedia.org/wiki/ISO_3166-2:ID) | [Indonesia](https://en.wikipedia.org/wiki/Indonesia) |
| [IE](https://en.wikipedia.org/wiki/ISO_3166-2:IE) | [Ireland](https://en.wikipedia.org/wiki/Republic_of_Ireland) |
| [IL](https://en.wikipedia.org/wiki/ISO_3166-2:IL) | [Israel](https://en.wikipedia.org/wiki/Israel) |
| [IM](https://en.wikipedia.org/wiki/ISO_3166-2:IM) | [Isle of Man](https://en.wikipedia.org/wiki/Isle_of_Man) |
| [IN](https://en.wikipedia.org/wiki/ISO_3166-2:IN) | [India](https://en.wikipedia.org/wiki/India) |
| [IO](https://en.wikipedia.org/wiki/ISO_3166-2:IO) | [British Indian Ocean Territory](https://en.wikipedia.org/wiki/British_Indian_Ocean_Territory) |
| [IQ](https://en.wikipedia.org/wiki/ISO_3166-2:IQ) | [Iraq](https://en.wikipedia.org/wiki/Iraq) |
| [IR](https://en.wikipedia.org/wiki/ISO_3166-2:IR) | [Iran, Islamic Republic of](https://en.wikipedia.org/wiki/Iran) |
| [IS](https://en.wikipedia.org/wiki/ISO_3166-2:IS) | [Iceland](https://en.wikipedia.org/wiki/Iceland) |
| [IT](https://en.wikipedia.org/wiki/ISO_3166-2:IT) | [Italy](https://en.wikipedia.org/wiki/Italy) |
| [JE](https://en.wikipedia.org/wiki/ISO_3166-2:JE) | [Jersey](https://en.wikipedia.org/wiki/Jersey) |
| [JM](https://en.wikipedia.org/wiki/ISO_3166-2:JM) | [Jamaica](https://en.wikipedia.org/wiki/Jamaica) |
| [JO](https://en.wikipedia.org/wiki/ISO_3166-2:JO) | [Jordan](https://en.wikipedia.org/wiki/Jordan) |
| [JP](https://en.wikipedia.org/wiki/ISO_3166-2:JP) | [Japan](https://en.wikipedia.org/wiki/Japan) |
| [KE](https://en.wikipedia.org/wiki/ISO_3166-2:KE) | [Kenya](https://en.wikipedia.org/wiki/Kenya) |
| [KG](https://en.wikipedia.org/wiki/ISO_3166-2:KG) | [Kyrgyzstan](https://en.wikipedia.org/wiki/Kyrgyzstan) |
| [KH](https://en.wikipedia.org/wiki/ISO_3166-2:KH) | [Cambodia](https://en.wikipedia.org/wiki/Cambodia) |
| [KI](https://en.wikipedia.org/wiki/ISO_3166-2:KI) | [Kiribati](https://en.wikipedia.org/wiki/Kiribati) |
| [KM](https://en.wikipedia.org/wiki/ISO_3166-2:KM) | [Comoros](https://en.wikipedia.org/wiki/Comoros) |
| [KN](https://en.wikipedia.org/wiki/ISO_3166-2:KN) | [Saint Kitts and Nevis](https://en.wikipedia.org/wiki/Saint_Kitts_and_Nevis) |
| [KP](https://en.wikipedia.org/wiki/ISO_3166-2:KP) | [Korea, Democratic People's Republic of](https://en.wikipedia.org/wiki/North_Korea) |
| [KR](https://en.wikipedia.org/wiki/ISO_3166-2:KR) | [Korea, Republic of](https://en.wikipedia.org/wiki/South_Korea) |
| [KW](https://en.wikipedia.org/wiki/ISO_3166-2:KW) | [Kuwait](https://en.wikipedia.org/wiki/Kuwait) |
| [KY](https://en.wikipedia.org/wiki/ISO_3166-2:KY) | [Cayman Islands](https://en.wikipedia.org/wiki/Cayman_Islands) |
| [KZ](https://en.wikipedia.org/wiki/ISO_3166-2:KZ) | [Kazakhstan](https://en.wikipedia.org/wiki/Kazakhstan) |
| [LA](https://en.wikipedia.org/wiki/ISO_3166-2:LA) | [Lao People's Democratic Republic](https://en.wikipedia.org/wiki/Laos) |
| [LB](https://en.wikipedia.org/wiki/ISO_3166-2:LB) | [Lebanon](https://en.wikipedia.org/wiki/Lebanon) |
| [LC](https://en.wikipedia.org/wiki/ISO_3166-2:LC) | [Saint Lucia](https://en.wikipedia.org/wiki/Saint_Lucia) |
| [LI](https://en.wikipedia.org/wiki/ISO_3166-2:LI) | [Liechtenstein](https://en.wikipedia.org/wiki/Liechtenstein) |
| [LK](https://en.wikipedia.org/wiki/ISO_3166-2:LK) | [Sri Lanka](https://en.wikipedia.org/wiki/Sri_Lanka) |
| [LR](https://en.wikipedia.org/wiki/ISO_3166-2:LR) | [Liberia](https://en.wikipedia.org/wiki/Liberia) |
| [LS](https://en.wikipedia.org/wiki/ISO_3166-2:LS) | [Lesotho](https://en.wikipedia.org/wiki/Lesotho) |
| [LT](https://en.wikipedia.org/wiki/ISO_3166-2:LT) | [Lithuania](https://en.wikipedia.org/wiki/Lithuania) |
| [LU](https://en.wikipedia.org/wiki/ISO_3166-2:LU) | [Luxembourg](https://en.wikipedia.org/wiki/Luxembourg) |
| [LV](https://en.wikipedia.org/wiki/ISO_3166-2:LV) | [Latvia](https://en.wikipedia.org/wiki/Latvia) |
| [LY](https://en.wikipedia.org/wiki/ISO_3166-2:LY) | [Libya](https://en.wikipedia.org/wiki/Libya) |
| [MA](https://en.wikipedia.org/wiki/ISO_3166-2:MA) | [Morocco](https://en.wikipedia.org/wiki/Morocco) |
| [MC](https://en.wikipedia.org/wiki/ISO_3166-2:MC) | [Monaco](https://en.wikipedia.org/wiki/Monaco) |
| [MD](https://en.wikipedia.org/wiki/ISO_3166-2:MD) | [Moldova, Republic of](https://en.wikipedia.org/wiki/Moldova) |
| [ME](https://en.wikipedia.org/wiki/ISO_3166-2:ME) | [Montenegro](https://en.wikipedia.org/wiki/Montenegro) |
| [MF](https://en.wikipedia.org/wiki/ISO_3166-2:MF) | [Saint Martin (French part)](https://en.wikipedia.org/wiki/Collectivity_of_Saint_Martin) |
| [MG](https://en.wikipedia.org/wiki/ISO_3166-2:MG) | [Madagascar](https://en.wikipedia.org/wiki/Madagascar) |
| [MH](https://en.wikipedia.org/wiki/ISO_3166-2:MH) | [Marshall Islands](https://en.wikipedia.org/wiki/Marshall_Islands) |
| [MK](https://en.wikipedia.org/wiki/ISO_3166-2:MK) | [North Macedonia](https://en.wikipedia.org/wiki/North_Macedonia) |
| [ML](https://en.wikipedia.org/wiki/ISO_3166-2:ML) | [Mali](https://en.wikipedia.org/wiki/Mali) |
| [MM](https://en.wikipedia.org/wiki/ISO_3166-2:MM) | [Myanmar](https://en.wikipedia.org/wiki/Myanmar) |
| [MN](https://en.wikipedia.org/wiki/ISO_3166-2:MN) | [Mongolia](https://en.wikipedia.org/wiki/Mongolia) |
| [MO](https://en.wikipedia.org/wiki/ISO_3166-2:MO) | [Macao](https://en.wikipedia.org/wiki/Macau) |
| [MP](https://en.wikipedia.org/wiki/ISO_3166-2:MP) | [Northern Mariana Islands](https://en.wikipedia.org/wiki/Northern_Mariana_Islands) |
| [MQ](https://en.wikipedia.org/wiki/ISO_3166-2:MQ) | [Martinique](https://en.wikipedia.org/wiki/Martinique) |
| [MR](https://en.wikipedia.org/wiki/ISO_3166-2:MR) | [Mauritania](https://en.wikipedia.org/wiki/Mauritania) |
| [MS](https://en.wikipedia.org/wiki/ISO_3166-2:MS) | [Montserrat](https://en.wikipedia.org/wiki/Montserrat) |
| [MT](https://en.wikipedia.org/wiki/ISO_3166-2:MT) | [Malta](https://en.wikipedia.org/wiki/Malta) |
| [MU](https://en.wikipedia.org/wiki/ISO_3166-2:MU) | [Mauritius](https://en.wikipedia.org/wiki/Mauritius) |
| [MV](https://en.wikipedia.org/wiki/ISO_3166-2:MV) | [Maldives](https://en.wikipedia.org/wiki/Maldives) |
| [MW](https://en.wikipedia.org/wiki/ISO_3166-2:MW) | [Malawi](https://en.wikipedia.org/wiki/Malawi) |
| [MX](https://en.wikipedia.org/wiki/ISO_3166-2:MX) | [Mexico](https://en.wikipedia.org/wiki/Mexico) |
| [MY](https://en.wikipedia.org/wiki/ISO_3166-2:MY) | [Malaysia](https://en.wikipedia.org/wiki/Malaysia) |
| [MZ](https://en.wikipedia.org/wiki/ISO_3166-2:MZ) | [Mozambique](https://en.wikipedia.org/wiki/Mozambique) |
| [NA](https://en.wikipedia.org/wiki/ISO_3166-2:NA) | [Namibia](https://en.wikipedia.org/wiki/Namibia) |
| [NC](https://en.wikipedia.org/wiki/ISO_3166-2:NC) | [New Caledonia](https://en.wikipedia.org/wiki/New_Caledonia) |
| [NE](https://en.wikipedia.org/wiki/ISO_3166-2:NE) | [Niger](https://en.wikipedia.org/wiki/Niger) |
| [NF](https://en.wikipedia.org/wiki/ISO_3166-2:NF) | [Norfolk Island](https://en.wikipedia.org/wiki/Norfolk_Island) |
| [NG](https://en.wikipedia.org/wiki/ISO_3166-2:NG) | [Nigeria](https://en.wikipedia.org/wiki/Nigeria) |
| [NI](https://en.wikipedia.org/wiki/ISO_3166-2:NI) | [Nicaragua](https://en.wikipedia.org/wiki/Nicaragua) |
| [NL](https://en.wikipedia.org/wiki/ISO_3166-2:NL) | [Netherlands, Kingdom of the](https://en.wikipedia.org/wiki/Kingdom_of_the_Netherlands) |
| [NO](https://en.wikipedia.org/wiki/ISO_3166-2:NO) | [Norway](https://en.wikipedia.org/wiki/Norway) |
| [NP](https://en.wikipedia.org/wiki/ISO_3166-2:NP) | [Nepal](https://en.wikipedia.org/wiki/Nepal) |
| [NR](https://en.wikipedia.org/wiki/ISO_3166-2:NR) | [Nauru](https://en.wikipedia.org/wiki/Nauru) |
| [NU](https://en.wikipedia.org/wiki/ISO_3166-2:NU) | [Niue](https://en.wikipedia.org/wiki/Niue) |
| [NZ](https://en.wikipedia.org/wiki/ISO_3166-2:NZ) | [New Zealand](https://en.wikipedia.org/wiki/New_Zealand) |
| [OM](https://en.wikipedia.org/wiki/ISO_3166-2:OM) | [Oman](https://en.wikipedia.org/wiki/Oman) |
| [PA](https://en.wikipedia.org/wiki/ISO_3166-2:PA) | [Panama](https://en.wikipedia.org/wiki/Panama) |
| [PE](https://en.wikipedia.org/wiki/ISO_3166-2:PE) | [Peru](https://en.wikipedia.org/wiki/Peru) |
| [PF](https://en.wikipedia.org/wiki/ISO_3166-2:PF) | [French Polynesia](https://en.wikipedia.org/wiki/French_Polynesia) |
| [PG](https://en.wikipedia.org/wiki/ISO_3166-2:PG) | [Papua New Guinea](https://en.wikipedia.org/wiki/Papua_New_Guinea) |
| [PH](https://en.wikipedia.org/wiki/ISO_3166-2:PH) | [Philippines](https://en.wikipedia.org/wiki/Philippines) |
| [PK](https://en.wikipedia.org/wiki/ISO_3166-2:PK) | [Pakistan](https://en.wikipedia.org/wiki/Pakistan) |
| [PL](https://en.wikipedia.org/wiki/ISO_3166-2:PL) | [Poland](https://en.wikipedia.org/wiki/Poland) |
| [PM](https://en.wikipedia.org/wiki/ISO_3166-2:PM) | [Saint Pierre and Miquelon](https://en.wikipedia.org/wiki/Saint_Pierre_and_Miquelon) |
| [PN](https://en.wikipedia.org/wiki/ISO_3166-2:PN) | [Pitcairn](https://en.wikipedia.org/wiki/Pitcairn_Islands) |
| [PR](https://en.wikipedia.org/wiki/ISO_3166-2:PR) | [Puerto Rico](https://en.wikipedia.org/wiki/Puerto_Rico) |
| [PS](https://en.wikipedia.org/wiki/ISO_3166-2:PS) | [Palestine, State of](https://en.wikipedia.org/wiki/State_of_Palestine) |
| [PT](https://en.wikipedia.org/wiki/ISO_3166-2:PT) | [Portugal](https://en.wikipedia.org/wiki/Portugal) |
| [PW](https://en.wikipedia.org/wiki/ISO_3166-2:PW) | [Palau](https://en.wikipedia.org/wiki/Palau) |
| [PY](https://en.wikipedia.org/wiki/ISO_3166-2:PY) | [Paraguay](https://en.wikipedia.org/wiki/Paraguay) |
| [QA](https://en.wikipedia.org/wiki/ISO_3166-2:QA) | [Qatar](https://en.wikipedia.org/wiki/Qatar) |
| [RE](https://en.wikipedia.org/wiki/ISO_3166-2:RE) | [Réunion](https://en.wikipedia.org/wiki/R%C3%A9union) |
| [RO](https://en.wikipedia.org/wiki/ISO_3166-2:RO) | [Romania](https://en.wikipedia.org/wiki/Romania) |
| [RS](https://en.wikipedia.org/wiki/ISO_3166-2:RS) | [Serbia](https://en.wikipedia.org/wiki/Serbia) |
| [RU](https://en.wikipedia.org/wiki/ISO_3166-2:RU) | [Russian Federation](https://en.wikipedia.org/wiki/Russia) |
| [RW](https://en.wikipedia.org/wiki/ISO_3166-2:RW) | [Rwanda](https://en.wikipedia.org/wiki/Rwanda) |
| [SA](https://en.wikipedia.org/wiki/ISO_3166-2:SA) | [Saudi Arabia](https://en.wikipedia.org/wiki/Saudi_Arabia) |
| [SB](https://en.wikipedia.org/wiki/ISO_3166-2:SB) | [Solomon Islands](https://en.wikipedia.org/wiki/Solomon_Islands) |
| [SC](https://en.wikipedia.org/wiki/ISO_3166-2:SC) | [Seychelles](https://en.wikipedia.org/wiki/Seychelles) |
| [SD](https://en.wikipedia.org/wiki/ISO_3166-2:SD) | [Sudan](https://en.wikipedia.org/wiki/Sudan) |
| [SE](https://en.wikipedia.org/wiki/ISO_3166-2:SE) | [Sweden](https://en.wikipedia.org/wiki/Sweden) |
| [SG](https://en.wikipedia.org/wiki/ISO_3166-2:SG) | [Singapore](https://en.wikipedia.org/wiki/Singapore) |
| [SH](https://en.wikipedia.org/wiki/ISO_3166-2:SH) | [Saint Helena, Ascension and Tristan da Cunha](https://en.wikipedia.org/wiki/Saint_Helena,_Ascension_and_Tristan_da_Cunha) |
| [SI](https://en.wikipedia.org/wiki/ISO_3166-2:SI) | [Slovenia](https://en.wikipedia.org/wiki/Slovenia) |
| [SJ](https://en.wikipedia.org/wiki/ISO_3166-2:SJ) | [Svalbard and Jan Mayen](https://en.wikipedia.org/wiki/Svalbard_and_Jan_Mayen) |
| [SK](https://en.wikipedia.org/wiki/ISO_3166-2:SK) | [Slovakia](https://en.wikipedia.org/wiki/Slovakia) |
| [SL](https://en.wikipedia.org/wiki/ISO_3166-2:SL) | [Sierra Leone](https://en.wikipedia.org/wiki/Sierra_Leone) |
| [SM](https://en.wikipedia.org/wiki/ISO_3166-2:SM) | [San Marino](https://en.wikipedia.org/wiki/San_Marino) |
| [SN](https://en.wikipedia.org/wiki/ISO_3166-2:SN) | [Senegal](https://en.wikipedia.org/wiki/Senegal) |
| [SO](https://en.wikipedia.org/wiki/ISO_3166-2:SO) | [Somalia](https://en.wikipedia.org/wiki/Somalia) |
| [SR](https://en.wikipedia.org/wiki/ISO_3166-2:SR) | [Suriname](https://en.wikipedia.org/wiki/Suriname) |
| [SS](https://en.wikipedia.org/wiki/ISO_3166-2:SS) | [South Sudan](https://en.wikipedia.org/wiki/South_Sudan) |
| [ST](https://en.wikipedia.org/wiki/ISO_3166-2:ST) | [Sao Tome and Principe](https://en.wikipedia.org/wiki/S%C3%A3o_Tom%C3%A9_and_Pr%C3%ADncipe) |
| [SV](https://en.wikipedia.org/wiki/ISO_3166-2:SV) | [El Salvador](https://en.wikipedia.org/wiki/El_Salvador) |
| [SX](https://en.wikipedia.org/wiki/ISO_3166-2:SX) | [Sint Maarten (Dutch part)](https://en.wikipedia.org/wiki/Sint_Maarten) |
| [SY](https://en.wikipedia.org/wiki/ISO_3166-2:SY) | [Syrian Arab Republic](https://en.wikipedia.org/wiki/Syria) |
| [SZ](https://en.wikipedia.org/wiki/ISO_3166-2:SZ) | [Eswatini](https://en.wikipedia.org/wiki/Eswatini) |
| [TC](https://en.wikipedia.org/wiki/ISO_3166-2:TC) | [Turks and Caicos Islands](https://en.wikipedia.org/wiki/Turks_and_Caicos_Islands) |
| [TD](https://en.wikipedia.org/wiki/ISO_3166-2:TD) | [Chad](https://en.wikipedia.org/wiki/Chad) |
| [TF](https://en.wikipedia.org/wiki/ISO_3166-2:TF) | [French Southern Territories](https://en.wikipedia.org/wiki/French_Southern_and_Antarctic_Lands) |
| [TG](https://en.wikipedia.org/wiki/ISO_3166-2:TG) | [Togo](https://en.wikipedia.org/wiki/Togo) |
| [TH](https://en.wikipedia.org/wiki/ISO_3166-2:TH) | [Thailand](https://en.wikipedia.org/wiki/Thailand) |
| [TJ](https://en.wikipedia.org/wiki/ISO_3166-2:TJ) | [Tajikistan](https://en.wikipedia.org/wiki/Tajikistan) |
| [TK](https://en.wikipedia.org/wiki/ISO_3166-2:TK) | [Tokelau](https://en.wikipedia.org/wiki/Tokelau) |
| [TL](https://en.wikipedia.org/wiki/ISO_3166-2:TL) | [Timor-Leste](https://en.wikipedia.org/wiki/East_Timor) |
| [TM](https://en.wikipedia.org/wiki/ISO_3166-2:TM) | [Turkmenistan](https://en.wikipedia.org/wiki/Turkmenistan) |
| [TN](https://en.wikipedia.org/wiki/ISO_3166-2:TN) | [Tunisia](https://en.wikipedia.org/wiki/Tunisia) |
| [TO](https://en.wikipedia.org/wiki/ISO_3166-2:TO) | [Tonga](https://en.wikipedia.org/wiki/Tonga) |
| [TR](https://en.wikipedia.org/wiki/ISO_3166-2:TR) | [Türkiye](https://en.wikipedia.org/wiki/Turkey) |
| [TT](https://en.wikipedia.org/wiki/ISO_3166-2:TT) | [Trinidad and Tobago](https://en.wikipedia.org/wiki/Trinidad_and_Tobago) |
| [TV](https://en.wikipedia.org/wiki/ISO_3166-2:TV) | [Tuvalu](https://en.wikipedia.org/wiki/Tuvalu) |
| [TW](https://en.wikipedia.org/wiki/ISO_3166-2:TW) | [Taiwan, Province of China](https://en.wikipedia.org/wiki/Taiwan,_China) |
| [TZ](https://en.wikipedia.org/wiki/ISO_3166-2:TZ) | [Tanzania, United Republic of](https://en.wikipedia.org/wiki/Tanzania) |
| [UA](https://en.wikipedia.org/wiki/ISO_3166-2:UA) | [Ukraine](https://en.wikipedia.org/wiki/Ukraine) |
| [UG](https://en.wikipedia.org/wiki/ISO_3166-2:UG) | [Uganda](https://en.wikipedia.org/wiki/Uganda) |
| [UM](https://en.wikipedia.org/wiki/ISO_3166-2:UM) | [United States Minor Outlying Islands](https://en.wikipedia.org/wiki/United_States_Minor_Outlying_Islands) |
| [US](https://en.wikipedia.org/wiki/ISO_3166-2:US) | [United States of America](https://en.wikipedia.org/wiki/United_States) |
| [UY](https://en.wikipedia.org/wiki/ISO_3166-2:UY) | [Uruguay](https://en.wikipedia.org/wiki/Uruguay) |
| [UZ](https://en.wikipedia.org/wiki/ISO_3166-2:UZ) | [Uzbekistan](https://en.wikipedia.org/wiki/Uzbekistan) |
| [VA](https://en.wikipedia.org/wiki/ISO_3166-2:VA) | [Holy See](https://en.wikipedia.org/wiki/Vatican_City) |
| [VC](https://en.wikipedia.org/wiki/ISO_3166-2:VC) | [Saint Vincent and the Grenadines](https://en.wikipedia.org/wiki/Saint_Vincent_and_the_Grenadines) |
| [VE](https://en.wikipedia.org/wiki/ISO_3166-2:VE) | [Venezuela, Bolivarian Republic of](https://en.wikipedia.org/wiki/Venezuela) |
| [VG](https://en.wikipedia.org/wiki/ISO_3166-2:VG) | [Virgin Islands (British)](https://en.wikipedia.org/wiki/British_Virgin_Islands) |
| [VI](https://en.wikipedia.org/wiki/ISO_3166-2:VI) | [Virgin Islands (U.S.)](https://en.wikipedia.org/wiki/United_States_Virgin_Islands) |
| [VN](https://en.wikipedia.org/wiki/ISO_3166-2:VN) | [Viet Nam](https://en.wikipedia.org/wiki/Vietnam) |
| [VU](https://en.wikipedia.org/wiki/ISO_3166-2:VU) | [Vanuatu](https://en.wikipedia.org/wiki/Vanuatu) |
| [WF](https://en.wikipedia.org/wiki/ISO_3166-2:WF) | [Wallis and Futuna](https://en.wikipedia.org/wiki/Wallis_and_Futuna) |
| [WS](https://en.wikipedia.org/wiki/ISO_3166-2:WS) | [Samoa](https://en.wikipedia.org/wiki/Samoa) |
| [YE](https://en.wikipedia.org/wiki/ISO_3166-2:YE) | [Yemen](https://en.wikipedia.org/wiki/Yemen) |
| [YT](https://en.wikipedia.org/wiki/ISO_3166-2:YT) | [Mayotte](https://en.wikipedia.org/wiki/Mayotte) |
| [ZA](https://en.wikipedia.org/wiki/ISO_3166-2:ZA) | [South Africa](https://en.wikipedia.org/wiki/South_Africa) |
| [ZM](https://en.wikipedia.org/wiki/ISO_3166-2:ZM) | [Zambia](https://en.wikipedia.org/wiki/Zambia) |
| [ZW](https://en.wikipedia.org/wiki/ISO_3166-2:ZW) | [Zimbabwe](https://en.wikipedia.org/wiki/Zimbabwe) |

## Card Type

|  |  |  |
| --- | --- | --- |
| **No.** | **Card Type** | **Code** |
| 1 | VISA | visa |
| 2 | MASTER CARD | mastercard |
| 3 | AMERICAN EXPRESS | amex |
| 4 | JCB | jcb |
| 5 | MAESTRO | maestro |
| 6 | DISCOVER | discover |
| 7 | UNION PAY | union-pay |
| 8 | DINERS | diners |

## Code example

### PHP

<?php

function generateSignature(array $data): string

{

$privateKey = openssl\_pkey\_get\_private("-----BEGIN PRIVATE KEY-----

MIIJQQIBADANBgkqhkiG9w0BAQEFAASCCSswggknAgEAAoICAQDwgPXMA8tv7boG

...

8S7wZ3XcsaLzlzMXrxhykwsDFBFg

-----END PRIVATE KEY-----

");

ksort($data, SORT\_NATURAL);

array\_walk\_recursive(

$data,

static function(&$field) {

$field = trim($field);

}

);

openssl\_sign(json\_encode($data), $signature,

$privateKey, 'md5WithRSAEncryption'

);

return base64\_encode($signature);

}

function verifySignature(array $data): bool

{

$publicKey = openssl\_pkey\_get\_public("-----BEGIN PUBLIC KEY-----

MIICIjANBgkqhkiG9w0BAQEFAAOCAg8AMIICCgKCAgEAxu+5iLOm7KCXoGxUMHX5

...

PR1ofGsEO8/N/qJuggQc2P8CAwEAAQ==

-----END PUBLIC KEY-----

");

$dataWithoutSignature = array\_filter($data, static function ($key) {

return $key !== 'signature';

}, ARRAY\_FILTER\_USE\_KEY);

$signature = $data['signature'];

ksort($dataWithoutSignature, SORT\_NATURAL);

array\_walk\_recursive(

$dataWithoutSignature,

static function(&$field) {

$field = trim($field);

}

);

$result = openssl\_verify(json\_encode($dataWithoutSignature), base64\_decode($signature), $publicKey, 'md5WithRSAEncryption');

return $result === 1;

}

### NodeJS

function generateSignature(data) {

const crypto = require('crypto');

const privateKey = `-----BEGIN PRIVATE KEY-----

MIIJQQIBADANBgkqhkiG9w0BAQEFAASCCSswggknAgEAAoICAQDwgPXMA8tv7boG

...

8S7wZ3XcsaLzlzMXrxhykwsDFBFg

-----END PRIVATE KEY-----

`;

const clonedData = JSON.parse(JSON.stringify(data));

Object.keys(clonedData).forEach(key => {

if (typeof clonedData[key] === 'number') {

clonedData[key] = String(clonedData[key]);

}

});

const sortedData = {};

Object.keys(clonedData)

.sort((a, b) => a.localeCompare(b, undefined, { numeric: true, sensitivity: 'base' }))

.forEach(key => {

sortedData[key] = clonedData[key];

});

const trimRecursive = (obj) => {

if (typeof obj !== 'object' || obj === null) {

return typeof obj === 'string' ? obj.trim() : obj;

}

if (Array.isArray(obj)) {

return obj.map(item => trimRecursive(item));

}

const result = {};

for (const key in obj) {

if (Object.prototype.hasOwnProperty.call(obj, key)) {

result[key] = trimRecursive(obj[key]);

}

}

return result;

};

const trimmedData = trimRecursive(sortedData);

const jsonData = JSON.stringify(trimmedData).replace(/\//g, '\\/');

const sign = crypto.createSign('md5WithRSAEncryption');

sign.update(jsonData);

const signature = sign.sign(privateKey, 'base64');

return signature;

}