



OpenAccount API Documentation

Version 1.0

Contents	2
Introduction	3
1. Introduction with API	4
1.1 Open a demo account	4
1.2 Open a trading account	6
2. Practical use	8
2.1 C#	8

Introduction

InstaForex OpenAccount API provides an opportunity to open live and demo accounts with InstaForex. API is the service using the SOAP protocol. WSDL document is available at

<http://client-api.instaforex.com/soapservices/OpenAccount.svc?wsdl>

1. Interaction with API

1.1 Open a demo account

Here is an example of the request to open a demo account and the response in case of successful operation.

Entry point:

<http://client-api.instaforex.com/soapservices/OpenAccount.svc>

Request:

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Header>
    <Action s:mustUnderstand="1"
xmlns="http://schemas.microsoft.com/ws/2005/05/addressing/none">http://client-api.
instaforex.com/OpenAccount/IOpenAccountService/OpenDemoAccount</Action>
  </s:Header>
  <s:Body>
    <OpenDemoAccount xmlns="http://client-api.instaforex.com/OpenAccount/">
      <demoAccount xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
        <AccountType>Standard</AccountType>
        <Address>Demo Address</Address>
        <City>Demo City</City>
        <Country>Poland</Country>
        <Email>demo@demo.demo</Email>
        <Ip>ipv4</Ip>
        <Leverage>100</Leverage>
        <Name>Demo account</Name>
        <Phone>1000000000</Phone>
        <State>Demo state</State>
        <ZipCode>00000</ZipCode>
        <InitialDeposit>1000</InitialDeposit>
        <InvestorPassword>qweqwe1</InvestorPassword>
        <Password>1qweqwe</Password>
      </demoAccount>
    </OpenDemoAccount>
  </s:Body>
</s:Envelope>
```

Response:

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Header />
  <s:Body>
    <OpenDemoAccountResponse xmlns="http://client-api.instaforex.com/OpenAccount/">
      <OpenDemoAccountResult xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
        <Login>60111111</Login>
      </OpenDemoAccountResult>
    </OpenDemoAccountResponse>
  </s:Body>
</s:Envelope>
```

Valid characters in fields Address, City, Email, Name, Phone, State, ZipCode:
Latin and Cyrillic letters, numbers, ".", ",", "+", gaps

1. Interaction with API

1.1 Open a demo account

AccountType – the type of trading account, valid entries are **Standard** and **Eurica**

Address - address of residence. From 6 to 127 characters

City - city of residence. From 2 to 31 characters

Country - country of residence. Valid entries look in CountryList.csv

Email – customer email. Up to 47 characters

Ip – client's IP address (IPv4). Allowed values from 0 up to 255, divided by dots

Leverage – account leverage. Valid entries: 1, 10, 20, 50, 100, 200, 300, 500

Name – customer name. From 6 to 127 characters

Phone – telephone number. From 6 to 31 characters

State – region of residence. From 2 to 31 characters

ZipCode – residential post code. From 5 to 15 characters

InitialDeposit – initial deposit. From 100 to 5,000

InvestorPassword –password of investor. From 7 to 15 characters. Be sure to enter at least one letter and at least one digit

Password –password of trader. From 7 to 15 characters. Be sure to enter at least one letter and at least one digit

Login – number of newly opened account

1. Interaction with API

1.2 Open a trading account

Here is an example of the request to open a trading account and the response in case of successful operation. All the information about the account will be sent to the specified email: investor password, trader password, PIN code, and phone password. If necessary, data can be changed in Client Cabinet.

Entry point:

<http://client-api.instaforex.com/soapservices/OpenAccount.svc>

Request:

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Header>
    <Action s:mustUnderstand="1"
xmlns="http://schemas.microsoft.com/ws/2005/05/addressing/none">http://client-api.
instaforex.com/OpenAccount/IOpenAccountService/OpenRealAccount</Action>
  </s:Header>
  <s:Body>
    <OpenRealAccount xmlns="http://client-api.instaforex.com/OpenAccount/">
      <realAccount xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
        <AccountType>Standard</AccountType>
        <Address>Real Address</Address>
        <City>Real City</City>
        <Country>Indonesia</Country>
        <Email>real@real.real</Email>
        <Ip>ipv4</Ip>
        <Leverage>100</Leverage>
        <Name>Real account</Name>
        <Phone>1000000000</Phone>
        <State>Real state</State>
        <ZipCode>00000</ZipCode>
        <AffiliateCode>F</AffiliateCode>
        <Currency>USD</Currency>
        <DateOfBirth>1992-12-27T15:53:41.4833984Z</DateOfBirth>
        <MobilePhoneOs>Android</MobilePhoneOs>
        <NotificationLanguage>En</NotificationLanguage>
        <SwapFree>false</SwapFree>
      </realAccount>
    </OpenRealAccount>
  </s:Body>
</s:Envelope>
```

Response:

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Header />
  <s:Body>
    <OpenRealAccountResponse xmlns="http://client-api.instaforex.com/OpenAccount/">
      <OpenRealAccountResult xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
        <Login>123456</Login>
      </OpenRealAccountResult>
    </OpenRealAccountResponse>
  </s:Body>
</s:Envelope>
```

Valid characters in fields Address, City, Email, Name, Phone, State, ZipCode:

Latin and Cyrillic letters, numbers, ".", ",", "+", gaps

1. Interaction with API

1.2 Open a trading account

AccountType – the type of trading account, valid entries are **Standard** and **Eurica**

Address - address of residence. From 6 to 127 characters

City - city of residence. From 2 to 31 characters

Country - country of residence. Valid entries look in CountryList.csv

Email – customer email. Up to 47 characters

Ip – client's IP address (IPv4). Allowed values from 0 up to 255, divided by dots

Leverage – account leverage. Valid entries: 1, 10, 20, 50, 100, 200, 300, 500

Name – customer name. From 6 to 127 characters

Phone – telephone number. From 6 to 31 characters

State – region of residence. From 2 to 31 characters

ZipCode – residential post code. From 5 to 15 characters

InitialDeposit – initial deposit. From 100 to 5,000

InvestorPassword –password of investor. From 7 to 15 characters. Be sure to enter at least one letter and at least one digit

Password –password of trader. From 7 to 15 characters. Be sure to enter at least one letter and at least one digit

Login – number of newly opened account

2. Practical use

2.1 C#

```
public static void OpenDemoAccount()
{
    var sc = new OpenAccountServiceClient();
    try
    {
        var newDemoAccount = new DemoAccount
        {
            AccountType = AccountType.Standard,
            Address = "Demo Address",
            City = "Demo City",
            Country = "Poland",
            Email = "demo@demo.demo",
            Ip = "ipv4",
            InitialDeposit = 1000,
            InvestorPassword = "qwe123qwe",
            Password = "123qwe123",
            Name = "Demo account",
            Leverage = 100,
            Phone = "1000000000",
            State = "Demo state",
            ZipCode = "00000"
        };
        var newOpenAccountLogin = sc.OpenDemoAccount(newDemoAccount);
        sc.Close();
    }
    finally
    {
        if(sc.State != CommunicationState.Closed)
            sc.Abort();
    }
}

public static void OpenRealAccount()
{
    var sc = new OpenAccountServiceClient();
    try
    {
        var newRealAccount = new RealAccount
        {
            AccountType = AccountType.Standard,
            Address = "Real Address",
            City = "Real City",
            Country = "Indonesia",
            Email = "real@real.real",
            Ip = "ipv4",
            Name = "Real account",
            Leverage = 100,
            Phone = "1000000000",
            State = "Real state",
            ZipCode = "00000",
            AffiliateCode = "F",
            Currency = AccountCurrency.USD,
            MobilePhoneOs = "Android",
            SwapFree = false,
            DateOfBirth = DateTime.UtcNow.AddYears(-20),
        };
    }
}
```


2. Примеры использования

2.1 C#

```
        NotificationLanguage = Language.En
    };
    var newOpenAccountLogin = sc.OpenRealAccount(newRealAccount);
    sc.Close();
}
finally
{
    if (sc.State != CommunicationState.Closed)
        sc.Abort();
}
}
```