

moneta | ru

MONETA.MerchantAPI

Содержание

Глава 1. Введение.....	7
Протоколы и стандарты интерфейса.....	7
SOAP.....	7
JSON.....	8
Глава 2. Безопасность.....	9
Передача данных по SSL.....	9
Идентификатор пользователя и пароль.....	9
Использование идентификатора пользователя.....	9
Клиентский сертификат.....	10
Получение клиентского сертификата.....	11
Глава 3. Обработка ошибок.....	12
Описание работы с ошибками.....	12
Коды ошибок.....	13
Глава 4. Описание Merchant API.....	27
Основные типы данных и форматы.....	27
Описание глобальных типов.....	28
AbstractAttributeObject Complex Type.....	28
AccountAccessInfo Complex Type.....	28
AccountId Simple Type.....	28
AccountInfo Complex Type.....	29
AccountPaymentPasswordType Simple Type.....	32
AccountRelation Complex Type.....	32
AccountStatement Complex Type.....	33
AccountStatementRecord Complex Type.....	35
AccountStatus Simple Type.....	36
AccountType Simple Type.....	36
AsyncStatus Simple Type.....	37
AuthoriseTransactionBatchRequestType Complex Type.....	37
AuthoriseTransactionRequestType Complex Type.....	37
BankAccount Complex Type.....	38
CTID Simple Type.....	38
CancelTransactionBatchRequestType Complex Type.....	39
CancelTransactionBatchResponseType Complex Type.....	39
CancelTransactionRequestType Complex Type.....	40
CancelTransactionResponseType Complex Type.....	40
CommonOperationTemplateParameters Complex Type.....	41
ConfirmTransactionBatchRequestType Complex Type.....	41
ConfirmTransactionRequestType Complex Type.....	41
Contract Complex Type.....	43
Currency Simple Type.....	43
Description Simple Type.....	44
DirectDebitOperationTemplateParameters Complex Type.....	44
Document Complex Type.....	44
DocumentType Simple Type.....	46
Email Simple Type.....	46
Entity Complex Type.....	46
EntityBatchRequestType Complex Type.....	47

Fee Simple Type.....	47
File Complex Type.....	47
ForecastTransactionResponseType Complex Type.....	48
InfoTariff Complex Type.....	50
InfoUrl Complex Type.....	50
InvoiceBatchRequestType Complex Type.....	51
InvoiceRequestType Complex Type.....	51
KeyValueApprovedAttribute Complex Type.....	53
KeyValueAttribute Complex Type.....	54
LegalInformation Complex Type.....	54
Money Simple Type.....	54
OperationAmountType Simple Type.....	55
OperationInfo Complex Type.....	55
OperationInfoBatchResponseType Complex Type.....	56
OperationInfoList Complex Type.....	57
OperationStatus Simple Type.....	57
OperationStatusState Simple Type.....	58
OperationTemplate Complex Type.....	58
OperationTemplateAmount Complex Type.....	61
OperationTemplateAmountInfo Complex Type.....	62
OperationTemplateAmountInfoType Simple Type.....	63
OperationTemplateAmountRange Complex Type.....	63
OperationTemplateAmountRest Complex Type.....	64
OperationTemplateReminderInfo Complex Type.....	64
OperationTemplateTimeInfo Complex Type.....	65
OperationTemplateTimeInfoType Simple Type.....	65
OperationTemplateType Simple Type.....	66
OperationTypeCategory Simple Type.....	66
Pager Complex Type.....	67
Password Simple Type.....	68
PaymentBatchRequestType Complex Type.....	68
PaymentPassword Complex Type.....	68
PaymentRequestType Complex Type.....	69
PaymentSystemInfoComplexType Complex Type.....	70
PersonalInformation Complex Type.....	72
Profile Complex Type.....	72
ProfileNotification Complex Type.....	74
ProfileNotificationFlag Complex Type.....	75
ProfileNotificationFlagType Simple Type.....	76
ProfileNotificationSelection Complex Type.....	76
ProfileNotificationType Simple Type.....	76
ProfileType Simple Type.....	76
ReferenceData Complex Type.....	77
RegularOperationTemplateParameters Complex Type.....	78
Report Complex Type.....	79
ReportInstance Complex Type.....	80
TransactionBatchRequestType Complex Type.....	81
TransactionBatchResponseType Complex Type.....	81
TransactionRequestType Complex Type.....	82
TransactionResponseType Complex Type.....	84
VerifyTransactionResponseType Complex Type.....	84
VerifyTransferResponseType Complex Type.....	85
Version Simple Type.....	86
Финансовые методы.....	86
AuthoriseTransactionBatch Endpoint.....	86
AuthoriseTransaction Endpoint.....	87
CancelTransactionBatch Endpoint.....	88
CancelTransaction Endpoint.....	88
ConfirmTransactionBatch Endpoint.....	89
ConfirmTransaction Endpoint.....	89

FindLastOperationsList Endpoint.....	90
FindOperationsListByCTID Endpoint.....	91
FindOperationsList Endpoint.....	92
ForecastTransaction Endpoint.....	95
GetAccountPaymentPasswordChallenge Endpoint.....	96
GetOperationDetailsById Endpoint.....	97
InvoiceBatch Endpoint.....	97
Invoice Endpoint.....	98
Payment Endpoint.....	98
PaymentBatch Endpoint.....	99
Refund Endpoint.....	100
SecureData Endpoint.....	101
TransferBatch Endpoint.....	102
Transfer Endpoint.....	103
VerifyPayment Endpoint.....	103
VerifyPaymentBatch Endpoint.....	104
VerifyTransaction Endpoint.....	104
VerifyTransfer Endpoint.....	106
PaymentSystemInfo Endpoint.....	107
Шаблоны операций.....	107
CreateOperationTemplate Endpoint.....	107
EditOperationTemplate Endpoint.....	108
FindOperationTemplates Endpoint.....	109
DeleteOperationTemplate Endpoint.....	111
Профайл пользователя.....	111
FindProfileInfoByAccountId Endpoint.....	111
EditProfile Endpoint.....	112
FindProfileInfo Endpoint.....	113
GetProfileInfo Endpoint.....	116
CreateProfile Endpoint.....	117
Управление счетами.....	118
FindAccountByAlias Endpoint.....	118
FindAccountById Endpoint.....	118
CreateAccount Endpoint.....	119
EditAccount Endpoint.....	122
FindAccountsList Endpoint.....	125
BlockAccount Endpoint.....	126
UnblockAccount Endpoint.....	127
Документы, юридические реквизиты, договора.....	128
CreateProfileDocument Endpoint.....	128
EditProfileDocument Endpoint.....	129
FindContracts Endpoint.....	130
FindLegalInformation Endpoint.....	131
FindProfileDocumentFiles Endpoint.....	131
FindProfileDocuments Endpoint.....	132
UploadProfileDocumentFile Endpoint.....	133
Банковские реквизиты.....	133
FindBankAccounts Endpoint.....	134
EditBankAccount Endpoint.....	134
CreateBankAccount Endpoint.....	135
Делегированный доступ к счетам.....	136
FindAccountRelations Endpoint.....	136
GetAccountRelation Endpoint.....	137
SaveAccountRelation Endpoint.....	138
DeleteAccountRelation Endpoint.....	138
Отчеты.....	139
GetTurnoverList Endpoint.....	139
GetFinancialFlowsList Endpoint.....	143
FindReports Endpoint.....	146
Идентификация пользователя.....	147

GetPersonificationCode Endpoint.....	148
VerifyPersonificationCode Endpoint.....	149
ConfirmPersonification Endpoint.....	150
SimplifiedIdentification Endpoint.....	151
Проверка телефона пользователя.....	153
ApprovePhoneApplyCode Endpoint.....	153
ApprovePhoneSendConfirmation Endpoint.....	154
Асинхронные запросы.....	156
Async Endpoint.....	156

Глава 5. Примеры использования.....157

Простое пополнение и списание.....	157
Оплата товаров и услуг.....	159
Оплата товаров и услуг.....	159
Однофазный платеж.....	159
Push-платежи.....	162
Двухфазный платеж.....	166
Выставление счета (инвойс).....	172
Оплата банковской картой.....	176
Возврат средств.....	181
Вывод средств.....	183
Введение.....	183
Вывод средств со счета в Moneta.Ru.....	183
Дополнительные параметры метода Payment.....	185
Получение параметров для вывода средств.....	187
Примеры вывода средств.....	187
Запросы в пакетном режиме.....	196
VerifyPaymentBatch.....	196
История платежей.....	199
GetOperationDetailsById - детали операции.....	199
FindOperationsList - список операций.....	202
FindLastOperationsList - последние операции.....	204
FindOperationsListByCTID - детали операции по внешнему идентификатору операции.....	206
GetTurnoverList - итоги по месяцам.....	207
GetTurnoverList (AsyncRequest) - итоги по месяцам.....	209
GetFinancialFlowsList - финансовые потоки.....	213
GetFinancialFlowsList (AsyncRequest) - финансовые потоки.....	215
Шаблоны операций.....	218
CreateOperationTemplate.....	218
EditOperationTemplate.....	231
FindOperationTemplates.....	233
Шаблоны операций на вывод средств.....	246
DeleteOperationTemplate.....	253
Работа с профайлом пользователя.....	255
Работа со счетами пользователя.....	260
Создание счета.....	269
Редактирование счета.....	270
Чтение счета.....	272
Счет с SMS платежным паролем.....	276
Блокировка/разблокирование счета.....	278
Информация о доступных платежных системах.....	280
Работа с документами пользователя.....	282
Работа с банковскими реквизитами.....	287
Работа с отчетами.....	292
Подтверждение сотового телефона.....	293
SimplifiedIdentificationRequest (AsyncRequest) - упрощенная идентификация.....	295

Глава 6. Примеры кода..... 305

SOAP запрос с помощью CURL.....	305
JSON запрос с помощью CURL.....	306
Пример Java API для XML Web Services (JAX-WS).....	308
Пример Java (SAAJ) для SOAP with Attachments.....	309
SOAP пример на C#.....	312
SOAP пример на PHP.....	313
SOAP пример на Python.....	314

Глава

1

Введение

- *Протоколы и стандарты интерфейса*

Протоколы и стандарты интерфейса

Взаимодействие между Moneta.Ru и внешней системой может быть построено на основе SOAP протокола или JSON формата и специфицировано в WSDL документе.

SOAP

WSDL описывает сервисы как набор портов (endpoint), доступных по сети. Порт ассоциирует сетевой адрес (network address) со связыванием данных (binding), коллекция портов описывает сервис. Сообщения являются абстрактными описаниями данных для обмена информацией, типы портов являются абстрактными коллекциями поддерживаемых действий. Конкретный протокол и спецификации формата данных для определенного типа порта составляют связывание данных, в котором действия и сообщения связаны с конкретным сетевым протоколом и форматом сообщения. Таким образом, WSDL описывает публичный интерфейс web-сервиса.

WSDL используется в комбинации с SOAP и XML Schema для предоставления web-сервисов через Internet. Клиентская система, связывающаяся с web-сервисом, может проанализировать WSDL, чтобы определить набор доступных на сервере функций. Все использующиеся особые типы данных описаны в WSDL в форме XML Schema. Клиент использует SOAP для вызова любой из функций, перечисленных в WSDL документе.

SOAP — это протокол обмена XML сообщениями по сети с использованием HTTPS. SOAP является базовым слоем стека протоколов web-сервисов и обеспечивает основные механизмы обмена сообщениями.

WSDL URL: <https://service.moneta.ru/services.wsdl>

Service URL: <https://service.moneta.ru/services>

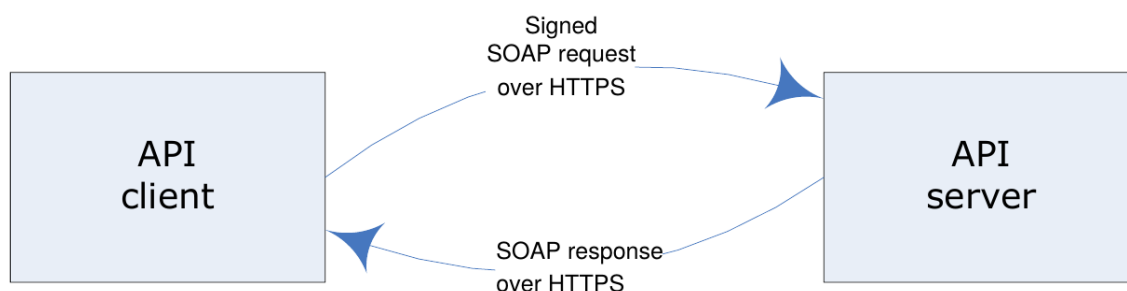
WSDL URL (x509): <https://service.moneta.ru:8443/services/x509.wsdl>

Service URL (x509): <https://service.moneta.ru:8443/services/x509>

HTTP method: POST

Content-type: text/xml;charset=UTF-8

SOAPAction: ""



JSON

JSON - текстовый формат обмена данными.

Описание используемых объектов:

WSDL URL: <https://service.moneta.ru/services.wsdl>

Вызов методов:

Service URL: <https://service.moneta.ru/services>

Service URL (x509): <https://service.moneta.ru:8443/services/x509>

При вызове методов следует указать:

HTTP method: POST

Content-type: application/json;charset=UTF-8

Глава

2

Безопасность

- [Передача данных по SSL](#)
- [Идентификатор пользователя и пароль](#)
- [Клиентский сертификат](#)

Доступ к использованию сервиса MONETA.MerchantAPI предоставляется только авторизованным пользователям Moneta.Ru.

Два основных уровня обеспечения безопасности:

- Передача данных с использованием Secure Sockets Layer (SSL)
- Идентификатор пользователя и пароль для доступа к API или клиентский сертификат

Невыполнение любого из условий запрещает доступ к MONETA.MerchantAPI.

Передача данных по SSL

Все данные должны передаваться по протоколу Secure Hyper Text Transport Protocol (HTTPS), основанному на протоколе Secure Sockets Layer (SSL).

Идентификатор пользователя и пароль

Moneta.Ru авторизует каждый запрос партнера при помощи идентификатора пользователя и пароля в заголовке запроса.

Подробную информацию по авторизации в SOAP протоколе можно найти: [Web Services Security UsernameToken Profile 1.1](#).

Если для взаимодействия с Moneta.Ru Вы используете JSON формат, то примеры, как указать идентификатор пользователя и пароль будут показаны ниже.

Если Вы используете идентификатор пользователя и пароль в заголовке запроса, то для работы с MONETA.MerchantAPI используйте следующие URL:

- WSDL URL: <https://service.moneta.ru/services.wsdl>
- Service URL: <https://service.moneta.ru/services>

Использование идентификатора пользователя

Идентификатор пользователя и пароль - это логин и пароль, которые Вы используете для входа в личный кабинет Moneta.Ru.

Ниже приведен пример с фрагментом данных, требующихся для авторизации SOAP запросов:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Header>
    <wsse:Security soapenv:mustUnderstand="1"
      xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-
        200401-wss-wssecurity-secext-1.0.xsd">
      <wsse:UsernameToken wsu:Id="UsernameToken"
        xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-
          200401-wss-wssecurity-utility-1.0.xsd">
        <wsse:Username>MERCHANT'S USERNAME</wsse:Username>
        <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-
          200401-wss-username-token-profile-
```

```

        1.0#PasswordText">PLAIN PASSWORD</wsse:Password>
      </wsse:UsernameToken>
    </wsse:Security>
  </soapenv:Header>
  <soapenv:Body>
    ...
  </soapenv:Body>
</soapenv:Envelope>

```

В случае неуспешной авторизации клиенту возвращается HTTP status равный 500 и ошибка безопасности SOAP:

```

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <SOAP-ENV:Fault>
      <faultcode>SOAP-ENV:Client</faultcode>
      <faultstring xml:lang="en">com.sun.xml.wss.impl.WssSoapFaultException:
        Authentication of Username Password Token Failed; nested exception
        is com.sun.xml.wss.XWSecurityException:
        com.sun.xml.wss.impl.WssSoapFaultException:
        Authentication of Username Password Token Failed</faultstring>
      <detail>
        <messages:faultDetail xmlns:messages="http://www.moneta.ru/schemas/
          messages.xsd">300.1</messages:faultDetail>
      </detail>
    </SOAP-ENV:Fault>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

Ниже приведен пример с фрагментом данных, требующихся для авторизации JSON запросов:

```

{"Envelope": {
  "Header": {
    "Security": {
      "UsernameToken": {
        "Username": "MERCHANT'S USERNAME",
        "Password": "PLAIN PASSWORD"
      }
    }
  },
  "Body": {
    ...
  }
}}

```

В случае неуспешной авторизации клиенту возвращается HTTP status равный 400 и ошибка безопасности JSON:

```

{"Envelope": {
  "Body": {
    "fault": {
      "detail": {
        "faultDetail": "300.1"
      },
      "faultcode": "Client",
      "faultstring": "wrong credentials"
    }
  }
}}

```

Клиентский сертификат

Вы можете получить клиентский сертификат для авторизации Ваших запросов в MONETA.MerchantAPI.

Если Вы используете клиентский сертификат, то для работы с MONETA.MerchantAPI используйте следующие URL:

- WSDL URL: <https://service.moneta.ru:8443/services/x509.wsdl>
- Service URL: <https://service.moneta.ru:8443/services/x509>

Получение клиентского сертификата

Если Вы хотите использовать клиентский сертификат для работы с MONETA.MerchantAPI, Вы должны получить клиентский сертификат в Moneta.Ru.

1. Для создания запроса на сертификат выполните следующие шаги:

- a) Установите OpenSSL.
- b) Создайте запрос на сертификат следующей командой:

```
openssl req -new -newkey rsa:2048 -sha512 -nodes -out request.txt -keyout private.key
```

OpenSSL создаст запрос на сертификат в файле request.txt. Используйте этот запрос на сертификат для получения клиентского сертификата в Moneta.Ru.

2. Получите клиентский сертификат в Moneta.Ru. Для этого выполните следующие шаги:

- a) Зайдите в личный кабинет на сайте Moneta.Ru.
- b) Перейдите на страницу Безопасность.
Перейдите **Мой счет** > **Безопасность**.
- c) Нажмите **добавить** Сертификат.
- d) Скопируйте запрос на сертификат из файла request.txt, который Вы создали на Шаге 1, в поле **Запрос на сертификат** и нажмите **Продолжить**.

Важно: Скопируйте все содержимое файла, включая строки BEGIN CERTIFICATE REQUEST и END CERTIFICATE REQUEST.

- e) Проверьте информацию указанную в сертификате и нажмите **Сохранить**.
Moneta.Ru создаст клиентский сертификат.
- f) Нажмите ссылку **Скачайте сертификат**.
- g) Вы можете создать хранилище ключей в формате PKCS12. Выполните следующую команду:

```
openssl pkcs12 -export -clcerts -in certificate.pem -inkey private.key -out wallet_name.p12 -name user_name
```

- h) Вы можете импортировать хранилище ключей в формате PKCS12 в Ваш браузер, чтобы заходить в личный кабинет Moneta.Ru без использования логина и пароля.

Глава

3

Обработка ошибок

- [Описание работы с ошибками](#)
- [Коды ошибок](#)

Описание работы с ошибками

Если во время обработки запроса произошла ошибка, то Moneta.Ru вернет HTTP status 500 (в случае SOAP) или HTTP status 400 (в случае JSON) и ответ будет содержать элемент Fault с описанием ошибки.

Следующий пример показывает как выглядит SOAP ответ, если произошла ошибка во время обработки запроса:

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <SOAP-ENV:Fault>
      <faultcode>Fault code</faultcode>
      <faultstring xml:lang="en">Error message</faultstring>
      <detail>
        <messages:faultDetail xmlns:messages="http://www.moneta.ru/schemas/messages.xsd">Error code</messages:faultDetail>
      </detail>
    </SOAP-ENV:Fault>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Следующий пример показывает как выглядит JSON ответ, если произошла ошибка во время обработки запроса:

```
{
  "Envelope": {
    "Body": {
      "fault": {
        "detail": {
          "faultDetail": "ERROR CODE"
        },
        "faultcode": "FAULT CODE",
        "faultstring": "ERROR MESSAGE"
      }
    }
  }
}
```

Каждый ответ с ошибкой содержит следующие элементы:

- **faultcode.** В случае SOAP состоит из префикса и имени. Префикс всегда равен SOAP-ENV. Следующая таблица описывает возможные значения имени:

Имя	Описание
Client	Запрос содержит неверные данные.
Server	Во время обработки запроса на сервере произошла ошибка.

Имя	Описание
VersionMismatch	Неверный namespace для объекта SOAPEnvelope.
MustUnderstand	Атрибут mustUnderstand в дочерних элементах SOAPHeader равен true, но Moneta.Ru либо не поняла, либо проигнорировала этот атрибут.

- **faultstring.** Текстовое описание ошибки.
- **faultDetail.** Код ошибки. Полный список кодов ошибок: [Коды ошибок](#).

Следующий пример показывает как выглядит SOAP ответ, если в запросе были переданы неверные данные:

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <SOAP-ENV:Fault>
      <faultcode>SOAP-ENV:Client</faultcode>
      <faultstring xml:lang="ru">Transaction is not found</faultstring>
      <detail>
        <messages:faultDetail xmlns:messages="http://www.moneta.ru/schemas/messages.xsd">500.1.5</messages:faultDetail>
      </detail>
    </SOAP-ENV:Fault>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Следующий пример показывает как выглядит JSON ответ, если в запросе были переданы неверные данные:

```
{
  "Envelope": {
    "Body": {
      "fault": {
        "detail": {
          "faultDetail": "500.1.5"
        },
        "faultcode": "Client",
        "faultstring": "Transaction is not found"
      }
    }
  }
}
```

Коды ошибок

Если во время работы сервиса произойдет ошибка, то ответ будет содержать элемент detail в котором будет элемент faultDetail. Для дополнительной информации смотрите http://www.w3.org/TR/2000/NOTE-SOAP-20000508/#_Toc478383507.

Элемент faultDetail содержит код ошибки.

Например, в SOAP ответ выглядит так:

```
<Fault>
  <faultcode>Client</faultcode>
  <faultstring>Операция не найдена</faultstring>
  <detail>
    <faultDetail>500.1.5</faultDetail>
  </detail>
</Fault>
```

В JSON ответ выглядит так:

```
{
  "fault": {
    "detail": {
```

```

        "faultDetail": "500.1.5"
      },
      "faultcode": "Client",
      "faultstring": "Операция не найдена"
    }
  }
}

```

Коды ошибок приведены ниже:

100	Сервис временно недоступен.
100.1	Указанные данные уже существуют.
100.2	Вы работаете с устаревшими данными.
100.3	Ошибки в профиле данной учетной единицы: <i>unitId</i> .
100.4	Ошибки в счетах данной учетной единицы: <i>unitId</i> .
100.5	Ошибки в структуре данной учетной единицы: <i>unitId</i> .
200	Ошибка валидации XSD схемы.
300.1	Неверное имя пользователя и пароль.
300.2	В доступе отказано.
300.3	В доступе отказано.
300.4.1	У Вас нет доступа к указанному шаблону.
300.4.2	Доступ к заказу запрещён.
300.4.3	Доступ к операции запрещён.
300.4.4	Данный счет нельзя редактировать.
300.4.5	Запрещено создавать профиль с указанным типом.
300.5	Пользователь не найден.
300.6	Нельзя изменять подтвержденные данные. Одно из значений, которое Вы пытаетесь изменить, имеет статус 'подтвержденное'.
400.1.1	Проведение операции невозможно.
400.1.2	Операция ожидает подтверждения.
400.1.3	Операция в процессе обработки.
400.1.4	Операция в обработке.
400.1.5	Оплата отменена.
400.1.6	Данная операция уже оплачена.
400.1.7	Проведение операции невозможно.
400.1.8.1	Невозможно сделать возврат так как операция не завершена.
400.1.8.2	Невозможно сделать возврат операции. Получатель не принимает платежи.
400.1.8.3	Невозможно сделать возврат операции.
400.1.8.4	Невозможно сделать возврат операции.
400.1.9	Ошибка при взаимодействии с внешней системой получателя платежа.
400.1.10	Перечисление на заданный счёт получателя невозможно либо указана неверная сумма.

400.1.11	Счет заблокирован.
400.1.12	Недостаточно средств для перевода.
400.1.13	Нельзя отменить данную операцию.
400.1.14	Последний платежный пароль зарезервирован для получения новой последовательности платежных паролей.
400.1.15	Закончился срок действия кода протекции.
400.1.16	Сумма в запросе не должна быть больше суммы в операции.
400.1.17	Ошибка при взаимодействии с внешней системой плательщика.
400.1.18	Сумма в заказе не совпадает с проверочной суммой.
400.1.19	Операция с таким внешним идентификатором находится в обработке или выполнена. Проведение еще одной операции с таким внешним идентификатором невозможно.
400.1.20	Нет прав на проведение операции.
400.1.21	Операция заморожена.
400.1.22	Выставленный счет к оплате недействителен.
400.1.23	Срок действия платежного пароля истек.
400.1.24	Рекуррентный платеж невозможен.
400.1.25	Повторная попытка оплаты с другим номером карты невозможна, транзакция отменена, вы можете оплатить еще раз, будет создана другая транзакция.
400.2	Операция поставлена в обработку.
400.2.1	Операция поставлена в очередь на обработку.
400.2.1.1	Проведение операции невозможно.
400.2.2	Операция поставлена в очередь на обработку.
400.2.2.1	Проведение операции невозможно.
400.2.3	Операция поставлена в очередь на обработку.
400.2.3.1	Проведение операции невозможно.
400.2.4	Превышены ограничения по сумме перечислений.
400.2.4.1	Превышены ограничения по сумме перечислений.
400.2.5	Операция поставлена в очередь на обработку.
400.2.5.1	Проведение операции невозможно.
400.2.6	Превышены ограничения по сумме перечислений.
400.2.6.1	Превышены ограничения по сумме перечислений.
400.2.7	Операция поставлена в очередь на обработку.
400.2.7.1	Проведение операции невозможно.
400.2.7.2	Недопустимое назначение платежа.
400.2.7.3	Недопустимое описание операции.

400.2.7.4	Оплата возможна только с российских карт с поддержкой 3DSecure.
400.2.7.5	Оплата возможна только с российских или 3DSecure карт.
400.2.7.6	Оплата возможна только с карт с поддержкой 3DSecure.
400.2.7.7	Данную операцию могут совершать только идентифицированные пользователи Монета.Ру. Для прохождения упрощенной идентификации укажите в личном кабинете: ФИО, паспортные данные, ИНН, СНИЛС, номер мобильного телефона.
400.2.7.8	Оплата с данной карты невозможна. Действует ограничение по стране, банк которой выпустил данную карту.
400.2.8.1	Ошибка во время шифрования данных.
400.2.8.2	Ошибка во время расшифровки данных.
500	Ошибки при валидации данных.
500.1	Обработка невозможна. Не найдены необходимые объекты.
500.1.1	Указанный счет не существует.
500.1.2	Не найден счет с псевдонимом: <i>alias</i> .
500.1.3	Не найден счет с номером: " <i>accountId</i> ".
500.1.4	Счет не найден.
500.1.5	Операция не найдена.
500.1.6	Не найден счет плательщика.
500.1.7	Не найден счет получателя.
500.1.8	Документ не найден.
500.1.8.1	Недопустимое имя файла. Поддерживаемые форматы: jpg,png,bmp,pdf,doc,odt.
500.1.9	Данные пользователя не найдены.
500.1.10	Неверный ID заказа.
500.1.11	Неверный ID операции.
500.1.12	Шаблон с указанным ID не найден.
500.1.13	В указанном шаблоне нет таких параметров.
500.1.14	Пользователь не найден.
500.1.15	Неверный идентификатор структуры.
500.1.16	Значение параметра SOURCEACCOUNTID должно быть равно номеру счета, к которому есть доступ.
500.1.17	Значение параметра SOURCEACCOUNTID должно быть равно либо {TARGETACCOUNTID}, либо номеру счета, к которому есть доступ.
500.1.18	Данные находятся в архиве и не могут быть получены в данный момент.

500.1.18.1	Данные находятся в архиве и не могут быть получены в данный момент. Укажите дату позднее <i>dd.mm.yyyy</i> .
500.1.19	Банковские реквизиты не найдены.
500.1.20	Не указан ID банковских реквизитов.
500.1.21	У пользователя не установлен Публичный идентификатор.
500.1.22	Данные Secure Token не найдены.
500.1.23	Обработка невозможна. Не найдены необходимые свойства объекта.
500.2	Неверное значение передаваемого параметра.
500.2.1	Элемент <i>element_name</i> обязателен для заполнения.
500.2.2	Обязательное поле.
500.2.3	The <i>element_name</i> element is required.
500.2.4	The <i>element_name</i> element must be null.
500.2.5	The <i>element_name</i> element must be null or empty.
500.2.6	The <i>element_name</i> and <i>element_name</i> values do not match in the request and response.
500.2.7	Неверный формат.
500.2.7.1	Неверный формат логической переменной: *.
500.2.7.2	Неверный формат числа: *.
500.3.1.1	Неверный платежный пароль.
500.3.1.2	Указанная сумма некорректна.
500.3.1.3	Счет плательщика задан неправильно.
500.3.1.4	Счет получателя задан неправильно.
500.3.1.5	Сумма плательщика задана неправильно.
500.3.1.6	Сумма получателя задана неправильно.
500.3.1.7	Сумма не должна превышать <i>amountcurrency</i> .
500.3.1.8	Сумма должна быть больше <i>amountcurrency</i> .
500.3.1.9	Невозможно определить - является ли переданная сумма суммой плательщика или получателя.
500.3.1.10	Нельзя использовать поле <i>transactionId</i> , если <i>transactional</i> = true.
500.3.1.11	Введен неверный код протекции.
500.3.1.12	<i>TransactionId</i> не сходится с <i>OperationInfo.ID</i> .
500.3.1.13	Сумма должна быть больше нуля.
500.3.1.17	Необходимо указать сумму. Сумма должна быть положительным числом.
500.3.1.18	Данный идентификатор операции уже существует.
500.3.1.19	Слишком много транзакций.
500.3.1.20	В пакетном режиме нельзя проводить операции с внешними платежными системами.
500.3.1.21	Не задана сумма заказа.

500.3.1.22	Не задан внутренний идентификатор заказа.
500.3.1.23	В операции указан другой внешний идентификатор плательщика.
500.3.1.24	В операции указан другой внешний идентификатор получателя.
500.3.1.25	Необходимо ввести номер карты.
500.3.1.26	Необходимо ввести срок действия карты.
500.3.1.27	Необходимо ввести код CVV2/CVC2.
500.3.1.28	Неверный формат номера карты.
500.3.1.29	Неверный формат срока действия карты (ММ/YYYY).
500.3.1.30	Неверный формат кода CVV2/CVC2 (3 цифры).
500.3.1.31	В одном пакетном запросе нельзя выполнить проведение новой операции и подтверждение существующей операции.
500.3.1.32	Нельзя установить код CVV2/CVC2 при создании инвойса.
500.3.1.33	Неверный формат имени владельца карты (латинские буквы, имя и фамилия через пробел).
500.3.1.34	Данный запрос не поддерживает работу в одной транзакции (transactional = true). Используйте transactional = false.
500.3.1.35	Необходимо ввести владельца карты.
500.3.1.36	Для работы с данными полями необходима PCI DSS сертификация.
500.3.1.37	Карта с истекшим сроком действия.
500.3.1.38	Слишком длинное имя владельца карты (максимум кол-во символов: *).
500.3.2.1	Неверный предыдущий платёжный пароль, поэтому вы не можете сохранить новый пароль.
500.3.2.2	Не указан псевдоним счета.
500.3.2.3	Неверно задан тип удостоверения личности.
500.3.2.4	Неизвестный тип профиля.
500.3.2.5	Неверная дата рождения.
500.3.2.6	Неверное значение пола.
500.3.2.7	Неверный формат номера телефона.
500.3.2.8	Неверный формат e-mail.
500.3.2.9	Длина платёжного пароля меньше 5 символов.
500.3.2.10	Платёжный пароль больше 32 символов.
500.3.2.11	Платёжный пароль должен состоять только из цифр.
500.3.2.12	Неверно задан URL.
500.3.2.13	URL должен начинаться с http[s]:// или template://.

500.3.2.14	Псевдоним счета должен быть уникальным. У вас уже есть другой счет с таким псевдонимом: <i>accountId</i> .
500.3.2.15	Ошибки при валидации полей документа.
500.3.2.16	Фамилия - обязательное поле.
500.3.2.17	Имя - обязательное поле.
500.3.2.18	Слишком длинный Псевдоним. Максимальная длина 64 символа.
500.3.2.19	Пользователи системы не должны быть моложе 14 лет.
500.3.2.20	Дата рождения должна быть больше 1900 года.
500.3.2.21	Требуется корректный 10-значный или 12-значный ИНН.
500.3.2.21.1	Требуется корректный 10-значный ИНН.
500.3.2.21.2	Требуется корректный 12-значный ИНН.
500.3.2.22	Неверно указана страна для региона.
500.3.2.23	Неверно указан регион для города.
500.3.2.24	Укажите номер телефона в международном формате (например, 71234567890).
500.3.2.25	Неверный адрес сайта. Пример правильного написания: http://www.site.com или http://site.com .
500.3.2.26	Неверно задан email. Пример правильного написания: email@company.ru .
500.3.2.27	Название организации - обязательное поле.
500.3.2.28	ФИО руководителя - обязательное поле.
500.3.2.29	Нельзя создать запрещающее правило для данного пользователя.
500.3.2.30	Неверный идентификатор статуса.
500.3.2.31	Такой псевдоним уже существует.
500.3.2.32	'Баланс меньше' указан неверно.
500.3.2.33	Не установлен ни один из прав доступа.
500.3.2.34	Укажите другой счет в качестве прототипа.
500.3.2.35	Указанный счет ссылается на другой счет-прототип. Укажите другой счет.
500.3.2.36	Нельзя создать счет в данной валюте или с данным типом.
500.3.2.37	Можно использовать только для счетов со статическим платежным паролем.
500.3.2.38	Неверный формат шаблона. Шаблон должен быть в формате: <code>template://template_id?param_name=param_value</code> .
500.3.2.39	Указанный счет имеет другой тип. Укажите другой счет.
500.3.2.40	Нельзя перенести структуру в саму себя.
500.3.2.41	Невозможно провести идентификацию.

500.3.2.42	Неверный код идентификации.
500.3.2.43	Адрес регистрации - обязательное поле.
500.3.2.44	Дата рождения - обязательное поле.
500.3.2.45	Нельзя изменять подтвержденное свойство.
500.3.2.46	Неверно указан СНИЛС.
500.3.2.47	Не указан сотовый телефон.
500.3.2.48	Телефон уже подтвержден.
500.3.2.48.1	Подтверждение телефона уже отменено.
500.3.2.49	Сообщение не содержит подстановки {CODE}.
500.3.2.50	Слишком большая длина sms сообщения.
500.3.2.51	Неверный код подтверждения.
500.3.2.51.1	Неверный код отмены подтверждения.
500.3.2.52.1	Необходим корректный БИК (9 цифр) в поле "*".
500.3.2.52.2	Номер расчётного счёта не корректный в поле "*". Введите правильный номер.
500.3.2.52.3	Указанный счёт предназначен для обслуживания физического лица в поле "*". Нельзя использовать его в целях приёма платежей.
500.3.2.52.4	Введён БИК, соответствующий РКЦ Банка России. В этом случае поле "Корреспондентский счёт" не заполняется.
500.3.2.52.5	Номер корреспондентского счёта не корректный. Введите правильный номер.
500.3.2.52.6	Необходим корректный КБК (число до 20 знаков) в поле "*".
500.3.2.52.7	Необходим корректный ОКТМО (8 или 11 значное число) в поле "*".
500.3.2.52.8	Необходим корректный КПП (9-значное число) в поле "*".
500.3.2.52.9	Укажите SWIFT в корректном формате (8 или 11 символов - заглавные латинские буквы и цифры).
500.3.2.52.10	Укажите IBAN в корректном формате в поле "*" (от 5 до 40 символов - заглавные латинские буквы и цифры).
500.3.2.53	Нельзя выбрать данный тип платежного пароля.
500.3.2.54	Требуемое поле - "element_name".
500.3.2.55	Слишком длинное значение в поле "element_name". Предел: "length".
500.3.2.56	Поле не может быть пустым - "element_name".
500.3.2.57	Неверный формат Secure Token.
500.3.2.58	Закончился срок действия Secure Token.
500.3.2.59.1	Неправильная подпись формы оплаты.
500.3.2.59.2	Подпись на форме оплаты обязательна.
500.3.2.60	Нельзя менять тип учетных единиц у записей данного вида.

500.3.2.61	Неверный тип счета.
500.3.2.62.1	Данный счет нельзя заблокировать, потому что он не является активным.
500.3.2.63.1	Данный счет нельзя разблокировать, потому что он не заблокирован.
500.3.2.63.2	Разблокировать счет невозможно, так как ранее Вами не был указан секретный вопрос. Обратитесь в службу поддержки.
500.3.2.63.3	Разблокировать счет невозможно, так как ранее Вами не был указан ответ на секретный вопрос. Обратитесь в службу поддержки.
500.3.2.63.4	Данный счет не является заблокированным.
500.3.2.63.5	Данный счет не может быть автоматически разблокирован. Обратитесь в службу поддержки.
500.3.2.63.6	Возможность использовать ответ на секретный вопрос заблокирована на один час. Попробуйте позднее или обратитесь в службу поддержки.
500.3.2.63.7	Возможность использовать платежный пароль для активации счета заблокирована на один час. Попробуйте позднее или обратитесь в службу поддержки.
500.3.2.63.8	Неправильный ответ на вопрос.
500.3.2.64.1	Неверно указан Тип интерфейса.
500.3.2.64.2	Неверно указан список платежных систем.
500.3.2.64.3	Неверно задана платежная система по умолчанию.
500.3.2.64.4	Неверно указан HTTP метод.
500.3.2.64.5	Неверно указан Target.
500.3.2.64.6	Превышено допустимое количество символов.
500.3.2.65.1	Неверный формат URL.
500.3.2.65.2	Ошибка установки соединения.
500.3.2.65.3	Срок действия серверного сертификата истек.
500.3.2.65.4	Ошибка проверки центра сертификации серверного сертификата.
500.3.2.65.5	Срок действия серверного сертификата еще не начался.
500.3.3.1	Сумма "от" больше суммы "до".
500.3.3.2	Дата "от" больше даты "до".
500.3.3.3	Максимальный интервал просмотра истории операций составляет 30 дней. Измените дату начала или конца периода просмотра.
500.3.3.4	Сумма не может быть отрицательной.
500.3.3.5	Если задана сумма, то следует указать валюту.
500.3.3.6	Неверно задан тип суммы операции.
500.3.3.7	Неверно задан код валюты.
500.3.3.8	Не указан период.

500.3.3.9	Указан неверный период.
500.3.3.10	Указан неверный тип операции.
500.3.3.11	Период просмотра финансовых потоков не может быть больше 3 месяцев.
500.3.3.12	Дата "начала периода" больше даты "конца периода".
500.3.3.13	Период просмотра Итогов по месяцам с "детальными по дням" не может быть больше месяца.
500.3.3.14	Период просмотра финансовых потоков с "детальными по дням" не может быть больше месяца.
500.3.3.15	Период просмотра операций не может быть больше одного года.
500.3.3.16	Платеж с идентичного счета невозможен.
500.4.1.1	Указанный вами лицевой счет не найден в реестре начислений. Проверьте корректность лицевого счета или обратитесь к поставщику услуг.
500.4.1.2	Некорректный формат поля "*".
500.4.2.1	Не указана сумма.
500.4.2.2	Сумма платежа должна быть в пределах от * до *.
500.4.2.3	Данное начисление уже оплачено.
500.4.2.4	Начислений по данному запросу не обнаружено.
500.4.2.5	Ошибка контрольной суммы, проверьте правильность ввода УИН *.
500.4.2.6	Ошибка поиска начисления.
500.4.4.1	Не найден провайдер товаров или услуг.
500.4.4.2	Данный провайдер отключен.
500.4.4.3	Произошла ошибка в системе ГИС ГМП. Просим Вас повторить запрос позже.
500.4.4.4	Провайдер "*" не поддерживает данный способ вызова.
500.4.4.5	Произошла ошибка в системе расчёта стоимости ОСАГО. Просим Вас повторить запрос позже.
500.5.1	Счёт к списанию и счёт к зачислению не должны совпадать.
500.5.2	Вы не имеете доступ ни к одному из счетов.
500.5.3	Создание шаблона операции невозможно.
500.5.4	Название не должно превышать * символов.
500.5.5	Метки могут содержать слова, цифры, символы "_", "-", пробел.
500.5.6	Превышено допустимое количество символов в одной из меток.
500.5.7	Сумма должна быть больше нуля.
500.5.8	Шаблон не может быть регулярным платежом.
500.5.9	Дата и время выполнения указаны раньше текущего времени.

500.5.10	Дата не соответствует последнему дню месяца.
500.5.11	Дата окончания выполнения указана раньше даты начала.
500.5.12	Дата окончания выполнения указана раньше текущего времени.
500.5.13	Количество часов в уведомлении о статусе платежа должно быть больше нуля.
500.5.14	Неверный тип суммы.
500.5.15	Минимальное значение суммы должно быть положительным числом.
500.5.16	Максимальное значение суммы должно быть больше нуля.
500.5.17	Максимальное значение суммы должно быть больше её минимального значения.
500.5.18	Значение остатка баланса должно быть больше нуля.
500.5.19	Определите, как минимум, одну из границ интервала.
500.5.20	Неверный шаблон.
500.6.1.1	Отсутствуют банковские реквизиты.
500.6.1.2	Отсутствуют подтвержденные банковские реквизиты.
500.6.1.3	Неверные банковские реквизиты.
500.6.1.4	Укажите расчетный счет, на который хотите перевести средства.
500.6.1.5	Укажите наименование банка, в котором находится Ваш расчетный счет.
500.6.1.6	Укажите БИК банка, в котором находится Ваш расчетный счет.
500.6.1.7	Укажите корреспондентский счет банка, в котором находится Ваш расчетный счет.
500.6.1.8	Укажите получателя.
500.6.1.9	Неверный индекс документа. Укажите в корректном формате.
500.6.1.10	Неверный идентификатор плательщика. Укажите в корректном формате.
500.6.1.11	Должен быть указан УИН (индекс документа) или идентификатор плательщика.
500.6.1.12	Укажите ИНН получателя или ИНН банка.
500.6.1.13	Укажите номер договора.
500.6.1.14	Неверный номер расчетного счета. Укажите номер в корректном формате (20 цифр).
500.6.1.15	Перевод на счет 40821* невозможен. Пожалуйста, укажите другой счет получателя.
500.6.1.16	Неверный БИК банка. Укажите БИК в корректном формате (9 цифр).

500.6.1.17	Указанный БИК банка отсутствует в справочнике.
500.6.1.18	Указанные БИК банка и счет не соответствуют друг другу.
500.6.1.19	Неверный корреспондентский счет банка. Укажите счет в корректном формате (20 цифр).
500.6.1.20	Неверный КПП. Укажите в корректном формате (9 цифр).
500.6.1.21	Неверный КБК. Укажите в корректном формате (20 цифр).
500.6.1.22	Неверный ОКТМО. Укажите в корректном формате (2-8 или 11 цифр).
500.6.1.23	параметры КПП, КБК и ОКТМО должны либо присутствовать все сразу, либо отсутствовать.
500.6.1.24	Неверное наименование получателя. Укажите в корректном формате.
500.6.1.25	Неверный ИНН. Укажите ИНН получателя (12 цифр) или ИНН банка (10 цифр).
500.6.1.26	Количество символов в наименовании получателя (*) превышает максимальное (*).
500.6.1.27	Количество символов в поле (*) превышает максимальное (*).
500.6.1.28	Количество символов в назначении платежа (*) превышает максимальное (*) для указанной валюты вывода.
500.6.1.29	Неверный формат назначения платежа.
500.6.1.30	Укажите номер карты.
500.6.1.31	Нельзя использовать латинские буквы.
500.6.1.32	Поле "Наименование получателя" или "Назначение платежа" должно содержать "*".
500.6.1.33	Вывод на данный счет запрещен для анонимных пользователей.
500.6.1.34	Неверный SWIFT банка. Укажите в корректном формате (8 цифр или латинских букв).
500.6.1.35	Неверный IBAN банка. Укажите в корректном формате (5..40 цифр и/или заглавных латинских букв).
500.6.1.36	Указанный SWIFT банка отсутствует в справочнике.
500.6.1.37	Указанный БИК российского банка посредника отсутствует в справочнике.
500.6.1.38	Неверный SWIFT международного банка посредника. Укажите в корректном формате (8 цифр или латинских букв).
500.6.1.39	Указанный SWIFT международного банка посредника отсутствует в справочнике.
500.6.1.40	Укажите номер платежного поручения.
500.6.1.41	Укажите дату платежного поручения.

500.6.1.42	Номер платежного поручения не должен заканчиваться тремя нулями.
500.6.2.1	Укажите номер счета, на который вы хотите перечислить средства.
500.6.2.2	Неверный номер счета. Укажите номер счета в QIWI (11 или 12 цифр).
500.7.1	Во время прохождения упрощённой идентификации возникли ошибки. Обратитесь в службу поддержки МОНЕТА.РУ.
500.7.2	Упрощённая идентификация не проведена. Закончился срок ожидания результатов проверки. Для прохождения Упрощённой идентификации следует сделать еще один запрос.
500.7.3	Данные пользователя не прошли проверку в ЕСИА.
500.7.4	Для прохождения упрощённой идентификации укажите паспортные данные. Указанный Вами документ не является действующим паспортом.
500.7.5	Сотовый телефон не подтвержден.
500.7.6	Вами предоставлена недостаточно полная информация о себе. Для прохождения упрощённой идентификации укажите в личном кабинете: ФИО, паспортные данные, ИНН, СНИЛС, номер мобильного телефона.
500.7.7	Автоматическая упрощённая идентификация доступна только для граждан России.
500.7.8	Найдены похожие профили со статусом "Упрощённая идентификация". Обратитесь в службу поддержки МОНЕТА.РУ.
500.7.9	Не хватает или подтверждены не все персональные данные, необходимые для упрощённой идентификации.
500.7.10	Документ удостоверяющей личность не прошел проверку подтверждения достоверности. Обратитесь в службу поддержки МОНЕТА.РУ.
500.7.11	Пользователь уже имеет статус "Упрощённая идентификация".
500.7.12	Невозможно провести Упрощённую идентификацию.
500.8	Ошибка во внешней системе СМЭВ: *.
500.8.4769.1	Единая система идентификации и аутентификации (ЕСИА) не отвечает на запрос. Просим Вас повторить запрос позже.
500.8.4769.2	Во время выполнения запроса в Единой системе идентификации и аутентификации (ЕСИА) произошла ошибка.
500.9.1	Асинхронная задача не найдена.
500.9.2	Данная задача не поддерживает асинхронную обработку.
500.9.3	Несоответствие контекста асинхронного запроса и результата.

500.9.4

Запрос может быть вызван только в асинхронном режиме. Используйте `AsyncRequest` для работы с данным запросом.

Глава

4

Описание Merchant API

- [Основные типы данных и форматы](#)
- [Описание глобальных типов](#)
- [Финансовые методы](#)
- [Шаблоны операций](#)
- [Профайл пользователя](#)
- [Управление счетами](#)
- [Документы, юридические реквизиты, договора](#)
- [Банковские реквизиты](#)
- [Делегированный доступ к счетам](#)
- [Отчеты](#)
- [Идентификация пользователя](#)
- [Проверка телефона пользователя](#)
- [Асинхронные запросы](#)

Основные типы данных и форматы

Основные типы данных, использующиеся в API:

- **string**. Представляет символьные строки.
- **normalizedString**. Представляет нормализованные строки, не содержащие символов возврата каретки (carriage return; #xD), перевода строки (line feed; #xA) и табуляции (tab; #x9).
- **decimal**. Представляет подмножество действительных чисел, которые могут быть представлены десятичными цифрами, областью значений является множество чисел, получаемых умножением целого числа на неположительную степень десяти.
- **integer**. Тип, производный от decimal ограничением количества десятичных знаков (fractionDigits) до нуля и запрещением десятичного разделителя на конце.
- **long**. Тип, производный от integer ограничением максимального значения (maxInclusive) числом 9223372036854775807 и минимального значения (minInclusive) числом -9223372036854775808.
- **int**. Тип, производный от long ограничением максимального значения (maxInclusive) числом 2147483647 и минимального значения (minInclusive) числом -2147483648.
- **dateTime**. Представляет дату и время, область значений тесно связана с датой и временем, описанными в ISO 8601, лексически состоит из последовательности символов конечной длины в формате: '-'? yyyy '-' mm '-' dd 'T' hh ':' mm ':' ss ('.' s+)? (zzzzzz)?, часовой пояс сервера соответствует московскому времени (MSD; GMT+3).

| **Важно:** Сервис Merchant API предполагает, что все данные запросов передаются в кодировке UTF-8.

Описание глобальных типов

AbstractAttributeObject Complex Type

Тип, который позволяет работать с сущностями типа "ключ-значение".

XML Schema

```
<xsd:complexType abstract="true" name="AbstractAttributeObject">
  <xsd:sequence>
    <xsd:element maxOccurs="unbounded" minOccurs="0" name="attribute"
      type="tns:KeyValueApprovedAttribute"/>
  </xsd:sequence>
</xsd:complexType>
```

AccountAccessInfo Complex Type

Тип, описывающий доступ к счету.

XML Schema

```
<xsd:complexType name="AccountAccessInfo">
  <xsd:sequence>
    <xsd:element name="accessToWrite" type="xsd:boolean"/>
    <xsd:element name="accessToTakenOut" type="xsd:boolean"/>
    <xsd:element name="accessToTakenIn" type="xsd:boolean"/>
  </xsd:sequence>
</xsd:complexType>
```

Описание

accessToWrite

Доступ на изменение свойств счета.

Тип: boolean

accessToTakenOut

Доступ на снятие средств со счета.

Тип: boolean

accessToTakenIn

Доступ на зачисление средств на счет.

Тип: boolean

AccountId Simple Type

Тип, описывающий номер счета в системе МОНЕТА.РУ. Счет задается числом, минимальное значение которого равно 1, а максимальное состоит из 10 цифр.

Тип: long

XML Schema

```
<xsd:simpleType name="AccountId">
  <xsd:restriction base="xsd:long">
    <xsd:minInclusive value="1"/>
    <xsd:totalDigits value="10"/>
  </xsd:restriction>
</xsd:simpleType>
```

AccountInfo Complex Type

Тип, описывающий атрибуты счета в системе МОНЕТА.РУ.

XML Schema

```
<xsd:complexType name="AccountInfo">
  <xsd:sequence>
    <xsd:element name="id" type="tns:AccountId"/>
    <xsd:element minOccurs="0" name="currency" type="tns:Currency"/>
    <xsd:element minOccurs="0" name="balance" type="tns:Money"/>
    <xsd:element minOccurs="0" name="availableBalance" type="tns:Money"/>
    <xsd:element name="type" type="tns:AccountType"/>
    <xsd:element name="status" type="tns:AccountStatus"/>
    <xsd:element minOccurs="0" name="alias" type="xsd:string"/>
    <xsd:element minOccurs="0" name="onSuccessfulDebitUrl" type="xsd:string"/>
    <xsd:element minOccurs="0" name="onSuccessfulCreditUrl" type="xsd:string"/>
    <xsd:element minOccurs="0" name="signature" type="xsd:string"/>
    <xsd:element minOccurs="0" name="lowBalanceThreshold" type="tns:Money"/>
    <xsd:element minOccurs="0" name="highBalanceThreshold" type="tns:Money"/>
    <xsd:element minOccurs="0" name="accountAccess" type="tns:AccountAccessInfo"/>
    <xsd:element minOccurs="0" name="prototypeAccountId" type="xsd:long"/>
    <xsd:element minOccurs="0" name="onCancelledDebitUrl" type="xsd:string"/>
    <xsd:element minOccurs="0" name="onCancelledCreditUrl" type="xsd:string"/>
    <xsd:element maxOccurs="unbounded" minOccurs="0" name="attribute"
      type="tns:KeyValueApprovedAttribute"/>
  </xsd:sequence>
</xsd:complexType>
```

Описание

id

Номер счета в системе МОНЕТА.РУ.

Тип: [AccountId Simple Type](#)

currency

Валюта данного счета.

Необязательный элемент.

Тип: [Currency Simple Type](#)

balance

Баланс на данном счете.

Необязательный элемент.

Тип: [Money Simple Type](#)

availableBalance

Доступный баланс на данном счете.

Необязательный элемент.

Тип: [Money Simple Type](#)

type

Тип счета в системе МОНЕТА.РУ.

Тип: [AccountType Simple Type](#)

status

Статус счета в системе МОНЕТА.РУ.

Тип: [AccountStatus Simple Type](#)

alias

Название счета в системе МОНЕТА.РУ.

Необязательный элемент.

Тип: string

onSuccessfulDebitUrl

URL после списания средств.

Необязательный элемент.

Тип: string

onSuccessfulCreditUrl

URL после зачисления средств.

Необязательный элемент.

Тип: string

signature

Код проверки целостности данных.

Необязательный элемент.

Тип: string

lowBalanceThreshold

Если баланс счета меньше данного значения, то раз в сутки уходит уведомление об этом событии.

Необязательный элемент.

Тип: *Money Simple Type*

highBalanceThreshold

Если баланс счета больше данного значения, то раз в сутки уходит уведомление об этом событии.

Необязательный элемент.

Тип: *Money Simple Type*

accountAccess

Информация о доступе к счету. Информация отдается, если счет является делегированным.

Необязательный элемент.

Тип: *AccountAccessInfo Complex Type*

prototypeAccountId

Счет-прототип с которого берутся свойства "по умолчанию".

Необязательный элемент.

Тип: long

onCancelledDebitUrl

URL после отмены списания средств.

Необязательный элемент.

Тип: string

onCancelledCreditUrl

URL после отмены зачисления средств.

Необязательный элемент.

Тип: string

attribute

Дополнительные свойства счета.

Важно: Для получения этого поля в запросе необходимо выставить атрибут version равный или больше **VERSION_2**.

Возможные ключи:

- **paymentPasswordType** - тип платежного пароля.
 - **STATIC**. Статический платежный пароль.
 - **SEQUENCE_BY_ORDER**. Последовательность по порядку.
 - **SMS_SIMPLE**. SMS пароль.
 - **SMS_SESSION**. SMS пароль (сессионный).
 - **TOTP_RFC6238**. Time-based one-time password algorithm.
- **paymentPasswordChallengeRequired**. Если в ответе придет поле с ключом key="paymentPasswordChallengeRequired" и значением value="true", то перед проведением операции следует сделать запрос "GetAccountPaymentPasswordChallengeRequest", который вернет строку запроса для платежного пароля. Это значение необходимо использовать в поле "paymentPasswordChallenge" в запросах на проведение операции (например: "PaymentRequest", "TransferRequest", "AuthoriseTransactionRequest" и т.д.).
- **paymentPasswordExpirationDate**. Срок действия платежного пароля. Пример: 2017-01-09T14:11:24.000+03:00. Это поле возвращается, если на счете установлен "Срок действия платежного пароля" и тип платежного пароля (paymentPasswordType) статический (STATIC).
- **alias**. Название счета. В этом свойстве также можно узнать, является ли название счета "публичным".
- **primary**. Если счет является "Основным", то значение будет равно **true**.

Возвращается в методе FindAccountsListRequest. В методах FindAccountByIdRequest и FindAccountByAliasRequest это значение не возвращается.

- **delegated**. Если счет является делегированным, то значение будет равно **true**.
- **balanceChangesDate**. Дата изменения баланса. Пример: 2017-01-09T14:11:24.000+03:00
- **bankAccountForCredits**. Внутрибанковский счет для пополнения.
- **bankAccountForDebits**. Внутрибанковский счет для списания.
- **interfacetype**. Тип интерфейса.

1 - MONETA.Assistant.

- **testmode**. Тестовый режим (true|false).
- **paymentsystem_limitids**. Список платежных систем.
- **paymentsystem_unitid**. Платежная система по умолчанию.
- **checkurl**. Check URL.
- **payurl**. Pay URL.
- **httpmethod**. HTTP метод (PayUrl, CheckUrl).
- **signature**. Код проверки целостности данных.
- **issignaturemandatory**. Подпись формы оплаты обязательна (true|false).
- **redefinesettingsinurl**. Можно переопределять настройки в url (true|false).
- **successurl**. Success URL.
- **failurl**. Fail URL.
- **inprogressurl**. InProgress URL.
- **returnurl**. Return URL.
- **assistantformtarget**. Target (возврат для iframe).
- **onsuccessdebiturl**. URL после списания средств.
- **onsuccesscrediturl**. URL после зачисления средств.
- **oncancelleddebiturl**. URL после отмены списания средств.
- **oncancelledcrediturl**. URL после отмены зачисления средств.
- **onauthoriseurl**. URL после авторизации средств.

Необязательный элемент.

Тип: *KeyValueApprovedAttribute Complex Type*

AccountPaymentPasswordType Simple Type

Тип платежного пароля.

Тип: string

Возможные значения:

- **STATIC**.Статический платежный пароль.
- **SMS_SIMPLE**.SMS пароль.

XML Schema

```
<xsd:simpleType name="AccountPaymentPasswordType">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="STATIC"/>
    <xsd:enumeration value="SMS_SIMPLE"/>
  </xsd:restriction>
</xsd:simpleType>
```

AccountRelation Complex Type

Тип, описывающий делегированный доступ к счету.

XML Schema

```
<xsd:complexType name="AccountRelation">
  <xsd:sequence>
    <xsd:element name="accountId" type="tns:AccountId"/>
    <xsd:element name="principalEmail" type="tns:Email"/>
    <xsd:element minOccurs="0" name="canViewAccount" type="xsd:boolean"/>
    <xsd:element minOccurs="0" name="canEditAccount" type="xsd:boolean"/>
    <xsd:element minOccurs="0" name="canProcessOperation" type="xsd:boolean"/>
  </xsd:sequence>
</xsd:complexType>
```

Описание

accountId

Номер счета в системе МОНЕТА.РУ.

Тип: *AccountId Simple Type*

principalEmail

Email пользователя, которому предоставляется делегированный доступ к счету.

Тип: *Email Simple Type*

canViewAccount

Просмотр счета.

Необязательный элемент.

Тип: boolean

canEditAccount

Редактирование счета.

Необязательный элемент.

Тип: boolean

canProcessOperation

Проведение операций.

Необязательный элемент.

Тип: boolean

AccountStatement Complex Type

Выписка по счету.

XML Schema

```
<xsd:complexType name="AccountStatement">
  <xsd:sequence>
    <xsd:element minOccurs="1" name="ownerAccount" type="xsd:string"/>
    <xsd:element minOccurs="0" name="contractDate" type="xsd:dateTime"/>
    <xsd:element minOccurs="0" name="ownerAccountCurrency" type="xsd:string"/>
    <xsd:element minOccurs="0" name="ownerName" type="xsd:string"/>
    <xsd:element minOccurs="0" name="dateFrom" type="xsd:dateTime"/>
    <xsd:element minOccurs="0" name="dateTo" type="xsd:dateTime"/>
    <xsd:element minOccurs="0" name="lastOperationDate" type="xsd:dateTime"/>
    <xsd:element minOccurs="0" name="now" type="xsd:dateTime"/>
    <xsd:element minOccurs="0" name="inSaldo" type="tns:Money"/>
    <xsd:element minOccurs="0" name="outSaldo" type="tns:Money"/>
    <xsd:element minOccurs="0" name="inSaldoBlocked" type="tns:Money"/>
    <xsd:element minOccurs="0" name="outSaldoBlocked" type="tns:Money"/>
    <xsd:element minOccurs="0" name="debitAmountTotal" type="tns:Money"/>
    <xsd:element minOccurs="0" name="creditAmountTotal" type="tns:Money"/>
    <xsd:element name="totalSize" type="xsd:long"/>
    <xsd:element maxOccurs="unbounded" minOccurs="0" name="orders"
      type="tns:AccountStatementRecord"/>
  </xsd:sequence>
</xsd:complexType>
```

Описание

ownerAccount

Внутрибанковский счет.

Обязательный элемент.

Тип: string

contractDate

Дата договора.

Необязательный элемент.

Тип: dateTime

ownerAccountCurrency

Валюта счета.

Необязательный элемент.

Тип: string

ownerName

Владелец счета.

Необязательный элемент.

Тип: string

dateFrom

Дата начала выборки.

Необязательный элемент.

Тип: dateTime

dateTo

Дата окончания выборки.

Необязательный элемент.

Тип: `dateTime`

lastOperationDate

Дата последней операции.

Необязательный элемент.

Тип: `dateTime`

now

Дата актуальности.

Необязательный элемент.

Тип: `dateTime`

inSaldo

Входящий остаток.

Необязательный элемент.

Тип: *Money Simple Type*

outSaldo

Исходящий остаток.

Необязательный элемент.

Тип: *Money Simple Type*

inSaldoBlocked

Заблокированная входящая сумма.

Необязательный элемент.

Тип: *Money Simple Type*

outSaldoBlocked

Заблокированная исходящая сумма.

Необязательный элемент.

Тип: *Money Simple Type*

debitAmountTotal

Итоговые обороты по дебету.

Необязательный элемент.

Тип: *Money Simple Type*

creditAmountTotal

Итоговые обороты по кредиту.

Необязательный элемент.

Тип: *Money Simple Type*

totalSize

Общее количество записей, которое можно получить в данной выборке.

Тип: `long`

orders

Информация об операциях.

Необязательный элемент.

Тип: *AccountStatementRecord Complex Type*

AccountStatementRecord Complex Type

Информация об операции.

XML Schema

```
<xsd:complexType name="AccountStatementRecord">
  <xsd:sequence>
    <xsd:element minOccurs="0" name="date" type="xsd:dateTime"/>
    <xsd:element minOccurs="1" name="operationId" type="xsd:long"/>
    <xsd:element minOccurs="0" name="operationView" type="xsd:string"/>
    <xsd:element minOccurs="0" name="correspondentName" type="xsd:string"/>
    <xsd:element minOccurs="0" name="correspondentAccountId" type="tns:AccountId"/>
    <xsd:element minOccurs="0" name="correspondentAccount" type="xsd:string"/>
    <xsd:element minOccurs="0" name="debitAmount" type="tns:Money"/>
    <xsd:element minOccurs="0" name="creditAmount" type="tns:Money"/>
    <xsd:element minOccurs="0" name="description" type="xsd:string"/>
  </xsd:sequence>
</xsd:complexType>
```

Описание

date

Дата операции.

Необязательный элемент.

Тип: dateTime

operationId

Номер операции в системе MONETA.PY.

Обязательный элемент.

Тип: long

operationView

Вид операции.

Необязательный элемент.

Тип: string

correspondentName

Наименование контрагента.

Необязательный элемент.

Тип: string

correspondentAccountId

Номер счета контрагента в системе MONETA.RU.

Необязательный элемент.

Тип: *AccountId Simple Type*

correspondentAccount

Номер корреспондирующего счета контрагента.

Необязательный элемент.

Тип: string

debitAmount

Сумма по дебету счета.

Необязательный элемент.

Тип: *Money Simple Type*

creditAmount

Сумма по кредиту счета.

Необязательный элемент.

Тип: *Money Simple Type*

description

Назначение операции.

Необязательный элемент.

Тип: string

AccountStatus Simple Type

Тип, описывающий статусы счетов в системе МОНЕТА.РУ. Счет может быть либо активным, либо заблокированным.

Тип: int

Возможные значения:

- 1.Активный счет.
- 2.Заблокированный счет.
- 3.Закрытый счет.

XML Schema

```
<xsd:simpleType name="AccountStatus">
  <xsd:restriction base="xsd:int">
    <xsd:enumeration value="1"/>
    <xsd:enumeration value="2"/>
    <xsd:enumeration value="3"/>
  </xsd:restriction>
</xsd:simpleType>
```

AccountType Simple Type

Типы счетов в системе МОНЕТА.РУ.

Тип: int

Возможные значения:

- 1.Обычный счет.
- 2.Расширенный счет.
- 3.Бонус-счет.
- 4.Займовый счет.
- 5.Целевой счет.
- 6.Комиссионный счет.
- 7.Расчетный счет.

XML Schema

```
<xsd:simpleType name="AccountType">
  <xsd:restriction base="xsd:int">
    <xsd:enumeration value="1"/>
    <xsd:enumeration value="2"/>
    <xsd:enumeration value="3"/>
    <xsd:enumeration value="4"/>
    <xsd:enumeration value="5"/>
    <xsd:enumeration value="6"/>
    <xsd:enumeration value="7"/>
  </xsd:restriction>
</xsd:simpleType>
```

AsyncStatus Simple Type

Тип, описывающий статусы асинхронных запросов в системе МОНЕТА.РУ. Данный тип может иметь только определенные значения, описанные ниже.

Тип: string

Возможные значения:

- **INPROGRESS**.Асинхронная задача в обработке.
- **CREATED**.Асинхронная задача создана.

XML Schema

```
<xsd:simpleType name="AsyncStatus">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="INPROGRESS"/>
    <xsd:enumeration value="CREATED"/>
  </xsd:restriction>
</xsd:simpleType>
```

AuthoriseTransactionBatchRequestType Complex Type

Тип, описывающий параметры операции в запросах в пакетном режиме.

Тип: *EntityBatchRequestType Complex Type*.

XML Schema

```
<xsd:complexType name="AuthoriseTransactionBatchRequestType">
  <xsd:complexContent>
    <xsd:extension base="tns:EntityBatchRequestType">
      <xsd:sequence>
        <xsd:element maxOccurs="unbounded" minOccurs="1" name="transaction"
          type="tns:AuthoriseTransactionRequestType"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Описание transaction

Набор операций, которые необходимо выполнить в одном пакете. Операции выполняются в том порядке, в котором они переданы в запросе.

Обязательный элемент.

Тип: *AuthoriseTransactionRequestType Complex Type*

AuthoriseTransactionRequestType Complex Type

Тип для запроса на регистрацию операции с блокировкой средств на счете плательщика.

Тип: *TransactionRequestType Complex Type*.

XML Schema

```
<xsd:complexType name="AuthoriseTransactionRequestType">
  <xsd:complexContent>
    <xsd:extension base="tns:TransactionRequestType">
      <xsd:sequence>
        <xsd:element minOccurs="0" name="transactionId" type="xsd:long"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Описание

transactionId

Номер операции.

Необязательный элемент.

Тип: long

BankAccount Complex Type

Банковские реквизиты в системе МОНЕТА.РУ. Данные представляются в виде "ключ-значение".

Возможные ключи:

- **is_international.** Международные банковские реквизиты (true/false).
- **bank.** Банк.
- **bik.** БИК.
- **account.** Расчетный счет.
- **corr_bank.** Корреспондентский банк.
- **corr_account.** Корреспондентский счет.
- **country.** Страна.
- **state.** Регион/республика.
- **city.** Город.
- **kbk.** КБК.
- **oktmo.** Общероссийский классификатор территорий муниципальных образований.
- **kpp.** КПП.
- **username.** Наименование получателя.

XML Schema

```
<xsd:complexType name="BankAccount">
  <xsd:sequence>
    <xsd:element minOccurs="0" name="id" type="xsd:long"/>
    <xsd:element maxOccurs="unbounded" minOccurs="0" name="attribute"
      type="tns:KeyValueApprovedAttribute"/>
  </xsd:sequence>
</xsd:complexType>
```

Описание

id

ID банковских реквизитов.

Необязательный элемент.

Тип: long

attribute

Необязательный элемент.

Тип: *KeyValueApprovedAttribute Complex Type*

CTID Simple Type

Тип, представляющий внешний номер операции в учетной системе партнера.

Тип: string

Максимальная длина: 255

XML Schema

```
<xsd:simpleType name="CTID">
  <xsd:restriction base="xsd:string">
```

```
<xsd:maxLength value="255"/>
</xsd:restriction>
</xsd:simpleType>
```

CancelTransactionBatchRequestType Complex Type

Тип для запроса на отмену операций в пакетном режиме.

Тип: [EntityBatchRequestType Complex Type](#).

XML Schema

```
<xsd:complexType name="CancelTransactionBatchRequestType">
  <xsd:complexContent>
    <xsd:extension base="tns:EntityBatchRequestType">
      <xsd:sequence>
        <xsd:element maxOccurs="unbounded" minOccurs="1" name="transaction"
          type="tns:CancelTransactionRequestType"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Описание

transaction

Набор операций, которые необходимо выполнить в одном пакете. Операции выполняются в том порядке, в котором они переданы в запросе.

Обязательный элемент.

Тип: [CancelTransactionRequestType Complex Type](#)

CancelTransactionBatchResponseType Complex Type

Тип, описывающий атрибуты при ответе на запрос отмены операций в пакетной режиме.

XML Schema

```
<xsd:complexType name="CancelTransactionBatchResponseType">
  <xsd:sequence>
    <xsd:element minOccurs="0" name="error" type="xsd:string"/>
    <xsd:element minOccurs="0" name="transaction"
      type="tns:CancelTransactionResponseType"/>
    <xsd:element minOccurs="0" name="errorCode" type="xsd:string"/>
  </xsd:sequence>
</xsd:complexType>
```

Описание

error

Поле содержит описание ошибки, если операция не была проведена. В этом случае поле transaction - пустое.

Если операция проведена, то поле error пустое, а поле transaction содержит детали операции.

Необязательный элемент.

Тип: string

transaction

Необязательный элемент.

Тип: [CancelTransactionResponseType Complex Type](#)

errorCode

Поле содержит код ошибки. Поле заполнено только тогда, когда есть описание ошибки в поле error.

Для получения этого поля в запросе необходимо выставить атрибут version равный или больше **VERSION_2**.

Коды ошибок совпадают со значением элемента faultDetail (смотрите описание для этого элемента).

Необязательный элемент.

Тип: string

CancelTransactionRequestType Complex Type

Тип для запроса на отмену операции.

XML Schema

```
<xsd:complexType name="CancelTransactionRequestType">
  <xsd:sequence>
    <xsd:element name="transactionId" type="xsd:long"/>
    <xsd:element minOccurs="0" name="protectionCode" type="xsd:string"/>
    <xsd:element minOccurs="0" name="description" type="tns:Description"/>
  </xsd:sequence>
</xsd:complexType>
```

Описание

transactionId

Номер операции.

Тип: long

protectionCode

Код протекции.

Необязательный элемент.

Тип: string

description

Описание операции.

Необязательный элемент.

Тип: [Description Simple Type](#)

CancelTransactionResponseType Complex Type

Тип, описывающий атрибуты при ответе на запрос отмены операции.

XML Schema

```
<xsd:complexType name="CancelTransactionResponseType">
  <xsd:sequence>
    <xsd:element name="transactionId" type="xsd:long"/>
    <xsd:element minOccurs="0" name="operationStatus" type="tns:OperationStatus"/>
  </xsd:sequence>
</xsd:complexType>
```

Описание

transactionId

Номер операции.

Тип: long

operationStatus

Статус операции.

Необязательный элемент.

Тип: *OperationStatus Simple Type*

CommonOperationTemplateParameters Complex Type

Параметры шаблона операции, если он имеет тип *OperationTemplateType.COMMON*.

Тип: *OperationTemplateAmount Complex Type*.

XML Schema

```

<xsd:complexType name="CommonOperationTemplateParameters">
  <xsd:complexContent>
    <xsd:extension base="tns:OperationTemplateAmount"/>
  </xsd:complexContent>
</xsd:complexType>

```

ConfirmTransactionBatchRequestType Complex Type

Тип для запроса подтверждения операции в пакетном режиме.

Тип: *EntityBatchRequestType Complex Type*.

XML Schema

```

<xsd:complexType name="ConfirmTransactionBatchRequestType">
  <xsd:complexContent>
    <xsd:extension base="tns:EntityBatchRequestType">
      <xsd:sequence>
        <xsd:element maxOccurs="unbounded" minOccurs="1" name="transaction"
          type="tns:ConfirmTransactionRequestType"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>

```

Описание**transaction**

Набор операций, которые необходимо выполнить в одном пакете. Операции выполняются в том порядке, в котором они переданы в запросе.

Обязательный элемент.

Тип: *ConfirmTransactionRequestType Complex Type*

ConfirmTransactionRequestType Complex Type

Тип для запроса подтверждения операции.

XML Schema

```

<xsd:complexType name="ConfirmTransactionRequestType">
  <xsd:sequence>
    <xsd:element name="transactionId" type="xsd:long"/>
    <xsd:element minOccurs="0" name="paymentPassword" type="tns:Password"/>
    <xsd:element minOccurs="0" name="protectionCode" type="xsd:string"/>
    <xsd:element minOccurs="0" name="operationInfo" type="tns:OperationInfo"/>
    <xsd:element minOccurs="0" name="amount" type="tns:Money"/>
    <xsd:element minOccurs="0" name="isPayerAmount" type="xsd:boolean"/>
    <xsd:element minOccurs="0" name="paymentPasswordChallenge" type="xsd:string"/>
    <xsd:element minOccurs="0" name="clientTransaction" type="tns:CTID"/>
  </xsd:sequence>
</xsd:complexType>

```

```
<xsd:element minOccurs="0" name="personalInformation"
type="tns:PersonalInformation"/>
</xsd:sequence>
</xsd:complexType>
```

Описание

transactionId

Номер операции.

Тип: long

paymentPassword

Платежный пароль счета плательщика.

Необязательный элемент.

Тип: *Password Simple Type*

protectionCode

Код протекции.

Необязательный элемент.

Тип: string

operationInfo

Набор полей, которые необходимо сохранить в качестве атрибутов операции. Значения дат в формате dd.MM.yyyy HH:mm:ss

Необязательный элемент.

Тип: *OperationInfo Complex Type*

amount

Сумма операции.

- Если флаг isPayerAmount отсутствует и пользователь имеет доступ только к счету плательщика, то интерпретируется как сумма к списанию (в валюте плательщика).
- Если флаг isPayerAmount отсутствует и пользователь имеет доступ только к счету получателя, то интерпретируется как сумма к зачислению (в валюте получателя).

Необязательный элемент.

Тип: *Money Simple Type*

isPayerAmount

Если пользователь имеет доступ как к счету плательщика, так и к счету получателя, то флаг isPayerAmount обязателен.

- Если флаг isPayerAmount установлен (true), то amount используется как сумма к списанию (в валюте плательщика).
- Если флаг isPayerAmount сброшен (false), то amount используется как сумма к зачислению (в валюте получателя).

Необязательный элемент.

Тип: boolean

paymentPasswordChallenge

Запрос для платежного пароля.

Необязательный элемент.

Тип: string

clientTransaction

Внешний номер операции.

Необязательный элемент.

Тип: *CTID Simple Type*

personalInformation

Персональные данные пользователя.

Необязательный элемент.

Тип: *PersonalInformation Complex Type*

Contract Complex Type

Договор в системе МОНЕТА.РУ. Данные представляются в виде "ключ-значение".

Возможные ключи договора:

- **num.** Номер договора.
- **unitid.** ID юнита, у которого создан договор.
- **own.** true-создан для указанного в запросе юнита, false-создан для родительского юнита.
- **unitname.** если own=false, то данное поле содержит имя юнита.
- **datestart.** Дата начала договора.
- **datestop.** Дата окончания договора.
- **status.** Статус договора:
 - **CREATED.** Сформирован.
 - **SIGNED.** Подписан.
 - **ACTIVE.** Активный.
 - **PAUSED.** Приостановлен.
 - **INACTIVE.** Расторгнут.
- **type.** Тип договора:
 - **PARTNER.** Договор.
 - **VIRTUAL.** Виртуальный договор.

XML Schema

```
<xsd:complexType name="Contract">
  <xsd:sequence>
    <xsd:element minOccurs="0" name="id" type="xsd:long"/>
    <xsd:element maxOccurs="unbounded" minOccurs="0" name="attribute"
      type="tns:KeyValueApprovedAttribute"/>
  </xsd:sequence>
</xsd:complexType>
```

Описание

id

ID договора.

Необязательный элемент.

Тип: long

attribute

Необязательный элемент.

Тип: *KeyValueApprovedAttribute Complex Type*

Currency Simple Type

Тип, описывающий валюты, используемые в системе МОНЕТА.РУ. Данный тип может иметь только строго определенные значения, описанные ниже.

Тип: string

Возможные значения:

- RUB
- USD
- EUR
- GBP

XML Schema

```
<xsd:simpleType name="Currency">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="RUB"/>
    <xsd:enumeration value="USD"/>
    <xsd:enumeration value="EUR"/>
    <xsd:enumeration value="GBP"/>
  </xsd:restriction>
</xsd:simpleType>
```

Description Simple Type

Описание или комментарий, применяющийся в сложных типах.

Тип: `normalizedString`

Максимальная длина: 255

XML Schema

```
<xsd:simpleType name="Description">
  <xsd:restriction base="xsd:normalizedString">
    <xsd:maxLength value="255"/>
  </xsd:restriction>
</xsd:simpleType>
```

DirectDebitOperationTemplateParameters Complex Type

Параметры шаблона операции, если он имеет тип *OperationTemplateType.DIRECT_DEBIT*.

Тип: *OperationTemplateAmountRange Complex Type*.

XML Schema

```
<xsd:complexType name="DirectDebitOperationTemplateParameters">
  <xsd:complexContent>
    <xsd:extension base="tns:OperationTemplateAmountRange">
      <xsd:sequence>
        <xsd:element name="active" type="xsd:boolean"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Описание

active

Поле указывает - является ли шаблон прямого дебетования активным или нет.

Тип: `boolean`

Document Complex Type

Тип, описывающий документ в системе МОНЕТА.РУ.

XML Schema

```
<xsd:complexType name="Document">
```

```

<xsd:sequence>
  <xsd:element minOccurs="0" name="id" type="xsd:long"/>
  <xsd:element minOccurs="0" name="type" type="tns:DocumentType"/>
  <xsd:element maxOccurs="unbounded" minOccurs="0" name="attribute"
    type="tns:KeyValueApprovedAttribute"/>
  <xsd:element minOccurs="0" name="hasAttachedFiles" type="xsd:boolean"/>
</xsd:sequence>
</xsd:complexType>

```

Описание

id

Идентификатор документа в системе МОНЕТА.РУ.

Необязательный элемент.

Тип: long

type

Тип документа.

Необязательный элемент.

Тип: *DocumentType Simple Type*

attribute

Поля документа в системе МОНЕТА.РУ.

Данные представляются в виде "ключ-значение" и признака подтвержденности.

В зависимости от типа документа возвращаются следующие поля.

Для документов типа PASSPORT, MILITARY_ID:

- **SERIES.** Серия документа.
- **NUMBER.** Номер документа.
- **ISSUER.** Кем выдан документ.
- **ISSUED.** Когда выдан документ.
- **COMMENTS.** Комментарии (необязательное поле).

Для документов типа DRIVING_LICENCE:

- **SERIES.** Серия документа.
- **NUMBER.** Номер документа.
- **ISSUER.** Кем выдан документ.
- **ISSUED.** Когда выдан документ.
- **EXPIRATIONDATE.** Срок действия.
- **COMMENTS.** Комментарии (необязательное поле).

Для OTHER:

- **COMMENTS.** Комментарии, пояснения, описание.

Для всех типов документов:

- **customfield:***. Произвольный набор значений. В документе их может быть несколько. Полный ключ атрибута состоит из префикса ("customfield:") и тэга (32 символа). Например, "customfield:name".
- **MODIFICATIONDATE.** Последняя дата редактирования документа.

Необязательный элемент.

Тип: *KeyValueApprovedAttribute Complex Type*

hasAttachedFiles

Имеет ли документ прикрепленные файлы.

Подсказка: Для получения прикрепленных файлов используйте вызов FindProfileDocumentFilesRequest.

Необязательный элемент.

Тип: boolean

DocumentType Simple Type

Типы документов, поддерживаемые в системе МОНЕТА.РУ.

Тип: string

Возможные значения:

- **PASSPORT**.Паспорт.
- **DRIVING_LICENCE**.Водительское удостоверение.
- **MILITARY_ID**.Военный билет.
- **OTHER**.Другой документ.
- **PARTNER_LICENCE**.Лицензия для деятельности партнера.
- **IDENTITY_DOCUMENT**.Документ PASSPORT для нерезидентов.
- **ADMINISTRATIVE**.Административные документ, доступный только для админу.
- **CUSTOM**

XML Schema

```
<xsd:simpleType name="DocumentType">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="PASSPORT"/>
    <xsd:enumeration value="DRIVING_LICENCE"/>
    <xsd:enumeration value="MILITARY_ID"/>
    <xsd:enumeration value="OTHER"/>
    <xsd:enumeration value="PARTNER_LICENCE"/>
    <xsd:enumeration value="IDENTITY_DOCUMENT"/>
    <xsd:enumeration value="ADMINISTRATIVE"/>
    <xsd:enumeration value="CUSTOM"/>
  </xsd:restriction>
</xsd:simpleType>
```

Email Simple Type

Email.

Тип: string

XML Schema

```
<xsd:simpleType name="Email">
  <xsd:restriction base="xsd:string">
    <xsd:minLength value="5"/>
    <xsd:whiteSpace value="collapse"/>
  </xsd:restriction>
</xsd:simpleType>
```

Entity Complex Type

Базовый тип, содержащий атрибут **version**.

XML Schema

```
<xsd:complexType name="Entity">
  <xsd:attribute default="VERSION_1" name="version" type="tns:Version" use="optional"/>
</xsd:complexType>
```

EntityBatchRequestType Complex Type

Тип, описывающий параметры в запросах в пакетном режиме.

Тип: *Entity Complex Type*.

XML Schema

```
<xsd:complexType name="EntityBatchRequestType">
  <xsd:complexContent>
    <xsd:extension base="tns:Entity">
      <xsd:sequence>
        <xsd:element name="transactional" type="xsd:boolean"/>
        <xsd:element name="exitOnFailure" type="xsd:boolean"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Описание

transactional

Флаг, указывающий выполнять ли все денежные переводы в одной транзакции. Если transactional = true, то:

- При возникновении ошибки все проведенные операции будут отменены.
- Можно проводить только операции со счетами монеты. Нельзя выводить деньги на внешние платежные системы.

Если transactional = false, то:

- При возникновении ошибки все проведенные операции будут сохранены.
- Можно проводить любые операции, которые доступны в TransferRequest.
- Если выставить флаг exitOnFailure = false, то при возникновении ошибки можно пропустить операцию и продолжить выполнение операций дальше.

Тип: boolean

exitOnFailure

Флаг, указывающий прерывать ли выполнение пакета операций, если произошла ошибка. Используется только при transactional = false.

Тип: boolean

Fee Simple Type

Тип, представляющий значение суммы вознаграждения.

Тип: decimal

XML Schema

```
<xsd:simpleType name="Fee">
  <xsd:restriction base="xsd:decimal">
    <xsd:fractionDigits value="3"/>
  </xsd:restriction>
</xsd:simpleType>
```

File Complex Type

Тип, описывающий данные в виде бинарного файла.

XML Schema

```
<xsd:complexType name="File">
  <xsd:sequence>
```

```

<xsd:element name="documentId" type="xsd:long"/>
<xsd:element minOccurs="0" name="blob" type="xsd:base64Binary"/>
<xsd:element minOccurs="0" name="approved" type="xsd:boolean"/>
<xsd:element minOccurs="0" name="fileId" type="xsd:long"/>
<xsd:element minOccurs="0" name="mimeType" type="xsd:string"/>
<xsd:element minOccurs="0" name="title" type="xsd:string"/>
<xsd:element minOccurs="0" name="requestUri" type="xsd:string"/>
</xsd:sequence>
</xsd:complexType>

```

Описание

documentId

ID документа, которому принадлежит данный файл.

Тип: long

blob

Данные файла. При передаче/получении данных используйте MTOM (Message Transmission Optimization Mechanism).

Необязательный элемент.

Тип: base64Binary

approved

Проверен или нет данный файл.

Необязательный элемент.

Тип: boolean

fileId

ID файла.

Необязательный элемент.

Тип: long

mimeType

Mime type файла (например: image/jpeg).

Необязательный элемент.

Тип: string

title

Имя файла или описание.

Необязательный элемент.

Тип: string

requestUri

Ссылка для получения бинарного данных

Необязательный элемент.

Тип: string

ForecastTransactionResponseType Complex Type

Тип, описывающий суммы и комиссии в предварительном расчете операции.

XML Schema

```

<xsd:complexType name="ForecastTransactionResponseType">
  <xsd:sequence>

```



```

<xsd:element name="payer" type="xsd:long"/>
<xsd:element name="payerCurrency" type="tns:Currency"/>
<xsd:element name="payerAmount" type="tns:Money"/>
<xsd:element minOccurs="0" name="payerFee" type="tns:Money"/>
<xsd:element name="payee" type="xsd:long"/>
<xsd:element name="payeeCurrency" type="tns:Currency"/>
<xsd:element name="payeeAmount" type="tns:Money"/>
<xsd:element minOccurs="0" name="payeeFee" type="tns:Money"/>
<xsd:element minOccurs="0" name="payerAlias" type="xsd:string"/>
<xsd:element minOccurs="0" name="payeeAlias" type="xsd:string"/>
</xsd:sequence>
</xsd:complexType>

```

Описание

payer

Номер счета плательщика.

Тип: long

payerCurrency

Валюта счета плательщика.

Тип: *Currency Simple Type*

payerAmount

Сумма к списанию.

Тип: *Money Simple Type*

payerFee

Комиссия списания средств.

Необязательный элемент.

Тип: *Money Simple Type*

payee

Номер счета получателя.

Тип: long

payeeCurrency

Валюта счета получателя.

Тип: *Currency Simple Type*

payeeAmount

Сумма к зачислению.

Тип: *Money Simple Type*

payeeFee

Комиссия зачисления средств.

Необязательный элемент.

Тип: *Money Simple Type*

payerAlias

Название счета плательщика.

Необязательный элемент.

Тип: string

payeeAlias

Название счета получателя.

Необязательный элемент.

Тип: string

InfoTariff Complex Type

Информация о тарифе

XML Schema

```
<xsd:complexType name="InfoTariff">
  <xsd:sequence>
    <xsd:element minOccurs="0" name="sourceRangeCurrency" type="tns:Currency"/>
    <xsd:element minOccurs="0" name="sourceAmountMin" type="tns:Money"/>
    <xsd:element minOccurs="0" name="sourceAmountMax" type="tns:Money"/>
    <xsd:element minOccurs="0" name="targetRangeCurrency" type="tns:Currency"/>
    <xsd:element minOccurs="0" name="targetAmountMin" type="tns:Money"/>
    <xsd:element minOccurs="0" name="targetAmountMax" type="tns:Money"/>
  </xsd:sequence>
</xsd:complexType>
```

Описание

sourceRangeCurrency

Валюта счета плательщика для которого рассчитывается лимит суммы операции.

Необязательный элемент.

Тип: *Currency Simple Type*

sourceAmountMin

Минимальная сумма для плательщика, допустимая в операции.

Необязательный элемент.

Тип: *Money Simple Type*

sourceAmountMax

Максимальная сумма для плательщика, допустимая в операции.

Необязательный элемент.

Тип: *Money Simple Type*

targetRangeCurrency

Валюта счета получателя платежа для которого рассчитывается лимит суммы операции.

Необязательный элемент.

Тип: *Currency Simple Type*

targetAmountMin

Минимальная сумма для получателя платежа, допустимая в операции.

Необязательный элемент.

Тип: *Money Simple Type*

targetAmountMax

Максимальная сумма для получателя платежа, допустимая в операции.

Необязательный элемент.

Тип: *Money Simple Type*

InfoUrl Complex Type

Расширенная информация об URL платежной системы.

XML Schema

```
<xsd:complexType name="InfoUrl">
  <xsd:sequence>
    <xsd:element minOccurs="0" name="href" type="xsd:string"/>
    <xsd:element minOccurs="0" name="text" type="xsd:string"/>
  </xsd:sequence>
</xsd:complexType>
```

Описание

href

URL платежной системы.

Необязательный элемент.

Тип: string

text

Представление URL платежной системы.

Необязательный элемент.

Тип: string

InvoiceBatchRequestType Complex Type

Тип, описывающий параметры операции в запросах в пакетном режиме.

Тип: *EntityBatchRequestType Complex Type*.

XML Schema

```
<xsd:complexType name="InvoiceBatchRequestType">
  <xsd:complexContent>
    <xsd:extension base="tns:EntityBatchRequestType">
      <xsd:sequence>
        <xsd:element maxOccurs="unbounded" minOccurs="1" name="transaction"
          type="tns:InvoiceRequestType"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Описание

transaction

Набор операций, которые необходимо выполнить в одном пакете. Операции выполняются в том порядке, в котором они переданы в запросе.

Обязательный элемент.

Тип: *InvoiceRequestType Complex Type*

InvoiceRequestType Complex Type

Тип, описывающий параметры операции для выставления счета к оплате.

Тип: *Entity Complex Type*.

XML Schema

```
<xsd:complexType name="InvoiceRequestType">
  <xsd:complexContent>
    <xsd:extension base="tns:Entity">
      <xsd:sequence>
        <xsd:element minOccurs="0" name="payer" type="xsd:string"/>
        <xsd:element name="payee" type="xsd:long"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

```

<xsd:element minOccurs="0" name="amount" type="tns:Money"/>
<xsd:element name="clientTransaction" type="tns:CTID"/>
<xsd:element minOccurs="0" name="description" type="tns:Description"/>
<xsd:element minOccurs="0" name="mnt_custom1" type="xsd:string"/>
<xsd:element minOccurs="0" name="mnt_custom2" type="xsd:string"/>
<xsd:element minOccurs="0" name="mnt_custom3" type="xsd:string"/>
<xsd:element minOccurs="0" name="operationInfo" type="tns:OperationInfo"/>
</xsd:sequence>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>

```

Описание

payer

Номер счета плательщика.

Необязательный элемент.

Тип: string

payee

Номер счета получателя.

Тип: long

amount

Сумма.

Необязательный элемент.

Тип: *Money Simple Type*

clientTransaction

Внешний номер операции.

Тип: *CTID Simple Type*

description

Описание операции.

Необязательный элемент.

Тип: *Description Simple Type*

mnt_custom1

Произвольный параметр.

Необязательный элемент.

Тип: string

mnt_custom2

Произвольный параметр.

Необязательный элемент.

Тип: string

mnt_custom3

Произвольный параметр.

Необязательный элемент.

Тип: string

operationInfo

Набор полей, которые необходимо сохранить в качестве атрибутов операции. Значения дат в формате dd.MM.yyyy HH:mm:ss. Для того чтобы провести прямое дебетование, следует передать атрибут с ключом "SUBSCRIBERID" и в значении указать ID подписчика. Прямое дебетование будет проведено, если ID подписчика указано верно,

если пользователь разрешил проводить прямое дебетование, если указанная сумма есть на балансе пользователя. В данном случае деньги будут списаны со счета пользователя, в ответе на запрос в поле Статус будет указано INPROGRESS, зачисление денег на счет получателя будет проведено в асинхронном режиме. Если прямое дебетование не может быть проведено или во время его выполнения произойдет ошибка, то инвойс будет создан и в ответе на запрос в поле Статус будет указано CREATED. USERCONTACT - ссылка на оплату счета будет отправлена в письме или SMS (можно перечислить получателей через запятую).

Необязательный элемент.

Тип: *OperationInfo Complex Type*

KeyValueApprovedAttribute Complex Type

Тип, расширяющий KeyValueAttribute.

Тип: *KeyValueAttribute Complex Type*.

XML Schema

```
<xsd:complexType name="KeyValueApprovedAttribute">
  <xsd:complexContent>
    <xsd:extension base="tns:KeyValueAttribute">
      <xsd:sequence>
        <xsd:element maxOccurs="1" minOccurs="0" name="approved"
type="xsd:boolean"/>
        <xsd:element maxOccurs="1" minOccurs="0" name="published"
type="xsd:boolean"/>
        <xsd:element maxOccurs="1" minOccurs="0" name="fromPrototype"
type="xsd:boolean"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Описание

approved

Поле approved показывает проверено или нет данное свойство. Поле approved - не обязательное.

Необязательный элемент.

Тип: boolean

published

Поле published показывает доступно или нет чтение свойства для всех пользователей. Поле published - не обязательное.

Важно: Для чтения этого атрибута в запросе необходимо выставить атрибут version равный или больше **VERSION_2**. Для записи version можно не указывать.

Необязательный элемент.

Тип: boolean

fromPrototype

Поле fromPrototype показывает, что свойство прочитано из "объекта-прототипа".

Поле только для чтения.

Важно: Для чтения этого свойства в запросе необходимо выставить атрибут version равный или больше **VERSION_3**.

Необязательный элемент.

Тип: boolean

KeyValueAttribute Complex Type

Тип, который позволяет работать с сущностями типа "ключ-значение".

XML Schema

```
<xsd:complexType name="KeyValueAttribute">
  <xsd:sequence>
    <xsd:element name="key" type="xsd:string"/>
    <xsd:element name="value" type="xsd:string"/>
  </xsd:sequence>
</xsd:complexType>
```

LegalInformation Complex Type

Юридические реквизиты в системе МОНЕТА.РУ. Данные представляются в виде "ключ-значение".

Возможные ключи:

- **inn.** ИНН.
- **kpp.** КПП.
- **ogrn.** ОГРН.
- **ogrnip.** ОГРНИП.
- **okpo.** ОКПО.
- **okved.** ОКВЭД.

XML Schema

```
<xsd:complexType name="LegalInformation">
  <xsd:sequence>
    <xsd:element minOccurs="0" name="id" type="xsd:long"/>
    <xsd:element maxOccurs="unbounded" minOccurs="0" name="attribute"
      type="tns:KeyValueApprovedAttribute"/>
  </xsd:sequence>
</xsd:complexType>
```

Описание

id

ID юридических реквизитов.

Необязательный элемент.

Тип: long

attribute

Необязательный элемент.

Тип: [KeyValueApprovedAttribute Complex Type](#)

Money Simple Type

Тип, представляющий значение суммы в операции.

Тип: decimal

XML Schema

```
<xsd:simpleType name="Money">
  <xsd:restriction base="xsd:decimal">
    <xsd:fractionDigits value="2"/>
  </xsd:restriction>
</xsd:simpleType>
```

OperationAmountType Simple Type

Тип операции.

Тип: string

Возможные значения:

- **INCOME**. Приход
- **EXPENSE**. Расход

XML Schema

```
<xsd:simpleType name="OperationAmountType">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="INCOME"/>
    <xsd:enumeration value="EXPENSE"/>
  </xsd:restriction>
</xsd:simpleType>
```

OperationInfo Complex Type

Тип, описывающий операцию в системе МОНЕТА.РУ. Он представляет собой номер (id) операции и набор полей, которые присутствуют у данной операции. Поля представлены в виде "ключ-значение".

XML Schema

```
<xsd:complexType name="OperationInfo">
  <xsd:sequence>
    <xsd:element minOccurs="0" name="id" type="xsd:long"/>
    <xsd:element maxOccurs="unbounded" minOccurs="0" name="attribute"
      type="tns:KeyValueAttribute"/>
  </xsd:sequence>
</xsd:complexType>
```

Описание

id

Номер операции.

Необязательный элемент.

Тип: long

attribute

Поля операции. Полей в операции может быть 0 и более. Список полей:

- **clienttransaction**. Внешний (не в системе МОНЕТА.РУ) номер операции.
- **statusid**. Статус операции.
- **typeid**. Тип операции:
 - **2,10**. Оплата с пользовательского счета.
 - **3,19**. Оплата с внешней платежной системы.
 - **7,14**. Пополнение счета.
 - **4,13**. Вывод на внешнюю платежную систему.
 - **11**. Вывод на пользовательский счет.
 - **12**. Внутренний перевод.
 - **17,18**. Возврат на внешнюю платежную систему.
- **category**. Категория операции:
 - DEPOSIT - ввод средств
 - WITHDRAWAL - вывод средств
 - TRANSFER - внутренний перевод
 - BUSINESS - товары и услуги

- **modified.** Время последнего изменения операции.
- **sourceaccountid.** Номер счета, с которого произведена операция.
- **sourcecurrencycode.** Валюта счета.
- **sourceamount.** Сумма по операции.
- **sourceamountfee.** Сумма комиссии.
- **sourceamounttotal.** Общая сумма с учетом комиссии.
- **targetaccountid.** Корреспондентский счет.
- **targetalias.** Название корреспондентского счета.
- **isreversed.**
 - **true.** sourceaccountid=получатель, targetaccountid=платательщик.
 - **false.** sourceaccountid=платательщик, targetaccountid=получатель.
- **customfield: custom_attribute_name.** Произвольный набор значений. В операции их может быть несколько.

Примечание: Полный ключ атрибута состоит из префикса ("customfield:") и тэга (32 символа). Например, "customfield:name".

Необязательный элемент.

Тип: *KeyValueAttribute Complex Type*

OperationInfoBatchResponseType Complex Type

Тип, описывающий атрибуты операции в ответах в пакетном запросе.

XML Schema

```
<xsd:complexType name="OperationInfoBatchResponseType">
  <xsd:sequence>
    <xsd:element minOccurs="0" name="error" type="xsd:string"/>
    <xsd:element minOccurs="0" name="transaction" type="tns:OperationInfo"/>
    <xsd:element minOccurs="0" name="errorCode" type="xsd:string"/>
  </xsd:sequence>
</xsd:complexType>
```

Описание

error

Поле содержит описание ошибки, если операция не была проведена. В этом случае поле transaction - пустое.

Если операция проведена, то поле error пустое, а поле transaction содержит детали операции.

Необязательный элемент.

Тип: string

transaction

Необязательный элемент.

Тип: *OperationInfo Complex Type*

errorCode

Поле содержит код ошибки. Поле заполнено только тогда, когда есть описание ошибки в поле error.

Для получения этого поля в запросе необходимо выставить атрибут version равный или больше **VERSION_2**.

Коды ошибок совпадают со значением элемента faultDetail (смотрите описание для этого элемента).

Необязательный элемент.

Тип: string

OperationInfoList Complex Type

Тип, представляющий список операций. Содержит разбиение по страницам для отображения длинных списков.

XML Schema

```
<xsd:complexType name="OperationInfoList">
  <xsd:sequence>
    <xsd:element name="pageSize" type="xsd:long"/>
    <xsd:element name="pageNumber" type="xsd:long"/>
    <xsd:element name="pagesCount" type="xsd:long"/>
    <xsd:element name="size" type="xsd:long"/>
    <xsd:element name="totalSize" type="xsd:long"/>
    <xsd:element maxOccurs="unbounded" minOccurs="0" name="operation"
      type="tns:OperationInfo"/>
  </xsd:sequence>
</xsd:complexType>
```

Описание

pageSize

Количество операций, возвращаемых в результате запроса.

Тип: long

pageNumber

Номер текущей страницы. Нумерация начинается с 1.

Тип: long

pagesCount

Максимальное количество страниц с операциями по данному запросу.

Тип: long

size

Количество операций на текущей странице. Меньше или равно pageSize. Последняя страница может содержать операций меньше, чем pageSize.

Тип: long

totalSize

Общее количество операций, которое можно получить в данной выборке.

Тип: long

operation

Список операций.

Необязательный элемент.

Тип: *OperationInfo Complex Type*

OperationStatus Simple Type

Тип, описывающий статусы операций в системе МОНЕТА.РУ. Данный тип может иметь только определенные значения, описанные ниже.

Тип: string

Возможные значения:

- **INPROGRESS**. В обработке.
- **SUCCEED**. Операция обработана.

- **CANCELED.**Операция отменена.
- **TAKENIN_NOTSENT.**Средства зачислены на счет получателя. Уведомление по операции не отправлено (или не принято).
- **CREATED.**Операция создана.
- **FROZEN.**Операция заморожена.
- **TAKENOUT.**Средства списаны.

XML Schema

```
<xsd:simpleType name="OperationStatus">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="INPROGRESS"/>
    <xsd:enumeration value="SUCCEED"/>
    <xsd:enumeration value="CANCELED"/>
    <xsd:enumeration value="TAKENIN_NOTSENT"/>
    <xsd:enumeration value="CREATED"/>
    <xsd:enumeration value="FROZEN"/>
    <xsd:enumeration value="TAKENOUT"/>
  </xsd:restriction>
</xsd:simpleType>
```

OperationStatusState Simple Type

Тип: string

Возможные значения:

- **DEBITED.**Средства списаны.
- **CREDITED.**Средства зачислены.
- **COMPLETED.**Выполнено.
- **FROZEN.**Заморожено.
- **CANCELED.**Отменено.

XML Schema

```
<xsd:simpleType name="OperationStatusState">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="DEBITED"/>
    <xsd:enumeration value="CREDITED"/>
    <xsd:enumeration value="COMPLETED"/>
    <xsd:enumeration value="FROZEN"/>
    <xsd:enumeration value="CANCELED"/>
  </xsd:restriction>
</xsd:simpleType>
```

OperationTemplate Complex Type

Шаблон операции.

Тип: [AbstractAttributeObject Complex Type](#).

XML Schema

```
<xsd:complexType name="OperationTemplate">
  <xsd:complexContent>
    <xsd:extension base="tns:AbstractAttributeObject">
      <xsd:sequence>
        <xsd:element minOccurs="0" name="id" type="xsd:long"/>
        <xsd:element minOccurs="0" name="unitId" type="xsd:long"/>
        <xsd:element minOccurs="0" name="type" type="tns:OperationTemplateType"/>
        <xsd:element minOccurs="0" name="name" type="xsd:string"/>
        <xsd:element minOccurs="0" name="payer" type="xsd:long"/>
        <xsd:element minOccurs="0" name="payee" type="xsd:long"/>
        <xsd:element minOccurs="0" name="operationTypeCategory"
          type="tns:OperationTypeCategory"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

```

<xsd:element minOccurs="0" name="description" type="tns:Description"/>
<xsd:element minOccurs="0" name="prototypeOperationId" type="xsd:long"/>
<xsd:element minOccurs="0" name="tags" type="xsd:string"/>
<xsd:element minOccurs="0" name="favorite" type="xsd:boolean"/>
<xsd:choice minOccurs="0">
  <xsd:element name="commonParameters"
type="tns:CommonOperationTemplateParameters"/>
  <xsd:element name="regularParameters"
type="tns:RegularOperationTemplateParameters"/>
  <xsd:element name="directDebitParameters"
type="tns:DirectDebitOperationTemplateParameters"/>
</xsd:choice>
<xsd:element maxOccurs="unbounded" minOccurs="0" name="operationInfo"
type="tns:KeyValueApprovedAttribute"/>
<xsd:element maxOccurs="unbounded" minOccurs="0" name="additionalInfo"
type="tns:KeyValueApprovedAttribute"/>
</xsd:sequence>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>

```

Описание

id

Идентификатор шаблона операции. При создании шаблона операции (*CreateOperationTemplateRequest*) должен иметь пустое значение.

Необязательный элемент.

Тип: long

unitId

Пользователь, которому принадлежит шаблон операции.

Если при создании шаблона операции это поле не задано, то используется текущий пользователь.

При редактировании шаблона операции это поле передавать нельзя.

Необязательный элемент.

Тип: long

type

Тип шаблона операции.

При редактировании шаблона этот элемент можно не передавать. При создании шаблона этот элемент обязателен.

При редактировании шаблона можно менять его тип. То есть регулярный шаблон можно сделать нерегулярным и наоборот. При смене типа шаблона следует передать все необходимые параметры шаблона: *commonParameters* или *regularParameters*.

Необязательный элемент.

Тип: *OperationTemplateType Simple Type*

name

Название шаблона.

Максимальная длина 250 символов.

При редактировании шаблона этот элемент можно не передавать. При создании шаблона этот элемент обязателен.

Необязательный элемент.

Тип: string

payer

Номер счета плательщика.

Этот элемент следует обязательно передавать при создании нового шаблона. Если шаблон создается из проведенной операции или редактируется, то этот элемент необязателен.

Необязательный элемент.

Тип: long

payee

Номер счета получателя.

Этот элемент следует обязательно передавать при создании нового шаблона. Если шаблон создается из проведенной операции или редактируется, то этот элемент необязателен.

Необязательный элемент.

Тип: long

operationTypeCategory

Категория операции.

Возвращается при чтении шаблона операции.

Необязательный элемент.

Тип: *OperationTypeCategory Simple Type*

description

Описание шаблона.

Максимальная длина 255 символов.

Необязательный элемент.

Тип: *Description Simple Type*

prototypeOperationId

Номер операции, на основе которой был создан шаблон.

Необязательный элемент.

Тип: long

tags

Метки шаблона. Для логической группировки шаблонов и быстрого поиска по нужной метке.

Метки могут содержать русские или английские символы, цифры, символы "_" и "-", пробел.

Метки могут разделяться запятыми. Длина одной метки не может быть больше 128 байт.

Например: "банковские переводы, еженедельные выплаты, сбербанк".

Необязательный элемент.

Тип: string

favorite

Шаблон операции добавлен в категорию "избранное".

Необязательный элемент.

Тип: boolean

commonParameters

Заполняется, если тип шаблона *COMMON*.

При создании шаблона из операции (с указанием *prototypeOperationId*), элемент *commonParameters* можно не указывать - тогда нужная сумма возьмётся из операции.

Если шаблон редактируется, то этот элемент необязателен.

Тип: [CommonOperationTemplateParameters Complex Type](#)

regularParameters

Заполняется, если тип шаблона *REGULAR*.

Этот элемент следует обязательно передавать при создании нового шаблона или шаблона из проведенной операции. Если шаблон редактируется, то этот элемент необязателен.

Тип: [RegularOperationTemplateParameters Complex Type](#)

directDebitParameters

Заполняется, если тип шаблона *DIRECT_DEBIT*.

Тип: [DirectDebitOperationTemplateParameters Complex Type](#)

operationInfo

Набор полей, которые необходимо сохранить в качестве атрибутов операции.

Необязательный элемент.

Тип: [KeyValueApprovedAttribute Complex Type](#)

additionalInfo

Дополнительные поля шаблона операции. Только для чтения.

Например:

- **executionlastdatetime.** Время последнего выполнения шаблона. Может возвращаться, если поиск был по "идентификатору шаблона".
- **executionlastoperationid.** Номер операции, которая была создана во время последнего выполнения шаблона. Может возвращаться, если поиск был по "идентификатору шаблона".
- **executionlastmessage.** Если во время последнего выполнения шаблона была ошибка, то это поле может содержать текст ошибки. Может возвращаться, если поиск был по "идентификатору шаблона".
- **payer_alias.** Название счета плательщика. Может возвращаться, если поиск был по "фильтру".
- **payee_alias.** Название счета получателя. Может возвращаться, если поиск был по "фильтру".
- **payer_currency.** Валюта счета плательщика. Может возвращаться, если поиск был по "фильтру".
- **payee_currency.** Валюта счета получателя. Может возвращаться, если поиск был по "фильтру".

Необязательный элемент.

Тип: [KeyValueApprovedAttribute Complex Type](#)

OperationTemplateAmount Complex Type

Тип: [AbstractAttributeObject Complex Type](#).

XML Schema

```
<xsd:complexType name="OperationTemplateAmount">
  <xsd:complexContent>
    <xsd:extension base="tns:AbstractAttributeObject">
      <xsd:sequence>
        <xsd:element minOccurs="0" name="amount" type="tns:Money"/>
        <xsd:element minOccurs="0" name="isPayerAmount" type="xsd:boolean"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

```
</xsd:complexContent>
</xsd:complexType>
```

Описание

amount

Сумма операции.

Необязательный элемент.

Тип: [Money Simple Type](#)

isPayerAmount

- Если флаг isPayerAmount установлен (**true**), то amount используется как сумма к списанию (в валюте плательщика).
- Если флаг isPayerAmount сброшен (**false**), то amount используется как сумма к зачислению (в валюте получателя).

Необязательный элемент.

Тип: boolean

OperationTemplateAmountInfo Complex Type

Информация о сумме операции в шаблоне.

Тип: [AbstractAttributeObject Complex Type](#).

XML Schema

```
<xsd:complexType name="OperationTemplateAmountInfo">
  <xsd:complexContent>
    <xsd:extension base="tns:AbstractAttributeObject">
      <xsd:sequence>
        <xsd:element name="type" type="tns:OperationTemplateAmountInfoType"/>
        <xsd:choice minOccurs="0">
          <xsd:element name="amount" type="tns:OperationTemplateAmount"/>
          <xsd:element name="range" type="tns:OperationTemplateAmountRange"/>
          <xsd:element name="rest" type="tns:OperationTemplateAmountRest"/>
        </xsd:choice>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Описание

type

Тип суммы в шаблоне операции.

Тип: [OperationTemplateAmountInfoType Simple Type](#)

amount

Указывается, если тип суммы *OperationTemplateAmountInfoType.AMOUNT*.

Тип: [OperationTemplateAmount Complex Type](#)

range

Указывается, если тип суммы *OperationTemplateAmountInfoType.RANGE*.

Тип: [OperationTemplateAmountRange Complex Type](#)

rest

Указывается, если тип суммы *OperationTemplateAmountInfoType.REST*.

Тип: [OperationTemplateAmountRest Complex Type](#)

OperationTemplateAmountInfoType Simple Type

Тип суммы в шаблоне операции.

Тип: string

Возможные значения:

- **AMOUNT.**

Во время выполнения шаблона будет использоваться указанная сумма.

При использовании данного типа внутри *OperationTemplateAmountInfo*, необходимо дополнительно указать поле *amount*.

- **BALANCE.**

Сумма к списанию будет равна всей доступной сумме на Вашем счёте.

- **PAYMENTS.**

Сумма к списанию будет равна платёжному сальдо, в зависимости от периода оплаты.

- **RANGE.**

Сумма к списанию будет равна всей доступной сумме на Вашем счёте, если значение этой суммы на момент выполнения операции будет входить в заданный интервал.

При использовании данного типа внутри *OperationTemplateAmountInfo*, необходимо дополнительно указать поле *range*.

- **REST.**

Сумма к списанию будет равна всей доступной сумме на Вашем счёте за вычетом указанного остатка.

При использовании данного типа внутри *OperationTemplateAmountInfo*, необходимо дополнительно указать поле *rest*.

- **CREDIT.**

Сумма к списанию будет равна сумме всех зачислений на счёт, в зависимости от периода оплаты.

XML Schema

```
<xsd:simpleType name="OperationTemplateAmountInfoType">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="AMOUNT"/>
    <xsd:enumeration value="BALANCE"/>
    <xsd:enumeration value="PAYMENTS"/>
    <xsd:enumeration value="RANGE"/>
    <xsd:enumeration value="REST"/>
    <xsd:enumeration value="CREDIT"/>
  </xsd:restriction>
</xsd:simpleType>
```

OperationTemplateAmountRange Complex Type

Тип: [AbstractAttributeObject Complex Type](#).

XML Schema

```
<xsd:complexType name="OperationTemplateAmountRange">
  <xsd:complexContent>
    <xsd:extension base="tns:AbstractAttributeObject">
      <xsd:sequence>
        <xsd:element minOccurs="0" name="amountMinValue" type="tns:Money"/>
        <xsd:element minOccurs="0" name="amountMaxValue" type="tns:Money"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Описание

amountMinValue

Минимальное значение баланса на счете списания.

Необязательный элемент.

Тип: *Money Simple Type*

amountMaxValue

Максимальное значение баланса на счете списания.

Необязательный элемент.

Тип: *Money Simple Type*

OperationTemplateAmountRest Complex Type

Тип: *AbstractAttributeObject Complex Type*.

XML Schema

```
<xsd:complexType name="OperationTemplateAmountRest">
  <xsd:complexContent>
    <xsd:extension base="tns:AbstractAttributeObject">
      <xsd:sequence>
        <xsd:element name="amount" type="tns:Money"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Описание

amount

Остаток баланса, который должен остаться на счете после выполнения шаблона операции.

Тип: *Money Simple Type*

OperationTemplateReminderInfo Complex Type

Напоминание о выполнении регулярного шаблона.

Если в разделе "Мой счет" - "Персональные данные" - "Уведомления", добавить уведомление с флагом "Регулярные платежи", то можно получать напоминания о начале выполнения шаблона.

Тип: *AbstractAttributeObject Complex Type*.

XML Schema

```
<xsd:complexType name="OperationTemplateReminderInfo">
  <xsd:complexContent>
    <xsd:extension base="tns:AbstractAttributeObject">
      <xsd:sequence>
        <xsd:element name="remind" type="xsd:boolean"/>
        <xsd:element minOccurs="0" name="hoursBeforeExecution" type="xsd:int"/>
        <xsd:element maxOccurs="10" minOccurs="0" name="notification"
          type="tns:ProfileNotificationSelection"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Описание

remind

- **true** - отсылать напоминания.
- **false** - не отсылать напоминания.

Тип: boolean

hoursBeforeExecution

За сколько часов до выполнения шаблона высылать напоминание.

Необязательный элемент.

Тип: int

notification

Список уведомлений, куда уходят напоминания. Только для чтения.

Необязательный элемент.

Тип: [ProfileNotificationSelection Complex Type](#)

OperationTemplateTimeInfo Complex Type

Информация о времени выполнения шаблона.

Тип: [AbstractAttributeObject Complex Type](#).

XML Schema

```
<xsd:complexType name="OperationTemplateTimeInfo">
  <xsd:complexContent>
    <xsd:extension base="tns:AbstractAttributeObject">
      <xsd:sequence>
        <xsd:element name="type" type="tns:OperationTemplateTimeInfoType"/>
        <xsd:element name="startDateTime" type="xsd:dateTime"/>
        <xsd:element minOccurs="0" name="endDateTime" type="xsd:dateTime"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Описание

type

Периодичность выполнения шаблона.

Тип: [OperationTemplateTimeInfoType Simple Type](#)

startDateTime

Начало работы шаблона.

Время, которое здесь указывается, должно быть кратно часу. Если будут указаны минуты, то они будут "обрезаны". Например, если указать, что время выполнения шаблона каждый день в 13:29, то шаблон будет выполняться каждый день в 13:00.

Время, которое здесь указывается, должно быть больше текущего времени.

Тип: dateTime

endDateTime

Дата и время окончания работы шаблона. Необязательное поле.

Необязательный элемент.

Тип: dateTime

OperationTemplateTimeInfoType Simple Type

Периодичность выполнения шаблона.

Тип: string

Возможные значения:

- **ONCE.**Один раз.
- **EVERY_DAY.**Каждый день.
- **EVERY_WORKDAY.**Каждый рабочий день.
- **EVERY_WEEK.**Каждую неделю.
- **EVERY_MONTH.**Каждый месяц.
- **EVERY_LAST_DAY_OF_MONTH.**Каждый последний день месяца.
- **EVERY_YEAR.**Каждый год.

XML Schema

```
<xsd:simpleType name="OperationTemplateTimeInfoType">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="ONCE"/>
    <xsd:enumeration value="EVERY_DAY"/>
    <xsd:enumeration value="EVERY_WORKDAY"/>
    <xsd:enumeration value="EVERY_WEEK"/>
    <xsd:enumeration value="EVERY_MONTH"/>
    <xsd:enumeration value="EVERY_LAST_DAY_OF_MONTH"/>
    <xsd:enumeration value="EVERY_YEAR"/>
  </xsd:restriction>
</xsd:simpleType>
```

OperationTemplateType Simple Type

Тип шаблона операции.

Тип: string

Возможные значения:

- **COMMON.**

Шаблон операции не будет выполняться на регулярной основе. Для таких шаблонов нельзя указать время выполнения. Шаблон можно выполнить только вручную.

При создании такого типа шаблонов, необходимо указать элемент *commonParameters*.

- **REGULAR.**

Шаблон операции будет выполняться автоматически в указанное время.

При создании такого типа шаблонов, необходимо указать элемент *regularParameters*.

- **DIRECT_DEBIT.**Шаблоны прямого дебетования. Такие типы шаблонов только для чтения. Их нельзя создать через MONETA.MerchantAPI.

XML Schema

```
<xsd:simpleType name="OperationTemplateType">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="COMMON"/>
    <xsd:enumeration value="REGULAR"/>
    <xsd:enumeration value="DIRECT_DEBIT"/>
  </xsd:restriction>
</xsd:simpleType>
```

OperationTypeCategory Simple Type

Категория операции.

Тип: string

Возможные значения:

- **DEPOSIT.**Ввод средств.
- **WITHDRAWAL.**Вывод средств.

- **TRANSFER.**Внутренний перевод.
- **BUSINESS.**Товары и услуги.

XML Schema

```
<xsd:simpleType name="OperationTypeCategory">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="DEPOSIT"/>
    <xsd:enumeration value="WITHDRAWAL"/>
    <xsd:enumeration value="TRANSFER"/>
    <xsd:enumeration value="BUSINESS"/>
  </xsd:restriction>
</xsd:simpleType>
```

Pager Complex Type

Тип, позволяющий задать необходимую страницу при отображении длинных списков.

XML Schema

```
<xsd:complexType name="Pager">
  <xsd:sequence>
    <xsd:element default="1" name="pageNumber">
      <xsd:simpleType>
        <xsd:restriction base="xsd:int">
          <xsd:minInclusive value="1"/>
        </xsd:restriction>
      </xsd:simpleType>
    </xsd:element>
    <xsd:element default="25" name="pageSize">
      <xsd:simpleType>
        <xsd:restriction base="xsd:int">
          <xsd:minInclusive value="1"/>
          <xsd:maxInclusive value="1000"/>
        </xsd:restriction>
      </xsd:simpleType>
    </xsd:element>
  </xsd:sequence>
</xsd:complexType>
```

Описание

pageNumber

Номер страницы, которую нужно получить.

Минимальное значение равно 1.

Значение по умолчанию равно 1.

Тип: int

Значение по умолчанию: 1

pageSize

Сколько записей необходимо получить на одной странице.

Минимальное значение равно 1.

Максимальное значение равно 1000.

Значение по умолчанию равно 25.

Тип: int

Значение по умолчанию: 25

Password Simple Type

Строка, представляющая различные виды паролей.

Тип: `normalizedString`

Максимальная длина: 255

XML Schema

```
<xsd:simpleType name="Password">
  <xsd:restriction base="xsd:normalizedString">
    <xsd:maxLength value="255"/>
  </xsd:restriction>
</xsd:simpleType>
```

PaymentBatchRequestType Complex Type

Тип, описывающий параметры операции в запросах в пакетном режиме.

Тип: *EntityBatchRequestType Complex Type*.

XML Schema

```
<xsd:complexType name="PaymentBatchRequestType">
  <xsd:complexContent>
    <xsd:extension base="tns:EntityBatchRequestType">
      <xsd:sequence>
        <xsd:element maxOccurs="unbounded" minOccurs="1" name="transaction"
          type="tns:PaymentRequestType"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Описание

transaction

Набор операций, которые необходимо выполнить в одном пакете. Операции выполняются в том порядке, в котором они переданы в запросе.

Обязательный элемент.

Тип: *PaymentRequestType Complex Type*

PaymentPassword Complex Type

Платежный пароль счета плательщика.

Тип: *AbstractAttributeObject Complex Type*.

XML Schema

```
<xsd:complexType name="PaymentPassword">
  <xsd:complexContent>
    <xsd:extension base="tns:AbstractAttributeObject">
      <xsd:sequence>
        <xsd:element name="paymentPassword" type="tns:Password"/>
        <xsd:element minOccurs="0" name="paymentPasswordChallenge"
          type="xsd:string"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Описание

paymentPassword

Платежный пароль.

Тип: *Password Simple Type*

paymentPasswordChallenge

Запрос для платежного пароля.

Необязательный элемент.

Тип: string

PaymentRequestType Complex Type

Тип, описывающий параметры операции в запросах. Отличается от TransactionRequestType тем, что в качестве получателя можно указывать не только номер счета.

Тип: *Entity Complex Type*.

XML Schema

```

<xsd:complexType name="PaymentRequestType">
  <xsd:complexContent>
    <xsd:extension base="tns:Entity">
      <xsd:sequence>
        <xsd:element name="payer" type="xsd:string"/>
        <xsd:element name="payee" type="xsd:string"/>
        <xsd:element minOccurs="0" name="amount" type="tns:Money"/>
        <xsd:element minOccurs="0" name="isPayerAmount" type="xsd:boolean"/>
        <xsd:element minOccurs="0" name="paymentPassword" type="tns:Password"/>
        <xsd:element minOccurs="0" name="clientTransaction" type="tns:CTID"/>
        <xsd:element minOccurs="0" name="description" type="tns:Description"/>
        <xsd:element minOccurs="0" name="operationInfo" type="tns:OperationInfo"/>
        <xsd:element minOccurs="0" name="paymentPasswordChallenge"
type="xsd:string"/>
        <xsd:element minOccurs="0" name="personalInformation"
type="tns:PersonalInformation"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>

```

Описание

payer

Номер счета плательщика.

Тип: string

payee

Может принимать следующие значения:

- Е-mail получателя. Сумма зачисляется на "основной счет" получателя, найденного по данному Е-mail.
- Номер операции - если значение этого поля начинается с "0". То есть значение 012345678 означает оплату операции 12345678.
- Телефон получателя - если значение этого поля начинается со знака "+". Сумма зачисляется на "основной счет" получателя, найденного по данному телефону.
- Номер счета получателя

Тип: string

amount

Сумма операции.

- Если флаг isPayerAmount отсутствует и пользователь имеет доступ только к счету плательщика, то интерпретируется как сумма к списанию (в валюте плательщика).
- Если флаг isPayerAmount отсутствует и пользователь имеет доступ только к счету получателя, то интерпретируется как сумма к зачислению (в валюте получателя).

Необязательный элемент.

Тип: *Money Simple Type*

isPayerAmount

Если пользователь имеет доступ как к счету плательщика, так и к счету получателя, то флаг isPayerAmount обязателен.

- Если флаг isPayerAmount установлен (true), то amount используется как сумма к списанию (в валюте плательщика).
- Если флаг isPayerAmount сброшен (false), то amount используется как сумма к зачислению (в валюте получателя).

Необязательный элемент.

Тип: boolean

paymentPassword

Платежный пароль счета плательщика.

Необязательный элемент.

Тип: *Password Simple Type*

clientTransaction

Внешний номер операции.

Необязательный элемент.

Тип: *CTID Simple Type*

description

Описание операции.

Необязательный элемент.

Тип: *Description Simple Type*

operationInfo

Набор полей, которые необходимо сохранить в качестве атрибутов операции. Значения дат в формате dd.MM.yyyy HH:mm:ss

Необязательный элемент.

Тип: *OperationInfo Complex Type*

paymentPasswordChallenge

Запрос для платежного пароля.

Необязательный элемент.

Тип: string

personalInformation

Персональные данные пользователя.

Необязательный элемент.

Тип: *PersonalInformation Complex Type*

PaymentSystemInfoComplexType Complex Type

Тип, описывающий представления информации о платежной системе.

XML Schema

```
<xsd:complexType name="PaymentSystemInfoComplexType">
  <xsd:sequence>
    <xsd:element minOccurs="0" name="unitId" type="xsd:long"/>
    <xsd:element minOccurs="0" name="name" type="xsd:string"/>
    <xsd:element minOccurs="0" name="icon" type="xsd:string"/>
    <xsd:element minOccurs="0" name="logicalGroup" type="xsd:string"/>
    <xsd:element minOccurs="0" name="infoUrl" type="tns:InfoUrl"/>
    <xsd:element minOccurs="0" name="referenceData" type="tns:ReferenceData"/>
    <xsd:element minOccurs="0" name="infoTariff" type="tns:InfoTariff"/>
    <xsd:element minOccurs="0" name="currencies" type="xsd:string"/>
    <xsd:element minOccurs="0" name="psAccountIds" type="xsd:string"/>
  </xsd:sequence>
</xsd:complexType>
```

Описание

unitId

ID пользователя в системе МОНЕТА.РУ.

Необязательный элемент.

Тип: long

name

Наименование платежной системы.

Необязательный элемент.

Тип: string

icon

URL логотипа платежной системы.

Необязательный элемент.

Тип: string

logicalGroup

Наименование логической группы, к которой принадлежит платежная система.

Необязательный элемент.

Тип: string

infoUrl

Расширенная информация об URL платежной системы.

Необязательный элемент.

Тип: *InfoUrl Complex Type*

referenceData

Справочная информация о платежной системе.

Необязательный элемент.

Тип: *ReferenceData Complex Type*

infoTariff

Краткая информация о тарифе.

Необязательный элемент.

Тип: *InfoTariff Complex Type*

currencies

Список валюты счетов платежной системы. Располагаются в том же порядке, что и идентификаторы счетов в psAccountIds.

Необязательный элемент.

Тип: string

psAccountIds

Список счетов платежной системы. Располагаются в том же порядке, что и валюты счетов в currencies.

Необязательный элемент.

Тип: string

PersonalInformation Complex Type

Персональные данные пользователя.

XML Schema

```
<xsd:complexType name="PersonalInformation">
  <xsd:sequence>
    <xsd:element name="profile" type="tns:Profile"/>
    <xsd:element minOccurs="0" name="document" type="tns:Document"/>
  </xsd:sequence>
</xsd:complexType>
```

Profile Complex Type

Данные пользователя в системе МОНЕТА.РУ. Данные представляются в виде "ключ-значение" и признака подтвержденности. Можно получить доступ либо к собственным, либо к публично доступным данным (то есть к тем данным, которые пользователь сам разрешил для просмотра).

Возможные ключи для пользователей с типом "client":

- **unitid.** ID пользователя в системе МОНЕТА.РУ.
- **last_name.** Фамилия.
- **first_name.** Имя.
- **middle_initial_name.** Отчество.
- **alias.** Псевдоним.
- **country.** Страна.
- **state.** Область, республика, штат.
- **city.** Город.
- **zip.** Индекс.
- **address.** Адрес.
- **email_for_notifications.** Адрес электронной почты.
- **phone.** Номер телефона.
- **cell_phone.** Номер сотового телефона.
- **url.** Url сайта.
- **sex.** Пол. MALE - мужской, FEMALE - женский.
- **date_of_birth.** Дата рождения (формат: уууу-mm-dd).
- **inn.** Номер ИНН.
- **snils.** СНИЛС.
- **timezone.** Часовой пояс пользователя.

Примечание: Список идентификаторов временных зон, которые используются в системе, доступен на странице wikipedia.org.

- **ui_language.** Язык интерфейса пользователя. RU - русский, EN - английский.
- **customfield:*** Произвольный набор значений. В данных пользователя их может быть несколько. Полный ключ атрибута состоит из префикса ("customfield:") и тэга (32 символа). Например, "customfield:name".

Возможные ключи для пользователей с типом "organization":

- **unitid.** ID пользователя в системе МОНЕТА.РУ.

- **rf_resident.** Тип партнера [резидент/нерезидент], если нет значения - партнер старого формата, записывается при регистрации партнера.
- **goal_and_business_relationships.** Цель установления и предполагаемый характер деловых отношений с НКО.
- **planned_turnovers.** Планируемые ежемесячные обороты.
- **planned_payers.** Планируемые плательщики.
- **tariff.** Тарифный план [PAY_MYSELF, PAY_TOGETHER, ZKH, NOT_RF_RESIDENT], задается при регистрации партнера.
- **incorporation_form_ru.** Вид партнера [организация или ИП], задается при регистрации партнера-нерезидента.
- **international_name.** Название на иностранном языке.
- **alias.** Псевдоним.
- **organization_name.** Полное название организации или ФИО индивидуального предпринимателя, задается при регистрации партнера.
- **organization_name_short.** Сокращенное название организации, задается при регистрации партнера.
- **position_director.** Должность руководителя (для организации).
- **position_director_details.** Детали для должности директора (для организации).
- **acting_document.** Документ, на основании которого действует директор (для организации).
- **acting_document_details.** Подробности по документу, на основании которого действует директор (для организации).
- **attorney_number_ru.** Номер (доверенность, для организации).
- **attorney_date_from_ru.** Дата от (доверенность, для организации).
- **attorney_date_to_ru.** Дата срок действия до (доверенность, для организации).
- **agreement_signer_fio.** ФИО для подписи в договоре (для организации).
- **registration_date_ru.** Дата (свидетельство о регистрации).
- **registration_series_ru.** Серия (свидетельство о регистрации).
- **registration_number_ru.** Номер (свидетельство о регистрации).
- **registration_authority_ru.** Наименование регистрирующего органа (свидетельство о регистрации).
- **registration_state_place_ru.** Место государственной регистрации (свидетельство о регистрации).
- **agreement_signer_fio_ie.** ФИО для подписи в договоре (для ИП).
- **fio_accountant.** ФИО контакта по финансовым вопросам.
- **position_accountant.** Должность контакта по финансовым вопросам.
- **fio_contact.** ФИО контакта по техническим вопросам.
- **position_contact.** Должность контакта по техническим вопросам.
- **joint_governing_body.** Коллегиальный орган управления (исполнительный орган).
- **joint_body_members.** ФИО и должность членов коллегиального органа.
- **business_activity.** Вид деятельности для категории "Другое".
- **url.** Url сайта, задается при регистрации партнера.
- **where_from_do_you_know.** Откуда Вы узнали.
- **where_from_do_you_know_details.** Подробности, откуда Вы узнали.
- **promocode.** Промо-код, задается при регистрации партнера.
- **inn.** Номер ИНН, задается при регистрации партнера-резидента.
- **fio_director.** ФИО руководителя.
- **phone_director.** Контактный телефон директора.
- **country** Страна. Использовать isocode [RUS], isocode2L [RU] или имя [Россия]
- **legal_address.** Адрес постоянно действующего органа управления.
- **post_address.** Почтовый адрес.
- **phone_contact.** Контактный телефон.
- **phone_accountant.** Телефон по финансовым вопросам.
- **phone_support.** Телефон по техническим вопросам.
- **fax.** Факс.
- **contact_email.** Контактный e-mail, задается при регистрации партнера.
- **finance_email.** E-mail финансовой службы.
- **technical_email.** E-mail технической поддержки.

- **timezone.** Часовой пояс пользователя.

Примечание: Список идентификаторов временных зон, которые используются в системе, доступен на странице wikipedia.org.

- **ui_language.** язык интерфейса пользователя. RU - русский, EN - английский.
- **email_for_notifications.** Адрес электронной почты.
- **conditions_payee.** Статус "На сайте присутствует информация о получателе платежей, доступная без регистрации плательщика".
- **conditions_payer.** Статус "На сайте присутствует контактная информация для плательщиков, доступная без регистрации".
- **conditions_site.** Статус "Сайт полностью функционален и заполнен контентом".
- **conditions_payment_info.** Статус "Описание порядка и процесса оплаты, а также информация о временном промежутке между оплатой и отгрузкой (доставкой) товара или предоставлением услуги доступны без регистрации".
- **conditions_correct_data.** Статус "Все необходимые данные внесены и являются актуальными".
- **conditions_payee_details.** Подробности для "На сайте присутствует информация о получателе платежей, доступная без регистрации плательщика".
- **conditions_payer_details.** Подробности для "На сайте присутствует контактная информация для плательщиков, доступная без регистрации".
- **conditions_site_details.** Подробности для "Сайт полностью функционален и заполнен контентом".
- **conditions_payment_info_details.** Подробности для "Описание порядка и процесса оплаты, а также информация о временном промежутке между оплатой и отгрузкой (доставкой) товара или предоставлением услуги доступны без регистрации".
- **conditions_correct_data_details.** Подробности для "Все необходимые данные внесены и являются актуальными".
- **agreement_sent_date.** Дата отправки Заявления.
- **agreement_sent_method.** Способ отправки Заявления.
- **capital_type.** Вид капитала.
- **registered_capital_size.** Размер зарегистрированного капитала.
- **paid_capital_size.** Размер оплаченного капитала.
- **budget_arrears_absense.** Сведения о наличии задолженности перед бюджетом по налогам и сборам.
- **customfield:*** Произвольный набор значений. В данных пользователя их может быть несколько. Полный ключ атрибута состоит из префикса ("customfield:") и тэга (32 символа). Например, "customfield:name".

XML Schema

```
<xsd:complexType name="Profile">
  <xsd:sequence>
    <xsd:element maxOccurs="unbounded" minOccurs="0" name="attribute"
      type="tns:KeyValueApprovedAttribute"/>
  </xsd:sequence>
</xsd:complexType>
```

ProfileNotification Complex Type

Тип: *AbstractAttributeObject Complex Type*.

XML Schema

```
<xsd:complexType name="ProfileNotification">
  <xsd:complexContent>
    <xsd:extension base="tns:AbstractAttributeObject">
      <xsd:sequence>
        <xsd:element minOccurs="0" name="id" type="xsd:long"/>
        <xsd:element minOccurs="0" name="unitId" type="xsd:long"/>
        <xsd:element minOccurs="0" name="type" type="tns:ProfileNotificationType"/>
        <xsd:element maxOccurs="10" minOccurs="0" name="recipient"
          type="xsd:string"/>
        <xsd:element maxOccurs="unbounded" minOccurs="0" name="flag"
```

```
type="tns:ProfileNotificationFlag"/>
    </xsd:sequence>
  </xsd:extension>
</xsd:complexContent>
</xsd:complexType>
```

Описание

id

Идентификатор уведомления.

Необязательный элемент.

Тип: long

unitId

Пользователь, которому принадлежат уведомления.

Необязательный элемент.

Тип: long

type

Тип уведомления.

Необязательный элемент.

Тип: [ProfileNotificationType Simple Type](#)

recipient

Список получателей уведомлений.

Необязательный элемент.

Тип: string

flag

Флаги уведомлений.

Необязательный элемент.

Тип: [ProfileNotificationFlag Complex Type](#)

ProfileNotificationFlag Complex Type

Тип: [AbstractAttributeObject Complex Type](#).

XML Schema

```
<xsd:complexType name="ProfileNotificationFlag">
  <xsd:complexContent>
    <xsd:extension base="tns:AbstractAttributeObject">
      <xsd:sequence>
        <xsd:element name="flag" type="tns:ProfileNotificationFlagType"/>
        <xsd:element name="enabled" type="xsd:boolean"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Описание

flag

Флаг уведомлений.

Тип: [ProfileNotificationFlagType Simple Type](#)

enabled

- **true** - флаг уведомлений установлен.

- **false** - флаг уведомлений не установлен.

Тип: boolean

ProfileNotificationFlagType Simple Type

Флаг уведомлений.

Тип: string

XML Schema

```
<xsd:simpleType name="ProfileNotificationFlagType">
  <xsd:restriction base="xsd:string"/>
</xsd:simpleType>
```

ProfileNotificationSelection Complex Type

Тип: *ProfileNotification Complex Type*.

XML Schema

```
<xsd:complexType name="ProfileNotificationSelection">
  <xsd:complexContent>
    <xsd:extension base="tns:ProfileNotification">
      <xsd:sequence>
        <xsd:element name="selected" type="xsd:boolean"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Описание

selected

Выбрано данное уведомление или нет.

Тип: boolean

ProfileNotificationType Simple Type

Тип уведомлений.

Тип: string

Возможные значения:

- **EMAIL**

XML Schema

```
<xsd:simpleType name="ProfileNotificationType">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="EMAIL"/>
  </xsd:restriction>
</xsd:simpleType>
```

ProfileType Simple Type

Тип данных пользователя.

Тип: string

Возможные значения:

- **organization**. Тип данных "Организация". В данных такого типа присутствуют поля, характерные для организаций, например: "Название организации", "ФИО директора" и т.п.

- **client**. Тип данных "Клиент". В данных такого типа присутствуют поля, характерные для обычных пользователей, например: "фамилия", "имя", "отчество" и т.д.

XML Schema

```
<xsd:simpleType name="ProfileType">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="organization"/>
    <xsd:enumeration value="client"/>
  </xsd:restriction>
</xsd:simpleType>
```

ReferenceData Complex Type

Справочная информация о платежной системе.

XML Schema

```
<xsd:complexType name="ReferenceData">
  <xsd:sequence>
    <xsd:element minOccurs="0" name="period" type="xsd:string"/>
    <xsd:element minOccurs="0" name="percentage" type="xsd:string"/>
    <xsd:element minOccurs="0" name="percentageExt" type="xsd:string"/>
    <xsd:element minOccurs="0" name="sourceFee" type="tns:Fee"/>
    <xsd:element minOccurs="0" name="sourceFeeExt" type="tns:Fee"/>
    <xsd:element minOccurs="0" name="targetFee" type="tns:Fee"/>
    <xsd:element minOccurs="0" name="targetFeeExt" type="tns:Fee"/>
  </xsd:sequence>
</xsd:complexType>
```

Описание

period

Срок проведения банковской операции.

Необязательный элемент.

Тип: string

percentage

Комиссия в процентах.

Необязательный элемент.

Тип: string

percentageExt

Внешняя комиссия в процентах.

Необязательный элемент.

Тип: string

sourceFee

Комиссия с плательщика (сумма).

Необязательный элемент.

Тип: *Fee Simple Type*

sourceFeeExt

Внешняя комиссия с плательщика (сумма).

Необязательный элемент.

Тип: *Fee Simple Type*

targetFee

Комиссия с получателя платежа (сумма).

Необязательный элемент.

Тип: *Fee Simple Type*

targetFeeExt

Внешняя комиссия с получателя платежа (сумма).

Необязательный элемент.

Тип: *Fee Simple Type*

RegularOperationTemplateParameters Complex Type

Параметры шаблона операции, если он имеет тип *OperationTemplateType.REGULAR*.

Тип: *AbstractAttributeObject Complex Type*.

XML Schema

```
<xsd:complexType name="RegularOperationTemplateParameters">
  <xsd:complexContent>
    <xsd:extension base="tns:AbstractAttributeObject">
      <xsd:sequence>
        <xsd:element minOccurs="0" name="amountInfo"
          type="tns:OperationTemplateAmountInfo"/>
        <xsd:element minOccurs="0" name="timeInfo"
          type="tns:OperationTemplateTimeInfo"/>
        <xsd:element minOccurs="0" name="reminderInfo"
          type="tns:OperationTemplateReminderInfo"/>
        <xsd:element maxOccurs="10" minOccurs="0"
          name="operationsReportNotifications"
          type="tns:ProfileNotificationSelection"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Описание

amountInfo

Информация о сумме операции в шаблоне.

Этот элемент следует обязательно передавать при создании нового шаблона или шаблона из проведенной операции. Если шаблон редактируется, то этот элемент необязателен.

Необязательный элемент.

Тип: *OperationTemplateAmountInfo Complex Type*

timeInfo

Информация о времени выполнения шаблона.

Этот элемент следует обязательно передавать при создании нового шаблона или шаблона из проведенной операции. Если шаблон редактируется, то этот элемент необязателен.

Необязательный элемент.

Тип: *OperationTemplateTimeInfo Complex Type*

reminderInfo

Напоминание о выполнении регулярного шаблона.

Необязательный элемент.

Тип: *OperationTemplateReminderInfo Complex Type*

operationsReportNotifications

Список e-mail уведомлений. Уведомления могут иметь только тип *ProfileNotificationType.EMAIL*.

Уведомления используются только:

- При выводе в банк.
- Если сумма имеет тип (*regularParameters.amountInfo.type*):
 - *OperationTemplateAmountInfoType.PAYMENTS*
 - *OperationTemplateAmountInfoType.CREDIT*

В этом случае, на указанные уведомления, будет приходиться список операций, которые вошли в период вывода.

Для указания уведомлений (при создании или редактировании шаблона), следует передать *id* уведомления и поле *selected = true*.

Необязательный элемент.

Тип: *ProfileNotificationSelection Complex Type*

Report Complex Type

Отчет без указания конкретной даты.

XML Schema

```
<xsd:complexType name="Report">
  <xsd:sequence>
    <xsd:element minOccurs="0" name="id" type="xsd:long"/>
    <xsd:element minOccurs="0" name="name" type="xsd:string"/>
    <xsd:element name="unitId" type="xsd:long"/>
    <xsd:element name="typeId" type="xsd:long"/>
    <xsd:element maxOccurs="unbounded" minOccurs="0" name="reportInstance"
      type="tns:ReportInstance"/>
    <xsd:element maxOccurs="unbounded" minOccurs="0" name="attribute"
      type="tns:KeyValueApprovedAttribute"/>
  </xsd:sequence>
</xsd:complexType>
```

Описание

id

ID отчета.

Необязательный элемент.

Тип: long

name

Название отчета.

Необязательный элемент.

Тип: string

unitId

ID пользователя в системе МОНЕТА.РУ.

Тип: long

typeId

Тип отчета.

Пока имеет одно значение: 3.

Тип: long

reportInstance

Список экземпляров отчета.

Необязательный элемент.

Тип: *ReportInstance Complex Type*

attribute

Необязательный элемент.

Тип: *KeyValueApprovedAttribute Complex Type*

ReportInstance Complex Type

Экземпляр отчета за конкретный месяц в системе МОНЕТА.РУ.

XML Schema

```
<xsd:complexType name="ReportInstance">
  <xsd:sequence>
    <xsd:element minOccurs="0" name="id" type="xsd:long"/>
    <xsd:element name="reportId" type="xsd:long"/>
    <xsd:element name="year" type="xsd:int"/>
    <xsd:element name="month" type="xsd:int"/>
    <xsd:element minOccurs="0" name="url" type="xsd:string"/>
    <xsd:element minOccurs="0" name="urlExpirationDate" type="xsd:dateTime"/>
    <xsd:element maxOccurs="unbounded" minOccurs="0" name="attribute"
      type="tns:KeyValueApprovedAttribute"/>
  </xsd:sequence>
</xsd:complexType>
```

Описание

id

ID экземпляра отчета.

Необязательный элемент.

Тип: long

reportId

ID отчета.

Тип: long

year

Год.

Тип: int

month

Месяц (1-12).

Тип: int

url

Ссылка, по которой можно скачать данный экземпляр отчета в формате PDF.

Ссылка для скачивания будет сгенерирована только в том случае, если у пользователя, который вызывает данный метод, в разделе "Мой счет" - "Безопасность" создан "Публичный идентификатор".

Ссылка для скачивания действует в течении 30 минут. В элементе urlExpirationDate указано время действия ссылки.

Если время действия ссылки закончилось, то в ответ будет отдаваться HTTP status code: 400 (Bad Request).

Если в ссылке для скачивания будут изменены параметры, то в ответ будет отдаваться HTTP status code: 400 (Bad Request).

Необязательный элемент.

Тип: string

urlExpirationDate

Время действия ссылки для скачивания отчета.

Необязательный элемент.

Тип: dateTime

attribute

Необязательный элемент.

Тип: *KeyValueApprovedAttribute Complex Type*

TransactionBatchRequestType Complex Type

Тип, описывающий параметры операции в запросах в пакетном режиме.

Тип: *EntityBatchRequestType Complex Type*.

XML Schema

```
<xsd:complexType name="TransactionBatchRequestType">
  <xsd:complexContent>
    <xsd:extension base="tns:EntityBatchRequestType">
      <xsd:sequence>
        <xsd:element maxOccurs="unbounded" minOccurs="1" name="transaction"
          type="tns:TransactionRequestType"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Описание

transaction

Набор операций, которые необходимо выполнить в одном пакете. Операции выполняются в том порядке, в котором они переданы в запросе.

Обязательный элемент.

Тип: *TransactionRequestType Complex Type*

TransactionBatchResponseType Complex Type

Тип, описывающий атрибуты операции в ответах в пакетном запросе.

XML Schema

```
<xsd:complexType name="TransactionBatchResponseType">
  <xsd:sequence>
    <xsd:element minOccurs="0" name="error" type="xsd:string"/>
    <xsd:element minOccurs="0" name="transaction" type="tns:TransactionResponseType"/>
  >
  <xsd:element minOccurs="0" name="errorCode" type="xsd:string"/>
</xsd:sequence>
</xsd:complexType>
```

Описание

error

Поле содержит описание ошибки, если операция не была проведена. В этом случае поле transaction - пустое. Если операция проведена, то поле error пустое, а поле transaction содержит детали операции.

Необязательный элемент.

Тип: string

transaction

Необязательный элемент.

Тип: [TransactionResponseType Complex Type](#)

errorCode

Поле содержит код ошибки. Поле заполнено только тогда, когда есть описание ошибки в поле error.

Для получения этого поля в запросе необходимо выставить атрибут version равный или больше **VERSION_2**.

Коды ошибок совпадают со значением элемента faultDetail (смотрите описание для этого элемента).

Необязательный элемент.

Тип: string

TransactionRequestType Complex Type

Тип, описывающий параметры операции в запросах.

Тип: [Entity Complex Type](#).

XML Schema

```
<xsd:complexType name="TransactionRequestType">
  <xsd:complexContent>
    <xsd:extension base="tns:Entity">
      <xsd:sequence>
        <xsd:element name="payer" type="xsd:string"/>
        <xsd:element name="payee" type="xsd:string"/>
        <xsd:element minOccurs="0" name="amount" type="tns:Money"/>
        <xsd:element minOccurs="0" name="isPayerAmount" type="xsd:boolean"/>
        <xsd:element minOccurs="0" name="paymentPassword" type="tns:Password"/>
        <xsd:element minOccurs="0" name="clientTransaction" type="tns:CTID"/>
        <xsd:element minOccurs="0" name="description" type="tns:Description"/>
        <xsd:element minOccurs="0" name="operationInfo" type="tns:OperationInfo"/>
        <xsd:element minOccurs="0" name="paymentPasswordChallenge"
type="xsd:string"/>
        <xsd:element minOccurs="0" name="personalInformation"
type="tns:PersonalInformation"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Описание

payer

Номер счета плательщика.

Тип: string

payee

Номер счета получателя.

Тип: string

amount

Сумма операции.

- Если флаг isPayerAmount отсутствует и пользователь имеет доступ только к счету плательщика, то интерпретируется как сумма к списанию (в валюте плательщика).
- Если флаг isPayerAmount отсутствует и пользователь имеет доступ только к счету получателя, то интерпретируется как сумма к зачислению (в валюте получателя).

Подсказка: Всегда используйте элемент isPayerAmount, чтобы избежать ошибки по интерпретации суммы.

Необязательный элемент.

Тип: *Money Simple Type*

isPayerAmount

Если пользователь имеет доступ как к счету плательщика, так и к счету получателя, то флаг isPayerAmount обязателен.

- Если флаг isPayerAmount установлен (**true**), то amount используется как сумма к списанию (в валюте плательщика).
- Если флаг isPayerAmount сброшен (**false**), то amount используется как сумма к зачислению (в валюте получателя).

Необязательный элемент.

Тип: boolean

paymentPassword

Платежный пароль счета плательщика.

Необязательный элемент.

Тип: *Password Simple Type*

clientTransaction

Внешний номер операции.

Необязательный элемент.

Тип: *CTID Simple Type*

description

Описание операции.

Необязательный элемент.

Тип: *Description Simple Type*

operationInfo

Набор полей, которые необходимо сохранить в качестве атрибутов операции. Значения дат в формате dd.MM.yyyy HH:mm:ss

Необязательный элемент.

Тип: *OperationInfo Complex Type*

paymentPasswordChallenge

Запрос для платежного пароля.

Необязательный элемент.

Тип: string

personalInformation

Персональные данные пользователя.

Необязательный элемент.

Тип: *PersonalInformation Complex Type*

TransactionResponseType Complex Type

Тип, описывающий атрибуты операции в ответах.

XML Schema

```
<xsd:complexType name="TransactionResponseType">
  <xsd:sequence>
    <xsd:element name="status" type="xsd:string"/>
    <xsd:element name="dateTime" type="xsd:dateTime"/>
    <xsd:element name="transaction" type="xsd:long"/>
    <xsd:element minOccurs="0" name="clientTransaction" type="tns:CTID"/>
    <xsd:element minOccurs="0" name="operationInfo" type="tns:OperationInfo"/>
  </xsd:sequence>
</xsd:complexType>
```

Описание

status

Текущий статус операции.

Тип: string

dateTime

Время последней модификации операции.

Тип: dateTime

transaction

Номер операции.

Тип: long

clientTransaction

Внешний номер операции.

Необязательный элемент.

Тип: [CTID Simple Type](#)

operationInfo

Дополнительные свойства операции. Для получения этого поля в запросе необходимо выставить атрибут version равный или больше VERSION_2 (только в InvoiceRequest).

Необязательный элемент.

Тип: [OperationInfo Complex Type](#)

VerifyTransactionResponseType Complex Type

Ответ на запрос проверки проведения операции в системе MOHETA.PY.

Тип: [VerifyTransferResponseType Complex Type](#).

XML Schema

```
<xsd:complexType name="VerifyTransactionResponseType">
  <xsd:complexContent>
    <xsd:extension base="tns:VerifyTransferResponseType">
      <xsd:sequence>
        <xsd:element minOccurs="0" name="transactionId" type="xsd:long"/>
        <xsd:element minOccurs="0" name="operationStatus"
          type="tns:OperationStatus"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

Описание

transactionId

Идентификатор операции в системе МОНЕТА.РУ.

Необязательный элемент.

Тип: long

operationStatus

Статус операции в системе МОНЕТА.РУ.

Необязательный элемент.

Тип: *OperationStatus Simple Type*

VerifyTransferResponseType Complex Type

Тип, описывающий состояние операции в системе МОНЕТА.РУ.

XML Schema

```

<xsd:complexType name="VerifyTransferResponseType">
  <xsd:sequence>
    <xsd:element name="isTransactionValid" type="xsd:boolean"/>
    <xsd:element minOccurs="0" name="description" type="xsd:string"/>
    <xsd:element minOccurs="0" name="forecast"
type="tns:ForecastTransactionResponseType"/>
    <xsd:element minOccurs="0" name="errorCode" type="xsd:string"/>
    <xsd:element minOccurs="0" name="operationInfo" type="tns:OperationInfo"/>
  </xsd:sequence>
</xsd:complexType>

```

Описание

isTransactionValid

Если поле равно **true**, то данная операция может быть проведена в системе МОНЕТА.РУ

Тип: boolean

description

Описание текущего статуса операции.

Необязательный элемент.

Тип: string

forecast

Если операция может быть проведена, то в данном поле содержатся детали операции.
Иначе - это поле пустое.

Необязательный элемент.

Тип: *ForecastTransactionResponseType Complex Type*

errorCode

Поле содержит код ошибки. Поле заполнено только тогда, когда есть описание ошибки в поле description.

Для получения этого поля в запросе необходимо выставить атрибут version равный или больше **VERSION_2**.

Коды ошибок совпадают со значением элемента faultDetail (смотрите описание для этого элемента).

Необязательный элемент.

Тип: string

operationInfo

Дополнительные свойства операции.

Важно: Для получения этого поля в запросе необходимо выставить атрибут `version` равный или больше **VERSION_2**.

Если в ответе придет поле с ключом **paymentPasswordChallengeRequired** и значением **true**, то перед проведением операции следует сделать запрос "GetAccountPaymentPasswordChallengeRequest", который вернет строку запроса для платежного пароля. Это значение необходимо использовать в поле "paymentPasswordChallenge" в запросах на проведение операции (например: "PaymentRequest", "TransferRequest", "AuthoriseTransactionRequest" и т.д.).

Необязательный элемент.

Тип: *OperationInfo Complex Type*

Version Simple Type

Список версий, который используется в MONETA.MerchantAPI. Если функциональность зависит от конкретной версии, то в описании запроса это будет указано дополнительно.

Если версия не указана, то по умолчанию используется VERSION_1.

Тип: string

Возможные значения:

- **VERSION_1**
- **VERSION_2**
- **VERSION_3**

XML Schema

```
<xsd:simpleType name="Version">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="VERSION_1"/>
    <xsd:enumeration value="VERSION_2"/>
    <xsd:enumeration value="VERSION_3"/>
  </xsd:restriction>
</xsd:simpleType>
```

Финансовые методы

AuthoriseTransactionBatch Endpoint

Входящее сообщение: AuthoriseTransactionBatchRequest

Запрос на регистрацию операции с блокировкой средств на счете плательщика в пакетном режиме.

Тип: *AuthoriseTransactionBatchRequestType Complex Type*

XML Schema

```
<xsd:element name="AuthoriseTransactionBatchRequest">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="tns:AuthoriseTransactionBatchRequestType"/>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>
```

Исходящее сообщение: AuthoriseTransactionBatchResponse

Ответ на запрос регистрации операций в пакетном режиме.

XML Schema

```
<xsd:element name="AuthoriseTransactionBatchResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element maxOccurs="unbounded" minOccurs="1" name="transaction"
        type="tns:OperationInfoBatchResponseType"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

Описание

transaction

Детали проведенных операций, либо описание ошибок, если операция не проведена. Порядок соответствует набору операций, переданных в AuthoriseTransactionBatchRequest.

Обязательный элемент.

Тип: *OperationInfoBatchResponseType Complex Type*

AuthoriseTransaction Endpoint

Входящее сообщение: AuthoriseTransactionRequest

Запрос на регистрацию операции с блокировкой средств на счете плательщика.

Завершить (подтвердить) операцию можно с помощью ConfirmTransactionRequest. Если операция не будет закончена с помощью ConfirmTransactionRequest, то операция будет отменена автоматически по истечению срока блокировки.

Тип: *AuthoriseTransactionRequestType Complex Type*

XML Schema

```
<xsd:element name="AuthoriseTransactionRequest">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="tns:AuthoriseTransactionRequestType"/>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>
```

Исходящее сообщение: AuthoriseTransactionResponse

Ответ на запрос регистрации операции.

Тип: *OperationInfo Complex Type*

XML Schema

```
<xsd:element name="AuthoriseTransactionResponse">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="tns:OperationInfo"/>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>
```

Описание

В список атрибутов операции добавятся два значения:

- **protectioncode.** Код протекции, который необходимо использовать в ConfirmTransactionRequest.
- **protectioncodeexpirationdate.** Дата, до которой работает код протекции, по истечении срока операция отменяется.

CancelTransactionBatch Endpoint

Входящее сообщение: CancelTransactionBatchRequest

Запрос на отмену операций в пакетном режиме.

Тип: *CancelTransactionBatchRequestType Complex Type*

XML Schema

```
<xsd:element name="CancelTransactionBatchRequest">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="tns:CancelTransactionBatchRequestType"/>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>
```

Исходящее сообщение: CancelTransactionBatchResponse

Ответ на запрос CancelTransactionBatchRequest.

XML Schema

```
<xsd:element name="CancelTransactionBatchResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element maxOccurs="unbounded" minOccurs="1" name="transaction"
        type="tns:CancelTransactionBatchResponseType"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

Описание

transaction

Детали проведенных операций, либо описание ошибок, если операция не проведена.
Порядок соответствует набору операций, переданных в CancelTransactionBatchRequest.

Обязательный элемент.

Тип: *CancelTransactionBatchResponseType Complex Type*

CancelTransaction Endpoint

Входящее сообщение: CancelTransactionRequest

Запрос на отмену операции. Отменять операцию можно при выполнении всех условий:

- Незавершенные операции.
- Если есть доступ к счету плательщика и счету получателя (то есть свои внутренние операции).

Тип: *CancelTransactionRequestType Complex Type*

XML Schema

```
<xsd:element name="CancelTransactionRequest">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="tns:CancelTransactionRequestType"/>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>
```

Исходящее сообщение: CancelTransactionResponse

Ответ на запрос CancelTransactionRequest.

Тип: *CancelTransactionResponseType Complex Type*

XML Schema

```
<xsd:element name="CancelTransactionResponse">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="tns:CancelTransactionResponseType"/>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>
```

ConfirmTransactionBatch Endpoint

Входящее сообщение: ConfirmTransactionBatchRequest

Запрос на завершение (подтверждение) операций в пакетном режиме.

Тип: *ConfirmTransactionBatchRequestType Complex Type*

XML Schema

```
<xsd:element name="ConfirmTransactionBatchRequest">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="tns:ConfirmTransactionBatchRequestType"/>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>
```

Исходящее сообщение: ConfirmTransactionBatchResponse

Ответ на запрос завершения операций в пакетном режиме.

XML Schema

```
<xsd:element name="ConfirmTransactionBatchResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element maxOccurs="unbounded" minOccurs="1" name="transaction"
        type="tns:OperationInfoBatchResponseType"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

Описание

transaction

Детали проведенных операций, либо описание ошибок, если операция не проведена.
Порядок соответствует набору операций, переданных в ConfirmTransactionBatchRequest.

Обязательный элемент.

Тип: *OperationInfoBatchResponseType Complex Type*

ConfirmTransaction Endpoint

Входящее сообщение: ConfirmTransactionRequest

Запрос на завершение (подтверждение) операции.

Тип: *ConfirmTransactionRequestType Complex Type*

XML Schema

```
<xsd:element name="ConfirmTransactionRequest">
```

```

<xsd:complexType>
  <xsd:complexContent>
    <xsd:extension base="tns:ConfirmTransactionRequestType"/>
  </xsd:complexContent>
</xsd:complexType>
</xsd:element>

```

Исходящее сообщение: ConfirmTransactionResponse

Ответ на запрос завершения операции.

Тип: *OperationInfo Complex Type*

XML Schema

```

<xsd:element name="ConfirmTransactionResponse">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="tns:OperationInfo"/>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>

```

Описание

Описание операции.

Операция полностью проведена, если statusid=**SUCCEED**.

FindLastOperationsList Endpoint

Входящее сообщение: FindLastOperationsListRequest

Запрос на получение списка последних операций.

Если данные не найдены, то size в ответе равен 0.

XML Schema

```

<xsd:element name="FindLastOperationsListRequest">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element minOccurs="0" name="unitId" type="xsd:long"/>
      <xsd:element default="5" minOccurs="0" name="transactionsQuantity">
        <xsd:simpleType>
          <xsd:restriction base="xsd:int">
            <xsd:minInclusive value="1"/>
            <xsd:maxInclusive value="100"/>
          </xsd:restriction>
        </xsd:simpleType>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>

```

Описание

unitId

ID пользователя в системе МОНЕТА.РУ.

Если это поле не задано, то используется текущий пользователь.

Необязательный элемент.

Тип: long

transactionsQuantity

Количество операций.

Необязательный элемент.

Тип: int

Значение по умолчанию: 5

Исходящее сообщение: FindLastOperationsListResponse

Ответ на запрос FindLastOperationsListRequest. В результате возвращается список последних операций.

Тип: *OperationInfoList Complex Type*

XML Schema

```
<xsd:element name="FindLastOperationsListResponse">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="tns:OperationInfoList"/>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>
```

FindOperationsListByCTID Endpoint

Входящее сообщение: FindOperationsListByCTIDRequest

Запрос на получение данных по внешнему номеру операции (номеру не в системе МОНЕТА.РУ).

По внешнему номеру операции может быть найдено несколько операций в системе МОНЕТА.РУ.

Поэтому операции возвращаются списком, разбитым на страницы.

Размером страницы можно управлять через поле pager. Если данные не найдены, то size в ответе равен 0.

XML Schema

```
<xsd:element name="FindOperationsListByCTIDRequest">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element minOccurs="0" name="pager" type="tns:Pager"/>
      <xsd:element name="accountId" type="tns:AccountId"/>
      <xsd:element name="clientTransaction">
        <xsd:simpleType>
          <xsd:restriction base="tns:CTID">
            <xsd:minLength value="1"/>
          </xsd:restriction>
        </xsd:simpleType>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

Описание

pager

Настройки страницы данных.

Необязательный элемент.

Тип: *Pager Complex Type*

accountId

Номер счета в системе МОНЕТА.РУ.

Тип: *AccountId Simple Type*

clientTransaction

Внешний номер операции.

Тип: *CTID Simple Type*

Минимальная длина: 1

Исходящее сообщение: FindOperationsListByCTIDResponse

Ответ на запрос FindOperationsListByCTIDRequest.

В результате возвращается список операций, разбитый по страницам.

Тип: *OperationInfoList Complex Type*

XML Schema

```
<xsd:element name="FindOperationsListByCTIDResponse">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="tns:OperationInfoList"/>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>
```

FindOperationsList Endpoint

Входящее сообщение: FindOperationsListRequest

Запрос на получение списка операций по заданному фильтру.

Обязательными в фильтре являются только поля с датами периода.

Все остальные поля в фильтре необязательные.

Если данные не найдены, то size в ответе равен 0.

XML Schema

```
<xsd:element name="FindOperationsListRequest">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element minOccurs="0" name="pager" type="tns:Pager"/>
      <xsd:element name="filter">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element minOccurs="0" name="unitId" type="xsd:long"/>
            <xsd:element minOccurs="0" name="accountId" type="tns:AccountId"/>
            <xsd:element minOccurs="1" name="dateFrom" type="xsd:dateTime"/>
            <xsd:element minOccurs="1" name="dateTo" type="xsd:dateTime"/>
            <xsd:element minOccurs="0" name="operationId" type="xsd:long"/>
            <xsd:element minOccurs="0" name="amountFrom" type="tns:Money"/>
            <xsd:element minOccurs="0" name="amountTo" type="tns:Money"/>
            <xsd:element minOccurs="0" name="currencyCode" type="tns:Currency"/>
            <xsd:element minOccurs="0" name="targetAccountId"
type="tns:AccountId"/>
            <xsd:element minOccurs="0" name="operationStatus"
type="tns:OperationStatus"/>
            <xsd:element minOccurs="0" name="clientTransaction" type="tns:CTID"/>
            <xsd:element minOccurs="0" name="operationAmountTypeId"
type="xsd:int"/>
            <xsd:element minOccurs="0" name="propertyName" type="xsd:string"/>
            <xsd:element minOccurs="0" name="propertyValue" type="xsd:string"/>
            <xsd:element minOccurs="0" name="operationCategoryId"
type="xsd:long"/>
            <xsd:element minOccurs="0" name="accountingPeriodDate"
type="xsd:dateTime"/>
            <xsd:element default="false" minOccurs="0" name="searchInArchive"
type="xsd:boolean"/>
            <xsd:element minOccurs="0" name="attributeNamesInResponse"
type="xsd:string"/>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

```

        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
  </xsd:sequence>
</xsd:complexType>
</xsd:element>

```

Описание

pager

Настройки страницы данных.

Необязательный элемент.

Тип: *Pager Complex Type*

filter

Фильтр, по которому ищем операции.

unitId

ID пользователя в системе МОНЕТА.РУ.

Необязательный элемент.

Тип: long

accountId

Номер счета.

Необязательный элемент.

Тип: *AccountId Simple Type*

dateFrom

Дата начала периода.

Обязательный элемент.

Тип: dateTime

dateTo

Дата конца периода.

Обязательный элемент.

Тип: dateTime

operationId

Номер операции в системе МОНЕТА.РУ.

Необязательный элемент.

Тип: long

amountFrom

Сумма от... Ищутся все операции, которые проходили на большую или равную сумму.

Если указано это поле, то обязательно надо заполнить Currency.

Необязательный элемент.

Тип: *Money Simple Type*

amountTo

Сумма до... Ищутся все операции, которые проходили на меньшую или равную сумму.

Если указано это поле, то обязательно надо заполнить Currency.

Необязательный элемент.

Тип: *Money Simple Type*

currencyCode

Валюта, по которой проходили операции.

Необязательный элемент.

Тип: *Currency Simple Type*

targetAccountId

Корреспондентский номер счета в системе МОНЕТА.РУ.

Необязательный элемент.

Тип: *AccountId Simple Type*

operationStatus

Статус операции.

Необязательный элемент.

Тип: *OperationStatus Simple Type*

clientTransaction

Внешний номер операции.

Необязательный элемент.

Тип: *CTID Simple Type*

operationAmountTypeId

- 1. Все операции.
- 2. Операции, где деньги были получены (сумма > 0).
- 3. Операции, где деньги были потрачены (сумма < 0).

Необязательный элемент.

Тип: int

propertyName

Название свойства операции. Может принимать следующие значения:

- **YANDEXACCOUNT**. Счет в системе Яндекс.Деньги.
- **WEBMONEYWMID**. WebMoney WMID.
- **WEBMONEYPURSE**. WebMoney кошелек.
- **CONTACTTRANSFERORDERNUMBER**. Номер перевода в системе Contact.
- **WIRETRANSFERORDERNUMBER**. Номер платежного поручения в банковском переводе.

Значение свойства указывается в поле propertyValue.

Необязательный элемент.

Тип: string

propertyValue

Значение свойства операции.

Поиск происходит по прямому совпадению. Для задания маски можно указать спец-символы - "*" или "?".

Необязательный элемент.

Тип: string

operationCategoryId

- 1 ввод средств
- 2 вывод средств

- 3 внутренний перевод
- 4 товары и услуги

Необязательный элемент.

Тип: long

accountingPeriodDate

Дата расчетного периода.

Необязательный элемент.

Тип: dateTime

searchInArchive

Поиск операций в архиве.

Необязательный элемент.

Тип: boolean

Значение по умолчанию: false

attributeNamesInResponse

Список свойств операции, разделённый запятыми, которые будут возвращены в ответе.

Стандартные свойства операции передавать не обязательно, потому что они возвращаются всегда. Это свойства: modified, statusid, typeid, category, sourceaccountid, sourceamount, sourcecurrencycode, targetaccountid, targetamount, targetcurrencycode, description, clienttransaction.

Например, если для банковского перевода требуется получить в ответе не все свойства, то можно указать: wirebankbik, wirebankaccount, wireusername, customfield:email

Необязательный элемент.

Тип: string

Исходящее сообщение: FindOperationsListResponse

Ответ на запрос FindOperationsListRequest.

В результате возвращается список операций, разбитый по страницам.

Тип: *OperationInfoList Complex Type*

XML Schema

```
<xsd:element name="FindOperationsListResponse">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="tns:OperationInfoList"/>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>
```

ForecastTransaction Endpoint

Входящее сообщение: ForecastTransactionRequest

Запрос предварительного расчета сумм и комиссий по параметрам операции.

Тип: *TransactionRequestType Complex Type*

XML Schema

```
<xsd:element name="ForecastTransactionRequest">
  <xsd:complexType>
    <xsd:complexContent>
```

```

        <xsd:extension base="tns:TransactionRequestType"/>
      </xsd:complexContent>
    </xsd:complexType>
  </xsd:element>

```

Исходящее сообщение: ForecastTransactionResponse

Ответ на запрос предварительного расчета сумм и комиссий. Может использоваться для информирования пользователя о приблизительной сумме и комиссии операции. Фактические значения могут отличаться, если к моменту проведения операции доступны новые правила тарификации или новые курсы обмена валют.

Тип: *ForecastTransactionResponseType Complex Type*

XML Schema

```

<xsd:element name="ForecastTransactionResponse">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="tns:ForecastTransactionResponseType"/>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>

```

GetAccountPaymentPasswordChallenge Endpoint

Входящее сообщение: GetAccountPaymentPasswordChallengeRequest

Получить запрос для платежного пароля.

XML Schema

```

<xsd:element name="GetAccountPaymentPasswordChallengeRequest">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element minOccurs="1" name="accountId" type="tns:AccountId"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>

```

Описание

accountId

Номер счета.

Обязательный элемент.

Тип: *AccountId Simple Type*

Исходящее сообщение: GetAccountPaymentPasswordChallengeResponse

Ответ содержит строку запроса для платежного пароля.

XML Schema

```

<xsd:element name="GetAccountPaymentPasswordChallengeResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element minOccurs="0" name="paymentPasswordChallenge" type="xsd:string"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>

```

Описание

paymentPasswordChallenge

Запрос для платежного пароля.

Необязательный элемент.

Тип: string

GetOperationDetailsById Endpoint

Входящее сообщение: GetOperationDetailsByIdRequest

Запрос на получение данных по номеру операции (номер в системе МОНЕТА.РУ).

Если данные не найдены, то возникает Exception.

XML Schema

```
<xsd:element name="GetOperationDetailsByIdRequest">
  <xsd:simpleType>
    <xsd:restriction base="xsd:long"/>
  </xsd:simpleType>
</xsd:element>
```

Исходящее сообщение: GetOperationDetailsByIdResponse

Ответ на запрос получения информации по операции.

XML Schema

```
<xsd:element name="GetOperationDetailsByIdResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="operation" type="tns:OperationInfo"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

Описание

operation

Тип: *OperationInfo Complex Type*

InvoiceBatch Endpoint

Входящее сообщение: InvoiceBatchRequest

Выставление счета к оплате в пакетном режиме.

Тип: *InvoiceBatchRequestType Complex Type*

XML Schema

```
<xsd:element name="InvoiceBatchRequest">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="tns:InvoiceBatchRequestType"/>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>
```

Исходящее сообщение: InvoiceBatchResponse

Ответ на выставление счета к оплате в пакетном режиме.

XML Schema

```
<xsd:element name="InvoiceBatchResponse">
```

```

<xsd:complexType>
  <xsd:sequence>
    <xsd:element maxOccurs="unbounded" minOccurs="1" name="transaction"
      type="tns:TransactionBatchResponseType"/>
  </xsd:sequence>
</xsd:complexType>
</xsd:element>

```

Описание

transaction

Детали проведенных операций, либо описание ошибок, если операция не проведена. Порядок соответствует набору операций, переданных в InvoiceBatchRequest.

Обязательный элемент.

Тип: *TransactionBatchResponseType Complex Type*

Invoice Endpoint

Входящее сообщение: InvoiceRequest

Выставление счета к оплате.

Тип: *InvoiceRequestType Complex Type*

XML Schema

```

<xsd:element name="InvoiceRequest">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="tns:InvoiceRequestType"/>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>

```

Исходящее сообщение: InvoiceResponse

Ответ на выставление счета к оплате.

Тип: *TransactionResponseType Complex Type*

XML Schema

```

<xsd:element name="InvoiceResponse">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="tns:TransactionResponseType"/>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>

```

Payment Endpoint

Входящее сообщение: PaymentRequest

Запрос на перевод денежных средств. Отличается от TransferRequest тем, что в качестве получателя можно указывать не только номер счета.

Тип: *PaymentRequestType Complex Type*

XML Schema

```

<xsd:element name="PaymentRequest">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="tns:PaymentRequestType"/>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>

```

```

    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>

```

Исходящее сообщение: **PaymentResponse**

Ответ на запрос перевода денежных средств.

Тип: *OperationInfo Complex Type*

XML Schema

```

<xsd:element name="PaymentResponse">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="tns:OperationInfo"/>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>

```

Описание

Описание операции.

Операция полностью проведена, если statusid=**SUCCEED**

PaymentBatch Endpoint

Входящее сообщение: **PaymentBatchRequest**

Запрос на перевод денежных средств в пакетном режиме.

Тип: *PaymentBatchRequestType Complex Type*

XML Schema

```

<xsd:element name="PaymentBatchRequest">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="tns:PaymentBatchRequestType"/>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>

```

Исходящее сообщение: **PaymentBatchResponse**

Ответ на запрос перевода денежных средств в пакетном режиме.

XML Schema

```

<xsd:element name="PaymentBatchResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element maxOccurs="unbounded" minOccurs="1" name="transaction"
        type="tns:OperationInfoBatchResponseType"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>

```

Описание

transaction

Детали проведенных операций, либо описание ошибок, если операция не проведена.
Порядок соответствует набору операций, переданных в PaymentBatchRequest.

Обязательный элемент.

Тип: *OperationInfoBatchResponseType Complex Type*

Refund Endpoint

Входящее сообщение: RefundRequest

Возврат средств по указанной операции.

XML Schema

```
<xsd:element name="RefundRequest">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="transactionId" type="xsd:long"/>
      <xsd:element minOccurs="0" name="amount" type="tns:Money"/>
      <xsd:element minOccurs="0" name="paymentPassword" type="tns:Password"/>
      <xsd:element minOccurs="0" name="clientTransaction" type="tns:CTID"/>
      <xsd:element minOccurs="0" name="description" type="tns:Description"/>
      <xsd:element minOccurs="0" name="operationInfo" type="tns:OperationInfo"/>
      <xsd:element minOccurs="0" name="paymentPasswordChallenge" type="xsd:string"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

Описание

transactionId

Номер операции в системе МОНЕТА.РУ, по которой необходимо вернуть деньги.

Тип: long

amount

Сумма, которую необходимо возратить. Если сумма не указана, то сумма для возврата берется из указанной операции.

Необязательный элемент.

Тип: *Money Simple Type*

paymentPassword

Платежный пароль.

Необязательный элемент.

Тип: *Password Simple Type*

clientTransaction

Внешний номер операции.

Необязательный элемент.

Тип: *CTID Simple Type*

description

Описание операции.

Необязательный элемент.

Тип: *Description Simple Type*

operationInfo

Набор полей, которые необходимо сохранить в качестве атрибутов операции. Значения дат в формате dd.MM.yyyy HH:mm:ss

Необязательный элемент.

Тип: *OperationInfo Complex Type*

paymentPasswordChallenge

Запрос для платежного пароля.

Необязательный элемент.

Тип: string

Исходящее сообщение: RefundResponse

Ответ по запросу возврата средств, содержащий информацию по возвратной операции.

Тип: *OperationInfo Complex Type*

XML Schema

```
<xsd:element name="RefundResponse">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="tns:OperationInfo"/>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>
```

Описание

Описание операции.

Операция полностью проведена, если statusid=**SUCCEED**.

SecureData Endpoint

Входящее сообщение: SecureDataRequest

Запрос на сохранение свойств, которые можно использовать в дальнейшем с помощью SECURETOKEN.

XML Schema

```
<xsd:element name="SecureDataRequest">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="publicId" type="xsd:string"/>
      <xsd:element minOccurs="0" name="accountId" type="tns:AccountId"/>
      <xsd:element maxOccurs="unbounded" minOccurs="1" name="attribute"
        type="tns:KeyValueAttribute"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

Описание

publicId

Публичный идентификатор.

Тип: string

accountId

Номер счета.

Поле следует заполнить, если данные этого запроса будут использоваться для проведения финансовой операции. Например, PaymentRequest, TransferRequest, AuthoriseTransactionRequest, ConfirmTransactionRequest и т.д. При проведении операции следует указать этот же самый номер счета - для другого счета эти данные использовать нельзя.

Необязательный элемент.

Тип: *AccountId Simple Type*

attribute

Свойства, которые надо сохранить.

Обязательный элемент.

Тип: *KeyValueAttribute Complex Type*

Исходящее сообщение: SecureDataResponse

Ответ, который приходит на запрос SecureDataRequest.

XML Schema

```
<xsd:element name="SecureDataResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="secureToken" type="xsd:string"/>
      <xsd:element minOccurs="0" name="expirationDate" type="xsd:dateTime"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

Описание

secureToken

Secure Token.

Тип: string

expirationDate

Срок действия Secure Token.

Необязательный элемент.

Тип: dateTime

TransferBatch Endpoint

Входящее сообщение: TransferBatchRequest

Запрос на перевод денежных средств в пакетном режиме.

Тип: *TransactionBatchRequestType Complex Type*

XML Schema

```
<xsd:element name="TransferBatchRequest">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="tns:TransactionBatchRequestType"/>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>
```

Исходящее сообщение: TransferBatchResponse

Ответ на запрос перевода денежных средств в пакетном режиме.

XML Schema

```
<xsd:element name="TransferBatchResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element maxOccurs="unbounded" minOccurs="1" name="transaction"
        type="tns:TransactionBatchResponseType"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

Описание transaction

Детали проведенных операций, либо описание ошибок, если операция не проведена. Порядок соответствует набору операций, переданных в TransferBatchRequest.

Обязательный элемент.

Тип: *TransactionBatchResponseType Complex Type*

Transfer Endpoint

Входящее сообщение: TransferRequest

Запрос на перевод денежных средств.

Тип: *TransactionRequestType Complex Type*

XML Schema

```
<xsd:element name="TransferRequest">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="tns:TransactionRequestType"/>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>
```

Исходящее сообщение: TransferResponse

Ответ на запрос перевода денежных средств.

Тип: *TransactionResponseType Complex Type*

XML Schema

```
<xsd:element name="TransferResponse">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="tns:TransactionResponseType"/>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>
```

VerifyPayment Endpoint

Входящее сообщение: VerifyPaymentRequest

Запрос возможности провести операцию в системе МОНЕТА.РУ. Отличается от VerifyTransferRequest тем, что в качестве получателя можно указывать не только номер счета.

Тип: *PaymentRequestType Complex Type*

XML Schema

```
<xsd:element name="VerifyPaymentRequest">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="tns:PaymentRequestType"/>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>
```

Исходящее сообщение: VerifyPaymentResponse

Ответ на запрос проверки проведения операции в системе МОНЕТА.РУ.

Тип: *VerifyTransactionResponseType Complex Type*

XML Schema

```
<xsd:element name="VerifyPaymentResponse">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="tns:VerifyTransactionResponseType"/>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>
```

VerifyPaymentBatch Endpoint

Входящее сообщение: VerifyPaymentBatchRequest

Запрос возможности провести операцию в системе МОНЕТА.РУ. Запрос выполняется в пакетном режиме.

Тип: *PaymentBatchRequestType Complex Type*

XML Schema

```
<xsd:element name="VerifyPaymentBatchRequest">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="tns:PaymentBatchRequestType"/>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>
```

Исходящее сообщение: VerifyPaymentBatchResponse

Ответ на запрос проверки проведения операции в системе МОНЕТА.РУ. Ответ в пакетном режиме.

XML Schema

```
<xsd:element name="VerifyPaymentBatchResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element maxOccurs="unbounded" minOccurs="1" name="transaction"
        type="tns:VerifyTransactionResponseType"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

Описание

transaction

Детали операций, либо описание ошибок. Порядок соответствует набору операций, переданных в VerifyPaymentBatchRequest.

Обязательный элемент.

Тип: *VerifyTransactionResponseType Complex Type*

VerifyTransaction Endpoint

Входящее сообщение: VerifyTransactionRequest

Запрос возможности провести операцию в системе МОНЕТА.РУ.

Тип: *Entity Complex Type*

XML Schema

```
<xsd:element name="VerifyTransactionRequest">
```



```

<xsd:complexType>
  <xsd:complexContent>
    <xsd:extension base="tns:Entity">
      <xsd:sequence>
        <xsd:element name="transactionId" type="xsd:long"/>
        <xsd:element minOccurs="0" name="payer" type="xsd:string"/>
        <xsd:element minOccurs="0" name="amount" type="tns:Money"/>
        <xsd:element minOccurs="0" name="isPayerAmount" type="xsd:boolean"/>
        <xsd:element minOccurs="0" name="operationInfo"
type="tns:OperationInfo"/>
        <xsd:element minOccurs="0" name="paymentPassword" type="tns:Password"/>
        <xsd:element minOccurs="0" name="paymentPasswordChallenge"
type="xsd:string"/>
        <xsd:element minOccurs="0" name="clientTransaction" type="tns:CTID"/>
        <xsd:element minOccurs="0" name="personalInformation"
type="tns:PersonalInformation"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
</xsd:element>

```

Описание

transactionId

Идентификатор операции в системе МОНЕТА.РУ, которую надо проверить.

Тип: long

payer

Номер счета плательщика.

Необязательный элемент.

Тип: string

amount

Сумма операции.

- Если флаг isPayerAmount отсутствует и пользователь имеет доступ только к счету плательщика, то интерпретируется как сумма к списанию (в валюте плательщика).
- Если флаг isPayerAmount отсутствует и пользователь имеет доступ только к счету получателя, то интерпретируется как сумма к зачислению (в валюте получателя).

Необязательный элемент.

Тип: *Money Simple Type*

isPayerAmount

Если пользователь имеет доступ как к счету плательщика, так и к счету получателя, то флаг isPayerAmount обязателен.

- Если флаг isPayerAmount установлен (true), то amount используется как сумма к списанию (в валюте плательщика).
- Если флаг isPayerAmount сброшен (false), то amount используется как сумма к зачислению (в валюте получателя).

Необязательный элемент.

Тип: boolean

operationInfo

Набор полей, которые необходимо использовать в качестве атрибутов операции.

Значения дат в формате dd.MM.yyyy HH:mm:ss

Необязательный элемент.

Тип: *OperationInfo Complex Type*

paymentPassword

Платежный пароль счета плательщика.

Необязательный элемент.

Тип: *Password Simple Type*

paymentPasswordChallenge

Запрос для платежного пароля

Необязательный элемент.

Тип: string

clientTransaction

Внешний номер операции.

Необязательный элемент.

Тип: *CTID Simple Type*

personalInformation

Персональные данные пользователя.

Необязательный элемент.

Тип: *PersonalInformation Complex Type*

Исходящее сообщение: VerifyTransactionResponse

Ответ на запрос проверки проведения операции в системе МОНЕТА.РУ.

Тип: *VerifyTransactionResponseType Complex Type*

XML Schema

```
<xsd:element name="VerifyTransactionResponse">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="tns:VerifyTransactionResponseType"/>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>
```

VerifyTransfer Endpoint

Входящее сообщение: VerifyTransferRequest

Запрос возможности провести операцию в системе МОНЕТА.РУ.

Тип: *TransactionRequestType Complex Type*

XML Schema

```
<xsd:element name="VerifyTransferRequest">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="tns:TransactionRequestType"/>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>
```

Исходящее сообщение: VerifyTransferResponse

Ответ на запрос проверки проведения операции в системе МОНЕТА.РУ.

Тип: *VerifyTransferResponseType Complex Type*

XML Schema

```
<xsd:element name="VerifyTransferResponse">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="tns:VerifyTransferResponseType"/>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>
```

PaymentSystemInfo Endpoint

Входящее сообщение: PaymentSystemInfoRequest

Запрос информации об платежных системах.

XML Schema

```
<xsd:element name="PaymentSystemInfoRequest">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="accountId" type="tns:AccountId"/>
      <xsd:element minOccurs="0" name="amount" type="tns:Money"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

Описание

accountId

Номер счета в системе МОНЕТА.РУ.

Тип: *AccountId Simple Type*

amount

Необязательный элемент.

Тип: *Money Simple Type*

Исходящее сообщение: PaymentSystemInfoResponse

Ответ, который приходит на запрос PaymentSystemInfoRequest.

XML Schema

```
<xsd:element name="PaymentSystemInfoResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element maxOccurs="unbounded" minOccurs="0" name="items"
        type="tns:PaymentSystemInfoComplexType"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

Шаблоны операций

CreateOperationTemplate Endpoint

Входящее сообщение: CreateOperationTemplateRequest

Запрос на создание шаблона операции.

Можно создавать шаблоны с типами:

- *OperationTemplateType.COMMON*
- *OperationTemplateType.REGULAR*

Тип: *OperationTemplate Complex Type*

XML Schema

```
<xsd:element name="CreateOperationTemplateRequest">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="tns:OperationTemplate">
        <xsd:sequence>
          <xsd:element minOccurs="0" name="paymentPassword"
            type="tns:PaymentPassword"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>
```

Описание

paymentPassword

Платежный пароль счета плательщика.

Поле необходимо указывать, если создается шаблон для регулярного выполнения (с типом *OperationTemplateType.REGULAR*).

Необязательный элемент.

Тип: *PaymentPassword Complex Type*

Исходящее сообщение: CreateOperationTemplateResponse

Ответ на запрос CreateOperationTemplateRequest.

XML Schema

```
<xsd:element name="CreateOperationTemplateResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="id" type="xsd:long"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

Описание

id

Идентификатор созданного шаблона операции.

Тип: long

EditOperationTemplate Endpoint

Входящее сообщение: EditOperationTemplateRequest

Запрос на редактирование шаблона операции.

Тип: *OperationTemplate Complex Type*

XML Schema

```
<xsd:element name="EditOperationTemplateRequest">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="tns:OperationTemplate">
```

```

        <xsd:sequence>
            <xsd:element minOccurs="0" name="paymentPassword"
type="tns:PaymentPassword"/>
        </xsd:sequence>
    </xsd:extension>
</xsd:complexContent>
</xsd:complexType>
</xsd:element>

```

Описание

paymentPassword

Платежный пароль счета плательщика.

Поле необходимо указывать, если редактируется шаблон для регулярного выполнения (с типом *OperationTemplateType.REGULAR*) и в запросе есть хотя бы один из элементов:

- *payer*
- *payee*
- *regularParameters.amountInfo*
- *regularParameters.timeInfo*
- *operationInfo*

Необязательный элемент.

Тип: *PaymentPassword Complex Type*

Исходящее сообщение: EditOperationTemplateResponse

Ответ на запрос EditOperationTemplateRequest.

В ответе нет никаких данных. Если в процессе сохранения шаблона произошла ошибка, то возникнет Exception. Если Exception не возник - значит шаблон операции гарантированно сохранен.

XML Schema

```

<xsd:element name="EditOperationTemplateResponse">
    <xsd:complexType/>
</xsd:element>

```

FindOperationTemplates Endpoint

Входящее сообщение: FindOperationTemplatesRequest

Запрос на поиск и чтение шаблонов.

Шаблоны можно искать либо по "идентификатору шаблона", либо по "фильтру".

Если поиск был по "идентификатору шаблона", то в результате будет список, состоящий из одного элемента. Если шаблон не найден, то будет ошибка с кодом: 500.5.20.

Если поиск был по "фильтру", то в результате будет список найденных шаблонов. Если шаблоны не найдены, будет пустой список. При поиске шаблонов по "фильтру" в ответе будут заполнены не все поля шаблона (например, не будет информации о "метках" и "operationInfo").

XML Schema

```

<xsd:element name="FindOperationTemplatesRequest">
    <xsd:complexType>
        <xsd:sequence>
            <xsd:choice>
                <xsd:element minOccurs="0" name="id" type="xsd:long"/>
                <xsd:sequence>
                    <xsd:element minOccurs="0" name="unitId" type="xsd:long"/>
                    <xsd:element minOccurs="0" name="operationTypeCategory"
type="tns:OperationTypeCategory"/>
                </xsd:sequence>
            </xsd:choice>
        </xsd:sequence>
    </xsd:complexType>
</xsd:element>

```

```

        <xsd:element minOccurs="0" name="type" type="tns:OperationTemplateType"/>
    >
        <xsd:element minOccurs="0" name="tag" type="xsd:string"/>
    </xsd:sequence>
</xsd:choice>
</xsd:sequence>
</xsd:complexType>
</xsd:element>

```

Описание

id

Идентификатор шаблона операции.

Необязательный элемент.

Тип: long

unitId

ID пользователя в системе МОНЕТА.РУ. Если это поле не задано, то используется текущий пользователь.

Необязательный элемент.

Тип: long

operationTypeCategory

Категория операции.

Необязательный элемент.

Тип: *OperationTypeCategory Simple Type*

type

Тип шаблона операции.

Необязательный элемент.

Тип: *OperationTemplateType Simple Type*

tag

Метка шаблона.

Необязательный элемент.

Тип: string

Исходящее сообщение: FindOperationTemplatesResponse

Ответ на запрос FindOperationTemplatesRequest.

XML Schema

```

<xsd:element name="FindOperationTemplatesResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element maxOccurs="unbounded" minOccurs="0" name="operationTemplate"
        type="tns:OperationTemplate"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>

```

Описание

operationTemplate

Список шаблонов.

Необязательный элемент.

Тип: *OperationTemplate Complex Type*

DeleteOperationTemplate Endpoint

Входящее сообщение: DeleteOperationTemplateRequest

Запрос на удаление шаблона операции.

XML Schema

```
<xsd:element name="DeleteOperationTemplateRequest">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="id" type="xsd:long"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

Описание

id

Идентификатор шаблона операции.

Тип: long

Исходящее сообщение: DeleteOperationTemplateResponse

Ответ на запрос DeleteOperationTemplateRequest.

В ответе нет никаких данных. Если в процессе удаления шаблона произошла ошибка, то возникнет Exception. Если Exception не возник - значит шаблон операции гарантированно удален.

XML Schema

```
<xsd:element name="DeleteOperationTemplateResponse">
  <xsd:complexType/>
</xsd:element>
```

Профайл пользователя

FindProfileInfoByAccountId Endpoint

Входящее сообщение: FindProfileInfoByAccountIdRequest

Запрос на получение данных пользователя по номеру счета.

Если данные не найдены, возникает Exception.

Тип: *AccountId Simple Type*

XML Schema

```
<xsd:element name="FindProfileInfoByAccountIdRequest">
  <xsd:complexType>
    <xsd:simpleContent>
      <xsd:extension base="tns:AccountId">
        <xsd:attribute name="version" type="tns:Version" use="optional"/>
      </xsd:extension>
    </xsd:simpleContent>
  </xsd:complexType>
</xsd:element>
```

Описание

Исходящее сообщение: FindProfileInfoByAccountIdResponse

Ответ на запрос FindProfileInfoByAccountIdRequest.

XML Schema

```

<xsd:element name="FindProfileInfoByAccountIdResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="accountId" type="tns:AccountId"/>
      <xsd:element name="currency" type="tns:Currency"/>
      <xsd:element name="profile" type="tns:Profile"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>

```

Описание

accountId

Номер счета пользователя в системе МОНЕТА.РУ.

Тип: *AccountId Simple Type*

currency

Валюта счета.

Тип: *Currency Simple Type*

profile

Данные пользователя в системе МОНЕТА.РУ.

Тип: *Profile Complex Type*

EditProfile Endpoint

Входящее сообщение: EditProfileRequest

Запрос на редактирование данных пользователя в системе МОНЕТА.РУ.

XML Schema

```

<xsd:element name="EditProfileRequest">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="unitId" type="xsd:long"/>
      <xsd:element minOccurs="0" name="profileId" type="xsd:long"/>
      <xsd:element minOccurs="0" name="profile" type="tns:Profile"/>
      <xsd:element minOccurs="0" name="parentId" type="xsd:long"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>

```

Описание

unitId

ID пользователя в системе МОНЕТА.РУ, данные которого надо отредактировать.

Тип: long

profileId

Необязательный элемент.

Тип: long

profile

Данные редактируемого пользователя. Данные представлены в виде "ключ-значение".

Ключи данных зависят от типа редактируемого пользователя (client или organization).

Необязательный элемент.

Тип: *Profile Complex Type*

parentId

ID структуры, куда необходимо перенести данного пользователя.

К указанной структуре должен быть доступ, а также действующий договор должен разрешать создание пользователя с нужным типом (client или organization).

Необязательный элемент.

Тип: long

Исходящее сообщение: EditProfileResponse

Ответ, который приходит на запрос EditProfileRequest.

В ответе нет никаких данных. Если в процессе сохранения пользователя произошла ошибка, то возникнет Exception. Если Exception не возник - значит пользователь гарантированно сохранен.

XML Schema

```
<xsd:element name="EditProfileResponse">
  <xsd:complexType/>
</xsd:element>
```

FindProfileInfo Endpoint

Входящее сообщение: FindProfileInfoRequest

Поиск пользователя в системе МОНЕТА.РУ по заданному фильтру.

В результате возвратится список пользователей, разбитый на страницы.

Тип: *Entity Complex Type*

XML Schema

```
<xsd:element name="FindProfileInfoRequest">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="tns:Entity">
        <xsd:sequence>
          <xsd:element minOccurs="0" name="pager" type="tns:Pager"/>
          <xsd:element minOccurs="1" name="filter">
            <xsd:complexType>
              <xsd:sequence>
                <xsd:element minOccurs="0" name="unitId" type="xsd:long"/>
                <xsd:element minOccurs="0" name="name" type="xsd:string"/>
                <xsd:element minOccurs="0" name="email" type="xsd:string"/>
                <xsd:element minOccurs="0" name="accountBalanceFrom"
type="tns:Money"/>
                <xsd:element minOccurs="0" name="accountBalanceTo"
type="tns:Money"/>
                <xsd:element minOccurs="0" name="accountCurrencyCode"
type="tns:Currency"/>
                <xsd:element minOccurs="0" name="profileId" type="xsd:long"/>
              </xsd:sequence>
            </xsd:complexType>
          </xsd:element>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
```

```
</xsd:element>
```

Описание

pager

Настройки страницы данных.

Необязательный элемент.

Тип: *Pager Complex Type*

filter

Фильтр, по которому ищем данные.

Обязательный элемент.

unitId

ID пользователя в системе МОНЕТА.РУ. ID пользователя "от которого" начинается поиск.

Если это поле не задано, то ищем пользователей, начиная от идентифицированного пользователя.

Необязательный элемент.

Тип: long

name

Имя, по которому происходит поиск.

Если поле задано, то оно должно быть не меньше 2-х символов.

Поиск происходит по прямому совпадению. Для задания маски можно указать спец-символы - "*" или "?".

Для пользователей с типом "organization" поиск происходит по полю "organization_name".
Для пользователей с типом "client" - по общему полю "last_name first_name middle_initial_name". То есть, если нужно найти пользователя "Иванов Андрей Сергеевич", то следует ввести имя для поиска именно так. Если нужно найти всех пользователей с фамилией "Иванов", то запрос должен быть "Иванов*". Если нужно найти всех пользователей с именем "Андрей", то запрос должен быть "*Андрей*".

Необязательный элемент.

Тип: string

email

Email, по которому происходит поиск.

Если поле задано, то оно должно быть не меньше 2-х символов.

Поиск происходит по прямому совпадению. Для задания маски можно указать спец-символы - "*" или "?".

Необязательный элемент.

Тип: string

accountBalanceFrom

Возвращаются пользователи у которых есть счета с балансом большим или равным указанному значению.

Необязательный элемент.

Тип: *Money Simple Type*

accountBalanceTo

Возвращаются пользователи у которых есть счета с балансом меньшим или равным указанному значению.

Необязательный элемент.

Тип: *Money Simple Type*

accountCurrencyCode

При указании границ баланса в поиске можно уточнить валюту счета.

Необязательный элемент.

Тип: *Currency Simple Type*

profileId

Необязательный элемент.

Тип: long

Исходящее сообщение: FindProfileInfoResponse

Ответ, который приходит на запрос FindProfileInfoRequest.

Данные в ответе разбиты по страницам.

Если данные не найдены, то size в ответе равен 0.

XML Schema

```
<xsd:element name="FindProfileInfoResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="pageSize" type="xsd:long"/>
      <xsd:element name="pageNumber" type="xsd:long"/>
      <xsd:element name="pagesCount" type="xsd:long"/>
      <xsd:element name="size" type="xsd:long"/>
      <xsd:element name="totalSize" type="xsd:long"/>
      <xsd:element maxOccurs="unbounded" minOccurs="0" name="profile"
type="tns:Profile"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

Описание

pageSize

Количество пользователей, возвращаемых в результате запроса.

Тип: long

pageNumber

Номер текущей страницы. Нумерация начинается с 1.

Тип: long

pagesCount

Сколько всего страниц с пользователями возможно по данному запросу.

Тип: long

size

Размер текущей страницы. Он всегда <= pageSize. На последней странице может быть < pageSize.

Тип: long

totalSize

Общее количество пользователей, которое можно получить в данной выборке.

Тип: long

profile

Список пользователей.

Необязательный элемент.

Тип: *Profile Complex Type*

GetProfileInfo Endpoint**Входящее сообщение: GetProfileInfoRequest**

Запрос на получение данных пользователя по ID пользователя.

Если данные не найдены, возникает Exception.

Тип: *Entity Complex Type*

XML Schema

```

<xsd:element name="GetProfileInfoRequest">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="tns:Entity">
        <xsd:sequence>
          <xsd:element minOccurs="0" name="unitId" type="xsd:long"/>
          <xsd:element minOccurs="0" name="profileId" type="xsd:long"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>

```

Описание**unitId**

ID пользователя в системе МОНЕТА.РУ.

Если это поле не задано, то используется текущий пользователь.

Необязательный элемент.

Тип: long

profileId

Необязательный элемент.

Тип: long

Исходящее сообщение: GetProfileInfoResponse

Ответ на запрос GetProfileInfoRequest.

Тип: *Profile Complex Type*

XML Schema

```

<xsd:element name="GetProfileInfoResponse">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="tns:Profile"/>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>

```

Описание

CreateProfile Endpoint

Входящее сообщение: CreateProfileRequest

Запрос на создание пользователя в системе МОНЕТА.РУ.

XML Schema

```

<xsd:element name="CreateProfileRequest">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element minOccurs="0" name="unitId" type="xsd:long"/>
      <xsd:element minOccurs="0" name="profileId" type="xsd:long"/>
      <xsd:element name="profileType" type="tns:ProfileType"/>
      <xsd:element name="profile" type="tns:Profile"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>

```

Описание

unitId

ID родительского пользователя в системе МОНЕТА.РУ, к которому будет принадлежать создаваемый пользователь.

Если поле не задано, то пользователь будет принадлежать тому пользователю, от имени которого работает web service.

Необязательный элемент.

Тип: long

profileId

Необязательный элемент.

Тип: long

profileType

Тип пользователя. Возможные значения:

- client
- organization

Тип: *ProfileType Simple Type*

profile

Данные создаваемого пользователя. Данные представлены в виде "ключ-значение".

Ключи данных зависят от типа пользователя (ProfileType).

Тип: *Profile Complex Type*

Исходящее сообщение: CreateProfileResponse

Ответ на запрос CreateProfileRequest, содержащий ID нового пользователя.

XML Schema

```

<xsd:element name="CreateProfileResponse">
  <xsd:simpleType>
    <xsd:restriction base="xsd:long"/>
  </xsd:simpleType>
</xsd:element>

```

Управление счетами

FindAccountByAlias Endpoint

Входящее сообщение: FindAccountByAliasRequest

Запрос на поиск счета по его названию.

Счет должен принадлежать текущему пользователю. Название в запросе не может быть пустым. Если счет не найден, то возникает Exception.

XML Schema

```
<xsd:element name="FindAccountByAliasRequest">
  <xsd:complexType>
    <xsd:simpleContent>
      <xsd:extension base="xsd:string">
        <xsd:attribute name="version" type="tns:Version" use="optional"/>
      </xsd:extension>
    </xsd:simpleContent>
  </xsd:complexType>
</xsd:element>
```

Исходящее сообщение: FindAccountByAliasResponse

Ответ, который приходит на запрос FindAccountByAliasRequest. В ответе содержится информация по счету.

XML Schema

```
<xsd:element name="FindAccountByAliasResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="account" type="tns:AccountInfo"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

FindAccountById Endpoint

Входящее сообщение: FindAccountByIdRequest

Запрос на поиск счета по его номеру.

Счет должен принадлежать текущему пользователю. Если счет не найден, то возникает Exception.

Тип: [AccountId Simple Type](#)

XML Schema

```
<xsd:element name="FindAccountByIdRequest">
  <xsd:complexType>
    <xsd:simpleContent>
      <xsd:extension base="tns:AccountId">
        <xsd:attribute name="version" type="tns:Version" use="optional"/>
      </xsd:extension>
    </xsd:simpleContent>
  </xsd:complexType>
</xsd:element>
```

Исходящее сообщение: FindAccountByIdResponse

Ответ, который приходит на запрос FindAccountByIdRequest. В ответе содержится информация по счету.

XML Schema

```
<xsd:element name="FindAccountByIdResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="account" type="tns:AccountInfo"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

CreateAccount Endpoint

Входящее сообщение: CreateAccountRequest

Запрос на создание счета. Счет создается для указанного пользователя или, если это поле не указано - для текущего пользователя.

XML Schema

```
<xsd:element name="CreateAccountRequest">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element default="2" minOccurs="0" name="type" type="tns:AccountType"/>
      <xsd:element name="currency" type="tns:Currency"/>
      <xsd:element minOccurs="0" name="alias" type="xsd:string"/>
      <xsd:element minOccurs="0" name="paymentPasswordType"
type="tns:AccountPaymentPasswordType"/>
      <xsd:element minOccurs="0" name="paymentPassword">
        <xsd:simpleType>
          <xsd:restriction base="xsd:string">
            <xsd:minLength value="5"/>
            <xsd:whiteSpace value="collapse"/>
          </xsd:restriction>
        </xsd:simpleType>
      </xsd:element>
      <xsd:element minOccurs="0" name="paymentPasswordExpirationDate"
type="xsd:boolean"/>
      <xsd:element minOccurs="0" name="unitId" type="xsd:long"/>
      <xsd:element minOccurs="0" name="onSuccessfulDebitUrl" type="xsd:string"/>
      <xsd:element minOccurs="0" name="onSuccessfulCreditUrl" type="xsd:string"/>
      <xsd:element minOccurs="0" name="signature" type="xsd:string"/>
      <xsd:element minOccurs="0" name="lowBalanceThreshold" type="tns:Money"/>
      <xsd:element minOccurs="0" name="highBalanceThreshold" type="tns:Money"/>
      <xsd:element minOccurs="0" name="prototypeAccountId" type="xsd:long"/>
      <xsd:element minOccurs="0" name="onCancelledDebitUrl" type="xsd:string"/>
      <xsd:element minOccurs="0" name="onCancelledCreditUrl" type="xsd:string"/>
      <xsd:element minOccurs="0" name="onAuthoriseUrl" type="xsd:string"/>
      <xsd:element maxOccurs="unbounded" minOccurs="0" name="attribute"
type="tns:KeyValueApprovedAttribute"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

Описание

type

Тип счета.

Значение по умолчанию равно 2 (Расширенный счет).

Необязательный элемент.

Тип: *AccountType Simple Type*

Значение по умолчанию: 2

currency

Валюта счета.

Тип: *Currency Simple Type*

alias

Название счета. Название счета должно быть уникальным среди счетов одного пользователя.

Необязательный элемент.

Тип: string

paymentPasswordType

Тип платежного пароля. По умолчанию равен STATIC.

Необязательный элемент.

Тип: *AccountPaymentPasswordType Simple Type*

paymentPassword

Платежный пароль. Минимальная длина - 5 символов. Это поле используется, если тип платежного пароля равен STATIC.

Необязательный элемент.

Тип: string

Минимальная длина: 5

paymentPasswordExpirationDate

Включить срок действия платежного пароля. Срок действия пароля равен одному году. Это поле используется, если тип платежного пароля равен STATIC.

Необязательный элемент.

Тип: boolean

unitId

Пользователь, которому будет принадлежать данный счет. Если это поле не задано, то счет создается для текущего пользователя.

Необязательный элемент.

Тип: long

onSuccessfulDebitUrl

URL после списания средств.

Необязательный элемент.

Тип: string

onSuccessfulCreditUrl

URL после зачисления средств.

Необязательный элемент.

Тип: string

signature

Код проверки целостности данных.

Необязательный элемент.

Тип: string

lowBalanceThreshold

Если баланс счета меньше данного значения, то раз в сутки уходит уведомление об этом событии.

Необязательный элемент.

Тип: *Money Simple Type*

highBalanceThreshold

Если баланс счета больше данного значения, то раз в сутки уходит уведомление об этом событии.

Необязательный элемент.

Тип: *Money Simple Type*

prototypeAccountId

Счет-прототип с которого берутся свойства "по умолчанию".

Необязательный элемент.

Тип: long

onCancelledDebitUrl

URL после отмены списания средств.

Необязательный элемент.

Тип: string

onCancelledCreditUrl

URL после отмены зачисления средств.

Необязательный элемент.

Тип: string

onAuthoriseUrl

URL после авторизации средств.

Необязательный элемент.

Тип: string

attribute

Свойства счета.

Данные представляются в виде "ключ-значение". Возможные ключи:

- **alias.** Название счета. Название счета должно быть уникальным среди счетов одного пользователя.

Если необходимо Название счета (Псевдоним) сделать публичным, то в элементе attribute для поля alias следует передать элемент published со значение true.

- **interfacetype.** Тип интерфейса.

1 - MONETA.Assistant.

- **testmode.** Тестовый режим (true|false).
- **paymentsystem_limitids.** Список платежных систем.
- **paymentsystem_unitid.** Платежная система по умолчанию.
- **checkurl.** Check URL.
- **payurl.** Pay URL.
- **httpmethod.** HTTP метод для PayUrl, CheckUrl (GET | POST).
- **issignaturemandatory.** Подпись формы оплаты обязательна (true|false).
- **redefinesettingsinurl.** Можно переопределять настройки в url (true|false).
- **successurl.** Success URL.
- **failurl.** Fail URL.
- **inprogressurl.** InProgress URL.
- **returnurl.** Return URL.
- **assistantformtarget.** Target (возврат для iframe). Возможные значения: _blank, _self, _top, _parent.

Необязательный элемент.

Тип: *KeyValueApprovedAttribute Complex Type*

Исходящее сообщение: CreateAccountResponse

Ответ, который приходит на запрос CreateAccountRequest. В ответе содержится номер (AccountId) созданного счета.

XML Schema

```
<xsd:element name="CreateAccountResponse">
  <xsd:simpleType>
    <xsd:restriction base="tns:AccountId"/>
  </xsd:simpleType>
</xsd:element>
```

EditAccount Endpoint

Входящее сообщение: EditAccountRequest

Запрос на редактирование счета. У счета можно изменить свойства "псевдоним" и "платежный пароль".

XML Schema

```
<xsd:element name="EditAccountRequest">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="id" type="tns:AccountId"/>
      <xsd:element minOccurs="0" name="alias" type="xsd:string"/>
      <xsd:element minOccurs="0" name="paymentPassword">
        <xsd:simpleType>
          <xsd:restriction base="xsd:string">
            <xsd:minLength value="5"/>
            <xsd:whiteSpace value="collapse"/>
          </xsd:restriction>
        </xsd:simpleType>
      </xsd:element>
      <xsd:element minOccurs="0" name="paymentPasswordExpirationDate"
type="xsd:boolean"/>
      <xsd:element minOccurs="0" name="oldPaymentPassword">
        <xsd:simpleType>
          <xsd:restriction base="xsd:string">
            <xsd:minLength value="5"/>
            <xsd:whiteSpace value="collapse"/>
          </xsd:restriction>
        </xsd:simpleType>
      </xsd:element>
      <xsd:element minOccurs="0" name="onSuccessfulDebitUrl" type="xsd:string"/>
      <xsd:element minOccurs="0" name="onSuccessfulCreditUrl" type="xsd:string"/>
      <xsd:element minOccurs="0" name="signature" type="xsd:string"/>
      <xsd:element minOccurs="0" name="lowBalanceThreshold" type="tns:Money"/>
      <xsd:element minOccurs="0" name="highBalanceThreshold" type="tns:Money"/>
      <xsd:element minOccurs="0" name="prototypeAccountId" type="xsd:long"/>
      <xsd:element minOccurs="0" name="oldPaymentPasswordChallenge"
type="xsd:string"/>
      <xsd:element minOccurs="0" name="onCancelledDebitUrl" type="xsd:string"/>
      <xsd:element minOccurs="0" name="onCancelledCreditUrl" type="xsd:string"/>
      <xsd:element minOccurs="0" name="onAuthoriseUrl" type="xsd:string"/>
      <xsd:element
maxOccurs="unbounded" minOccurs="0" name="attribute"
type="tns:KeyValueApprovedAttribute"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

Описание

id

Идентификатор редактируемого счета.

Тип: *AccountId Simple Type*

alias

Новый псевдоним счета.

Псевдоним счета должен быть уникальным среди счетов пользователя.

Необязательный элемент.

Тип: string

paymentPassword

Новый платежный пароль.

Для того чтобы изменить платежный пароль, необходимо указать предыдущий платежный пароль.

Необязательный элемент.

Тип: string

Минимальная длина: 5

paymentPasswordExpirationDate

Включить/выключить срок действия платежного пароля. Это поле используется, если в запросе пришел элемент paymentPassword.

Необязательный элемент.

Тип: boolean

oldPaymentPassword

Текущий платежный пароль.

Для того чтобы изменить платежный пароль на новый, необходимо указать текущий платежный пароль.

Необязательный элемент.

Тип: string

Минимальная длина: 5

onSuccessfulDebitUrl

URL после списания средств.

Необязательный элемент.

Тип: string

onSuccessfulCreditUrl

URL после зачисления средств.

Необязательный элемент.

Тип: string

signature

Код проверки целостности данных.

Необязательный элемент.

Тип: string

lowBalanceThreshold

Если баланс счета меньше данного значения, то раз в сутки уходит уведомление об этом событии.

Необязательный элемент.

Тип: *Money Simple Type*

highBalanceThreshold

Если баланс счета больше данного значения, то раз в сутки уходит уведомление об этом событии.

Необязательный элемент.

Тип: *Money Simple Type*

prototypeAccountId

Счет-прототип с которого берутся свойства "по умолчанию".

Необязательный элемент.

Тип: long

oldPaymentPasswordChallenge

Запрос для платежного пароля.

Необязательный элемент.

Тип: string

onCancelledDebitUrl

URL после отмены списания средств.

Необязательный элемент.

Тип: string

onCancelledCreditUrl

URL после отмены зачисления средств.

Необязательный элемент.

Тип: string

onAuthoriseUrl

URL после авторизации средств.

Необязательный элемент.

Тип: string

attribute

Свойства счета.

Данные представляются в виде "ключ-значение". Возможные ключи:

- **alias.** Название счета. Название счета должно быть уникальным среди счетов одного пользователя.

Если необходимо Название счета (Псевдоним) сделать публичным, то в элементе attribute для поля alias следует передать элемент published со значение true.

- **interfacetype.** Тип интерфейса.

1 - MONETA.Assistant.

- **testmode.** Тестовый режим (true|false).
- **paymentsystem_limitids.** Список платежных систем.
- **paymentsystem_unitid.** Платежная система по умолчанию.
- **checkurl.** Check URL.
- **payurl.** Pay URL.
- **httpmethod.** HTTP метод для PayUrl, CheckUrl (GET | POST).
- **issignaturemandatory.** Подпись формы оплаты обязательна (true|false).
- **redefinesettingsinurl.** Можно переопределять настройки в url (true|false).
- **successurl.** Success URL.
- **failurl.** Fail URL.

- **inprogressurl**. InProgress URL.
- **returnurl**. Return URL.
- **assistantformtarget**. Target (возврат для iframe). Возможные значения: `_blank`, `_self`, `_top`, `_parent`.

Для того чтобы удалить какое-то значение счета, требуется передать `key` и пустое значение `value`.

Необязательный элемент.

Тип: *KeyValueApprovedAttribute Complex Type*

Исходящее сообщение: **EditAccountResponse**

Ответ, который приходит на запрос `EditAccountRequest`. В ответе нет никаких данных. Если в процессе сохранения счета произошла ошибка, то возникнет `Exception`. Если `Exception` не возник - значит счет гарантированно сохранен.

XML Schema

```
<xsd:element name="EditAccountResponse">
  <xsd:complexType/>
</xsd:element>
```

FindAccountsList Endpoint

Входящее сообщение: **FindAccountsListRequest**

Запрос на получение списка счетов по заданным условиям поиска.

Тип: *Entity Complex Type*

XML Schema

```
<xsd:element name="FindAccountsListRequest">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="tns:Entity">
        <xsd:sequence>
          <xsd:element minOccurs="0" name="unitId" type="xsd:long"/>
          <xsd:element minOccurs="0" name="alias" type="xsd:string"/>
          <xsd:element minOccurs="0" name="currency" type="tns:Currency"/>
          <xsd:element minOccurs="0" name="isDelegatedAccount" type="xsd:boolean"/>
        >
          <xsd:element minOccurs="0" name="showBankAccount" type="xsd:boolean"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>
```

Описание

unitId

Пользователь, которому принадлежат счета. Если это поле не задано, то используется текущий пользователь.

Необязательный элемент.

Тип: `long`

alias

Название счета. Поиск происходит по прямому совпадению. Для задания маски можно указать спец-символы `"*"` или `"?"`.

Необязательный элемент.

Тип: string

currency

Валюта счета.

Необязательный элемент.

Тип: *Currency Simple Type*

isDelegatedAccount

Является ли счет делегированным.

- Поле не указано - выбрать все счета.
- **false**. Выбрать только недеlegированные счета.
- **true**. Выбрать только делегированные счета.

Необязательный элемент.

Тип: boolean

showBankAccount

Если значение равно **true**, в ответе (в элементе attribute) вернутся поля **bankAccountForCredits** и **bankAccountForDebits**.

Необязательный элемент.

Тип: boolean

Исходящее сообщение: FindAccountsListResponse

Ответ, который содержит список счетов.

XML Schema

```
<xsd:element name="FindAccountsListResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element maxOccurs="unbounded" minOccurs="0" name="account"
        type="tns:AccountInfo"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

BlockAccount Endpoint

Входящее сообщение: BlockAccountRequest

Блокировка счета.

Тип: *AbstractAttributeObject Complex Type*

XML Schema

```
<xsd:element name="BlockAccountRequest">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="tns:AbstractAttributeObject">
        <xsd:sequence>
          <xsd:element name="id" type="tns:AccountId"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>
```

Описание

id

Идентификатор счета.

Тип: *AccountId Simple Type*

Исходящее сообщение: BlockAccountResponse

Ответ, который приходит на запрос BlockAccountRequest. В ответе нет никаких данных. Если в процессе блокирования счета произошла ошибка, то возникнет Exception. Если Exception не возник - значит счет гарантированно заблокирован.

Тип: *AbstractAttributeObject Complex Type*

XML Schema

```
<xsd:element name="BlockAccountResponse">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="tns:AbstractAttributeObject"/>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>
```

UnblockAccount Endpoint

Входящее сообщение: UnblockAccountRequest

Разблокирование счета.

Если счет был заблокирован с помощью метода BlockAccountRequest или в личном кабинете пользователя, то для разблокирования счета нужно передать в запросе "ответ на секретный вопрос" и "платежный пароль" этого счета. Разблокировать счет может только тот пользователь (с тем же логином), который заблокировал счет.

Если счет был заблокирован после ввода неверного платежного пароля (например, во время оплаты), то для разблокирования счета требуется только ответ на секретный вопрос.

Тип: *AbstractAttributeObject Complex Type*

XML Schema

```
<xsd:element name="UnblockAccountRequest">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="tns:AbstractAttributeObject">
        <xsd:sequence>
          <xsd:element name="id" type="tns:AccountId"/>
          <xsd:element name="secretAnswer" type="xsd:string"/>
          <xsd:element minOccurs="0" name="paymentPassword"
type="tns:PaymentPassword"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>
```

Описание

id

Идентификатор счета.

Тип: *AccountId Simple Type*

secretAnswer

Ответ на секретный вопрос (в разделе Безопасность).

Тип: string

paymentPassword

Платежный пароль.

Платежный пароль требуется, если счет был заблокирован с помощью метода BlockAccountRequest или в личном кабинете пользователя.

Если счет был заблокирован после ввода неверного платежного пароля, то этот элемент не требуется.

Необязательный элемент.

Тип: *PaymentPassword Complex Type*

Исходящее сообщение: UnblockAccountResponse

Ответ, который приходит на запрос UnblockAccountRequest. В ответе нет никаких данных. Если в процессе разблокирования счета произошла ошибка, то возникнет Exception. Если Exception не возник - значит счет гарантированно разблокирован.

Тип: *AbstractAttributeObject Complex Type*

XML Schema

```

<xsd:element name="UnblockAccountResponse">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="tns:AbstractAttributeObject"/>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>

```

Документы, юридические реквизиты, договора

CreateProfileDocument Endpoint

Входящее сообщение: CreateProfileDocumentRequest

Запрос на создание документа пользователя.

Тип: *Document Complex Type*

XML Schema

```

<xsd:element name="CreateProfileDocumentRequest">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="tns:Document">
        <xsd:sequence>
          <xsd:element minOccurs="0" name="unitId" type="xsd:long"/>
          <xsd:element minOccurs="0" name="profileId" type="xsd:long"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>

```

Описание

unitId

Пользователь, которому будет принадлежать данный документ.

Если это поле не задано, то документ создается для текущего пользователя.

Необязательный элемент.

Тип: long

profileId

Необязательный элемент.

Тип: long

Исходящее сообщение: CreateProfileDocumentResponse

Ответ на запрос создания документа в профиле.

XML Schema

```
<xsd:element name="CreateProfileDocumentResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="id" type="xsd:long"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

Описание

id

Идентификатор документа в системе МОНЕТА.РУ.

Тип: long

EditProfileDocument Endpoint

Входящее сообщение: EditProfileDocumentRequest

Запрос на редактирование документа пользователя.

Тип: *Document Complex Type*

XML Schema

```
<xsd:element name="EditProfileDocumentRequest">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="tns:Document">
        <xsd:sequence>
          <xsd:element minOccurs="0" name="unitId" type="xsd:long"/>
          <xsd:element minOccurs="0" name="profileId" type="xsd:long"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>
```

Описание

unitId

Пользователь, которому принадлежит данный документ.

Если это поле не задано, то документ ищется для текущего пользователя.

Необязательный элемент.

Тип: long

profileId

Необязательный элемент.

Тип: long

Исходящее сообщение: EditProfileDocumentResponse

Ответ, который приходит на запрос EditProfileDocumentRequest.

В ответе нет никаких данных. Если в процессе сохранения документа произошла ошибка, то возникнет Exception.

Если Exception не возник - значит документ гарантированно сохранен.

XML Schema

```

<xsd:element name="EditProfileDocumentResponse">
  <xsd:complexType/>
</xsd:element>

```

FindContracts Endpoint

Входящее сообщение: FindContractsRequest

Запрос на получение данных договора по ID пользователя. Если данные не найдены, возвращается пустой список.

Тип: *Entity Complex Type*

XML Schema

```

<xsd:element name="FindContractsRequest">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="tns:Entity">
        <xsd:sequence>
          <xsd:element minOccurs="0" name="unitId" type="xsd:long"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>

```

Описание

unitId

ID пользователя в системе МОНЕТА.РУ.

Если это поле не задано, то используется текущий пользователь.

Необязательный элемент.

Тип: long

Исходящее сообщение: FindContractsResponse

Ответ на запрос FindContractsRequest.

XML Schema

```

<xsd:element name="FindContractsResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element maxOccurs="unbounded" minOccurs="0" name="contract"
type="tns:Contract"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>

```

FindLegalInformation Endpoint

Входящее сообщение: FindLegalInformationRequest

Запрос на получение юридических реквизитов по ID пользователя. Если данные не найдены, возвращается пустой список.

Тип: *Entity Complex Type*

XML Schema

```
<xsd:element name="FindLegalInformationRequest">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="tns:Entity">
        <xsd:sequence>
          <xsd:element minOccurs="0" name="unitId" type="xsd:long"/>
          <xsd:element minOccurs="0" name="profileId" type="xsd:long"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>
```

Описание

unitId

ID пользователя в системе МОНЕТА.РУ.

Если это поле не задано, то используется текущий пользователь.

Необязательный элемент.

Тип: long

profileId

Необязательный элемент.

Тип: long

Исходящее сообщение: FindLegalInformationResponse

Ответ на запрос FindLegalInformationRequest.

XML Schema

```
<xsd:element name="FindLegalInformationResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element maxOccurs="unbounded" minOccurs="0" name="legalInformation"
        type="tns:LegalInformation"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

FindProfileDocumentFiles Endpoint

Входящее сообщение: FindProfileDocumentFilesRequest

Запрос на получение бинарных данных, ассоциированных с указанным документом.

XML Schema

```
<xsd:element name="FindProfileDocumentFilesRequest">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="documentId" type="xsd:long"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

```

    </xsd:sequence>
  </xsd:complexType>
</xsd:element>

```

Исходящее сообщение: FindProfileDocumentFilesResponse

Ответ, содержащий бинарные данные, ассоциированные с указанным в запросе документом.

XML Schema

```

<xsd:element name="FindProfileDocumentFilesResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element maxOccurs="unbounded" minOccurs="0" name="file" type="tns:File"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>

```

FindProfileDocuments Endpoint

Входящее сообщение: FindProfileDocumentsRequest

Запрос на получение документов пользователя.

XML Schema

```

<xsd:element name="FindProfileDocumentsRequest">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="unitId" type="xsd:long"/>
      <xsd:element minOccurs="0" name="profileId" type="xsd:long"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>

```

Описание

unitId

ID пользователя, документы которого запрашиваются.

Тип: long

profileId

Необязательный элемент.

Тип: long

Исходящее сообщение: FindProfileDocumentsResponse

Ответ на запрос FindProfileDocumentsRequest.

XML Schema

```

<xsd:element name="FindProfileDocumentsResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element maxOccurs="unbounded" minOccurs="0" name="document"
type="tns:Document"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>

```

Описание

document

Список найденных документов. Если документов нет - список пустой.

Необязательный элемент.

Тип: *Document Complex Type*

UploadProfileDocumentFile Endpoint

Входящее сообщение: UploadProfileDocumentFileRequest

Запрос на создание/редактирование файла документа.

XML Schema

```
<xsd:element name="UploadProfileDocumentFileRequest">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="file" type="tns:File"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

Описание

file

Если указано поле fileId - файл редактируется, иначе создается новый файл.

Поле approved в данном запросе не используется.

Если не указан mimeType, но в запросе есть имя файла (поле title), то mimeType будет вычисляться по расширению файла.

Тип: *File Complex Type*

Исходящее сообщение: UploadProfileDocumentFileResponse

Ответ на запрос создания/редактирования файла документа.

Если в процессе сохранения файла произошла ошибка, то возникнет Exception. Если Exception не возник - значит файл гарантированно сохранен.

XML Schema

```
<xsd:element name="UploadProfileDocumentFileResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="fileId" type="xsd:long"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

Описание

fileId

ID файла.

Тип: long

Банковские реквизиты

FindBankAccounts Endpoint

Входящее сообщение: FindBankAccountsRequest

Запрос на получение банковских реквизитов по ID пользователя. Если данные не найдены, возвращается пустой список.

Тип: *Entity Complex Type*

XML Schema

```
<xsd:element name="FindBankAccountsRequest">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="tns:Entity">
        <xsd:sequence>
          <xsd:element minOccurs="0" name="unitId" type="xsd:long"/>
          <xsd:element minOccurs="0" name="profileId" type="xsd:long"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>
```

Описание

unitId

ID пользователя в системе МОНЕТА.РУ. Если это поле не задано, то используется текущий пользователь.

Необязательный элемент.

Тип: long

profileId

Необязательный элемент.

Тип: long

Исходящее сообщение: FindBankAccountsResponse

Ответ на запрос FindBankAccountsRequest.

XML Schema

```
<xsd:element name="FindBankAccountsResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element maxOccurs="unbounded" minOccurs="0" name="bankAccount"
type="tns:BankAccount"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

EditBankAccount Endpoint

Входящее сообщение: EditBankAccountRequest

Запрос на редактирование банковских реквизитов пользователя.

Тип: *BankAccount Complex Type*

XML Schema

```
<xsd:element name="EditBankAccountRequest">
  <xsd:complexType>
    <xsd:complexContent>
```

```

<xsd:extension base="tns:BankAccount">
  <xsd:sequence>
    <xsd:element minOccurs="0" name="unitId" type="xsd:long"/>
    <xsd:element minOccurs="0" name="profileId" type="xsd:long"/>
  </xsd:sequence>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>
</xsd:element>

```

Описание

unitId

Пользователь, которому принадлежат банковские реквизиты.

Если это поле не задано, то банковские реквизиты ищутся для текущего пользователя.

Необязательный элемент.

Тип: long

profileId

Необязательный элемент.

Тип: long

Исходящее сообщение: EditBankAccountResponse

Ответ, который приходит на запрос EditBankAccountRequest.

В ответе нет никаких данных. Если в процессе сохранения банковских реквизитов произошла ошибка, то возникнет Exception. Если Exception не возник - значит банковские реквизиты гарантированно сохранены.

XML Schema

```

<xsd:element name="EditBankAccountResponse">
  <xsd:complexType/>
</xsd:element>

```

CreateBankAccount Endpoint

Входящее сообщение: CreateBankAccountRequest

Запрос на создание банковских реквизитов.

Тип: *BankAccount Complex Type*

XML Schema

```

<xsd:element name="CreateBankAccountRequest">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="tns:BankAccount">
        <xsd:sequence>
          <xsd:element minOccurs="0" name="unitId" type="xsd:long"/>
          <xsd:element minOccurs="0" name="profileId" type="xsd:long"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>

```

Описание

unitId

Пользователь, которому будут принадлежать банковские реквизиты.

Если это поле не задано, то банковские реквизиты создаются для текущего пользователя.

Необязательный элемент.

Тип: long

profileId

Необязательный элемент.

Тип: long

Исходящее сообщение: CreateBankAccountResponse

Ответ на запрос создания банковских реквизитов.

XML Schema

```
<xsd:element name="CreateBankAccountResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="id" type="xsd:long"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

Описание

id

Идентификатор банковских реквизитов в системе МОНЕТА.РУ.

Тип: long

Делегированный доступ к счетам

FindAccountRelations Endpoint

Входящее сообщение: FindAccountRelationsRequest

Запрос на получение списка пользователей, которые имеют делегированный доступ к указанному счету.

XML Schema

```
<xsd:element name="FindAccountRelationsRequest">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="accountId" type="tns:AccountId"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

Описание

accountId

Номер счета в системе МОНЕТА.РУ.

Тип: *AccountId Simple Type*

Исходящее сообщение: FindAccountRelationsResponse

Ответ на запрос FindAccountRelationsRequest.

XML Schema

```
<xsd:element name="FindAccountRelationsResponse">
```



```

<xsd:complexType>
  <xsd:sequence>
    <xsd:element maxOccurs="unbounded" minOccurs="0" name="accountRelation"
      type="tns:AccountRelation"/>
  </xsd:sequence>
</xsd:complexType>
</xsd:element>

```

Описание

accountRelation

Список найденных пользователей, которые имеют делегированный доступ к указанному счету. Если пользователей нет - список пустой.

Необязательный элемент.

Тип: *AccountRelation Complex Type*

GetAccountRelation Endpoint

Входящее сообщение: GetAccountRelationRequest

Запрос на получение данных по делегированному доступу.

XML Schema

```

<xsd:element name="GetAccountRelationRequest">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="accountId" type="tns:AccountId"/>
      <xsd:element name="principalEmail" type="tns:Email"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>

```

Описание

accountId

Номер счета в системе МОНЕТА.РУ.

Тип: *AccountId Simple Type*

principalEmail

Email пользователя.

Тип: *Email Simple Type*

Исходящее сообщение: GetAccountRelationResponse

Ответ на запрос GetAccountRelationRequest.

XML Schema

```

<xsd:element name="GetAccountRelationResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element maxOccurs="1" minOccurs="0" name="accountRelation"
        type="tns:AccountRelation"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>

```

SaveAccountRelation Endpoint

Входящее сообщение: SaveAccountRelationRequest

Запрос на сохранение делегированного доступа к счету.

Тип: *AccountRelation Complex Type*

XML Schema

```
<xsd:element name="SaveAccountRelationRequest">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="tns:AccountRelation">
        <xsd:sequence>
          <xsd:element name="paymentPassword" type="tns:Password"/>
          <xsd:element minOccurs="0" name="paymentPasswordChallenge"
type="xsd:string"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>
```

Описание

paymentPassword

Платежный пароль.

Тип: *Password Simple Type*

paymentPasswordChallenge

Запрос для платежного пароля.

Необязательный элемент.

Тип: string

Исходящее сообщение: SaveAccountRelationResponse

Ответ, который приходит на запрос SaveAccountRelationRequest.

В ответе нет никаких данных. Если в процессе сохранения счета произошла ошибка, то возникнет Exception. Если Exception не возник - значит делегированный доступ гарантированно сохранен.

XML Schema

```
<xsd:element name="SaveAccountRelationResponse">
  <xsd:complexType/>
</xsd:element>
```

DeleteAccountRelation Endpoint

Входящее сообщение: DeleteAccountRelationRequest

Запрос на удаление делегированного доступа к счету.

XML Schema

```
<xsd:element name="DeleteAccountRelationRequest">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="accountId" type="tns:AccountId"/>
      <xsd:element name="principalEmail" type="tns:Email"/>
    </xsd:sequence>
  </xsd:complexType>
```

```
</xsd:element>
```

Описание

accountId

Номер счета в системе МОНЕТА.РУ.

Тип: *AccountId Simple Type*

principalEmail

Email пользователя.

Тип: *Email Simple Type*

Исходящее сообщение: DeleteAccountRelationResponse

Ответ, который приходит на запрос DeleteAccountRelationRequest.

В ответе нет никаких данных. Если в процессе сохранения счета произошла ошибка, то возникнет Exception. Если Exception не возник - значит делегированный доступ гарантированно удален.

XML Schema

```
<xsd:element name="DeleteAccountRelationResponse">
  <xsd:complexType/>
</xsd:element>
```

Отчеты

GetTurnoverList Endpoint

Входящее сообщение: GetTurnoverListRequest

Получение информации об итогах по месяцам.

Если данные не найдены, то size в ответе равен 0.

XML Schema

```
<xsd:element name="GetTurnoverListRequest">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element minOccurs="0" name="unitId" type="xsd:long"/>
      <xsd:element name="dateFrom" type="xsd:date"/>
      <xsd:element minOccurs="0" name="dateTo" type="xsd:date"/>
      <xsd:element maxOccurs="unbounded" minOccurs="0" name="accountIds"
type="tns:AccountId"/>
      <xsd:element minOccurs="0" name="currencyCode" type="tns:Currency"/>
      <xsd:element default="true" minOccurs="0" name="groupByCurrency"
type="xsd:boolean"/>
      <xsd:element minOccurs="0" name="operationTypeCategory"
type="tns:OperationTypeCategory"/>
      <xsd:element minOccurs="0" name="categoryDetails" type="xsd:boolean"/>
      <xsd:element maxOccurs="unbounded" minOccurs="0" name="attribute"
type="tns:KeyValueAttribute"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

Описание

unitId

ID пользователя в системе МОНЕТА.РУ.

Если это поле не задано, то используется текущий пользователь.

Необязательный элемент.

Тип: long

dateFrom

Дата начала периода. Дата берется с начала указанного месяца.

Тип: date

dateTo

Дата окончания периода. В дате устанавливается конец указанного месяца. Если дата не установлена, то берется конец месяца из даты dateFrom.

Необязательный элемент.

Тип: date

accountIds

Список счетов, которые учитываются в итогах по месяцам.

Необязательный элемент.

Тип: *AccountId Simple Type*

currencyCode

Валюта, которая учитывается в итогах по месяцам.

Необязательный элемент.

Тип: *Currency Simple Type*

groupByCurrency

Если groupByCurrency=true, то в ответе делается группировка по валюте.

Необязательный элемент.

Тип: boolean

Значение по умолчанию: true

operationTypeCategory

Категория операций, которая учитывается в итогах по месяцам.

Необязательный элемент.

Тип: *OperationTypeCategory Simple Type*

categoryDetails

Если categoryDetails=true, то в ответе делается группировка по категориям операций.

Необязательный элемент.

Тип: boolean

attribute

Дополнительные атрибуты запроса.

Необязательный элемент.

Тип: *KeyValueAttribute Complex Type*

Исходящее сообщение: GetTurnoverListResponse

Ответ на запрос информации об итогах по месяцам.

XML Schema

```
<xsd:element name="GetTurnoverListResponse">
```

```

<xsd:complexType>
  <xsd:sequence>
    <xsd:element default="true" minOccurs="0" name="groupByCurrency"
type="xsd:boolean"/>
    <xsd:element maxOccurs="unbounded" minOccurs="0" name="turnover">
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element minOccurs="0" name="date" type="xsd:date"/>
          <xsd:element minOccurs="0" name="account" type="xsd:string"/>
          <xsd:element name="currency" type="tns:Currency"/>
          <xsd:element minOccurs="0" name="operationTypeCategory"
type="tns:OperationTypeCategory"/>
          <xsd:element name="incomeAmount" type="tns:Money"/>
          <xsd:element name="incomeCommission" type="tns:Money"/>
          <xsd:element name="incomeTransactionsCount" type="xsd:long"/>
          <xsd:element minOccurs="0" name="incomeLockedBalance"
type="tns:Money"/>
          <xsd:element name="expenseAmount" type="tns:Money"/>
          <xsd:element name="expensesIncludingCommission" type="tns:Money"/>
          <xsd:element name="expensesExtraCommission" type="tns:Money"/>
          <xsd:element name="expenseTransactionsCount" type="xsd:long"/>
          <xsd:element minOccurs="0" name="expenseLockedBalance"
type="tns:Money"/>
          <xsd:element name="total" type="tns:Money"/>
          <xsd:element name="openingBalance" type="tns:Money"/>
          <xsd:element name="closingBalance" type="tns:Money"/>
        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
  </xsd:sequence>
</xsd:complexType>
</xsd:element>

```

Описание

groupByCurrency

Если groupByCurrency=true, то в ответе делается группировка по валюте.

Необязательный элемент.

Тип: boolean

Значение по умолчанию: true

turnover

Необязательный элемент.

date

Дата.

Необязательный элемент.

Тип: date

account

Счет.

Необязательный элемент.

Тип: string

currency

Валюта.

Тип: *Currency Simple Type*

operationTypeCategory

Категория операций (возвращается, если в запросе categoryDetails=true).

Необязательный элемент.

Тип: *OperationTypeCategory Simple Type*

incomeAmount

Приход. Сумма.

Тип: *Money Simple Type*

incomeCommission

Приход. Комиссия.

Тип: *Money Simple Type*

incomeTransactionsCount

Приход. Количество.

Тип: long

incomeLockedBalance

Приход. Заблокированная сумма.

Необязательный элемент.

Тип: *Money Simple Type*

expenseAmount

Расход. Сумма.

Тип: *Money Simple Type*

expensesIncludingCommission

Расход. В том числе комиссия.

Тип: *Money Simple Type*

expensesExtraCommission

Расход. Дополнительная комиссия.

Тип: *Money Simple Type*

expenseTransactionsCount

Расход. Количество.

Тип: long

expenseLockedBalance

Расход. Заблокированная сумма.

Необязательный элемент.

Тип: *Money Simple Type*

total

Итого.

Тип: *Money Simple Type*

openingBalance

Входящий остаток.

Тип: *Money Simple Type*

closingBalance

Исходящий остаток.

Тип: *Money Simple Type*

GetFinancialFlowsList Endpoint

Входящее сообщение: GetFinancialFlowsListRequest

Получение информации о финансовых потоках.

Если данные не найдены, то size в ответе равен 0.

XML Schema

```
<xsd:element name="GetFinancialFlowsListRequest">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element minOccurs="0" name="unitId" type="xsd:long"/>
      <xsd:element name="dateFrom" type="xsd:date"/>
      <xsd:element minOccurs="0" name="dateTo" type="xsd:date"/>
      <xsd:element maxOccurs="unbounded" minOccurs="0" name="accountIds"
type="tns:AccountId"/>
      <xsd:element minOccurs="0" name="currencyCode" type="tns:Currency"/>
      <xsd:element minOccurs="0" name="operationTypeCategory"
type="tns:OperationTypeCategory"/>
      <xsd:element minOccurs="0" name="operationAmountType"
type="tns:OperationAmountType"/>
      <xsd:element minOccurs="0" name="categoryDetails" type="xsd:boolean"/>
      <xsd:element maxOccurs="unbounded" minOccurs="0" name="attribute"
type="tns:KeyValueAttribute"/>
      <xsd:element minOccurs="0" name="operationStatusState"
type="tns:OperationStatusState"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

Описание

unitId

ID пользователя в системе МОНЕТА.РУ.

Если это поле не задано, то используется текущий пользователь.

Необязательный элемент.

Тип: long

dateFrom

Дата начала финансового потока. Дата берется с начала указанного месяца.

Тип: date

dateTo

Дата окончания финансового потока. В дате устанавливается конец указанного месяца.

Если дата не установлена, то берется конец месяца из даты dateFrom. Период просмотра финансовых потоков не может быть больше 3 месяцев.

Необязательный элемент.

Тип: date

accountIds

Список счетов, которые учитываются в финансовом потоке.

Необязательный элемент.

Тип: *AccountId Simple Type*

currencyCode

Валюта, которая учитывается в финансовом потоке.

Необязательный элемент.

Тип: *Currency Simple Type*

operationTypeCategory

Категория операций, которая учитывается в финансовом потоке.

Необязательный элемент.

Тип: *OperationTypeCategory Simple Type*

operationAmountType

Расходные или приходные операции. При отсутствии этого поля учитываются все операции.

Необязательный элемент.

Тип: *OperationAmountType Simple Type*

categoryDetails

Если categoryDetails=true, то в ответе делается группировка по категориям операций.

Необязательный элемент.

Тип: boolean

attribute

Дополнительные атрибуты запроса.

Необязательный элемент.

Тип: *KeyValueAttribute Complex Type*

operationStatusState

Состояние операций в данном финансовом потоке. Если это поле не установлено - то потоки со всеми состояниями.

Необязательный элемент.

Тип: *OperationStatusState Simple Type*

Исходящее сообщение: GetFinancialFlowsListResponse

Ответ на запрос информации о финансовых потоках.

XML Schema

```
<xsd:element name="GetFinancialFlowsListResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element maxOccurs="unbounded" minOccurs="0" name="financialFlow">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element minOccurs="0" name="date" type="xsd:string"/>
            <xsd:element name="payerSideAccess" type="xsd:boolean"/>
            <xsd:element name="payerFlowId" type="xsd:long"/>
            <xsd:element name="payerFlowName" type="xsd:string"/>
            <xsd:element name="payerCurrencyCode" type="tns:Currency"/>
            <xsd:element name="payeeSideAccess" type="xsd:boolean"/>
            <xsd:element name="payeeFlowId" type="xsd:long"/>
            <xsd:element name="payeeFlowName" type="xsd:string"/>
            <xsd:element name="payeeCurrencyCode" type="tns:Currency"/>
            <xsd:element minOccurs="0" name="payerDebited" type="tns:Money"/>
            <xsd:element minOccurs="0" name="payerWithheld" type="tns:Money"/>
            <xsd:element minOccurs="0" name="payerFee" type="tns:Money"/>
            <xsd:element minOccurs="0" name="payerExchangeFee" type="tns:Money"/>
            <xsd:element minOccurs="0" name="payeeCredited" type="tns:Money"/>
            <xsd:element minOccurs="0" name="payeeFee" type="tns:Money"/>
            <xsd:element name="transactionsCount" type="xsd:long"/>
            <xsd:element minOccurs="0" name="operationTypeCategory"
              type="tns:OperationTypeCategory"/>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```



```

        <xsd:element name="operationStatusState"
type="tns:OperationStatusState"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>

```

Описание

financialFlow

Необязательный элемент.

date

Дата финансового потока (формат уууу-mm)

Необязательный элемент.

Тип: string

payerSideAccess

Есть ли доступ к финансовому потоку плательщика.

Тип: boolean

payerFlowId

ID финансового потока плательщика.

Тип: long

payerFlowName

Название финансового потока плательщика.

Тип: string

payerCurrencyCode

Валюта финансового потока плательщика.

Тип: *Currency Simple Type*

payeeSideAccess

Есть ли доступ к финансовому потоку получателя.

Тип: boolean

payeeFlowId

ID финансового потока получателя.

Тип: long

payeeFlowName

Название финансового потока получателя.

Тип: string

payeeCurrencyCode

Валюта финансового потока получателя.

Тип: *Currency Simple Type*

payerDebited

Сумма, списанная с плательщика.

Необязательный элемент.

Тип: *Money Simple Type*

payerWithheld

Сумма, удержанная с плательщика.

Необязательный элемент.

Тип: *Money Simple Type*

payerFee

Сумма комиссии плательщика.

Необязательный элемент.

Тип: *Money Simple Type*

payerExchangeFee

Сумма обменной комиссии плательщика.

Необязательный элемент.

Тип: *Money Simple Type*

payeeCredited

Сумма, зачисленная получателю.

Необязательный элемент.

Тип: *Money Simple Type*

payeeFee

Сумма комиссии получателя.

Необязательный элемент.

Тип: *Money Simple Type*

transactionsCount

Количество операций в данном финансовом потоке.

Тип: long

operationTypeCategory

Категория операций в данном финансовом потоке (возвращается, если в запросе categoryDetails=true).

Необязательный элемент.

Тип: *OperationTypeCategory Simple Type*

operationStatusState

Состояние операций в данном финансовом потоке.

Тип: *OperationStatusState Simple Type*

FindReports Endpoint

Входящее сообщение: FindReportsRequest

Запрос на получение отчетов по ID пользователя. Если данные не найдены, возвращается пустой список.

XML Schema

```

<xsd:element name="FindReportsRequest">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element minOccurs="0" name="unitId" type="xsd:long"/>
      <xsd:element minOccurs="0" name="reportId" type="xsd:long"/>
      <xsd:element minOccurs="0" name="year" type="xsd:int"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>

```

```
</xsd:complexType>
</xsd:element>
```

Описание

unitId

ID пользователя в системе МОНЕТА.РУ. Если это поле не задано, то используется текущий пользователь.

Необязательный элемент.

Тип: long

reportId

ID отчета.

Если в запросе указан ID отчета, то в ответе (FindReportsResponse) в списке отчетов (элемент report) будет один отчет.

Необязательный элемент.

Тип: long

year

Год, за который следует искать экземпляры отчета.

Если в запросе это поле не будет указано, то в ответе (FindReportsResponse) будут возвращены только отчеты, а экземпляры отчетов возвращены не будут. То есть, если в запросе не указать элемент year, то в ответе элемент reportInstance (список экземпляров отчета) для типа Report будет пустым.

Необязательный элемент.

Тип: int

Исходящее сообщение: FindReportsResponse

Ответ на запрос FindReportsRequest.

XML Schema

```
<xsd:element name="FindReportsResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element maxOccurs="unbounded" minOccurs="0" name="report"
type="tns:Report"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

Описание

report

Список отчетов.

Необязательный элемент.

Тип: *Report Complex Type*

Идентификация пользователя

1. Перед тем как начать процедуру идентификации пользователь должен заполнить следующие поля в своем профайле:

- Имя (first_name)
- Отчество (middle_initial_name)

- Фамилия (last_name)
 - Дата рождения (date_of_birth)
 - Адрес регистрации (legal_address)
 - ИНН (inn)
 - Сотовый телефон (cell_phone) должен быть подтвержден
 - Паспортные данные
- Для того чтобы получить уникальный код идентификации, пользователь может перейти по ссылке <https://www.moneta.ru/personificationCode.htm>, либо центр идентификации может вызвать метод `GetPersonificationCodeRequest`, чтобы получить код идентификации для пользователя.

Примечание: Код идентификации действует 7 дней. Если пользователь не закончил процедуру идентификации за это время, ему следует получить новый код.
 - Центр идентификации использует метод `VerifyPersonificationCodeRequest` для того, чтобы получить информацию о пользователе. Если пользователь предоставил недостаточно информации, то Moneta.Ru вернет ошибку.
 - После того как центр идентификации убедится, что предоставленная пользователем информация верна, центр идентификации использует метод `ConfirmPersonificationRequest` для завершения процедуры идентификации.

GetPersonificationCode Endpoint

Входящее сообщение: `GetPersonificationCodeRequest`

Запрос на получение кода, с помощью которого можно произвести идентификацию пользователя.

XML Schema

```
<xsd:element name="GetPersonificationCodeRequest">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:choice minOccurs="0">
        <xsd:element name="accountId" type="tns:AccountId"/>
        <xsd:element name="unitId" type="xsd:long"/>
      </xsd:choice>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

Описание

accountId

Номер счета в системе МОНЕТА.РУ.

Тип: *AccountId Simple Type*

unitId

ID пользователя в системе МОНЕТА.РУ.

Если это поле не задано, то используется текущий пользователь.

Тип: long

Исходящее сообщение: `GetPersonificationCodeResponse`

Ответ, который приходит на запрос `GetPersonificationCodeRequest`.

XML Schema

```
<xsd:element name="GetPersonificationCodeResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="personificationCode" type="xsd:string"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

```
</xsd:element>
```

Описание

personificationCode

Код, с помощью которого можно произвести идентификацию пользователя.

Тип: string

VerifyPersonificationCode Endpoint

Входящее сообщение: VerifyPersonificationCodeRequest

Проверка кода идентификации пользователя.

Тип: *Entity Complex Type*

XML Schema

```
<xsd:element name="VerifyPersonificationCodeRequest">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="tns:Entity">
        <xsd:sequence>
          <xsd:element name="personificationCode" type="xsd:string"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>
```

Описание

personificationCode

Код, с помощью которого можно произвести идентификацию пользователя.

Тип: string

Исходящее сообщение: VerifyPersonificationCodeResponse

Ответ, который приходит на запрос VerifyPersonificationCodeRequest.

Если код идентификации можно использовать (то есть, он найден и срок его действия не закончился), то в ответе придут данные пользователя, которые есть в системе Монета.Ру.

Если код идентификации использовать нельзя, то возникнет Exception.

XML Schema

```
<xsd:element name="VerifyPersonificationCodeResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="personalInformation" type="tns:PersonalInformation"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

Описание

personalInformation

В profile могут быть следующие поля (если они заполнены пользователем):

- first_name
- middle_initial_name
- last_name
- date_of_birth

- формат: yyyy-mm-dd
 - legal_address
 - inn
- Тип: *PersonalInformation Complex Type*

ConfirmPersonification Endpoint

Входящее сообщение: ConfirmPersonificationRequest

Запрос на проведение идентификации пользователя.

Тип: *Entity Complex Type*

XML Schema

```
<xsd:element name="ConfirmPersonificationRequest">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="tns:Entity">
        <xsd:sequence>
          <xsd:element name="personificationCode" type="xsd:string"/>
          <xsd:element name="personalInformation" type="tns:PersonalInformation"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>
```

Описание

personificationCode

Код, с помощью которого можно произвести идентификацию пользователя.

Тип: string

personalInformation

В profile должны быть заполнены следующие поля:

- first_name
- middle_initial_name (optional)
- last_name
- date_of_birth

формат: yyyy-mm-dd

- legal_address
- inn (optional)

В document должны быть заполнены следующие поля:

- type (всегда равен PASSPORT)
- series
- number
- issuer
- issued
- department

Тип: *PersonalInformation Complex Type*

Исходящее сообщение: ConfirmPersonificationResponse

Ответ, который приходит на запрос ConfirmPersonificationRequest.

XML Schema

```
<xsd:element name="ConfirmPersonificationResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="personalInformation" type="tns:PersonalInformation"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

Описание

personalInformation

Если идентификация пользователя прошла успешно, то данный элемент содержит текущие данные пользователя.

Тип: *PersonalInformation Complex Type*

SimplifiedIdentification Endpoint

Входящее сообщение: SimplifiedIdentificationRequest

Запрос на проведение Упрощённой идентификации пользователя.

Для прохождения упрощённой идентификации у пользователя в личном кабинете должны быть заполнены:

- Фамилия
- Имя
- Отчество
- Сотовый телефон
- СНИЛС
- ИНН (необязательно)
- Паспорт РФ (в разделе Документы)

Упрощённая идентификация доступна только для граждан России. Поэтому, если в личном кабинете указана страна, то это должна быть Россия.

Сотовый телефон в личном кабинете пользователя должен быть "подтвержден". Если он не подтвержден, то в ответе будет ошибка с кодом "500.7.5". Для подтверждения сотового телефона следует использовать запросы **ApprovePhoneSendConfirmationRequest** и **ApprovePhoneApplyCodeRequest**.

Если в личном кабинете пользователя все необходимые данные уже заполнены (например, запросами **EditProfileRequest**, **CreateProfileDocumentRequest** или **EditProfileDocumentRequest**), и сотовый телефон подтвержден, то в данном запросе элемент `personalInformation` можно не присылать. Если элемент `personalInformation` будет заполнен (полностью или частично), то все присланные данные будут сохранены в личном кабинете пользователя.

Упрощённая идентификация происходит обычно в течение часа.

Тип: *AbstractAttributeObject Complex Type*

XML Schema

```
<xsd:element name="SimplifiedIdentificationRequest">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="tns:AbstractAttributeObject">
        <xsd:sequence>
          <xsd:element minOccurs="0" name="unitId" type="xsd:long"/>
          <xsd:element minOccurs="0" name="personalInformation"
            type="tns:PersonalInformation"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
```

```
</xsd:element>
```

Описание

unitId

ID пользователя в системе МОНЕТА.РУ. Если это поле не задано, то используется текущий пользователь.

Необязательный элемент.

Тип: long

personalInformation

В profile должны быть заполнены следующие поля:

- first_name
- middle_initial_name
- last_name
- cell_phone
- snils
- inn (optional)

В document должны быть заполнены следующие поля:

- type (всегда равен PASSPORT)
- series
- number
- issuer
- issued
- department

Необязательный элемент.

Тип: *PersonalInformation Complex Type*

Исходящее сообщение: SimplifiedIdentificationResponse

Ответ, который приходит на запрос SimplifiedIdentificationRequest.

Тип: *AbstractAttributeObject Complex Type*

XML Schema

```
<xsd:element name="SimplifiedIdentificationResponse">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="tns:AbstractAttributeObject">
        <xsd:sequence>
          <xsd:element name="success" type="xsd:boolean"/>
          <xsd:element minOccurs="0" name="error" type="xsd:string"/>
          <xsd:element minOccurs="0" name="errorCode" type="xsd:string"/>
          <xsd:element name="personalInformation" type="tns:PersonalInformation"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>
```

Описание

success

- **true.** Упрощённая идентификация проведена успешно.
- **false.** Упрощённая идентификация не была проведена.

Тип: boolean

error

Поле содержит описание ошибки, если Упрощённая идентификация не была проведена.

Если Упрощённая идентификация проведена успешно, то поле error пустое.

Необязательный элемент.

Тип: string

errorCode

Поле содержит код ошибки, если Упрощённая идентификация не была проведена.

Если Упрощённая идентификация проведена успешно, то поле errorCode пустое.

Коды ошибок совпадают со значением элемента faultDetail (смотрите описание для этого элемента).

Необязательный элемент.

Тип: string

personalInformation

Текущие персональные данные пользователя.

В profile возвращаются следующие поля:

- first_name
- middle_initial_name
- last_name
- cell_phone
- snils
- inn (optional)

В document возвращаются следующие поля:

- type
- series
- number
- issuer
- issued
- department

Тип: *PersonalInformation Complex Type*

Проверка телефона пользователя

ApprovePhoneApplyCode Endpoint

Входящее сообщение: ApprovePhoneApplyCodeRequest

Запрос на подтверждение (approval=true) или отмену подтверждения (approval=false) сотового телефона пользователя (поле cell_phone объекта tns:Profile).

XML Schema

```
<xsd:element name="ApprovePhoneApplyCodeRequest">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element default="true" minOccurs="0" name="approval" type="xsd:boolean"/>
      <xsd:element minOccurs="0" name="unitId" type="xsd:long"/>
      <xsd:element minOccurs="0" name="profileId" type="xsd:long"/>
      <xsd:element name="confirmationCode" type="xsd:string"/>
    </xsd:sequence>
  </xsd:complexType>
```

```
</xsd:element>
```

Описание

approval

- **true.** Подтверждение сотового телефона.
- **false.** Отмена подтверждения сотового телефона.

Значение по умолчанию равно **true**.

Необязательный элемент.

Тип: boolean

Значение по умолчанию: true

unitId

ID пользователя в системе МОНЕТА.РУ. Если это поле не задано, то используется текущий пользователь.

Необязательный элемент.

Тип: long

profileId

Необязательный элемент.

Тип: long

confirmationCode

Код подтверждения, который пришел пользователю в тексте sms сообщения.

Тип: string

Исходящее сообщение: ApprovePhoneApplyCodeResponse

Ответ, который приходит на запрос ApprovePhoneApplyCodeRequest. В ответе нет никаких данных.

Если в процессе подтверждения (или отмены подтверждения) сотового телефона произошла ошибка, то возникнет Эксерption. Если Эксерption не возник - значит сотовый телефон гарантированно подтвержден (или подтверждение отменено).

XML Schema

```
<xsd:element name="ApprovePhoneApplyCodeResponse">
  <xsd:complexType/>
</xsd:element>
```

ApprovePhoneSendConfirmation Endpoint

Входящее сообщение: ApprovePhoneSendConfirmationRequest

Запрос на подтверждение (approval=true) или отмену подтверждения (approval=false) сотового телефона пользователя (поле cell_phone объекта tns:Profile).

В личном кабинете пользователя сотовый телефон должен быть указан обязательно.

Для подтверждения номера, телефон не должен быть подтвержден в личном кабинете. Для отмены подтверждения номера, телефон должен быть подтвержден в личном кабинете.

XML Schema

```
<xsd:element name="ApprovePhoneSendConfirmationRequest">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element default="true" minOccurs="0" name="approval" type="xsd:boolean"/>
      <xsd:element minOccurs="0" name="unitId" type="xsd:long"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

```

<xsd:element minOccurs="0" name="profileId" type="xsd:long"/>
<xsd:element minOccurs="0" name="text">
  <xsd:simpleType>
    <xsd:restriction base="xsd:string">
      <xsd:maxLength value="160"/>
    </xsd:restriction>
  </xsd:simpleType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>

```

Описание

approval

- **true.** Подтверждение сотового телефона.
- **false.** Отмена подтверждения сотового телефона.

Значение по умолчанию равно **true**.

Необязательный элемент.

Тип: boolean

Значение по умолчанию: true

unitId

ID пользователя в системе МОНЕТА.РУ. Если это поле не задано, то используется текущий пользователь.

Необязательный элемент.

Тип: long

profileId

Необязательный элемент.

Тип: long

text

Текст sms сообщения, которое отправляется пользователю. Текст должен содержать подстановку {CODE}, которая будет заменена настоящим кодом подтверждения.

Важно: Длина сообщения, если не содержит русских букв - 160 символов, если содержит русские буквы - 70 символов.

По умолчанию для подтверждения используется - Код подтверждения: {CODE}.

По умолчанию для отмены подтверждения используется - Код отмены подтверждения: {CODE}.

Необязательный элемент.

Тип: string

Максимальная длина: 160

Исходящее сообщение: ApprovePhoneSendConfirmationResponse

Ответ, который приходит на запрос ApprovePhoneSendConfirmationRequest.

XML Schema

```

<xsd:element name="ApprovePhoneSendConfirmationResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="phoneNumber" type="xsd:string"/>
    </xsd:sequence>
  </xsd:complexType>

```

```
</xsd:element>
```

Описание

phoneNumber

Номер телефона, на который было послано sms сообщение.

Тип: string

Асинхронные запросы

Async Endpoint

Входящее сообщение: AsyncRequest

XML Schema

```
<xsd:element name="AsyncRequest">
  <xsd:complexType>
    <xsd:choice>
      <xsd:element ref="tns:GetTurnoverListRequest"/>
      <xsd:element ref="tns:GetFinancialFlowsListRequest"/>
      <xsd:element ref="tns:SimplifiedIdentificationRequest"/>
      <xsd:element name="asyncId" type="xsd:long"/>
    </xsd:choice>
  </xsd:complexType>
</xsd:element>
```

Исходящее сообщение: AsyncResponse

XML Schema

```
<xsd:element name="AsyncResponse">
  <xsd:complexType>
    <xsd:choice>
      <xsd:element ref="tns:GetTurnoverListResponse"/>
      <xsd:element ref="tns:GetFinancialFlowsListResponse"/>
      <xsd:element ref="tns:SimplifiedIdentificationResponse"/>
    <xsd:sequence>
      <xsd:element name="asyncId" type="xsd:long"/>
      <xsd:element name="asyncStatus" type="tns:AsyncStatus"/>
      <xsd:element name="expirationDate" type="xsd:dateTime"/>
    </xsd:sequence>
  </xsd:choice>
</xsd:complexType>
</xsd:element>
```

Глава

5

Примеры использования

- *Простое пополнение и списание*
- *Оплата товаров и услуг*
- *Возврат средств*
- *Вывод средств*
- *Запросы в пакетном режиме*
- *История платежей*
- *Шаблоны операций*
- *Работа с профайлом пользователя*
- *Работа со счетами пользователя*
- *Работа с документами пользователя*
- *Работа с банковскими реквизитами*
- *Работа с отчетами*
- *Подтверждение сотового телефона*
- *SimplifiedIdentificationRequest (AsyncRequest) - упрощенная идентификация*

Простое пополнение и списание

Пополнение и списание средств оперирует со счетами МОНЕТА.РУ.

Главные требования:

- Счета плательщика и получателя зарегистрированы в системе МОНЕТА.РУ.
- На счете плательщика достаточно средств для проведения перевода.

Для счета плательщика должен быть указан платежный пароль для проведения финансовых операций по списанию средств.

Пользователь системы МОНЕТА.РУ может иметь несколько счетов в каждой из доступных валют. Все пользовательские счета ассоциированы с пользовательскими данными. Уникальным идентификатором, объединяющим все счета пользователя, является «unitid» в списке атрибутов типа [Profile](#). Сервис может идентифицировать пользователя по заданному счету при помощи вызова метода [FindProfileInfoByAccountId](#). Запрос пользовательской информации не является обязательным и может быть пропущен, если сервис уже запрашивал эти данные и сохранил их в своей системе, или сервис не зависит от уникальности представления пользователей.

Списание при помощи метода [Payment](#):

- «payer» и «paymentPassword» относятся к счету пользователя
- «payee» – счет сервиса

- «isPayerAmount» установлен в «false» для получения средств от пользователя
- «amount» – объем средств в валюте сервиса
- «clientTransaction» – уникальный идентификатор операции в системе сервиса

Пополнение при помощи метода *Payment*:

- «payer» и «paymentPassword» относятся к счету сервиса
- «payee» – счет пользователя
- «isPayerAmount» установлен в «true» для отправки средств пользователю
- «amount» – объем средств в валюте сервиса
- «clientTransaction» – уникальный идентификатор операции в системе сервиса

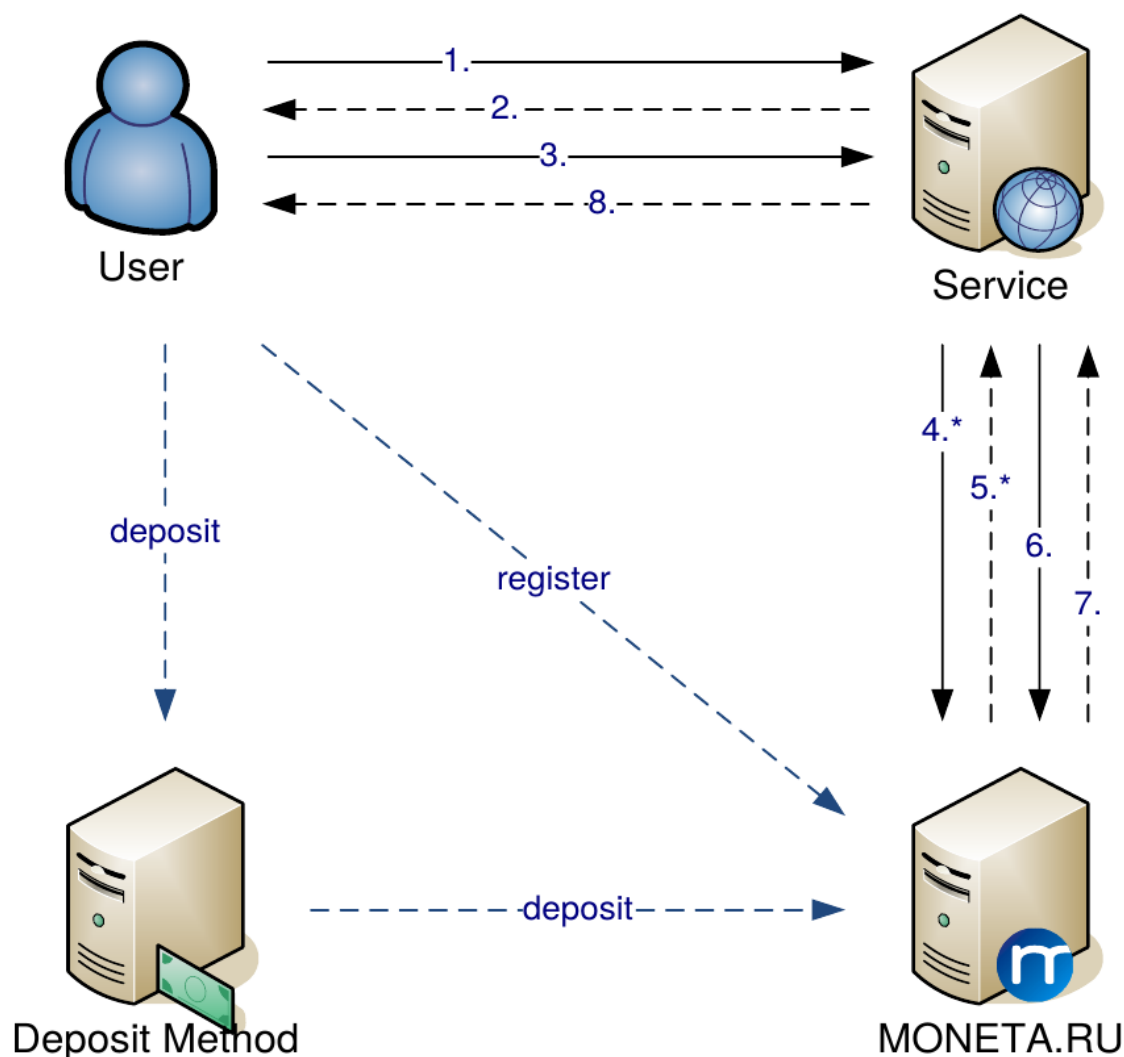


Рисунок 1. Общая диаграмма перевода средств

1. Пользователь запрашивает страницу пополнения/списания.
2. Сервис партнера подготавливает платежную форму.
3. Пользователь вводит номер счета, сумма и платежный пароль (пароль необходим только для списания).
4. Сервис партнера запрашивает информацию о возможности проведения перевода при помощи метода *VerifyPayment*.
5. МОНЕТА.РУ возвращает сервису информацию о возможности проведения перевода в структуре *VerifyPaymentResponse*.
6. Сервис партнера вызывает метод *Payment* для перевода средств.
7. МОНЕТА.РУ проводит операцию и возвращает результат обработки в структуре *PaymentResponse*.
8. Сервис партнера отображает результат и статус обработки операции.

Оплата товаров и услуг

Оплата товаров и услуг

Введение

Вы можете провести платеж одним из следующих методов: Payment или Transfer. Метод Transfer является устаревшим. В отличие от метода Transfer, метод Payment в качестве параметра может принимать номер операции, если Вы продолжаете платеж. Также метод Payment возвращает в ответе детали операции. Поэтому Вам не нужно вызывать метод GetOperationDetailsById после проведения каждой операции.

Перед тем как провести платеж, Вы можете дополнительно вызвать метод VerifyPayment или VerifyTransfer, которые проверяют параметры платежа.

Типы платежей

- Однофазный платеж.
- Двухфазный платеж.
- Push-платеж.

Однофазный платеж

Для проведения однофазного платежа выполните следующие шаги:

1. Рекомендуется (но необязательно) сначала выполнить запрос [VerifyPayment](#), чтобы проверить параметры платежа.

Следующий SOAP пример запроса VerifyPayment проверяет, можно ли провести платеж на 100 RUB со счета пользователя 12345678 в Moneta.Ru на счет магазина 87654321:

```
<VerifyPaymentRequest version="VERSION_2">
  <payer>12345678</payer>
  <payee>87654321</payee>
  <amount>100</amount>
  <isPayerAmount>true</isPayerAmount>
  <paymentPassword>*****</paymentPassword>
</VerifyPaymentRequest>
```

Элемент `<isPayerAmount>true</isPayerAmount>` в запросе означает, что сумма указана в валюте плательщика.

Важно: Если плательщик может получать платежный пароль в SMS, то установите в запросе версию равной VERSION_2. В этом случае, если плательщик использует SMS платежный пароль, то в ответе будет элемент `<paymentPasswordChallengeRequired>true</paymentPasswordChallengeRequired>`.

Следующий SOAP ответ показывает, что платеж можно провести:

```
<VerifyPaymentResponse>
  <isTransactionValid>true</isTransactionValid>
  <forecast>
    <payer>12345678</payer>
    <payerCurrency>RUB</payerCurrency>
    <payerAmount>100</payerAmount>
    <payee>87654321</payee>
    <payeeCurrency>RUB</payeeCurrency>
    <payeeAmount>100</payeeAmount>
    <payeeFee>0</payeeFee>
    <payerAlias>RUB</payerAlias>
    <payeeAlias>Test Store</payeeAlias>
  </forecast>
</VerifyPaymentResponse>
```

JSON запрос:

```
{
  "Envelope": {
    "Header": {
      "Security": {
        "UsernameToken": {
          "Username": "USERNAME",
          "Password": "PASSWORD"
        }
      }
    },
    "Body": {
      "VerifyPaymentRequest": {
        "version": "VERSION_2",
        "payer": "12345678",
        "payee": "87654321",
        "amount": 100,
        "isPayerAmount": true,
        "paymentPassword": "*****",
        "clientTransaction": "Merchant transaction Id"
      }
    }
  }
}
```

JSON ответ:

```
{
  "Envelope": {
    "Body": {
      "VerifyPaymentResponse": {
        "isTransactionValid": true,
        "forecast": {
          "payer": "12345678",
          "payerCurrency": "RUB",
          "payerAmount": 100,
          "payerFee": 0,
          "payee": "87654321",
          "payeeAmount": 100,
          "payeeCurrency": "RUB"
        },
        "operationInfo": {
          "attribute": [
            {
              "value": "true",
              "key": "paymentPasswordChallengeRequired"
            }
          ]
        }
      }
    }
  }
}
```

- Если плательщик получает платежный пароль в SMS, выполните метод [GetAccountPaymentPasswordChallenge](#) для получения плательщиком платежного пароля:

```
<GetAccountPaymentPasswordChallengeRequest>
  <accountId>12345678</accountId>
</GetAccountPaymentPasswordChallengeRequest>
```

JSON запрос:

```
{
  "Envelope": {
    "Header": {
      "Security": {
        "UsernameToken": {
          "Username": "USERNAME",
          "Password": "PASSWORD"
        }
      }
    },
    "Body": {
      "GetAccountPaymentPasswordChallengeRequest": {
        "accountId": "12345678"
      }
    }
  }
}
```



```

    "Body": {
      "GetAccountPaymentPasswordChallengeRequest": {
        "accountId": 12345678
      }
    }
  }
}

```

3. Сделайте вызов метода [Payment](#).

Следующий SOAP пример переводит 100 RUB со счета пользователя 12345678 на счет магазина 87654321:

```

<PaymentRequest>
  <payer>12345678</payer>
  <payee>87654321</payee>
  <amount>100</amount>
  <isPayerAmount>true</isPayerAmount>
  <paymentPassword>*****</paymentPassword>
  <clientTransaction>777777</clientTransaction>
</PaymentRequest>

```

Если операция проведена успешно, то SOAP ответ будет выглядеть так:

```

<PaymentResponse>
  <id>328268</id>
  <attribute>
    <key>category</key>
    <value>BUSINESS</value>
  </attribute>
  <attribute>
    <key>sourcecurrencycode</key>
    <value>RUB</value>
  </attribute>
  <attribute>
    <key>sourceamount</key>
    <value>100</value>
  </attribute>
  <attribute>
    <key>typeid</key>
    <value>10</value>
  </attribute>
  <attribute>
    <key>targetamount</key>
    <value>-100</value>
  </attribute>
  <attribute>
    <key>sourceamountfee</key>
    <value>0</value>
  </attribute>
  <attribute>
    <key>targetcurrencycode</key>
    <value>RUB</value>
  </attribute>
  <attribute>
    <key>statusid</key>
    <value>SUCCEED</value>
  </attribute>
  <attribute>
    <key>sourceamounttotal</key>
    <value>100</value>
  </attribute>
  <attribute>
    <key>modified</key>
    <value>2014-12-17T01:25:22.000+03:00</value>
  </attribute>
  <attribute>
    <key>clienttransaction</key>
    <value>777777</value>
  </attribute>
</attribute>

```

```

    <key>targetaccountid</key>
    <value>12345678</value>
  </attribute>
  <attribute>
    <key>sourceaccountid</key>
    <value>87654321</value>
  </attribute>
  <attribute>
    <key>isreversed</key>
    <value>true</value>
  </attribute>
</PaymentResponse>

```

JSON запрос:

```

{"Envelope": {
  "Header": {
    "Security": {
      "UsernameToken": {
        "Username": "USERNAME",
        "Password": "PASSWORD"
      }
    }
  },
  "Body": {
    "PaymentRequest": {
      "payer": "12345678",
      "payee": "87654321",
      "amount": 100,
      "isPayerAmount": true,
      "paymentPassword": "*****",
      "clientTransaction": "Merchant transaction id"
    }
  }
}}

```

JSON ответ:

```

{"Envelope": {
  "Body": {
    "PaymentResponse": {
      "id": 328268,
      "attribute": [
        ...
        {
          "value": "SUCCEED",
          "key": "statusid"
        }
        ...
      ]
    }
  }
}}

```

Push-платежи

Введение

В обычной ситуации оплата начинается на стороне получателя. Интернет-магазин создает заказ, его номер и сумму передает в платежную форму, при подтверждении оплаты информация о поступившем платеже, идентифицированным по номеру заказа, передается в интернет-магазин.

Push-платеж – операция по переводу денежных средств, инициированная на стороне плательщика. Например, в терминале пользователь выбирает кнопку интернет-магазина, вводит свой e-mail или телефон, зарегистрированный в интернет-магазине, и производит пополнение своего авансового счета. Для такого платежа пользователю не нужен номер заказа, достаточно знать свой логин в интернет-магазине.

Основная задача обработки таких платежей – однозначно идентифицировать конечного получателя. МОНЕТА.РУ предоставляет возможность принимать такие платежи с возможностью указать в реквизитах оплаты идентификатор конкретного получателя.

Для отправителей push-платежей

При работе через MONETA.MerchantAPI значение идентификатора конкретного получателя должно быть указано в дополнительном атрибуте «subscriberid».

В качестве значения этого параметра может быть передан уникальный символьный идентификатор конечного получателя платежа или подписчика услуги. Например, логин пользователя, его e-mail или телефон, зарегистрированный в интернет-магазине.

Для получателей push-платежей

Для решения задачи при взаимодействии через MONETA.Assistant предусмотрен параметр MNT_SUBSCRIBER_ID.

Этот параметр будет передаваться в проверочных запросах, в уведомлениях об оплате и будет участвовать в формировании подписи запросов для обеспечения проверки целостности передаваемых данных.

Поскольку push-платеж проводится по инициативе плательщика, уникальный идентификатор заказа/покупки, передаваемый в параметре MNT_TRANSACTION_ID, будет сгенерирован автоматически или может отсутствовать. Таким образом, автоматизированная система получателя должна быть готова к приёму несуществующего номера заказа MNT_TRANSACTION_ID в случае, если передан дополнительный параметр MNT_SUBSCRIBER_ID, однозначно идентифицирующий получателя.

Проведение push-платежа

Для проведения push-платежа выполните следующие действия:

1. Рекомендуется (но необязательно) сначала выполнить запрос [VerifyPayment](#), чтобы проверить параметры платежа.

Следующий SOAP пример запроса VerifyPayment проверяет, можно ли провести платеж на 100 RUB со счета 12345678 в Moneta.Ru на счет магазина 87654321, где идентификатор пользователя в магазине равен 007:

```
<VerifyPaymentRequest>
  <payer>12345678</payer>
  <payee>87654321</payee>
  <amount>100</amount>
  <isPayerAmount>true</isPayerAmount>
  <paymentPassword>*****</paymentPassword>
  <operationInfo>
    <attribute>
      <key>subscriberid</key>
      <value>007</value>
    </attribute>
  </operationInfo>
</VerifyPaymentRequest>
```

Элемент <isPayerAmount>true</isPayerAmount> в запросе означает, что сумма указана в валюте плательщика.

Следующий SOAP ответ показывает, что платеж можно провести:

```
<VerifyPaymentResponse>
  <isTransactionValid>true</isTransactionValid>
  <forecast>
    <payer>12345678</payer>
    <payerCurrency>RUB</payerCurrency>
    <payerAmount>100</payerAmount>
    <payee>87654321</payee>
    <payeeCurrency>RUB</payeeCurrency>
    <payeeAmount>100</payeeAmount>
```

```

    <payeeFee>0</payeeFee>
    <payerAlias>RUB</payerAlias>
    <payeeAlias>Test Store</payeeAlias>
  </forecast>
</VerifyPaymentResponse>

```

JSON запрос:

```

{
  "Envelope": {
    "Header": {
      "Security": {
        "UsernameToken": {
          "Username": "USERNAME",
          "Password": "PASSWORD"
        }
      }
    },
    "Body": {
      "VerifyPaymentRequest": {
        "version": "VERSION_2",
        "payer": "12345678",
        "payee": "87654321",
        "amount": 100,
        "isPayerAmount": true,
        "paymentPassword": "*****",
        "clientTransaction": "Merchant transaction Id",
        "operationInfo": {
          "attribute": [
            {
              "key": "subscriberid",
              "value": "007"
            }
          ]
        }
      }
    }
  }
}

```

JSON ответ:

```

{
  "Envelope": {
    "Body": {
      "VerifyPaymentResponse": {
        "isTransactionValid": true,
        "forecast": {
          "payer": "12345678",
          "payerCurrency": "RUB",
          "payerAmount": 100,
          "payerFee": 0,
          "payee": "87654321",
          "payeeAmount": 100,
          "payeeCurrency": "RUB"
        }
      }
    }
  }
}

```

2. Выполните метод [Payment](#).

Следующий SOAP пример переводит 100 RUB со счета 12345678 на счет магазина 87654321, где идентификатор пользователя в магазине равен 007:

```

<PaymentRequest>
  <payer>12345678</payer>
  <payee>87654321</payee>
  <amount>100</amount>
  <isPayerAmount>true</isPayerAmount>
  <paymentPassword>*****</paymentPassword>
  <operationInfo>
    <attribute>

```

```

    <key>subscriberid</key>
    <value>007</value>
  </attribute>
</operationInfo>
</PaymentRequest>

```

Если операция проведена успешно, то SOAP ответ будет выглядеть так:

```

<PaymentResponse>
  <id>333855</id>
  <attribute>
    <key>category</key>
    <value>BUSINESS</value>
  </attribute>
  <attribute>
    <key>sourcecurrencycode</key>
    <value>RUB</value>
  </attribute>
  <attribute>
    <key>sourceamount</key>
    <value>100</value>
  </attribute>
  <attribute>
    <key>typeid</key>
    <value>10</value>
  </attribute>
  <attribute>
    <key>targetamount</key>
    <value>-100</value>
  </attribute>
  <attribute>
    <key>subscriberid</key>
    <value>007</value>
  </attribute>
  <attribute>
    <key>sourceamountfee</key>
    <value>0</value>
  </attribute>
  <attribute>
    <key>targetcurrencycode</key>
    <value>RUB</value>
  </attribute>
  <attribute>
    <key>statusid</key>
    <value>SUCCEED</value>
  </attribute>
  <attribute>
    <key>sourceamounttotal</key>
    <value>100</value>
  </attribute>
  <attribute>
    <key>modified</key>
    <value>2015-01-31T17:15:46.000+03:00</value>
  </attribute>
  <attribute>
    <key>targetaccountid</key>
    <value>12345678</value>
  </attribute>
  <attribute>
    <key>sourceaccountid</key>
    <value>87654321</value>
  </attribute>
  <attribute>
    <key>isreversed</key>
    <value>true</value>
  </attribute>
</PaymentResponse>

```

JSON запрос:

```
{
  "Envelope": {
    "Header": {
      "Security": {
        "UsernameToken": {
          "Username": "USERNAME",
          "Password": "PASSWORD"
        }
      }
    },
    "Body": {
      "PaymentRequest": {
        "payer": "12345678",
        "payee": "87654321",
        "amount": 100,
        "isPayerAmount": true,
        "paymentPassword": "*****",
        "clientTransaction": "Merchant transaction id",
        "operationInfo": {
          "attribute": [
            {
              "key": "subscriberid",
              "value": "007"
            }
          ]
        }
      }
    }
  }
}
```

JSON ответ:

```
{
  "Envelope": {
    "Body": {
      "PaymentResponse": {
        "id": 328268,
        "attribute": [
          ...
          {
            "value": "SUCCEED",
            "key": "statusid"
          },
          {
            "key": "subscriberid",
            "value": "007"
          }
          ...
        ]
      }
    }
  }
}
```

Двухфазный платеж

Магазин может провести платеж в два этапа. Сначала магазин списывает средства со счета плательщика. Затем магазин зачисляет средства на счет получателя.

Первый этап

1. Рекомендуется (но необязательно) сначала выполнить запрос [VerifyPayment](#), чтобы проверить параметры платежа.

Следующий SOAP пример запроса VerifyPayment проверяет, можно ли провести платеж на 100 RUB со счета пользователя 12345678 в Moneta.Ru на счет магазина 87654321:

```
<VerifyPaymentRequest version="VERSION_2">
```

```

<payer>12345678</payer>
<payee>87654321</payee>
<amount>100</amount>
<isPayerAmount>true</isPayerAmount>
<paymentPassword>*****</paymentPassword>
</VerifyPaymentRequest>

```

Элемент `<isPayerAmount>true</isPayerAmount>` в запросе означает, что сумма указана в валюте плательщика.

Важно: Если плательщик может получать платежный пароль в SMS, то установите в запросе версию равной `VERSION_2`. В этом случае, если плательщик использует SMS платежный пароль, то в ответе будет элемент `<paymentPasswordChallengeRequired>true</paymentPasswordChallengeRequired>`.

Следующий SOAP ответ показывает, что платеж можно провести:

```

<VerifyPaymentResponse>
  <isTransactionValid>true</isTransactionValid>
  <forecast>
    <payer>12345678</payer>
    <payerCurrency>RUB</payerCurrency>
    <payerAmount>100</payerAmount>
    <payee>87654321</payee>
    <payeeCurrency>RUB</payeeCurrency>
    <payeeAmount>100</payeeAmount>
    <payeeFee>0</payeeFee>
    <payerAlias>RUB</payerAlias>
    <payeeAlias>Test Store</payeeAlias>
  </forecast>
</VerifyPaymentResponse>

```

JSON запрос:

```

{"Envelope": {
  "Header": {
    "Security": {
      "UsernameToken": {
        "Username": "USERNAME",
        "Password": "PASSWORD"
      }
    }
  },
  "Body": {
    "VerifyPaymentRequest": {
      "version": "VERSION_2",
      "payer": "12345678",
      "payee": "87654321",
      "amount": 100,
      "isPayerAmount": true,
      "paymentPassword": "*****",
      "clientTransaction": "Merchant transaction Id"
    }
  }
}}

```

JSON ответ:

```

{"Envelope": {
  "Body": {
    "VerifyPaymentResponse": {
      "isTransactionValid": true,
      "forecast": {
        "payer": "12345678",
        "payerCurrency": "RUB",
        "payerAmount": 100,
        "payerFee": 0,
        "payee": "87654321",
        "payeeAmount": 100,

```

```
{
  "payeeCurrency": "RUB",
},
"operationInfo": {
  "attribute": [
    {
      "value": "true",
      "key": "paymentPasswordChallengeRequired"
    }
  ]
}
}
```

2. Если плательщик получает платежный пароль в SMS, выполните метод [GetAccountPaymentPasswordChallenge](#) для получения плательщиком платежного пароля:

```
<GetAccountPaymentPasswordChallengeRequest>
  <accountId>12345678</accountId>
</GetAccountPaymentPasswordChallengeRequest>
```

JSON запрос:

```
{
  "Envelope": {
    "Header": {
      "Security": {
        "UsernameToken": {
          "Username": "USERNAME",
          "Password": "PASSWORD"
        }
      }
    }
  },
  "Body": {
    "GetAccountPaymentPasswordChallengeRequest": {
      "accountId": 12345678
    }
  }
}
```

3. Выполните метод *AuthoriseTransaction*, чтобы списать сумму со счета плательщика.

Следующий SOAP пример списывает 100 RUB со счета пользователя 12345678 на счет магазина 87654321:

```
<AuthoriseTransactionRequest>
  <payer>12345678</payer>
  <payee>87654321</payee>
  <amount>100</amount>
  <isPayerAmount>true</isPayerAmount>
  <paymentPassword>*****</paymentPassword>
  <clientTransaction>000002</clientTransaction>
  <description>Sample two-phase transaction</description>
</AuthoriseTransactionRequest>
```

Если списание прошло успешно, то SOAP ответ будет выглядеть так:

```
<AuthoriseTransactionResponse>
  <id>328805</id>
  <attribute>
    <key>category</key>
    <value>BUSINESS</value>
  </attribute>
  <attribute>
    <key>sourcecurrencycode</key>
    <value>RUB</value>
  </attribute>
  <attribute>
    <key>sourceamount</key>
    <value>100</value>
  </attribute>

```



```

<attribute>
  <key>description</key>
  <value>Sample two-phase transaction</value>
</attribute>
<attribute>
  <key>typeid</key>
  <value>10</value>
</attribute>
<attribute>
  <key>targetamount</key>
  <value>-100</value>
</attribute>
<attribute>
  <key>protectioncode</key>
  <value>18721</value>
</attribute>
<attribute>
  <key>targetcurrencycode</key>
  <value>RUB</value>
</attribute>
<attribute>
  <key>statusid</key>
  <value>INPROGRESS</value>
</attribute>
<attribute>
  <key>sourceamounttotal</key>
  <value>100</value>
</attribute>
<attribute>
  <key>modified</key>
  <value>2014-12-22T23:23:20.000+03:00</value>
</attribute>
<attribute>
  <key>clienttransaction</key>
  <value>000002</value>
</attribute>
<attribute>
  <key>targetaccountid</key>
  <value>12345678</value>
</attribute>
<attribute>
  <key>protectioncodeexpirationdate</key>
  <value>2014-12-23T23:23:19.000+03:00</value>
</attribute>
<attribute>
  <key>sourceaccountid</key>
  <value>87654321</value>
</attribute>
<attribute>
  <key>isreversed</key>
  <value>true</value>
</attribute>
</AuthoriseTransactionResponse>

```

Важно: По умолчанию списание средств действует в течении 24 часов. Если Вы не завершите операцию за это время, то операция отменится. Вы можете добавить элемент `protectioncodeexpirationdate` в Ваш запрос, чтобы переопределить время действия по умолчанию. Вы можете добавить элемент `protectioncode` в Ваш запрос, чтобы самим указать код подтверждения в методе `ConfirmTransaction`.

JSON запрос:

```

{"Envelope": {
  "Header": {
    "Security": {
      "UsernameToken": {
        "Username": "USERNAME",
        "Password": "PASSWORD"
      }
    }
  }
}

```

```

    },
    "Body": {
      "AuthoriseTransactionRequest": {
        "payer": "12345678",
        "payee": "87654321",
        "amount": 100,
        "isPayerAmount": true,
        "paymentPassword": "*****",
        "clientTransaction": "Merchant transaction Id",
        "description": "Sample two-phase transaction",
        "operationInfo": {
          "attribute": [
            {
              "key": "protectioncodeexpirationdate",
              "value": "29.09.2015 15:25:07"
            },
            {
              "key": "protectioncode",
              "value": "12345"
            }
          ]
        }
      }
    }
  }
}

```

JSON ответ:

```

{
  "Envelope": {
    "Body": {
      "AuthoriseTransactionResponse": {
        "id": 328805,
        "attribute": [
          {
            "value": "INPROGRESS",
            "key": "statusid"
          },
          {
            "value": "12345",
            "key": "protectioncode"
          },
          {
            "value": "2015-09-29T15:25:07.000+03:00",
            "key": "protectioncodeexpirationdate"
          }
        ]
      }
    }
  }
}

```

Второй этап

1. Выполните метод [ConfirmTransaction](#), чтобы зачислить средства на счет получателя.

Следующий SOAP пример показывает, как зачислить средства на счет получателя по операции 328805:

```

<ConfirmTransactionRequest>
  <transactionId>328805</transactionId>
  <protectionCode>18721</protectionCode>
</ConfirmTransactionRequest>

```

Если операция проведена успешно, то SOAP ответ будет выглядеть так:

```

<ConfirmTransactionResponse>
  <id>328805</id>

```

```

<attribute>
  <key>category</key>
  <value>BUSINESS</value>
</attribute>
<attribute>
  <key>sourcecurrencycode</key>
  <value>RUB</value>
</attribute>
<attribute>
  <key>sourceamount</key>
  <value>100</value>
</attribute>
<attribute>
  <key>description</key>
  <value>Sample two-phase transaction</value>
</attribute>
<attribute>
  <key>typeid</key>
  <value>10</value>
</attribute>
<attribute>
  <key>targetamount</key>
  <value>-100</value>
</attribute>
<attribute>
  <key>protectioncode</key>
  <value>*****</value>
</attribute>
<attribute>
  <key>sourceamountfee</key>
  <value>0</value>
</attribute>
<attribute>
  <key>targetcurrencycode</key>
  <value>RUB</value>
</attribute>
<attribute>
  <key>statusid</key>
  <value>SUCCEED</value>
</attribute>
<attribute>
  <key>sourceamounttotal</key>
  <value>100</value>
</attribute>
<attribute>
  <key>modified</key>
  <value>2014-12-23T00:44:37.000+03:00</value>
</attribute>
<attribute>
  <key>clienttransaction</key>
  <value>000002</value>
</attribute>
<attribute>
  <key>targetaccountid</key>
  <value>12345678</value>
</attribute>
<attribute>
  <key>protectioncodeexpirationdate</key>
  <value>2014-12-23T23:23:19.000+03:00</value>
</attribute>
<attribute>
  <key>sourceaccountid</key>
  <value>87654321</value>
</attribute>
<attribute>
  <key>isreversed</key>
  <value>true</value>
</attribute>
</ConfirmTransactionResponse>

```

JSON запрос:

```
{
  "Envelope": {
    "Header": {
      "Security": {
        "UsernameToken": {
          "Username": "USERNAME",
          "Password": "PASSWORD"
        }
      }
    },
    "Body": {
      "ConfirmTransactionRequest": {
        "transactionId": "328805",
        "protectionCode": "12345"
      }
    }
  }
}
```

JSON ответ:

```
{
  "Envelope": {
    "Body": {
      "ConfirmTransactionResponse": {
        "id": 328805,
        "attribute": [
          ...
          {
            "value": "SUCCEED",
            "key": "statusid"
          }
          ...
        ]
      }
    }
  }
}
```

Выставление счета (инвойс)

Вы можете выставить счет для оплаты и передать его пользователю. Пользователь может использовать различные методы для оплаты счета.

Создание инвойса

1. Выполните метод *Invoice*.

Следующий SOAP пример показывает создание инвойса на 100 RUB, который должен быть оплачен на счет магазина 87654321:

```
<InvoiceRequest>
  <payee>87654321</payee>
  <amount>100</amount>
  <clientTransaction>00001</clientTransaction>
  <description>Sample invoice</description>
</InvoiceRequest>
```

Пример SOAP ответа, если инвойс создан успешно:

```
<InvoiceResponse>
  <status>CREATED</status>
  <dateTime>2014-12-18T23:36:27.000+03:00</dateTime>
  <transaction>328498</transaction>
  <clientTransaction>000001</clientTransaction>
</InvoiceResponse>
```

JSON запрос:

```
{
  "Envelope": {
    "Header": {
      "Security": {
        "UsernameToken": {
          "Username": "USERNAME",
          "Password": "PASSWORD"
        }
      }
    },
    "Body": {
      "InvoiceRequest": {
        "payee": "87654321",
        "amount": 100,
        "clientTransaction": "Merchant transaction Id"
      }
    }
  }
}
```

JSON ответ:

```
{
  "Envelope": {
    "Body": {
      "InvoiceResponse": {
        "transaction": 328498,
        "dateTime": "2015-06-17T09:39:41.000+0300",
        "status": "CREATED",
        "clientTransaction": "Merchant transaction Id"
      }
    }
  }
}
```

- Получите в ответе элемент transaction и передайте его пользователю. Этот номер операции необходим для оплаты инвойса.

Подсказка: Вы также можете указать номер телефона или e-mail в Invoice запросе. В этом случае MONETA.MerchantAPI автоматически пошлет уведомление пользователю.

Оплата инвойса

Пользователь может оплатить инвойс, например, через терминал.

- Рекомендуется (но необязательно) сначала выполнить запрос [VerifyPayment](#), чтобы проверить параметры платежа.

Следующий SOAP пример проверяет существует ли инвойс 0328498 и можно ли его оплатить со счета 12345678:

```
<VerifyPaymentRequest>
  <payer>12345678</payer>
  <payee>0328498</payee>
  <paymentPassword>*****</paymentPassword>
  <clientTransaction>000001</clientTransaction>
</VerifyPaymentRequest>
```

Важно: Для того чтобы указать, что будет проведена оплата инвойса, следует подставить 0 перед номером операции и передать это значение в элементе payee запроса VerifyPayment.

Следующий SOAP ответ показывает, что параметры платежа указаны верно:

```
<VerifyPaymentResponse>
  <isTransactionValid>true</isTransactionValid>
  <forecast>
    <payer>12345678</payer>
    <payerCurrency>RUB</payerCurrency>
    <payerAmount>100</payerAmount>
    <payerFee>0</payerFee>
```

```

<payee>87654321</payee>
<payeeCurrency>RUB</payeeCurrency>
<payeeAmount>100</payeeAmount>
<payerAlias>RUB</payerAlias>
<payeeAlias>Test Store</payeeAlias>
</forecast>
<transactionId>328498</transactionId>
<operationStatus>CREATED</operationStatus>
</VerifyPaymentResponse>

```

JSON запрос:

```

{"Envelope": {
  "Header": {
    "Security": {
      "UsernameToken": {
        "Username": "USERNAME",
        "Password": "PASSWORD"
      }
    }
  },
  "Body": {
    "VerifyPaymentRequest": {
      "version": "VERSION_2",
      "payer": "12345678",
      "payee": "0328498",
      "paymentPassword": "*****",
      "clientTransaction": "Merchant transaction Id"
    }
  }
}}

```

JSON ответ:

```

{"Envelope": {
  "Body": {
    "VerifyPaymentResponse": {
      "isTransactionValid": true,
      "forecast": {
        "payer": "12345678",
        "payerCurrency": "RUB",
        "payerAmount": 100,
        "payerFee": 0,
        "payee": "87654321",
        "payeeAmount": 100,
        "payeeCurrency": "RUB"
      },
      "transactionId": "328498",
      "operationStatus": "CREATED"
    }
  }
}}

```

2. Выполните запрос *Payment*.

Важно: Инвойс не может быть оплачен с того же счета магазина на который создан инвойс.

Следующий SOAP пример показывает вызов метода *Payment* для оплаты инвойса 328498:

```

<PaymentRequest>
  <payer>12345678</payer>
  <payee>0328498</payee>
  <paymentPassword>*****</paymentPassword>
</PaymentRequest>

```

Важно: Для того чтобы указать, что будет проведена оплата инвойса, следует подставить 0 перед номером операции и передать это значение в элементе *payee* запроса *Payment*.

Следующий SOAP пример показывает ответ, если инвойс успешно оплачен:

```
<PaymentResponse xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd">
  <id>328498</id>
  <attribute>
    <key>category</key>
    <value>BUSINESS</value>
  </attribute>
  <attribute>
    <key>isinvoice</key>
    <value>1</value>
  </attribute>
  <attribute>
    <key>sourcecurrencycode</key>
    <value>RUB</value>
  </attribute>
  <attribute>
    <key>sourceamount</key>
    <value>-100</value>
  </attribute>
  <attribute>
    <key>description</key>
    <value>Sample invoice</value>
  </attribute>
  <attribute>
    <key>typeid</key>
    <value>2</value>
  </attribute>
  <attribute>
    <key>targetamount</key>
    <value>100</value>
  </attribute>
  <attribute>
    <key>targetcurrencycode</key>
    <value>RUB</value>
  </attribute>
  <attribute>
    <key>statusid</key>
    <value>SUCCEED</value>
  </attribute>
  <attribute>
    <key>sourceamounttotal</key>
    <value>-100</value>
  </attribute>
  <attribute>
    <key>modified</key>
    <value>2014-12-21T01:39:42.000+03:00</value>
  </attribute>
  <attribute>
    <key>targetaccountid</key>
    <value>87654321</value>
  </attribute>
  <attribute>
    <key>targettransaction</key>
    <value>000001</value>
  </attribute>
  <attribute>
    <key>sourceaccountid</key>
    <value>12345678</value>
  </attribute>
</PaymentResponse>
```

JSON запрос:

```
{ "Envelope": {
  "Header": {
    "Security": {
      "UsernameToken": {
        "Username": "USERNAME",
```

```

        "Password": "PASSWORD"
      }
    },
    "Body": {
      "PaymentRequest": {
        "payer": "12345678",
        "payee": "0328498",
        "paymentPassword": "*****"
      }
    }
  }
}

```

JSON ответ:

```

{"Envelope": {
  "Body": {
    "PaymentResponse": {
      "id": 328498,
      "attribute": [
        ...
        {
          "value": "SUCCEED",
          "key": "statusid"
        }
        ...
      ]
    }
  }
}
}

```

Важно: После обработки платежа Moneta.Ru посылает уведомление об оплате на Pay URL, установленный на счете магазина. Pay Url должен вернуть ответ в текстовом формате. Варианты ответа:

- SUCCESS. Магазин принимает оплату. Операция переходит в статус "Выполнено".
- FAIL. Магазин не принял платеж. Операция переходит в статус "Уведомление не доставлено". Moneta.Ru делает повторные попытки отослать уведомление об оплате.

Более подробную информацию смотрите в документации *MONETA.Assistant*.

Оплата банковской картой

Оплата банковской картой

Для оплаты банковской картой используется метод [Payment](#).

В запросе следует передать следующие поля:

- CARDNUMBER
- CARDEXPIRATION
- CARDCVV2

Пример SOAP запроса:

```

<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
    <wsse:Security
      xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-secext-1.0.xsd"
      xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-utility-1.0.xsd"
      soap:mustUnderstand="1">
      <wsse:UsernameToken wsu:Id="UsernameToken-42">
        <wsse:Username>USERNAME</wsse:Username>
        <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/
oasis-200401-wss-username-token-profile-1.0#PasswordText">PASSWORD</wsse:Password>
      </wsse:UsernameToken>
    </wsse:Security>
  </SOAP-ENV:Header>
  <body>
    <PaymentRequest>
      <payer>12345678</payer>
      <payee>0328498</payee>
      <paymentPassword>*****</paymentPassword>
    </PaymentRequest>
  </body>
</soap:Envelope>

```



```

    </wsse:Security>
  </SOAP-ENV:Header>
  <soap:Body>
    <PaymentRequest xmlns="http://www.moneta.ru/schemas/messages.xsd">
      <payer>159</payer>
      <payee>Merchant's account</payee>
      <amount>Amount</amount>
      <isPayerAmount>false</isPayerAmount>
      <clientTransaction>Merchant transaction ID</clientTransaction>
      <operationInfo>
        <attribute>
          <key>CARDNUMBER</key>
          <value>1111111111111111</value>
        </attribute>
        <attribute>
          <key>CARDEXPIRATION</key>
          <value>01/2001</value>
        </attribute>
        <attribute>
          <key>CARDCVV2</key>
          <value>012</value>
        </attribute>
      </operationInfo>
    </PaymentRequest>
  </soap:Body>
</soap:Envelope>

```

JSON запрос:

```

{"Envelope": {
  "Header": {
    "Security": {
      "UsernameToken": {
        "Username": "USERNAME",
        "Password": "PASSWORD"
      }
    }
  },
  "Body": {
    "PaymentRequest": {
      "payer": "159",
      "payee": "Merchant's account",
      "amount": "Amount",
      "isPayerAmount": false,
      "clientTransaction": "Merchant transaction id",
      "operationInfo": {
        "attribute": [
          {
            "key": "CARDNUMBER",
            "value": "1111111111111111"
          },
          {
            "key": "CARDEXPIRATION",
            "value": "01/2001"
          },
          {
            "key": "CARDCVV2",
            "value": "012"
          }
        ]
      }
    }
  }
}
}
}
}

```

Важно: В ответе придет статус операции statusid=CREATED.

Если карта без поддержки технологии 3DS, то операция будет проведена сразу. Вам следует только дождаться уведомления об оплате.

Если на карте включен 3DS, то в ответе придет ссылка, куда следует перенаправить пользователя за вводом 3DS подтверждения:

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <ns2:PaymentResponse xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd">
      <ns2:id>Moneta_transaction_id</ns2:id>
      ...
      <ns2:attribute>
        <ns2:key>auth3dsurl</ns2:key>
        <ns2:value>https://www.moneta.ru/process3ds.htm?
operationId=Moneta_transaction_id&clientTransaction=Merchant_transaction_ID</
ns2:value>
      </ns2:attribute>
      <ns2:attribute>
        <ns2:key>statusid</ns2:key>
        <ns2:value>CREATED</ns2:value>
      </ns2:attribute>
      ...
    </ns2:PaymentResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

JSON ответ:

```
{ "Envelope": {
  "Body": {
    "PaymentResponse": {
      "id": "Moneta_transaction_id",
      "attribute": [
        ...
        {
          "value": "https://www.moneta.ru/process3ds.htm?
operationId=Moneta_transaction_id&clientTransaction=Merchant_transaction_ID",
          "key": "auth3dsurl"
        },
        {
          "value": "CREATED",
          "key": "statusid"
        }
        ...
      ]
    }
  }
}}
```

После этого Вам следует дождаться уведомления об оплате.

Подсказка: Вы также можете проверять статус операции методом [GetOperationDetailsById](#) или ждать запроса по факту зачисления, который настраивается в счете/прототипе в «Действиях при зачислении/списании».

Запрос на рекуррентные платежи без ввода карточных данных

Важно: Если Вы хотите проводить рекуррентные платежи, то следует написать письмо об этом в коммерческий отдел, так как требуется дополнительная настройка в банке-эквайере.

Выполните метод [Invoice](#) с параметром PAYMENTTOKEN:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
    <wsse:Security
      xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-secext-1.0.xsd"
      xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-utility-1.0.xsd"
      soap:mustUnderstand="1">
      <wsse:UsernameToken wsu:Id="UsernameToken-42">
```

```

        <wsse:Username>USERNAME</wsse:Username>
        <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/
oasis-200401-wss-username-token-profile-1.0#PasswordText">PASSWORD</wsse:Password>
    </wsse:UsernameToken>
</wsse:Security>
</SOAP-ENV:Header>
<soap:Body>
    <InvoiceRequest xmlns="http://www.moneta.ru/schemas/messages.xsd">
        <payer>159</payer>
        <payee>Merchant's account</payee>
        <amount>Amount</amount>
        <clientTransaction>Merchant transaction ID</clientTransaction>
        <operationInfo>
            <attribute>
                <key>PAYMENTTOKEN</key>
                <value>request</value>
            </attribute>
        </operationInfo>
    </InvoiceRequest>
</soap:Body>
</soap:Envelope>

```

JSON запрос:

```

{"Envelope": {
  "Header": {
    "Security": {
      "UsernameToken": {
        "Username": "USERNAME",
        "Password": "PASSWORD"
      }
    }
  },
  "Body": {
    "InvoiceRequest": {
      "payer": "159",
      "payee": "Merchant's account",
      "amount": "Amount",
      "clientTransaction": "Merchant transaction ID",
      "operationInfo": {
        "attribute": [
          {
            "key": "PAYMENTTOKEN",
            "value": "request"
          }
        ]
      }
    }
  }
}
}
}

```

Подсказка: Атрибут PAYMENTTOKEN=request означает, что если платеж пройдет успешно, то мы должны сформировать токен для рекуррентных платежей и сложить его в этот же атрибут PAYMENTTOKEN. После проведения операции сформированный PAYMENTTOKEN придет в уведомлении об оплате. Либо его можно получить вызвав метод [GetOperationDetailsById](#). Затем этот PAYMENTTOKEN передаётся в методе Payment вместо карточных данных.

В результате вызова метода [Invoice](#) мы возвращаем transactionId – номер операции, который нужно использовать при переходе на платежную форму Moneta.Assistant в виде: [https://www.payanyway.ru/assistant.htm?operationId=\[полученный_номер_операции\]&paymentSystem.unitId=card&paymentSystem.limitIds=card&followup=true](https://www.payanyway.ru/assistant.htm?operationId=[полученный_номер_операции]&paymentSystem.unitId=card&paymentSystem.limitIds=card&followup=true)

Запрос на выставление счета с последующей авторизацией без подтверждения списания

Если Вам требуется, чтобы в результате оплаты деньги были не списаны с карты, а только заблокированы, то сначала следует вызвать метод [Invoice](#) с параметром AUTHORIZEONLY:

```

<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">

```

```

<SOAP-ENV:Header xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <wsse:Security
    xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-secext-1.0.xsd"
    xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-utility-1.0.xsd"
    soap:mustUnderstand="1">
    <wsse:UsernameToken wsu:Id="UsernameToken-42">
      <wsse:Username>USERNAME</wsse:Username>
      <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/
oasis-200401-wss-username-token-profile-1.0#PasswordText">PASSWORD</wsse:Password>
    </wsse:UsernameToken>
    </wsse:Security>
  </SOAP-ENV:Header>
  <soap:Body>
    <InvoiceRequest xmlns="http://www.moneta.ru/schemas/messages.xsd">
      <payer>159</payer>
      <payee>Merchant's account</payee>
      <amount>Amount</amount>
      <clientTransaction>Merchant transaction ID</clientTransaction>
      <operationInfo>
        <attribute>
          <key>AUTHORIZEONLY</key>
          <value>1</value>
        </attribute>
      </operationInfo>
    </InvoiceRequest>
  </soap:Body>
</soap:Envelope>

```

JSON запрос:

```

{"Envelope": {
  "Header": {
    "Security": {
      "UsernameToken": {
        "Username": "USERNAME",
        "Password": "PASSWORD"
      }
    }
  },
  "Body": {
    "InvoiceRequest": {
      "payer": "159",
      "payee": "Merchant's account",
      "amount": "Amount",
      "clientTransaction": "Merchant transaction ID",
      "operationInfo": {
        "attribute": [
          {
            "key": "AUTHORIZEONLY",
            "value": "1"
          }
        ]
      }
    }
  }
}
}
}

```

В результате вызова метода *Invoice* мы возвращаем transactionId – номер операции, который нужно использовать при переходе на платежную форму Moneta.Assistant в виде: [https://www.payanyway.ru/assistant.htm?operationId=\[полученный_номер_операции\]&paymentSystem.unitId=card&paymentSystem.limitIds=card&followup=true](https://www.payanyway.ru/assistant.htm?operationId=[полученный_номер_операции]&paymentSystem.unitId=card&paymentSystem.limitIds=card&followup=true)

Возврат средств

Примеры запросов на возврат средств

SOAP запрос:

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header>
    <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd" SOAP-ENV:mustUnderstand="1">
      <wsse:UsernameToken xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd" wsu:Id="XWSSGID-1453884616001-1949076214"
        xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd">
        <wsse:Username>USERNAME</wsse:Username>
        <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-username-token-profile-1.0#PasswordText">PASSWORD</wsse:Password>
      </wsse:UsernameToken>
    </wsse:Security>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <ns2:RefundRequest xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd">
      <ns2:transactionId>102314549</ns2:transactionId>
      <ns2:amount>2.16</ns2:amount>
      <ns2:paymentPassword>*****</ns2:paymentPassword>
      <ns2:clientTransaction>CLIENT_TRANSACTION</ns2:clientTransaction>
      <ns2:description>DESCRIPTION</ns2:description>
      <ns2:operationInfo>
        <ns2:attribute>
          <ns2:key>CUSTOMFIELD:comment</ns2:key>
          <ns2:value>***</ns2:value>
        </ns2:attribute>
      </ns2:operationInfo>
    </ns2:RefundRequest>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Минимальный SOAP запрос:

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header>
    <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd" SOAP-ENV:mustUnderstand="1">
      <wsse:UsernameToken xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd" wsu:Id="XWSSGID-1453884616001-1949076214"
        xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd">
        <wsse:Username>USERNAME</wsse:Username>
        <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-username-token-profile-1.0#PasswordText">PASSWORD</wsse:Password>
      </wsse:UsernameToken>
    </wsse:Security>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <ns2:RefundRequest xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd">
      <ns2:transactionId>102314549</ns2:transactionId>
      <ns2:paymentPassword>*****</ns2:paymentPassword>
    </ns2:RefundRequest>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

SOAP ответ:

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <ns2:RefundResponse xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd">
```

```

        <ns2:id>102314550</ns2:id>
        ...
        <ns2:attribute>
            <ns2:key>parentid</ns2:key>
            <ns2:value>102314549</ns2:value>
        </ns2:attribute>
        <ns2:attribute>
            <ns2:key>statusid</ns2:key>
            <ns2:value>SUCCEED</ns2:value>
        </ns2:attribute>
        <ns2:attribute>
            <ns2:key>isrefund</ns2:key>
            <ns2:value>1</ns2:value>
        </ns2:attribute>
        ...
    </ns2:RefundResponse>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

JSON запрос:

```

{
  "Envelope": {
    "Body": {
      "RefundRequest": {
        "amount": 2.16,
        "paymentPassword": "*****",
        "transactionId": 102314549,
        "description": "DESCRIPTION",
        "clientTransaction": "CLIENT_TRANSACTION",
        "operationInfo": {
          "attribute": [
            {
              "value": "****",
              "key": "CUSTOMFIELD:comment"
            }
          ]
        }
      }
    },
    "Header": {
      "Security": {
        "UsernameToken": {
          "Password": "PASSWORD",
          "Username": "USERNAME"
        }
      }
    }
  }
}

```

Минимальный JSON запрос:

```

{
  "Envelope": {
    "Body": {
      "RefundRequest": {
        "paymentPassword": "*****",
        "transactionId": 102314549
      }
    },
    "Header": {
      "Security": {
        "UsernameToken": {
          "Password": "PASSWORD",
          "Username": "USERNAME"
        }
      }
    }
  }
}

```

```
}
```

JSON ответ:

```
{
  "Envelope": {
    "Body": {
      "RefundResponse": {
        "id": 102314550,
        "attribute": [
          ...
          {
            "value": "102314549",
            "key": "parentid"
          },
          {
            "value": "SUCCEED",
            "key": "statusid"
          },
          {
            "value": "1",
            "key": "isrefund"
          }
          ...
        ]
      }
    }
  }
}
```

Вывод средств

Введение

Метод Payment можно использовать для вывода средств в различные платежные системы.

Для выбора платежной системы в методе Payment требуется указать нужный элемент payee.

В методе Payment передаются параметры необходимые для вывода средств, которые зависят от выбранной платежной системы. Параметры передаются в элементе operationInfo.

Для определения элемента payee и параметров платежной системы смотрите дополнительную информацию:

- [Дополнительные параметры метода Payment](#)
- [Получение параметров для вывода средств](#)
- [Примеры вывода средств](#)

Вывод средств со счета в Moneta.Ru

Для вывода средств со счета Moneta.Ru выполните следующие шаги:

1. Рекомендуется (но необязательно) сначала выполнить запрос VerifyPayment, чтобы проверить параметры платежа.

Смотрите подробнее о [VerifyPayment Endpoint](#).

Следующий SOAP ответ показывает, что платеж можно провести:

```
<VerifyPaymentResponse>
  <isTransactionValid>true</isTransactionValid>
  <forecast>
    <payer>12*****78</payer>
    <payerCurrency>RUB</payerCurrency>
    <payerAmount>1.23</payerAmount>
    <payerFee>0</payerFee>
    <payee>9000</payee>
    <payeeCurrency>RUB</payeeCurrency>
```

```

    <payeeAmount>1.23</payeeAmount>
    <payeeFee>0</payeeFee>
    <payeeAlias>Municipal Enterprise of Water Supply and Waste Water Treatment
(Yoshkar-Ola)</payeeAlias>
  </forecast>
</VerifyPaymentResponse>

```

JSON ответ:

```

{"Envelope": {
  "Body": {
    "VerifyPaymentResponse": {
      "isTransactionValid": true,
      "forecast": {
        "payer": "12*****78",
        "payerCurrency": "RUB",
        "payerAmount": 1.23,
        "payerFee": 0,
        "payee": "9000",
        "payeeAmount": 1.23,
        "payeeCurrency": "RUB",
        "payeeAlias": "Municipal Enterprise of Water Supply and Waste Water
Treatment (Yoshkar-Ola)"
      }
    }
  }
}}

```

2. Если плательщик получает платежный пароль в SMS, выполните метод `GetAccountPaymentPasswordChallenge` для получения плательщиком платежного пароля.

Смотрите подробнее о [GetAccountPaymentPasswordChallenge Endpoint](#)

3. Выполните метод `Payment`.

Смотрите подробнее о [Payment Endpoint](#)

Если операция проведена успешно, то SOAP ответ будет выглядеть так:

```

<PaymentResponse>
  <id>6*****</id>
  <attribute>
    <key>category</key>
    <value>BUSINESS</value>
  </attribute>
  <attribute>
    <key>sourcecurrencycode</key>
    <value>RUB</value>
  </attribute>
  <attribute>
    <key>sourceamount</key>
    <value>-103</value>
  </attribute>
  <attribute>
    <key>description</key>
    <value>Beeline</value>
  </attribute>
  <attribute>
    <key>typeid</key>
    <value>2</value>
  </attribute>
  <attribute>
    <key>targetamount</key>
    <value>100</value>
  </attribute>
  <attribute>
    <key>sourceamountcompensation</key>
    <value>3</value>
  </attribute>
  <attribute>
    <key>sourceamountfee</key>

```



```

        <value>0</value>
    </attribute>
    <attribute>
        <key>targetcurrencycode</key>
        <value>RUB</value>
    </attribute>
    <attribute>
        <key>statusid</key>
        <value>SUCCEED</value>
    </attribute>
    <attribute>
        <key>sourceamounttotal</key>
        <value>-103</value>
    </attribute>
    <attribute>
        <key>customfield:100</key>
        <value>9*****4</value>
    </attribute>
    <attribute>
        <key>modified</key>
        <value>2015-01-26T15:00:00.000+03:00</value>
    </attribute>
    <attribute>
        <key>targetalias</key>
        <value>Mobile Communication</value>
    </attribute>
    <attribute>
        <key>targetaccountid</key>
        <value>1144</value>
    </attribute>
    <attribute>
        <key>targettransaction</key>
        <value>4*****0</value>
    </attribute>
    <attribute>
        <key>subproviderid</key>
        <value>2</value>
    </attribute>
    <attribute>
        <key>sourceaccountid</key>
        <value>8*****7</value>
    </attribute>
</PaymentResponse>

```

JSON ответ:

```

{"Envelope": {
  "Body": {
    "PaymentResponse": {
      "id": "6*****",
      "attribute": [
        ...
        {
          "value": "SUCCEED",
          "key": "statusid"
        }
        ...
      ]
    }
  }
}}

```

Дополнительные параметры метода Payment

Следующая таблица показывает значения элемента `payee` и возможные значения элемента `operationInfo`, которые нужно передать для вывода средств со счета Moneta.Ru в различные платежные системы:

Платежная система	Получатель (payee)	Значения в OperationInfo
Яндекс.Деньги	13	<ul style="list-style-type: none"> • YANDEXACCOUNT Номер кошелька в системе Яндекс.Деньги.
Vkontakte	139	<ul style="list-style-type: none"> • VKONTAKTEID E-mail или имя пользователя ВКонтакте.
WebMoney	2, 3 или 4	<ul style="list-style-type: none"> • WEBMONEYWMID Идентификатор пользователя (WMID). • WEBMONEYPURSE Номер кошелька WMR (если payee = 2). Номер кошелька WMZ (если payee = 3). Номер кошелька WME (если payee = 4).
CyberPlat	1144	<ul style="list-style-type: none"> • CYBERPLATPROVIDERID Значение CYBERPLATPROVIDERID. • CUSTOMFIELD:100
Банковский перевод	5	<ul style="list-style-type: none"> • WIREUSERNAME Наименование получателя. • WIREUSERINN ИНН получателя. • WIREFPAYMENTPURPOSE Назначение платежа. • WIREBANKACCOUNT Номер расчетного счета. • WIREBANKBIK БИК банка. • WIREBANKNAME Наименование банка. • WIREBANKKS Корр. счет банка. • WIREKPP КПП. • WIREKVK КБК. • WIREOKTMO ОКТМО. • WIREDOCINDEX УИН (индекс документа).
Банковские карты (через Банк Русский Стандарт)	275	<ul style="list-style-type: none"> • PAYERCOUNTRY Код страны. Возможные значения: <ul style="list-style-type: none"> • RUS. Перевод средств на карту, выпущенную российским банком. • PAYEECARDNUMBER

Платежная система	Получатель	Значения в OperationInfo (payee)
Банковские карты (через Альфа-Банк)	279	<p>Полный номер карты.</p> <ul style="list-style-type: none"> • PAYEECARDNUMBER <p>Полный номер карты.</p> <ul style="list-style-type: none"> • CARDEXPIRATION <p>Срок действия карты в формате: ММ/YYYY</p>
Деньги@Mail.ru	268	<ul style="list-style-type: none"> • CUSTOMFIELD:RECIPIENT <p>Е-mail или 16-ти значный номер пользователя в Деньги@Mail.ru.</p> <ul style="list-style-type: none"> • CUSTOMFIELD:PAYMENT_PURPOSE <p>Назначение платежа.</p>
QIWI	255	<ul style="list-style-type: none"> • EXTERNALACCOUNTID <p>QIWI account.</p>

Получение параметров для вывода средств

Параметры для вывода средств в методе Payment зависят от платежной системы, куда выводятся средства.

1. Сделайте вывод небольшой суммы со счета в Moneta.Ru в нужную платежную систему через интерфейс пользователя для получения номера операции.
2. Используйте метод GetOperationDetailsById для получения деталей операции.

Подсказка: Вы можете использовать cURL для вызова метода GetOperationDetailsById. Смотрите [SOAP запрос с помощью CURL](#) или [JSON запрос с помощью CURL](#).

3. В ответе получите элемент OperationInfo. Этот элемент содержит все параметры необходимые для вывода средств через MONETA.MerchantAPI.

Теперь Вы можете отредактировать элемент OperationInfo и использовать его в методе Payment для вывода средств.

Примеры вывода средств

Примеры использования метода Payment для вывода средств на банковские реквизиты и другие платежные системы.

Яндекс.Деньги

SOAP запрос:

```
<PaymentRequest xmlns="http://www.moneta.ru/schemas/messages.xsd">
  <payer>MONETA.RU Account Number</payer>
  <payee>13</payee>
  <amount>withdrawal_amount</amount>
  <isPayerAmount>true</isPayerAmount>
  <paymentPassword>payment_password</paymentPassword>
  <operationInfo>
    <attribute>
      <key>YANDEXACCOUNT</key>
      <value>Yandex.Money_Wallet_Number</value>
    </attribute>
  </operationInfo>
</PaymentRequest>
```

JSON запрос:

```
{"Envelope": {
```

```
"Header": {
  "Security": {
    "UsernameToken": {
      "Username": "USERNAME",
      "Password": "PASSWORD"
    }
  }
},
"Body": {
  "PaymentRequest": {
    "payer": "MONETA.RU Account Number",
    "payee": "13",
    "amount": "withdrawal_amount",
    "isPayerAmount": true,
    "paymentPassword": "payment_password",
    "clientTransaction": "Merchant transaction Id",
    "description": "Transaction description",
    "operationInfo": {
      "attribute": [
        {
          "key": "YANDEXACCOUNT",
          "value": "Yandex.Money_Wallet_Number"
        }
      ]
    }
  }
}
}
```

VKontakte

SOAP запрос:

```
<PaymentRequest xmlns="http://www.moneta.ru/schemas/messages.xsd">
  <payer>MONETA.RU Account Number</payer>
  <payee>139</payee>
  <amount>payment_amount</amount>
  <isPayerAmount>true</isPayerAmount>
  <paymentPassword>payment_password</paymentPassword>
  <operationInfo>
    <attribute>
      <key>VKONTAKTEID</key>
      <value>E-mail or ID of the VKontakte user</value>
    </attribute>
  </operationInfo>
</PaymentRequest>
```

JSON запрос:

```
{
  "Envelope": {
    "Header": {
      "Security": {
        "UsernameToken": {
          "Username": "USERNAME",
          "Password": "PASSWORD"
        }
      }
    }
  },
  "Body": {
    "PaymentRequest": {
      "payer": "MONETA.RU Account Number",
      "payee": "139",
      "amount": "withdrawal_amount",
      "isPayerAmount": true,
      "paymentPassword": "payment_password",
      "clientTransaction": "Merchant transaction Id",
      "description": "Transaction description",
      "operationInfo": {
        "attribute": [

```

```
{
    {
        "key": "VKONTAKTEID",
        "value": "E-mail or ID of the VKontakte user"
    }
}
}
```

WebMoney

SOAP запрос:

```
<PaymentRequest xmlns="http://www.moneta.ru/schemas/messages.xsd">
  <payer>MONETA.RU Account Number</payer>
  <payee>2</payee>
  <amount>withdrawal_amount</amount>
  <isPayerAmount>true</isPayerAmount>
  <paymentPassword>payment_password</paymentPassword>
  <operationInfo>
    <attribute>
      <key>WEBMONEYWMID</key>
      <value>Identifier of WebMoney User (WMID)</value>
    </attribute>
    <attribute>
      <key>WEBMONEYPURSE</key>
      <value>WMR Purse Number</value>
    </attribute>
  </operationInfo>
</PaymentRequest>
```

JSON запрос:

```
{
  "Envelope": {
    "Header": {
      "Security": {
        "UsernameToken": {
          "Username": "USERNAME",
          "Password": "PASSWORD"
        }
      }
    },
    "Body": {
      "PaymentRequest": {
        "payer": "MONETA.RU Account Number",
        "payee": "2",
        "amount": "withdrawal_amount",
        "isPayerAmount": true,
        "paymentPassword": "payment_password",
        "clientTransaction": "Merchant transaction Id",
        "description": "Transaction description",
        "operationInfo": {
          "attribute": [
            {
              "key": "WEBMONEYWMID",
              "value": "Identifier of WebMoney User (WMID)"
            },
            {
              "key": "WEBMONEYPURSE",
              "value": "WMR Purse Number"
            }
          ]
        }
      }
    }
  }
}
```

CyberPlat

SOAP запрос:

```
<PaymentRequest xmlns="http://www.moneta.ru/schemas/messages.xsd">
  <payer>MONETA.RU Account Number</payer>
  <payee>1144</payee>
  <amount>withdrawal_amount</amount>
  <isPayerAmount>true</isPayerAmount>
  <paymentPassword>payment_password</paymentPassword>
  <operationInfo>
    <attribute>
      <key>CYBERPLATPROVIDERID</key>
      <value>CYBERPLATPROVIDERID_value</value>
    </attribute>
    <attribute>
      <key>CUSTOMFIELD:100</key>
      <value>8888888888</value>
    </attribute>
  </operationInfo>
</PaymentRequest>
```

JSON запрос:

```
{ "Envelope": {
  "Header": {
    "Security": {
      "UsernameToken": {
        "Username": "USERNAME",
        "Password": "PASSWORD"
      }
    }
  },
  "Body": {
    "PaymentRequest": {
      "payer": "MONETA.RU Account Number",
      "payee": "1144",
      "amount": "withdrawal_amount",
      "isPayerAmount": true,
      "paymentPassword": "payment_password",
      "clientTransaction": "Merchant transaction Id",
      "description": "Transaction description",
      "operationInfo": {
        "attribute": [
          {
            "key": "CYBERPLATPROVIDERID",
            "value": "CYBERPLATPROVIDERID_value"
          },
          {
            "key": "CUSTOMFIELD:100",
            "value": "8888888888"
          }
        ]
      }
    }
  }
}
```

Банковский перевод

SOAP запрос:

```
<PaymentRequest xmlns="http://www.moneta.ru/schemas/messages.xsd">
  <payer>MONETA.RU Account Number</payer>
  <payee>5</payee>
  <amount>withdrawal_amount</amount>
  <isPayerAmount>true</isPayerAmount>
  <paymentPassword>payment_password</paymentPassword>
  <operationInfo>
```

```

<attribute>
  <key>WIREUSERNAME</key>
  <value>beneficiary_name</value>
</attribute>
<attribute>
  <key>WIREUSERINN</key>
  <value>Tax Identification Number (TIN) of the bank</value>
</attribute>
<attribute>
  <key>WIREPAYMENTPURPOSE</key>
  <value>payment_purpose_description</value>
</attribute>
<attribute>
  <key>WIREBANKACCOUNT</key>
  <value>beneficiary_account</value>
</attribute>
<attribute>
  <key>WIREBANKBIK</key>
  <value>russian_central_bank_identififier_code</value>
</attribute>
<attribute>
  <key>WIREBANKNAME</key>
  <value>bank_name</value>
</attribute>
<attribute>
  <key>WIREBANKKS</key>
  <value>bank_correspondent_account</value>
</attribute>
<attribute>
  <key>WIREKPP</key>
  <value>tax_registration_reason_code</value>
</attribute>
<attribute>
  <key>WIREKBK</key>
  <value>budgetary_classification_code</value>
</attribute>
<attribute>
  <key>WIREOKTMO</key>
  <value>russian_classifier_of_municipal_unit_territories</value>
</attribute>
<attribute>
  <key>WIREDOCINDEX</key>
  <value>unique_identififier_of_charges</value>
</attribute>
</operationInfo>
</PaymentRequest>

```

JSON запрос:

```

{"Envelope": {
  "Header": {
    "Security": {
      "UsernameToken": {
        "Username": "USERNAME",
        "Password": "PASSWORD"
      }
    }
  },
  "Body": {
    "PaymentRequest": {
      "payer": "MONETA.RU Account Number",
      "payee": "5",
      "amount": "withdrawal_amount",
      "isPayerAmount": true,
      "paymentPassword": "payment_password",
      "clientTransaction": "Merchant transaction Id",
      "operationInfo": {
        "attribute": [
          {
            "key": "WIREUSERNAME",

```

```

    "value": "beneficiary_name"
  },
  {
    "key": "WIREUSERINN",
    "value": "Tax Identification Number (TIN) of the bank"
  },
  {
    "key": "WIREFPAYMENTPURPOSE",
    "value": "payment_purpose_description"
  },
  {
    "key": "WIREBANKACCOUNT",
    "value": "beneficiary_account"
  },
  {
    "key": "WIREBANKBIK",
    "value": "russian_central_bank_identifiler_code"
  },
  {
    "key": "WIREBANKNAME",
    "value": "bank_name"
  },
  {
    "key": "WIREBANKKS",
    "value": "bank_correspondent_account"
  },
  {
    "key": "WIREKPP",
    "value": "tax_registration_reason_code"
  },
  {
    "key": "WIREKBK",
    "value": "budgetary_classification_code"
  },
  {
    "key": "WIREOKTMO",
    "value": "russian_classifier_of_municipal_unit_territories"
  },
  {
    "key": "WIREDOCINDEX",
    "value": "unique_identifler_of_charges"
  }
}
]
}
}
}}

```

Банковские карты (через Банк Русский Стандарт)

SOAP запрос:

```
<PaymentRequest xmlns="http://www.moneta.ru/schemas/messages.xsd">
  <payer>MONETA.RU Account Number</payer>
  <payee>275</payee>
  <amount>withdrawal_amount</amount>
  <isPayerAmount>true</isPayerAmount>
  <paymentPassword>payment_password</paymentPassword>
  <clientTransaction>unique_transaction_identifier</clientTransaction>
  <operationInfo>
    <attribute>
      <key>PAYERCOUNTRY</key>
      <value>RUS</value>
    </attribute>
    <!-- Withdrawal to a Credit Card that was Issued in Russia -->
    <attribute>
      <key>PAYEECARDNUMBER</key>
      <value>full_credit_card_number</value>
    </attribute>
  </operationInfo>
</PaymentRequest>
```



```
</PaymentRequest>
```

JSON запрос:

```
{
  "Envelope": {
    "Header": {
      "Security": {
        "UsernameToken": {
          "Username": "USERNAME",
          "Password": "PASSWORD"
        }
      }
    },
    "Body": {
      "PaymentRequest": {
        "payer": "MONETA.RU Account Number",
        "payee": "275",
        "amount": "withdrawal_amount",
        "isPayerAmount": true,
        "paymentPassword": "payment_password",
        "clientTransaction": "Merchant transaction Id",
        "description": "Transaction description",
        "operationInfo": {
          "attribute": [
            {
              "key": "PAYERCOUNTRY",
              "value": "RUS"
            },
            {
              "key": "PAYEECARDNUMBER",
              "value": "full_credit_card_number"
            }
          ]
        }
      }
    }
  }
}
```

Банковские карты (через Альфа-Банк)

SOAP запрос:

```
<PaymentRequest xmlns="http://www.moneta.ru/schemas/messages.xsd">
  <payer>MONETA.RU Account Number</payer>
  <payee>279</payee>
  <amount>withdrawal_amount</amount>
  <isPayerAmount>true</isPayerAmount>
  <paymentPassword>payment_password</paymentPassword>
  <clientTransaction>unique_transaction_identifier</clientTransaction>
  <operationInfo>
    <attribute>
      <key>PAYEECARDNUMBER</key>
      <value>full_credit_card_number</value>
    </attribute>
    <attribute>
      <key>CARDEXPIRATION</key>
      <value>credit_card_expiration_date</value>
    </attribute>
    <!-- MM/YYYY -->
  </operationInfo>
</PaymentRequest>
```

JSON запрос:

```
{
  "Envelope": {
    "Header": {
      "Security": {
        "UsernameToken": {
          "Username": "USERNAME",
```

```
    "Password": "PASSWORD"
  }
}
},
"Body": {
  "PaymentRequest": {
    "payer": "MONETA.RU Account Number",
    "payee": "279",
    "amount": "withdrawal_amount",
    "isPayerAmount": true,
    "paymentPassword": "payment_password",
    "clientTransaction": "Merchant transaction Id",
    "description": "Transaction description",
    "operationInfo": {
      "attribute": [
        {
          "key": "PAYEECARDNUMBER",
          "value": "full_credit_card_number"
        },
        {
          "key": "CARDEXPIRATION",
          "value": "MM/YYYY"
        }
      ]
    }
  }
}
}
```

Деньги@Mail.ru

SOAP запрос:

```
<PaymentRequest xmlns="http://www.moneta.ru/schemas/messages.xsd">
  <payer>MONETA.RU Account Number</payer>
  <payee>268</payee>
  <amount>withdrawal_amount</amount>
  <!-- Withdrawal amount format: 12.34 -->
  <isPayerAmount>true</isPayerAmount>
  <paymentPassword>payment_password</paymentPassword>
  <clientTransaction>unique_transaction_identifier</clientTransaction>
  <operationInfo>
    <attribute>
      <key>CUSTOMFIELD:RECIPIENT</key>
      <value>E-mail address or 16-digit account number at Money@Mail.ru</value>
    </attribute>
    <attribute>
      <key>CUSTOMFIELD:PAYMENT_PURPOSE</key>
      <value>payment_purpose_description</value>
    </attribute>
  </operationInfo>
</PaymentRequest>
```

JSON запрос:

```
{
  "Envelope": {
    "Header": {
      "Security": {
        "UsernameToken": {
          "Username": "USERNAME",
          "Password": "PASSWORD"
        }
      }
    }
  },
  "Body": {
    "PaymentRequest": {
      "payer": "MONETA.RU Account Number",
      "payee": "268",
      "amount": "withdrawal_amount",
    }
  }
}
```

```

    "isPayerAmount": true,
    "paymentPassword": "payment_password",
    "clientTransaction": "Merchant transaction Id",
    "description": "Transaction description",
    "operationInfo": {
      "attribute": [
        {
          "key": "CUSTOMFIELD:RECIPIENT",
          "value": "E-mail address or 16-digit account number at
Money@Mail.ru"
        },
        {
          "key": "CUSTOMFIELD:PAYMENT_PURPOSE",
          "value": "payment_purpose_description"
        }
      ]
    }
  }
}
}
```

QIWI

SOAP запрос:

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header>
    <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd" SOAP-ENV:mustUnderstand="1">
      <wsse:UsernameToken
        xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd"
        xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd">
          <wsse:Username>USERNAME</wsse:Username>
          <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-username-token-profile-1.0#PasswordText">PASSWORD</wsse:Password>
        </wsse:UsernameToken>
      </wsse:Security>
    </SOAP-ENV:Header>
    <SOAP-ENV:Body>
      <ns2:PaymentRequest xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd"
        ns2:version="VERSION_2">
        <ns2:payer>MONETA.RU Account Number</ns2:payer>
        <ns2:payee>255</ns2:payee>
        <ns2:amount>AMOUNT</ns2:amount>
        <ns2:isPayerAmount>true</ns2:isPayerAmount>
        <ns2:paymentPassword>PAYMENT_PASSWORD</ns2:paymentPassword>
        <ns2:operationInfo>
          <ns2:attribute>
            <ns2:key>EXTERNALACCOUNTID</ns2:key>
            <ns2:value>QIWI_Account_Id</ns2:value>
          </ns2:attribute>
        </ns2:operationInfo>
      </ns2:PaymentRequest>
    </SOAP-ENV:Body>
  </SOAP-ENV:Envelope>
```

JSON запрос:

```
{
  "Envelope": {
    "Header": {
      "Security": {
        "UsernameToken": {
          "Username": "USERNAME",
          "Password": "PASSWORD"
        }
      }
    }
  },

```

```
"Body": {
  "PaymentRequest": {
    "payer": "MONETA.RU Account Number",
    "payee": "255",
    "amount": "withdrawal_amount",
    "isPayerAmount": true,
    "paymentPassword": "payment_password",
    "clientTransaction": "Merchant transaction Id",
    "description": "Transaction description",
    "operationInfo": {
      "attribute": [
        {
          "key": "EXTERNALACCOUNTID",
          "value": "QIWI_Account_Id"
        }
      ]
    }
  }
}
```

Запросы в пакетном режиме

VerifyPaymentBatch

Запрос возможности провести операцию в системе МОНЕТА.РУ. Запрос выполняется в пакетном режиме.

Примеры запросов

SOAP запрос:

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header>
    <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd" SOAP-ENV:mustUnderstand="1">
      <wsse:UsernameToken xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd"
        wsu:Id="XWSSGID-14563895631711980303723"
        xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd">
          <wsse:Username>USERNAME</wsse:Username>
          <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-username-token-profile-1.0#PasswordText">PASSWORD</wsse:Password>
        </wsse:UsernameToken>
      </wsse:Security>
    </SOAP-ENV:Header>
    <SOAP-ENV:Body>
      <ns2:VerifyPaymentBatchRequest xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd" ns2:version="VERSION_2" xmlns="">
        <ns2:transactional>false</ns2:transactional>
        <ns2:exitOnFailure>false</ns2:exitOnFailure>
        <ns2:transaction ns2:version="VERSION_2">
          <ns2:payer>12345678</ns2:payer>
          <ns2:payee>87654321</ns2:payee>
          <ns2:amount>100.87</ns2:amount>
          <ns2:isPayerAmount>false</ns2:isPayerAmount>
          <ns2:clientTransaction>Merchant transaction Id</ns2:clientTransaction>
        </ns2:transaction>
        <ns2:transaction ns2:version="VERSION_2">
          <ns2:payer>12345678</ns2:payer>
          <ns2:payee>87654321</ns2:payee>
          <ns2:amount>200.87</ns2:amount>
          <ns2:isPayerAmount>false</ns2:isPayerAmount>
          <ns2:clientTransaction>Merchant transaction Id</ns2:clientTransaction>
          <ns2:operationInfo>
            <ns2:attribute>
              <ns2:key>ISINVOICE</ns2:key>
            </ns2:attribute>
          </ns2:operationInfo>
        </ns2:transaction>
      </ns2:VerifyPaymentBatchRequest>
    </SOAP-ENV:Body>
  </SOAP-ENV:Envelope>
```

```

        <ns2:value>1</ns2:value>
      </ns2:attribute>
    </ns2:operationInfo>
  </ns2:transaction>
</ns2:VerifyPaymentBatchRequest>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

SOAP ответ:

```

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <ns2:VerifyPaymentBatchResponse xmlns:ns2="http://www.moneta.ru/schemas/
messages.xsd">
      <ns2:transaction>
        <ns2:isTransactionValid>true</ns2:isTransactionValid>
        <ns2:forecast>
          <ns2:payer>12345678</ns2:payer>
          <ns2:payerCurrency>USD</ns2:payerCurrency>
          <ns2:payerAmount>3.56</ns2:payerAmount>
          <ns2:payee>87654321</ns2:payee>
          <ns2:payeeCurrency>RUB</ns2:payeeCurrency>
          <ns2:payeeAmount>100.87</ns2:payeeAmount>
        </ns2:forecast>
      </ns2:transaction>
      <ns2:transaction>
        <ns2:isTransactionValid>true</ns2:isTransactionValid>
        <ns2:description>Transaction description</ns2:description>
        <ns2:forecast>
          <ns2:payer>12345678</ns2:payer>
          <ns2:payerCurrency>USD</ns2:payerCurrency>
          <ns2:payerAmount>6.17</ns2:payerAmount>
          <ns2:payee>87654321</ns2:payee>
          <ns2:payeeCurrency>RUB</ns2:payeeCurrency>
          <ns2:payeeAmount>200.87</ns2:payeeAmount>
        </ns2:forecast>
        <ns2:operationInfo>
          <ns2:attribute>
            <ns2:key>isinvoice</ns2:key>
            <ns2:value>1</ns2:value>
          </ns2:attribute>
        </ns2:operationInfo>
      </ns2:transaction>
    </ns2:VerifyPaymentBatchResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

SOAP ответ с ошибкой:

```

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <ns2:VerifyPaymentBatchResponse xmlns:ns2="http://www.moneta.ru/schemas/
messages.xsd">
      <ns2:transaction>
        <ns2:isTransactionValid>true</ns2:isTransactionValid>
        <ns2:forecast>
          <ns2:payer>12345678</ns2:payer>
          <ns2:payerCurrency>USD</ns2:payerCurrency>
          <ns2:payerAmount>3.56</ns2:payerAmount>
          <ns2:payee>87654321</ns2:payee>
          <ns2:payeeCurrency>RUB</ns2:payeeCurrency>
          <ns2:payeeAmount>100.87</ns2:payeeAmount>
        </ns2:forecast>
      </ns2:transaction>
      <ns2:transaction>
        <ns2:isTransactionValid>false</ns2:isTransactionValid>
        <ns2:description>Error description</ns2:description>
        <ns2:errorCode>500</ns2:errorCode>
      </ns2:transaction>
    </ns2:VerifyPaymentBatchResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

```

        </ns2:transaction>
    </ns2:VerifyPaymentBatchResponse>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

JSON запрос:

```

{
  "Envelope": {
    "Body": {
      "VerifyPaymentBatchRequest": {
        "version": "VERSION_2",
        "transactional": false,
        "exitOnFailure": false,
        "transaction": [
          {
            "version": "VERSION_2",
            "payer": "12345678",
            "payee": "87654321",
            "amount": 100.87,
            "isPayerAmount": false,
            "clientTransaction": "Merchant transaction Id"
          },
          {
            "version": "VERSION_2",
            "payer": "12345678",
            "payee": "87654321",
            "amount": 200.87,
            "isPayerAmount": false,
            "clientTransaction": "Merchant transaction Id",
            "operationInfo": {
              "attribute": [
                {
                  "value": "1",
                  "key": "ISINVOICE"
                }
              ]
            }
          }
        ]
      }
    }
  },
  "Header": {
    "Security": {
      "UsernameToken": {
        "Password": "PASSWORD",
        "Username": "USERNAME"
      }
    }
  }
}

```

JSON ответ:

```

{
  "Envelope": {
    "Body": {
      "VerifyPaymentBatchResponse": {
        "transaction": [
          {
            "isTransactionValid": true,
            "forecast": {
              "payer": 12345678,
              "payerCurrency": "USD",
              "payerAmount": 3.56,
              "payee": 87654321,
              "payeeCurrency": "RUB",
              "payeeAmount": 100.87
            }
          }
        ]
      }
    }
  }
}

```

```

    },
    {
      "isTransactionValid": true,
      "description": "Transaction description",
      "forecast": {
        "payer": 12345678,
        "payerCurrency": "USD",
        "payerAmount": 6.17,
        "payee": 87654321,
        "payeeCurrency": "RUB",
        "payeeAmount": 200.87
      },
      "operationInfo": {
        "attribute": [
          {
            "value": "1",
            "key": "isinvoice"
          }
        ]
      }
    }
  ]
}

```

JSON ответ с ошибкой:

```

{
  "Envelope": {
    "Body": {
      "VerifyPaymentBatchResponse": {
        "transaction": [
          {
            "isTransactionValid": true,
            "forecast": {
              "payer": 12345678,
              "payerCurrency": "USD",
              "payerAmount": 3.56,
              "payee": 87654321,
              "payeeCurrency": "RUB",
              "payeeAmount": 100.87
            }
          },
          {
            "isTransactionValid": false,
            "description": "Error description",
            "errorCode": "500"
          }
        ]
      }
    }
  }
}

```

История платежей

GetOperationDetailsById - детали операции

Получение данных операции по номеру операции.

Примеры запросов

SOAP запрос:

```
<soapenv:Envelope xmlns:mes="http://www.moneta.ru/schemas/messages.xsd"
```

```

        xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
    <soapenv:Header>
        <wsse:Security soapenv:mustUnderstand="1"
            xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-
wss-wssecurity-secext-1.0.xsd">
            <wsse:UsernameToken wsu:Id="UsernameToken-1"
                xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/
oasis-200401-wss-wssecurity-utility-1.0.xsd">
                <wsse:Username>USERNAME</wsse:Username>
                <wsse:Password
                    Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
username-token-profile-1.0#PasswordText">PASSWORD</wsse:Password>
            </wsse:UsernameToken>
        </wsse:Security>
    </soapenv:Header>
    <soapenv:Body>
        <mes:GetOperationDetailsByIdRequest>101370592</
mes:GetOperationDetailsByIdRequest>
    </soapenv:Body>
</soapenv:Envelope>

```

SOAP ответ:

```

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
    <SOAP-ENV:Header/>
    <SOAP-ENV:Body>
        <ns2:GetOperationDetailsByIdResponse xmlns:ns2="http://www.moneta.ru/schemas/
messages.xsd">
            <ns2:operation>
                <ns2:id>101370592</ns2:id>
                <ns2:attribute>
                    <ns2:key>category</ns2:key>
                    <ns2:value>BUSINESS</ns2:value>
                </ns2:attribute>
                <ns2:attribute>
                    <ns2:key>sourcecurrencycode</ns2:key>
                    <ns2:value>RUB</ns2:value>
                </ns2:attribute>
                <ns2:attribute>
                    <ns2:key>sourceamount</ns2:key>
                    <ns2:value>2.02</ns2:value>
                </ns2:attribute>
                <ns2:attribute>
                    <ns2:key>typeid</ns2:key>
                    <ns2:value>3</ns2:value>
                </ns2:attribute>
                <ns2:attribute>
                    <ns2:key>targetamount</ns2:key>
                    <ns2:value>-2.02</ns2:value>
                </ns2:attribute>
                <ns2:attribute>
                    <ns2:key>sourceamountfee</ns2:key>
                    <ns2:value>-0.03</ns2:value>
                </ns2:attribute>
                <ns2:attribute>
                    <ns2:key>targetcurrencycode</ns2:key>
                    <ns2:value>RUB</ns2:value>
                </ns2:attribute>
                <ns2:attribute>
                    <ns2:key>statusid</ns2:key>
                    <ns2:value>SUCCEED</ns2:value>
                </ns2:attribute>
                <ns2:attribute>
                    <ns2:key>sourceamounttotal</ns2:key>
                    <ns2:value>1.99</ns2:value>
                </ns2:attribute>
                <ns2:attribute>
                    <ns2:key>modified</ns2:key>
                    <ns2:value>2015-07-13T12:00:46.000+03:00</ns2:value>
                </ns2:attribute>
            </ns2:operation>
        </ns2:GetOperationDetailsByIdResponse>
    </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```



```

        <ns2:attribute>
          <ns2:key>targetaccountid</ns2:key>
          <ns2:value>14692426</ns2:value>
        </ns2:attribute>
        <ns2:attribute>
          <ns2:key>sourceaccountid</ns2:key>
          <ns2:value>96801571</ns2:value>
        </ns2:attribute>
        <ns2:attribute>
          <ns2:key>isreversed</ns2:key>
          <ns2:value>true</ns2:value>
        </ns2:attribute>
      </ns2:operation>
    </ns2:GetOperationDetailsByIdResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

JSON запрос:

```

{"Envelope": {
  "Header": {
    "Security": {
      "UsernameToken": {
        "Username": "USERNAME",
        "Password": "PASSWORD"
      }
    }
  },
  "Body": {
    "GetOperationDetailsByIdRequest": 101370592
  }
}}

```

JSON ответ:

```

{"Envelope": {
  "Body": {
    "GetOperationDetailsByIdResponse": {
      "operation": {
        "id": 101370592,
        "attribute": [
          {
            "value": "BUSINESS",
            "key": "category"
          },
          {
            "value": "RUB",
            "key": "sourcecurrencycode"
          },
          {
            "value": "2.02",
            "key": "sourceamount"
          },
          {
            "value": "3",
            "key": "typeid"
          },
          {
            "value": "-2.02",
            "key": "targetamount"
          },
          {
            "value": "-0.03",
            "key": "sourceamountfee"
          },
          {
            "value": "RUB",
            "key": "targetcurrencycode"
          }
        ]
      }
    }
  }
}}

```

```
        "value": "SUCCEED",  
        "key": "statusid"  
    },  
    {  
        "value": "1.99",  
        "key": "sourceamounttotal"  
    },  
    {  
        "value": "2015-07-13T12:00:46.000+03:00",  
        "key": "modified"  
    },  
    {  
        "value": "87654321",  
        "key": "targetaccountid"  
    },  
    {  
        "value": "12345678",  
        "key": "sourceaccountid"  
    },  
    {  
        "value": "true",  
        "key": "isreversed"  
    }  
]  
  
}  
  
}
```

FindOperationsList - список операций

Получение списка операций по фильтру.

Примеры запросов

SOAP запрос:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:mes="http://www.moneta.ru/schemas/messages.xsd">
  <soapenv:Header>
    <wsse:Security soapenv:mustUnderstand="1" xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd">
      <wsse:UsernameToken wsu:Id="UsernameToken-1" xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd">
        <wsse:Username>USERNAME</wsse:Username>
        <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-username-token-profile-1.0#PasswordText">PASSWORD</wsse:Password>
      </wsse:UsernameToken>
    </wsse:Security>
  </soapenv:Header>
  <soapenv:Body>
    <mes:FindOperationsListRequest>
      <mes:pager>
        <mes:pageNumber>1</mes:pageNumber>
        <mes:pageSize>10</mes:pageSize>
      </mes:pager>
      <mes:filter>
        <mes:unitId>123456</mes:unitId>
        <mes:dateFrom>2015-05-01T00:00:00.000+03:00</mes:dateFrom>
        <mes:dateTo>2015-05-03T23:59:59.999+03:00</mes:dateTo>
      </mes:filter>
    </mes:FindOperationsListRequest>
  </soapenv:Body>
</soapenv:Envelope>
```

SOAP ответ:

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
```

```

<SOAP-ENV:Body>
  <ns2:FindOperationsListResponse xmlns:ns2="http://www.moneta.ru/schemas/
messages.xsd">
    <ns2:pageSize>10</ns2:pageSize>
    <ns2:pageNumber>1</ns2:pageNumber>
    <ns2:pagesCount>57</ns2:pagesCount>
    <ns2:size>10</ns2:size>
    <ns2:totalSize>570</ns2:totalSize>
    <ns2:operation>
      <ns2:id>101093434</ns2:id>
      <ns2:attribute>
        <ns2:key>statusid</ns2:key>
        <ns2:value>SUCCEED</ns2:value>
      </ns2:attribute>
      ...
    </ns2:operation>
    ...
    <ns2:operation>
      <ns2:id>101093432</ns2:id>
      <ns2:attribute>
        <ns2:key>statusid</ns2:key>
        <ns2:value>SUCCEED</ns2:value>
      </ns2:attribute>
      ...
    </ns2:operation>
    ...
  </ns2:FindOperationsListResponse>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

JSON запрос:

```

{"Envelope": {
  "Header": {
    "Security": {
      "UsernameToken": {
        "Username": "USERNAME",
        "Password": "PASSWORD"
      }
    }
  },
  "Body": {
    "FindOperationsListRequest": {
      "pager": {
        "pageNumber": 1,
        "pageSize": 10
      },
      "filter": {
        "unitId": 123456,
        "dateFrom": "2015-05-01T00:00:00.000+03:00",
        "dateTo": "2015-05-03T23:59:59.999+03:00"
      }
    }
  }
}}

```

JSON ответ:

```

{"Envelope": {
  "Body": {
    "FindOperationsListResponse": {
      "pageSize": 10,
      "pagesCount": 57,
      "pageNumber": 1,
      "totalSize": 570,
      "size": 10,
      "operation": [
        {
          "id": 101093434,
          "attribute": [

```

```

    {
      "value": "SUCCEED",
      "key": "statusid"
    }
    ...
  ]
},
...
{
  "id": 101093433,
  "attribute": [
    {
      "value": "SUCCEED",
      "key": "statusid"
    }
    ...
  ]
}
...
]
}
}
}}
```

FindLastOperationsList - последние операции

Получение списка последних операций.

Примеры запросов

SOAP запрос:

```
<soapenv:Envelope xmlns:mes="http://www.moneta.ru/schemas/messages.xsd"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Header>
    <wsse:Security soapenv:mustUnderstand="1"
      xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-
wss-wssecurity-secext-1.0.xsd">
      <wsse:UsernameToken wsu:Id="UsernameToken-1"
        xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/
oasis-200401-wss-wssecurity-utility-1.0.xsd">
        <wsse:Username>USERNAME</wsse:Username>
        <wsse:Password
          Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
username-token-profile-1.0#PasswordText">PASSWORD</wsse:Password>
        </wsse:UsernameToken>
      </wsse:Security>
    </soapenv:Header>
    <soapenv:Body>
      <mes:FindLastOperationsListRequest>
        <mes:unitId>123456</mes:unitId>
        <mes:transactionsQuantity>5</mes:transactionsQuantity>
      </mes:FindLastOperationsListRequest>
    </soapenv:Body>
  </soapenv:Envelope>
```

SOAP ответ:

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <ns2:FindLastOperationsListResponse xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd">
      <ns2:pageSize>5</ns2:pageSize>
      <ns2:pageNumber>1</ns2:pageNumber>
      <ns2:pagesCount>1</ns2:pagesCount>
      <ns2:size>5</ns2:size>
      <ns2:totalSize>5</ns2:totalSize>
      <ns2:operation>
```

```

        <ns2:id>101851409</ns2:id>
        <ns2:attribute>
          <ns2:key>statusid</ns2:key>
          <ns2:value>TAKENIN_NOTSENT</ns2:value>
        </ns2:attribute>
        ...
      </ns2:operation>
      ...
    <ns2:operation>
      <ns2:id>101851410</ns2:id>
      <ns2:attribute>
        <ns2:key>statusid</ns2:key>
        <ns2:value>CREATED</ns2:value>
      </ns2:attribute>
      ...
    </ns2:operation>
    ...
  </ns2:FindLastOperationsListResponse>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

JSON запрос:

```

{"Envelope": {
  "Header": {
    "Security": {
      "UsernameToken": {
        "Username": "USERNAME",
        "Password": "PASSWORD"
      }
    }
  },
  "Body": {
    "FindLastOperationsListRequest": {
      "unitId": 123456,
      "transactionsQuantity": 5
    }
  }
}}

```

JSON ответ:

```

{"Envelope": {
  "Body": {
    "FindLastOperationsListResponse": {
      "pageSize": 5,
      "pagesCount": 1,
      "pageNumber": 1,
      "totalSize": 5,
      "size": 5,
      "operation": [
        {
          "id": 101851409,
          "attribute": [
            {
              "value": "TAKENIN_NOTSENT",
              "key": "statusid"
            }
            ...
          ]
        },
        ...
        {
          "id": 101851410,
          "attribute": [
            {
              "value": "CREATED",
              "key": "statusid"
            }
            ...
          ]
        }
      ]
    }
  }
}}

```

FindOperationsListByCTID - детали операции по внешнему идентификатору операции

Получение списка операций по номеру счета и внешнему идентификатору операции.

Примеры запросов

SOAP запрос:

```
<soapenv:Envelope xmlns:mes="http://www.moneta.ru/schemas/messages.xsd"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Header>
    <wsse:Security soapenv:mustUnderstand="1"
      xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-
wss-wssecurity-secext-1.0.xsd">
      <wsse:UsernameToken wsu:Id="UsernameToken-1"
        xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/
oasis-200401-wss-wssecurity-utility-1.0.xsd">
        <wsse:Username>USERNAME</wsse:Username>
        <wsse:Password
          Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
username-token-profile-1.0#PasswordText">PASSWORD</wsse:Password>
        </wsse:UsernameToken>
      </wsse:Security>
    </soapenv:Header>
    <soapenv:Body>
      <mes:FindOperationsListByCTIDRequest>
        <mes:pager>
          <mes:pageNumber>1</mes:pageNumber>
          <mes:pageSize>10</mes:pageSize>
        </mes:pager>
        <mes:accountId>12345678</mes:accountId>
        <mes:clientTransaction>Merchant transaction Id</mes:clientTransaction>
      </mes:FindOperationsListByCTIDRequest>
    </soapenv:Body>
  </soapenv:Envelope>
```

SOAP ответ:

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <ns2:FindOperationsListByCTIDResponse xmlns:ns2="http://www.moneta.ru/schemas/
messages.xsd">
      <ns2:pageSize>10</ns2:pageSize>
      <ns2:pageNumber>1</ns2:pageNumber>
      <ns2:pagesCount>1</ns2:pagesCount>
      <ns2:size>1</ns2:size>
      <ns2:totalSize>1</ns2:totalSize>
      <ns2:operation>
        <ns2:id>100944043</ns2:id>
        <ns2:attribute>
          <ns2:key>statusid</ns2:key>
          <ns2:value>SUCCEED</ns2:value>
        </ns2:attribute>
        <ns2:attribute>
          <ns2:key>clienttransaction</ns2:key>
          <ns2:value>Merchant transaction Id</ns2:value>
        </ns2:attribute>
        ...
      </ns2:operation>
    </ns2:FindOperationsListByCTIDResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

```

    </ns2:FindOperationsListByCTIDResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

JSON запрос:

```

{"Envelope": {
  "Header": {
    "Security": {
      "UsernameToken": {
        "Username": "USERNAME",
        "Password": "PASSWORD"
      }
    }
  },
  "Body": {
    "FindOperationsListByCTIDRequest": {
      "pager": {
        "pageNumber": 1,
        "pageSize": 10
      },
      "accountId": 12345678,
      "clientTransaction": "Merchant transaction Id"
    }
  }
}}

```

JSON ответ:

```

{"Envelope": {
  "Body": {
    "FindOperationsListByCTIDResponse": {
      "pageSize": 10,
      "pagesCount": 1,
      "pageNumber": 1,
      "totalSize": 1,
      "size": 1,
      "operation": [
        {
          "id": 100944043,
          "attribute": [
            {
              "value": "SUCCEED",
              "key": "statusid"
            },
            {
              "value": "Merchant transaction Id",
              "key": "clienttransaction"
            },
            ...
          ]
        }
      ]
    }
  }
}}

```

GetTurnoverList - итоги по месяцам

Получение информации об итогах по месяцам.

Примеры запросов

SOAP запрос:

```

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header>

```

```

    <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-
wss-wssecurity-secext-1.0.xsd" SOAP-ENV:mustUnderstand="1">
      <wsse:UsernameToken
        xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-utility-1.0.xsd"
        wsu:Id="XWSSGID-1491387795271-184820942"
        xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-
wss-wssecurity-secext-1.0.xsd">
        <wsse:Username>USERNAME</wsse:Username>
        <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/
oasis-200401-wss-username-token-profile-1.0#PasswordText">
          PASSWORD
        </wsse:Password>
      </wsse:UsernameToken>
    </wsse:Security>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <ns2:GetTurnoverListRequest xmlns:ns2="http://www.moneta.ru/schemas/
messages.xsd" xmlns="">
      <ns2:unitId>123456</ns2:unitId>
      <ns2:dateFrom>2017-03-01+03:00</ns2:dateFrom>
      <ns2:dateTo>2017-03-31+03:00</ns2:dateTo>
      <ns2:accountIds>12345678</ns2:accountIds>
      <ns2:currencyCode>RUB</ns2:currencyCode>
      <ns2:groupByCurrency>false</ns2:groupByCurrency>
      <ns2:operationTypeCategory>BUSINESS</ns2:operationTypeCategory>
      <ns2:categoryDetails>true</ns2:categoryDetails>
    </ns2:GetTurnoverListRequest>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

SOAP ответ:

```

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <ns2:GetTurnoverListResponse xmlns:ns2="http://www.moneta.ru/schemas/
messages.xsd">
      <ns2:groupByCurrency>false</ns2:groupByCurrency>
      <ns2:turnover>
        <ns2:date>2017-03-01+03:00</ns2:date>
        <ns2:account>12345678</ns2:account>
        <ns2:currency>RUB</ns2:currency>
        <ns2:operationTypeCategory>BUSINESS</ns2:operationTypeCategory>
        <ns2:incomeAmount>253675.62</ns2:incomeAmount>
        <ns2:incomeCommission>-2924.27</ns2:incomeCommission>
        <ns2:incomeTransactionsCount>2547</ns2:incomeTransactionsCount>
        <ns2:expenseAmount>3999.58</ns2:expenseAmount>
        <ns2:expensesIncludingCommission>5.19</ns2:expensesIncludingCommission>
        <ns2:expensesExtraCommission>115.66</ns2:expensesExtraCommission>
        <ns2:expenseTransactionsCount>323</ns2:expenseTransactionsCount>
        <ns2:total>246636.11</ns2:total>
        <ns2:openingBalance>0</ns2:openingBalance>
        <ns2:closingBalance>0</ns2:closingBalance>
      </ns2:turnover>
    </ns2:GetTurnoverListResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

JSON запрос:

```

{
  "Envelope": {
    "Header": {
      "Security": {
        "UsernameToken": {
          "Username": "USERNAME",
          "Password": "PASSWORD"
        }
      }
    }
  }
}

```



```

    },
    "Body": {
      "GetTurnoverListRequest": {
        "unitId": 123456,
        "dateFrom": "2017-03-01T00:00:00.000+03:00",
        "dateTo": "2017-03-31T00:00:00.000+03:00",
        "accountIds": [
          12345678
        ],
        "currencyCode": "RUB",
        "groupByCurrency": false,
        "categoryDetails": true,
        "operationTypeCategory": "BUSINESS"
      }
    }
  }
}

```

JSON ответ:

```

{
  "Envelope": {
    "Body": {
      "GetTurnoverListResponse": {
        "turnover": [
          {
            "total": 246636.11,
            "incomeCommission": -2924.27,
            "expenseAmount": 3999.58,
            "expenseTransactionsCount": 323,
            "date": "2017-03-01T00:00:00.000+03:00",
            "operationTypeCategory": "BUSINESS",
            "currency": "RUB",
            "expensesIncludingCommission": 5.19,
            "openingBalance": 0,
            "incomeAmount": 253675.62,
            "account": "12345678",
            "expensesExtraCommission": 115.66,
            "incomeTransactionsCount": 2547,
            "closingBalance": 0
          }
        ],
        "groupByCurrency": false
      }
    }
  }
}

```

GetTurnoverList (AsyncRequest) - итоги по месяцам

Асинхронное получение информации об итогах по месяцам.

Примеры запросов

SOAP запрос (начинаем запрос):

```

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header>
    <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd" SOAP-ENV:mustUnderstand="1">
      <wsse:UsernameToken
        xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd"
        wsu:Id="XWSSGID-1500891606651-1147232686"
        xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd">
        <wsse:Username>USERNAME</wsse:Username>
        <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-username-token-profile-1.0#PasswordText">PASSWORD</wsse:Password>

```

```

        </wsse:UsernameToken>
      </wsse:Security>
    </SOAP-ENV:Header>
    <SOAP-ENV:Body>
      <ns2:AsyncRequest xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd"
        xmlns="">
        <ns2:GetTurnoverListRequest xmlns:ns2="http://www.moneta.ru/schemas/
        messages.xsd" xmlns="">
          <ns2:unitId>123456</ns2:unitId>
          <ns2:dateFrom>2017-03-01+03:00</ns2:dateFrom>
          <ns2:dateTo>2017-03-31+03:00</ns2:dateTo>
          <ns2:accountIds>12345678</ns2:accountIds>
          <ns2:currencyCode>RUB</ns2:currencyCode>
          <ns2:groupByCurrency>false</ns2:groupByCurrency>
          <ns2:operationTypeCategory>BUSINESS</ns2:operationTypeCategory>
          <ns2:categoryDetails>true</ns2:categoryDetails>
        </ns2:GetTurnoverListRequest>
      </ns2:AsyncRequest>
    </SOAP-ENV:Body>
  </SOAP-ENV:Envelope>

```

SOAP ответ (status=CREATED):

```

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <ns2:AsyncResponse xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd">
      <ns2:asyncId>333333</ns2:asyncId>
      <ns2:asyncStatus>CREATED</ns2:asyncStatus>
      <ns2:expirationDate>2017-07-24T14:20:06.887+03:00</ns2:expirationDate>
    </ns2:AsyncResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

JSON запрос (начинаем запрос):

```

{
  "Envelope": {
    "Header": {
      "Security": {
        "UsernameToken": {
          "Username": "USERNAME",
          "Password": "PASSWORD"
        }
      }
    },
    "Body": {
      "AsyncRequest": {
        "GetTurnoverListRequest": {
          "unitId": 123456,
          "dateFrom": "2017-03-01T00:00:00.000+03:00",
          "dateTo": "2017-03-31T00:00:00.000+03:00",
          "accountIds": [
            12345678
          ],
          "currencyCode": "RUB",
          "groupByCurrency": false,
          "categoryDetails": true,
          "operationTypeCategory": "BUSINESS"
        }
      }
    }
  }
}

```

JSON ответ (status=CREATED):

```

{
  "Envelope": {

```

```

"Body": {
  "AsyncResponse": {
    "expirationDate": "2017-07-24T14:44:15.122+03:00",
    "asyncId": 333333,
    "asyncStatus": "CREATED"
  }
}
}
}

```

SOAP запрос (получение результата по asyncId):

```

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header>
    <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd" SOAP-ENV:mustUnderstand="1">
      <wsse:UsernameToken
        xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd"
        wsu:Id="XWSSGID-1500891607067-1295353757"
        xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd">
          <wsse:Username>USERNAME</wsse:Username>
          <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-username-token-profile-1.0#PasswordText">PASSWORD</wsse:Password>
        </wsse:UsernameToken>
      </wsse:Security>
    </SOAP-ENV:Header>
    <SOAP-ENV:Body>
      <ns2:AsyncRequest xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd"
        xmlns="">
        <ns2:asyncId>333333</ns2:asyncId>
      </ns2:AsyncRequest>
    </SOAP-ENV:Body>
  </SOAP-ENV:Envelope>

```

SOAP ответ (status=INPROGRESS):

```

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <ns2:AsyncResponse xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd">
      <ns2:asyncId>333333</ns2:asyncId>
      <ns2:asyncStatus>INPROGRESS</ns2:asyncStatus>
      <ns2:expirationDate>2017-07-24T14:39:55.000+03:00</ns2:expirationDate>
    </ns2:AsyncResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

SOAP ответ (результат запроса):

```

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <ns2:AsyncResponse xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd">
      <ns2:GetTurnoverListResponse xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd">
        <ns2:groupByCurrency>false</ns2:groupByCurrency>
        <ns2:turnover>
          <ns2:date>2017-03-01+03:00</ns2:date>
          <ns2:account>12345678</ns2:account>
          <ns2:currency>RUB</ns2:currency>
          <ns2:operationTypeCategory>BUSINESS</ns2:operationTypeCategory>
          <ns2:incomeAmount>253675.62</ns2:incomeAmount>
          <ns2:incomeCommission>-2924.27</ns2:incomeCommission>
          <ns2:incomeTransactionsCount>2547</ns2:incomeTransactionsCount>
          <ns2:expenseAmount>3999.58</ns2:expenseAmount>
          <ns2:expensesIncludingCommission>5.19</ns2:expensesIncludingCommission>
        </ns2:turnover>
      </ns2:GetTurnoverListResponse>
    </ns2:AsyncResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

```

        <ns2:expensesExtraCommission>115.66</ns2:expensesExtraCommission>
        <ns2:expenseTransactionsCount>323</ns2:expenseTransactionsCount>
        <ns2:total>246636.11</ns2:total>
        <ns2:openingBalance>0</ns2:openingBalance>
        <ns2:closingBalance>0</ns2:closingBalance>
    </ns2:turnover>
</ns2:GetTurnoverListResponse>
</ns2:AsyncResponse>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

JSON запрос (получение результата по asyncId):

```

{
  "Envelope": {
    "Header": {
      "Security": {
        "UsernameToken": {
          "Username": "USERNAME",
          "Password": "PASSWORD"
        }
      }
    },
    "Body": {
      "AsyncRequest": {
        "asyncId": 333333
      }
    }
  }
}

```

JSON ответ (status=INPROGRESS):

```

{
  "Envelope": {
    "Body": {
      "AsyncResponse": {
        "expirationDate": "2017-07-24T14:44:15.000+03:00",
        "asyncId": 333333,
        "asyncStatus": "INPROGRESS"
      }
    }
  }
}

```

JSON ответ (результат запроса):

```

{
  "Envelope": {
    "Body": {
      "AsyncResponse": {
        "GetTurnoverListResponse": {
          "turnover": [
            {
              "total": 246636.11,
              "incomeCommission": -2924.27,
              "expenseAmount": 3999.58,
              "expenseTransactionsCount": 323,
              "date": "2017-03-01T00:00:00.000+03:00",
              "operationTypeCategory": "BUSINESS",
              "currency": "RUB",
              "expensesIncludingCommission": 5.19,
              "openingBalance": 0,
              "incomeAmount": 253675.62,
              "account": "12345678",
              "expensesExtraCommission": 115.66,
              "incomeTransactionsCount": 2547,
              "closingBalance": 0
            }
          ]
        }
      }
    }
  }
}

```

```
    ],  
    "groupByCurrency": false  
  }  
}  
}  
}
```

GetFinancialFlowsList - финансовые потоки

Получение информации о финансовых потоках.

Примеры запросов

SOAP запрос:

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header>
    <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd" SOAP-ENV:mustUnderstand="1">
      <wsse:UsernameToken
        xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd"
        wsu:Id="XWSSGID-14913860452512108369323"
        xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd">
          <wsse:Username>USERNAME</wsse:Username>
          <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-username-token-profile-1.0#PasswordText">
            PASSWORD
          </wsse:Password>
        </wsse:UsernameToken>
      </wsse:Security>
    </SOAP-ENV:Header>
    <SOAP-ENV:Body>
      <ns2:GetFinancialFlowsListRequest xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd" xmlns="">
        <ns2:unitId>123456</ns2:unitId>
        <ns2:dateFrom>2017-03-01+03:00</ns2:dateFrom>
        <ns2:dateTo>2017-03-31+03:00</ns2:dateTo>
        <ns2:accountIds>12345678</ns2:accountIds>
        <ns2:currencyCode>RUB</ns2:currencyCode>
        <ns2:operationTypeCategory>BUSINESS</ns2:operationTypeCategory>
        <ns2:operationAmountType>INCOME</ns2:operationAmountType>
        <ns2:categoryDetails>true</ns2:categoryDetails>
        <ns2:operationStatusState>COMPLETED</ns2:operationStatusState>
      </ns2:GetFinancialFlowsListRequest>
    </SOAP-ENV:Body>
  </SOAP-ENV:Envelope>
```

SOAP ответ:

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <ns2:GetFinancialFlowsListResponse xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd">
      <ns2:financialFlow>
        <ns2:date>2017-03</ns2:date>
        <ns2:payerSideAccess>false</ns2:payerSideAccess>
        <ns2:payerFlowId>654321</ns2:payerFlowId>
        <ns2:payerFlowName>Payer name</ns2:payerFlowName>
        <ns2:payerCurrencyCode>RUB</ns2:payerCurrencyCode>
        <ns2:payeeSideAccess>true</ns2:payeeSideAccess>
        <ns2:payeeFlowId>123456</ns2:payeeFlowId>
        <ns2:payeeFlowName>Payee name</ns2:payeeFlowName>
        <ns2:payeeCurrencyCode>RUB</ns2:payeeCurrencyCode>
        <ns2:payeeCredited>1481.4</ns2:payeeCredited>
        <ns2:payeeFee>180</ns2:payeeFee>
      </ns2:financialFlow>
    </ns2:GetFinancialFlowsListResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

```

        <ns2:transactionsCount>12</ns2:transactionsCount>
        <ns2:operationTypeCategory>BUSINESS</ns2:operationTypeCategory>
        <ns2:operationStatusState>COMPLETED</ns2:operationStatusState>
    </ns2:financialFlow>
</ns2:GetFinancialFlowsListResponse>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

JSON запрос:

```

{
  "Envelope": {
    "Header": {
      "Security": {
        "UsernameToken": {
          "Username": "USERNAME",
          "Password": "PASSWORD"
        }
      }
    },
    "Body": {
      "GetFinancialFlowsListRequest": {
        "unitId": 123456,
        "dateFrom": "2017-03-01T00:00:00.000+03:00",
        "dateTo": "2017-03-31T00:00:00.000+03:00",
        "accountIds": [
          12345678
        ],
        "currencyCode": "RUB",
        "operationTypeCategory": "BUSINESS",
        "operationAmountType": "INCOME",
        "categoryDetails": true,
        "operationStatusState": "COMPLETED"
      }
    }
  }
}

```

JSON ответ:

```

{
  "Envelope": {
    "Body": {
      "GetFinancialFlowsListResponse": {
        "financialFlow": [
          {
            "payeeFlowId": 123456,
            "payerCurrencyCode": "RUB",
            "date": "2017-03",
            "operationTypeCategory": "BUSINESS",
            "transactionsCount": 12,
            "payerSideAccess": false,
            "payeeCurrencyCode": "RUB",
            "payeeFee": 180,
            "payeeSideAccess": true,
            "operationStatusState": "COMPLETED",
            "payeeCredited": 1481.4,
            "payerFlowName": "Payer name",
            "payeeFlowName": "Payee name",
            "payerFlowId": 654321
          }
        ]
      }
    }
  }
}

```

GetFinancialFlowsList (AsyncRequest) - финансовые потоки

Асинхронное получение информации о финансовых потоках.

Примеры запросов

SOAP запрос (начинаем запрос):

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header>
    <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd" SOAP-ENV:mustUnderstand="1">
      <wsse:UsernameToken
        xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd"
        wsu:Id="XWSSGID-1500901469981-677067966"
        xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd">
        <wsse:Username>USERNAME</wsse:Username>
        <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-username-token-profile-1.0#PasswordText">PASSWORD</wsse:Password>
      </wsse:UsernameToken>
    </wsse:Security>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <ns2:AsyncRequest xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd"
      xmlns="">
      <ns2:GetFinancialFlowsListRequest xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd" xmlns="">
        <ns2:unitId>123456</ns2:unitId>
        <ns2:dateFrom>2017-03-01+03:00</ns2:dateFrom>
        <ns2:dateTo>2017-03-31+03:00</ns2:dateTo>
        <ns2:accountIds>12345678</ns2:accountIds>
        <ns2:currencyCode>RUB</ns2:currencyCode>
        <ns2:operationTypeCategory>BUSINESS</ns2:operationTypeCategory>
        <ns2:operationAmountType>INCOME</ns2:operationAmountType>
        <ns2:categoryDetails>true</ns2:categoryDetails>
        <ns2:operationStatusState>COMPLETED</ns2:operationStatusState>
      </ns2:GetFinancialFlowsListRequest>
    </ns2:AsyncRequest>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

SOAP ответ (status=CREATED):

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <ns2:AsyncResponse xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd">
      <ns2:asyncId>333333</ns2:asyncId>
      <ns2:asyncStatus>CREATED</ns2:asyncStatus>
      <ns2:expirationDate>2017-07-24T18:04:32.357+03:00</ns2:expirationDate>
    </ns2:AsyncResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

JSON запрос (начинаем запрос):

```
{
  "Envelope": {
    "Header": {
      "Security": {
        "UsernameToken": {
          "Username": "USERNAME",
          "Password": "PASSWORD"
        }
      }
    }
  },
  "Body": {
```

```

    "AsyncRequest": {
      "GetFinancialFlowsListRequest": {
        "unitId": 123456,
        "dateFrom": "2017-03-01T00:00:00.000+03:00",
        "dateTo": "2017-03-31T00:00:00.000+03:00",
        "accountIds": [
          12345678
        ],
        "currencyCode": "RUB",
        "operationTypeCategory": "BUSINESS",
        "operationAmountType": "INCOME",
        "categoryDetails": true,
        "operationStatusState": "COMPLETED"
      }
    }
  }
}

```

JSON ответ (status=CREATED):

```

{
  "Envelope": {
    "Body": {
      "AsyncResponse": {
        "expirationDate": "2017-07-24T18:07:12.536+03:00",
        "asyncId": 333333,
        "asyncStatus": "CREATED"
      }
    }
  }
}

```

SOAP запрос (получение результата по asyncId):

```

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header>
    <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd" SOAP-ENV:mustUnderstand="1">
      <wsse:UsernameToken
        xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd"
        wsu:Id="XWSSGID-1500901472966-1144616882"
        xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd">
        <wsse:Username>USERNAME</wsse:Username>
        <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-username-token-profile-1.0#PasswordText">PASSWORD</wsse:Password>
      </wsse:UsernameToken>
    </wsse:Security>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <ns2:AsyncRequest xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd"
      xmlns="">
      <ns2:asyncId>333333</ns2:asyncId>
    </ns2:AsyncRequest>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

SOAP ответ (status=INPROGRESS):

```

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <ns2:AsyncResponse xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd">
      <ns2:asyncId>333333</ns2:asyncId>
      <ns2:asyncStatus>INPROGRESS</ns2:asyncStatus>
      <ns2:expirationDate>2017-07-24T18:04:32.000+03:00</ns2:expirationDate>
    </ns2:AsyncResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```



```
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

SOAP ответ (результат запроса):

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <ns2:AsyncResponse xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd">
      <ns2:GetFinancialFlowsListResponse xmlns:ns2="http://www.moneta.ru/schemas/
messages.xsd">
        <ns2:financialFlow>
          <ns2:date>2017-03</ns2:date>
          <ns2:payerSideAccess>false</ns2:payerSideAccess>
          <ns2:payerFlowId>654321</ns2:payerFlowId>
          <ns2:payerFlowName>Payer name</ns2:payerFlowName>
          <ns2:payerCurrencyCode>RUB</ns2:payerCurrencyCode>
          <ns2:payeeSideAccess>true</ns2:payeeSideAccess>
          <ns2:payeeFlowId>123456</ns2:payeeFlowId>
          <ns2:payeeFlowName>Payee name</ns2:payeeFlowName>
          <ns2:payeeCurrencyCode>RUB</ns2:payeeCurrencyCode>
          <ns2:payeeCredited>1481.4</ns2:payeeCredited>
          <ns2:payeeFee>180</ns2:payeeFee>
          <ns2:transactionsCount>12</ns2:transactionsCount>
          <ns2:operationTypeCategory>BUSINESS</ns2:operationTypeCategory>
          <ns2:operationStatusState>COMPLETED</ns2:operationStatusState>
        </ns2:financialFlow>
      </ns2:GetFinancialFlowsListResponse>
    </ns2:AsyncResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

JSON запрос (получение результата по asyncId):

```
{
  "Envelope": {
    "Header": {
      "Security": {
        "UsernameToken": {
          "Username": "USERNAME",
          "Password": "PASSWORD"
        }
      }
    },
    "Body": {
      "AsyncRequest": {
        "asyncId": 333333
      }
    }
  }
}
```

JSON ответ (status=INPROGRESS):

```
{
  "Envelope": {
    "Body": {
      "AsyncResponse": {
        "expirationDate": "2017-07-24T18:07:12.000+03:00",
        "asyncId": 333333,
        "asyncStatus": "INPROGRESS"
      }
    }
  }
}
```

JSON ответ (результат запроса):

```
{
```

```

"Envelope": {
  "Body": {
    "AsyncResponse": {
      "GetFinancialFlowsListResponse": {
        "financialFlow": [
          {
            "payeeFlowId": 123456,
            "payerCurrencyCode": "RUB",
            "date": "2017-03",
            "operationTypeCategory": "BUSINESS",
            "transactionsCount": 12,
            "payerSideAccess": false,
            "payeeCurrencyCode": "RUB",
            "payeeFee": 180,
            "payeeSideAccess": true,
            "operationStatusState": "COMPLETED",
            "payeeCredited": 1481.4,
            "payerFlowName": "Payer name",
            "payeeFlowName": "Payee name",
            "payerFlowId": 654321
          }
        ]
      }
    }
  }
}

```

Шаблоны операций

CreateOperationTemplate

Запрос на создание шаблона операции.

Подробнее: [CreateOperationTemplate Endpoint](#).

Создание нерегулярного шаблона

Шаблон операции не будет выполняться на регулярной основе. Для таких шаблонов нельзя указать время выполнения. Шаблон можно выполнить только вручную.

Примеры

Минимальный SOAP запрос:

```

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header>
    <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd" SOAP-ENV:mustUnderstand="1">
      <wsse:UsernameToken
        xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd"
        wsu:Id="XWSSGID-1495190341499-967371235"
        xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd">
        <wsse:Username>USERNAME</wsse:Username>
        <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-username-token-profile-1.0#PasswordText">PASSWORD</wsse:Password>
      </wsse:UsernameToken>
    </wsse:Security>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <ns2:CreateOperationTemplateRequest xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd" xmlns="">
      <ns2:type>COMMON</ns2:type>
      <ns2:name>TemplateName</ns2:name>
      <ns2:payer>12345678</ns2:payer>
      <ns2:payee>87654321</ns2:payee>
    </ns2:CreateOperationTemplateRequest>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

```

        <ns2:commonParameters>
            <ns2:amount>10.00</ns2:amount>
            <ns2:isPayerAmount>true</ns2:isPayerAmount>
        </ns2:commonParameters>
    </ns2:CreateOperationTemplateRequest>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

Минимальный JSON запрос:

```

{
  "Envelope": {
    "Header": {
      "Security": {
        "UsernameToken": {
          "Username": "USERNAME",
          "Password": "PASSWORD"
        }
      }
    },
    "Body": {
      "CreateOperationTemplateRequest": {
        "payer": 12345678,
        "name": "TemplateName",
        "payee": 87654321,
        "commonParameters": {
          "amount": 10.0,
          "isPayerAmount": true
        },
        "operationInfo": [],
        "type": "COMMON"
      }
    }
  }
}

```

Максимальный SOAP запрос:

```

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header>
    <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd" SOAP-ENV:mustUnderstand="1">
      <wsse:UsernameToken
        xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd"
        wsu:Id="XWSSGID-1495196689869-1728719953"
        xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd">
        <wsse:Username>USERNAME</wsse:Username>
        <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-username-token-profile-1.0#PasswordText">PASSWORD</wsse:Password>
      </wsse:UsernameToken>
    </wsse:Security>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <ns2:CreateOperationTemplateRequest xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd" xmlns="">
      <ns2:unitId>1111111</ns2:unitId>
      <ns2:type>COMMON</ns2:type>
      <ns2:name>TemplateName</ns2:name>
      <ns2:payer>12345678</ns2:payer>
      <ns2:payee>87654321</ns2:payee>
      <ns2:description>Template description</ns2:description>
      <ns2:tags>transfer, John, work</ns2:tags>
      <ns2:favorite>true</ns2:favorite>
      <ns2:commonParameters>
        <ns2:amount>10.00</ns2:amount>
        <ns2:isPayerAmount>true</ns2:isPayerAmount>
      </ns2:commonParameters>
    </ns2:CreateOperationTemplateRequest>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

```
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Максимальный JSON запрос:

```
{
  "Envelope": {
    "Header": {
      "Security": {
        "UsernameToken": {
          "Username": "USERNAME",
          "Password": "PASSWORD"
        }
      }
    },
    "Body": {
      "CreateOperationTemplateRequest": {
        "unitId": 11111111,
        "type": "COMMON",
        "name": "TemplateName",
        "payer": 12345678,
        "payee": 87654321,
        "description": "Template description",
        "tags": "transfer, John, work",
        "favorite": true,
        "commonParameters": {
          "amount": 10.0,
          "isPayerAmount": true
        },
        "operationInfo": []
      }
    }
  }
}
```

Создание регулярного шаблона

Шаблон операции будет выполняться автоматически в указанное время.

Примеры

SOAP запрос (amountType = AMOUNT):

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header>
    <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd" SOAP-ENV:mustUnderstand="1">
      <wsse:UsernameToken
        xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd"
        wsu:Id="XWSSGID-1495197812473-214663471"
        xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd">
        <wsse:Username>USERNAME</wsse:Username>
        <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-username-token-profile-1.0#PasswordText">PASSWORD</wsse:Password>
      </wsse:UsernameToken>
    </wsse:Security>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <ns2:CreateOperationTemplateRequest xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd" xmlns="">
      <ns2:unitId>11111111</ns2:unitId>
      <ns2:type>REGULAR</ns2:type>
      <ns2:name>Template name</ns2:name>
      <ns2:payer>12345678</ns2:payer>
      <ns2:payee>87654321</ns2:payee>
      <ns2:regularParameters>
        <ns2:amountInfo>
          <ns2:type>AMOUNT</ns2:type>
        </ns2:amountInfo>
      </ns2:regularParameters>
    </ns2:CreateOperationTemplateRequest>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

```

        <ns2:amount>
          <ns2:amount>10.00</ns2:amount>
          <ns2:isPayerAmount>true</ns2:isPayerAmount>
        </ns2:amount>
      </ns2:amountInfo>
      <ns2:timeInfo>
        <ns2:type>EVERY_DAY</ns2:type>
        <ns2:startDateTime>2099-09-04T13:00:00.000+03:00</
ns2:startDateTime>
        <ns2:endDateTime>2099-11-26T02:00:00.000+03:00</ns2:endDateTime>
      </ns2:timeInfo>
      <ns2:reminderInfo>
        <ns2:remind>true</ns2:remind>
        <ns2:hoursBeforeExecution>25</ns2:hoursBeforeExecution>
      </ns2:reminderInfo>
    </ns2:regularParameters>
    <ns2:paymentPassword>
      <ns2:paymentPassword>11111</ns2:paymentPassword>
    </ns2:paymentPassword>
  </ns2:CreateOperationTemplateRequest>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

JSON запрос (amountType = AMOUNT):

```

{
  "Envelope": {
    "Header": {
      "Security": {
        "UsernameToken": {
          "Username": "USERNAME",
          "Password": "PASSWORD"
        }
      }
    },
    "Body": {
      "CreateOperationTemplateRequest": {
        "payer": 12345678,
        "name": "Template name",
        "payee": 87654321,
        "operationInfo": [],
        "type": "REGULAR",
        "unitId": 11111111,
        "paymentPassword": {
          "paymentPassword": "11111"
        },
        "regularParameters": {
          "timeInfo": {
            "startDateTime": "2017-09-04T13:00:00.000+03:00",
            "type": "EVERY_DAY",
            "endDateTime": "2017-11-26T00:00:00.000+03:00"
          },
          "amountInfo": {
            "amount": {
              "amount": 10.0,
              "isPayerAmount": true
            },
            "type": "AMOUNT"
          },
          "reminderInfo": {
            "remind": true,
            "hoursBeforeExecution": 25
          }
        }
      }
    }
  }
}

```

SOAP запрос (amountType = BALANCE):

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header>
    <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd" SOAP-ENV:mustUnderstand="1">
      <wsse:UsernameToken
        xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd"
        wsu:Id="XWSSGID-1495197814475-906210923"
        xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd">
          <wsse:Username>USERNAME</wsse:Username>
          <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-username-token-profile-1.0#PasswordText">PASSWORD</wsse:Password>
        </wsse:UsernameToken>
      </wsse:Security>
    </SOAP-ENV:Header>
    <SOAP-ENV:Body>
      <ns2:CreateOperationTemplateRequest xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd" xmlns="">
        <ns2:unitId>1111111</ns2:unitId>
        <ns2:type>REGULAR</ns2:type>
        <ns2:name>Template name</ns2:name>
        <ns2:payer>12345678</ns2:payer>
        <ns2:payee>87654321</ns2:payee>
        <ns2:regularParameters>
          <ns2:amountInfo>
            <ns2:type>BALANCE</ns2:type>
          </ns2:amountInfo>
          <ns2:timeInfo>
            <ns2:type>EVERY_WORKDAY</ns2:type>
            <ns2:startDateTime>2099-09-04T13:00:00.000+03:00</
ns2:startDateTime>
            <ns2:endDateTime>2099-11-26T02:00:00.000+03:00</ns2:endDateTime>
          </ns2:timeInfo>
        </ns2:regularParameters>
        <ns2:paymentPassword>
          <ns2:paymentPassword>11111</ns2:paymentPassword>
        </ns2:paymentPassword>
      </ns2:CreateOperationTemplateRequest>
    </SOAP-ENV:Body>
  </SOAP-ENV:Envelope>
```

JSON запрос (amountType = BALANCE):

```
{
  "Envelope": {
    "Header": {
      "Security": {
        "UsernameToken": {
          "Username": "USERNAME",
          "Password": "PASSWORD"
        }
      }
    },
    "Body": {
      "CreateOperationTemplateRequest": {
        "paymentPassword": {
          "paymentPassword": "11111"
        },
        "regularParameters": {
          "amountInfo": {
            "type": "BALANCE"
          },
          "timeInfo": {
            "type": "EVERY_WORKDAY",
            "startDateTime": "2099-09-04T13:00:00.000+03:00",
            "endDateTime": "2099-11-26T02:00:00.000+03:00"
          }
        }
      }
    }
  }
}
```

```

    },
    "payer": 12345678,
    "name": "Template name",
    "payee": 87654321,
    "operationInfo": [],
    "type": "REGULAR",
    "unitId": 11111111
  }
}
}
}

```

SOAP запрос (amountType = PAYMENTS):

```

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header>
    <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd" SOAP-ENV:mustUnderstand="1">
      <wsse:UsernameToken
        xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd"
        wsu:Id="XWSSGID-1495197814475-906210923"
        xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd">
        <wsse:Username>USERNAME</wsse:Username>
        <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-username-token-profile-1.0#PasswordText">PASSWORD</wsse:Password>
      </wsse:UsernameToken>
    </wsse:Security>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <ns2:CreateOperationTemplateRequest xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd" xmlns="">
      <ns2:unitId>11111111</ns2:unitId>
      <ns2:type>REGULAR</ns2:type>
      <ns2:name>Template name</ns2:name>
      <ns2:payer>12345678</ns2:payer>
      <ns2:payee>87654321</ns2:payee>
      <ns2:regularParameters>
        <ns2:amountInfo>
          <ns2:type>PAYMENTS</ns2:type>
        </ns2:amountInfo>
        <ns2:timeInfo>
          <ns2:type>EVERY_MONTH</ns2:type>
          <ns2:startDateTime>2099-09-04T13:00:00.000+03:00</
ns2:startDateTime>
          <ns2:endDateTime>2099-11-26T02:00:00.000+03:00</ns2:endDateTime>
        </ns2:timeInfo>
        <ns2:operationsReportNotifications>
          <ns2:id>222222</ns2:id>
          <ns2:selected>true</ns2:selected>
        </ns2:operationsReportNotifications>
      </ns2:regularParameters>
      <ns2:paymentPassword>
        <ns2:paymentPassword>11111</ns2:paymentPassword>
      </ns2:paymentPassword>
    </ns2:CreateOperationTemplateRequest>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

JSON запрос (amountType = PAYMENTS):

```

{
  "Envelope": {
    "Header": {
      "Security": {
        "UsernameToken": {
          "Username": "USERNAME",
          "Password": "PASSWORD"
        }
      }
    }
  }
}

```

```

    }
  },
  "Body": {
    "CreateOperationTemplateRequest": {
      "paymentPassword": {
        "paymentPassword": "11111"
      },
      "regularParameters": {
        "amountInfo": {
          "type": "PAYMENTS"
        },
        "timeInfo": {
          "type": "EVERY_MONTH",
          "startDateTime": "2099-09-04T13:00:00.000+03:00",
          "endDateTime": "2099-11-26T02:00:00.000+03:00"
        },
        "operationsReportNotifications": [
          {
            "id": 222222,
            "selected": true
          }
        ]
      },
      "payer": 12345678,
      "name": "Template name",
      "payee": 87654321,
      "operationInfo": [],
      "type": "REGULAR",
      "unitId": 11111111
    }
  }
}

```

SOAP запрос (amountType = RANGE):

```

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header>
    <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd" SOAP-ENV:mustUnderstand="1">
      <wsse:UsernameToken
        xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd"
        wsu:Id="XWSSGID-1495197815450485075915"
        xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd">
        <wsse:Username>USERNAME</wsse:Username>
        <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-username-token-profile-1.0#PasswordText">PASSWORD</wsse:Password>
      </wsse:UsernameToken>
    </wsse:Security>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <ns2:CreateOperationTemplateRequest xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd" xmlns="">
      <ns2:unitId>11111111</ns2:unitId>
      <ns2:type>REGULAR</ns2:type>
      <ns2:name>Template name</ns2:name>
      <ns2:payer>12345678</ns2:payer>
      <ns2:payee>87654321</ns2:payee>
      <ns2:regularParameters>
        <ns2:amountInfo>
          <ns2:type>RANGE</ns2:type>
          <ns2:range>
            <ns2:amountMinValue>10.00</ns2:amountMinValue>
            <ns2:amountMaxValue>20.00</ns2:amountMaxValue>
          </ns2:range>
        </ns2:amountInfo>
        <ns2:timeInfo>
          <ns2:type>EVERY_LAST_DAY_OF_MONTH</ns2:type>

```



```

        <ns2:startDateTime>2099-09-30T13:00:00.000+03:00</
ns2:startDateTime>
        <ns2:endDateTime>2099-11-26T02:00:00.000+03:00</ns2:endDateTime>
        </ns2:timeInfo>
    </ns2:regularParameters>
    <ns2:paymentPassword>
        <ns2:paymentPassword>11111</ns2:paymentPassword>
    </ns2:paymentPassword>
</ns2:CreateOperationTemplateRequest>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

JSON запрос (amountType = RANGE):

```

{
  "Envelope": {
    "Header": {
      "Security": {
        "UsernameToken": {
          "Username": "USERNAME",
          "Password": "PASSWORD"
        }
      }
    },
    "Body": {
      "CreateOperationTemplateRequest": {
        "paymentPassword": {
          "paymentPassword": "11111"
        },
        "regularParameters": {
          "amountInfo": {
            "type": "RANGE",
            "range": {
              "amountMinValue": 10.0,
              "amountMaxValue": 20.0
            }
          },
          "timeInfo": {
            "type": "EVERY_LAST_DAY_OF_MONTH",
            "startDateTime": "2099-09-30T13:00:00.000+03:00",
            "endDateTime": "2099-11-26T02:00:00.000+03:00"
          }
        },
        "payer": 12345678,
        "name": "Template name",
        "payee": 87654321,
        "operationInfo": [],
        "type": "REGULAR",
        "unitId": 11111111
      }
    }
  }
}

```

SOAP запрос (amountType = REST):

```

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header>
    <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-
wss-wssecurity-secext-1.0.xsd" SOAP-ENV:mustUnderstand="1">
      <wsse:UsernameToken
        xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-utility-1.0.xsd"
        wsu:Id="XWSSGID-1495197816260838370495"
        xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-
wss-wssecurity-secext-1.0.xsd">
        <wsse:Username>USERNAME</wsse:Username>
        <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/
oasis-200401-wss-username-token-profile-1.0#PasswordText">PASSWORD</wsse:Password>
      </wsse:UsernameToken>
    </wsse:Security>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <ns2:CreateOperationTemplateRequest>
      <ns2:regularParameters>
        <ns2:amountInfo>
          <ns2:amountType>REST</ns2:amountType>
          <ns2:amountMinValue>10.0</ns2:amountMinValue>
          <ns2:amountMaxValue>20.0</ns2:amountMaxValue>
        </ns2:amountInfo>
        <ns2:timeInfo>
          <ns2:timeType>EVERY_LAST_DAY_OF_MONTH</ns2:timeType>
          <ns2:startDateTime>2099-09-30T13:00:00.000+03:00</ns2:startDateTime>
          <ns2:endDateTime>2099-11-26T02:00:00.000+03:00</ns2:endDateTime>
        </ns2:timeInfo>
        <ns2:paymentPassword>11111</ns2:paymentPassword>
      </ns2:regularParameters>
      <ns2:payer>12345678</ns2:payer>
      <ns2:name>Template name</ns2:name>
      <ns2:payee>87654321</ns2:payee>
      <ns2:operationInfo></ns2:operationInfo>
      <ns2:type>REGULAR</ns2:type>
      <ns2:unitId>11111111</ns2:unitId>
    </ns2:CreateOperationTemplateRequest>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

```

        </wsse:Security>
    </SOAP-ENV:Header>
    <SOAP-ENV:Body>
        <ns2:CreateOperationTemplateRequest xmlns:ns2="http://www.moneta.ru/schemas/
messages.xsd" xmlns="">
            <ns2:unitId>11111111</ns2:unitId>
            <ns2:type>REGULAR</ns2:type>
            <ns2:name>Template name</ns2:name>
            <ns2:payer>12345678</ns2:payer>
            <ns2:payee>87654321</ns2:payee>
            <ns2:regularParameters>
                <ns2:amountInfo>
                    <ns2:type>REST</ns2:type>
                    <ns2:rest>
                        <ns2:amount>10.00</ns2:amount>
                    </ns2:rest>
                </ns2:amountInfo>
                <ns2:timeInfo>
                    <ns2:type>ONCE</ns2:type>
                    <ns2:startDateTime>2099-09-04T13:00:00.000+03:00</
ns2:startDateTime>
                </ns2:timeInfo>
            </ns2:regularParameters>
            <ns2:paymentPassword>
                <ns2:paymentPassword>11111</ns2:paymentPassword>
            </ns2:paymentPassword>
        </ns2:CreateOperationTemplateRequest>
    </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

JSON запрос (amountType = REST):

```

{
  "Envelope": {
    "Header": {
      "Security": {
        "UsernameToken": {
          "Username": "USERNAME",
          "Password": "PASSWORD"
        }
      }
    },
    "Body": {
      "CreateOperationTemplateRequest": {
        "paymentPassword": {
          "paymentPassword": "11111"
        },
        "regularParameters": {
          "amountInfo": {
            "type": "REST",
            "rest": {
              "amount": 10.0
            }
          },
          "timeInfo": {
            "type": "ONCE",
            "startDateTime": "2099-09-04T13:00:00.000+03:00"
          }
        },
        "payer": 12345678,
        "name": "Template name",
        "payee": 87654321,
        "operationInfo": [],
        "type": "REGULAR",
        "unitId": 11111111
      }
    }
  }
}

```

SOAP запрос (amountType = CREDIT):

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header>
    <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd" SOAP-ENV:mustUnderstand="1">
      <wsse:UsernameToken
        xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd"
        wsu:Id="XWSSGID-1495197814475-906210923"
        xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd">
          <wsse:Username>USERNAME</wsse:Username>
          <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-username-token-profile-1.0#PasswordText">PASSWORD</wsse:Password>
        </wsse:UsernameToken>
      </wsse:Security>
    </SOAP-ENV:Header>
    <SOAP-ENV:Body>
      <ns2:CreateOperationTemplateRequest xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd" xmlns="">
        <ns2:unitId>1111111</ns2:unitId>
        <ns2:type>REGULAR</ns2:type>
        <ns2:name>Template name</ns2:name>
        <ns2:payer>12345678</ns2:payer>
        <ns2:payee>87654321</ns2:payee>
        <ns2:regularParameters>
          <ns2:amountInfo>
            <ns2:type>CREDIT</ns2:type>
          </ns2:amountInfo>
          <ns2:timeInfo>
            <ns2:type>EVERY_WEEK</ns2:type>
            <ns2:startDateTime>2099-09-04T13:00:00.000+03:00</
ns2:startDateTime>
            <ns2:endDateTime>2099-11-26T02:00:00.000+03:00</ns2:endDateTime>
          </ns2:timeInfo>
          <ns2:operationsReportNotifications>
            <ns2:id>222222</ns2:id>
            <ns2:selected>true</ns2:selected>
          </ns2:operationsReportNotifications>
        </ns2:regularParameters>
        <ns2:paymentPassword>
          <ns2:paymentPassword>11111</ns2:paymentPassword>
        </ns2:paymentPassword>
      </ns2:CreateOperationTemplateRequest>
    </SOAP-ENV:Body>
  </SOAP-ENV:Envelope>
```

JSON запрос (amountType = CREDIT):

```
{
  "Envelope": {
    "Header": {
      "Security": {
        "UsernameToken": {
          "Username": "USERNAME",
          "Password": "PASSWORD"
        }
      }
    },
    "Body": {
      "CreateOperationTemplateRequest": {
        "paymentPassword": {
          "paymentPassword": "11111"
        },
        "regularParameters": {
          "amountInfo": {
            "type": "CREDIT"
          },
          "timeInfo": {
```

```

        "type": "EVERY_WEEK",
        "startDateTime": "2099-09-04T13:00:00.000+03:00",
        "endDateTime": "2099-11-26T02:00:00.000+03:00"
    },
    "operationsReportNotifications": [
        {
            "id": 222222,
            "selected": true
        }
    ]
},
"payer": 12345678,
"name": "Template name",
"payee": 87654321,
"operationInfo": [],
"type": "REGULAR",
"unitId": 11111111
}
}
}
}
}

```

Создание шаблона из проведенной операции

Для создания шаблона на основе операции в запросе следует передать элемент `prototypeOperationId`. В нем нужно указать номер операции.

Важно: Операция должна быть в статусе "Выполнена".

Операция должна быть создана тем же пользователем, который создает шаблон.

Если в шаблоне требуется указать свойства отличные от данных в операции, то их можно передать в этом запросе.

Примеры

Минимальный SOAP запрос (создание нерегулярного шаблона):

```

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header>
    <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd" SOAP-ENV:mustUnderstand="1">
      <wsse:UsernameToken
        xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd"
        wsu:Id="XWSSGID-1496927745339-258987445"
        xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd">
        <wsse:Username>USERNAME</wsse:Username>
        <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-username-token-profile-1.0#PasswordText">PASSWORD</wsse:Password>
      </wsse:UsernameToken>
    </wsse:Security>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <ns2:CreateOperationTemplateRequest xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd" xmlns="">
      <ns2:type>COMMON</ns2:type>
      <ns2:name>Template</ns2:name>
      <ns2:prototypeOperationId>88888888</ns2:prototypeOperationId>
    </ns2:CreateOperationTemplateRequest>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

Минимальный JSON запрос (создание нерегулярного шаблона):

```

{
  "Envelope": {
    "Header": {
      "Security": {

```

```

    "UsernameToken": {
      "Username": "USERNAME",
      "Password": "PASSWORD"
    }
  },
  "Body": {
    "CreateOperationTemplateRequest": {
      "type": "COMMON",
      "name": "Template Name",
      "prototypeOperationId": 88888888
    }
  }
}

```

SOAP запрос с указанием дополнительных свойств (создание регулярного шаблона):

```

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header>
    <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd" SOAP-ENV:mustUnderstand="1">
      <wsse:UsernameToken
        xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd"
        wsu:Id="XWSSGID-1496928543165-1438707752"
        xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd">
        <wsse:Username>USERNAME</wsse:Username>
        <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-username-token-profile-1.0#PasswordText">PASSWORD</wsse:Password>
      </wsse:UsernameToken>
    </wsse:Security>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <ns2:CreateOperationTemplateRequest xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd" xmlns="">
      <ns2:type>REGULAR</ns2:type>
      <ns2:name>Template Name</ns2:name>
      <ns2:payer>12345678</ns2:payer>
      <ns2:prototypeOperationId>88888888</ns2:prototypeOperationId>
      <ns2:regularParameters>
        <ns2:amountInfo>
          <ns2:type>AMOUNT</ns2:type>
          <ns2:amount>
            <ns2:amount>10.00</ns2:amount>
            <ns2:isPayerAmount>false</ns2:isPayerAmount>
          </ns2:amount>
        </ns2:amountInfo>
        <ns2:timeInfo>
          <ns2:type>EVERY_DAY</ns2:type>
          <ns2:startDateTime>2099-09-04T13:29:52.000+03:00</
ns2:startDateTime>
          <ns2:endDateTime>2099-11-26T02:01:39.000+03:00</ns2:endDateTime>
        </ns2:timeInfo>
      </ns2:regularParameters>
      <ns2:operationInfo>
        <ns2:key>WIREPAYMENTPURPOSE</ns2:key>
        <ns2:value>payment purpose value</ns2:value>
      </ns2:operationInfo>
      <ns2:paymentPassword>
        <ns2:paymentPassword>11111</ns2:paymentPassword>
      </ns2:paymentPassword>
    </ns2:CreateOperationTemplateRequest>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

JSON с указанием дополнительных свойств (создание регулярного шаблона):

```

{

```

```

"Envelope": {
  "Header": {
    "Security": {
      "UsernameToken": {
        "Username": "USERNAME",
        "Password": "PASSWORD"
      }
    }
  },
  "Body": {
    "CreateOperationTemplateRequest": {
      "paymentPassword": {
        "paymentPassword": "11111"
      },
      "type": "REGULAR",
      "name": "Template Name",
      "prototypeOperationId": 88888888,
      "payer": 12345678,
      "regularParameters": {
        "timeInfo": {
          "type": "EVERY_DAY",
          "startDateTime": "2099-09-04T13:29:52.000+03:00",
          "endDateTime": "2099-11-26T02:01:39.000+03:00"
        },
        "amountInfo": {
          "type": "AMOUNT",
          "amount": {
            "amount": 10.0,
            "isPayerAmount": false
          }
        }
      }
    },
    "operationInfo": [
      {
        "key": "WIREFPAYMENTPURPOSE",
        "value": "payment purpose value"
      }
    ]
  }
}

```

Ответ на создание шаблона операции

Примеры

SOAP ответ:

```

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <ns2:CreateOperationTemplateResponse xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd">
      <ns2:id>12345</ns2:id>
    </ns2:CreateOperationTemplateResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

JSON ответ:

```

{
  "Envelope": {
    "Body": {
      "CreateOperationTemplateResponse": {
        "id": 12345
      }
    }
  }
}

```

}

EditOperationTemplate

Запрос на редактирование шаблона операции.

Подробнее: [EditOperationTemplate Endpoint](#).

Запрос на редактирование шаблона операции

Примеры

Минимальный SOAP запрос (изменение названия шаблона):

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header>
    <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd" SOAP-ENV:mustUnderstand="1">
      <wsse:UsernameToken
        xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd"
        wsu:Id="XWSSGID-1497944564654271860547"
        xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd">
        <wsse:Username>USERNAME</wsse:Username>
        <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-username-token-profile-1.0#PasswordText">PASSWORD</wsse:Password>
      </wsse:UsernameToken>
    </wsse:Security>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <ns2:EditOperationTemplateRequest xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd" xmlns="">
      <ns2:id>12345</ns2:id>
      <ns2:name>New name</ns2:name>
    </ns2:EditOperationTemplateRequest>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Минимальный JSON запрос (изменение названия шаблона):

```
{
  "Envelope": {
    "Header": {
      "Security": {
        "UsernameToken": {
          "Username": "USERNAME",
          "Password": "PASSWORD"
        }
      }
    },
    "Body": {
      "EditOperationTemplateRequest": {
        "id": 12345,
        "name": "New name"
      }
    }
  }
}
```

Максимальный SOAP запрос:

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header>
    <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd" SOAP-ENV:mustUnderstand="1">
      <wsse:UsernameToken
        xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd"
```

```

wsu:Id="XWSSGID-14979452346211167390080"
xmlns:wss="http://docs.oasis-open.org/wss/2004/01/oasis-200401-
wss-wssecurity-secext-1.0.xsd">
  <wsse:Username>USERNAME</wsse:Username>
  <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/
oasis-200401-wss-username-token-profile-1.0#PasswordText">PASSWORD</wsse:Password>
  </wsse:UsernameToken>
</wsse:Security>
</SOAP-ENV:Header>
<SOAP-ENV:Body>
  <ns2:EditOperationTemplateRequest xmlns:ns2="http://www.moneta.ru/schemas/
messages.xsd" xmlns="">
    <ns2:id>12345</ns2:id>
    <ns2:type>REGULAR</ns2:type>
    <ns2:name>New name</ns2:name>
    <ns2:payer>12345678</ns2:payer>
    <ns2:description>New description</ns2:description>
    <ns2:tags>New tag</ns2:tags>
    <ns2:favorite>true</ns2:favorite>
    <ns2:regularParameters>
      <ns2:amountInfo>
        <ns2:type>PAYMENTS</ns2:type>
      </ns2:amountInfo>
      <ns2:timeInfo>
        <ns2:type>EVERY_WEEK</ns2:type>
        <ns2:startDateTime>2099-09-04T13:29:52.000+03:00</
ns2:startDateTime>
        <ns2:endDateTime>2099-11-26T02:01:39.000+03:00</ns2:endDateTime>
      </ns2:timeInfo>
      <ns2:operationsReportNotifications>
        <ns2:id>222222</ns2:id>
        <ns2:selected>false</ns2:selected>
      </ns2:operationsReportNotifications>
      <ns2:operationsReportNotifications>
        <ns2:id>333333</ns2:id>
        <ns2:selected>true</ns2:selected>
      </ns2:operationsReportNotifications>
    </ns2:regularParameters>
    <ns2:paymentPassword>
      <ns2:paymentPassword>11111</ns2:paymentPassword>
    </ns2:paymentPassword>
  </ns2:EditOperationTemplateRequest>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

Максимальный JSON запрос:

```

{
  "Envelope": {
    "Header": {
      "Security": {
        "UsernameToken": {
          "Username": "USERNAME",
          "Password": "PASSWORD"
        }
      }
    },
    "Body": {
      "EditOperationTemplateRequest": {
        "id": 12345,
        "type": "REGULAR",
        "name": "New name",
        "payer": 12345678,
        "description": "New description",
        "tags": "New tag",
        "favorite": true,
        "regularParameters": {
          "amountInfo": {
            "type": "PAYMENTS"
          }
        }
      }
    }
  }
}

```



```

        "timeInfo": {
            "type": "EVERY_WEEK",
            "startDateTime": "2099-09-04T13:29:52.000+03:00",
            "endDateTime": "2099-11-26T02:01:39.000+03:00"
        },
        "operationsReportNotifications": [
            {
                "id": 222222,
                "selected": false
            },
            {
                "id": 333333,
                "selected": true
            }
        ]
    },
    "operationInfo": [],
    "paymentPassword": {
        "paymentPassword": "11111"
    }
}
}
}
}
}

```

Ответ на редактирование шаблона операции

Примеры

SOAP ответ:

```

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <ns2:EditOperationTemplateResponse xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd"/>
    </SOAP-ENV:Body>
  </SOAP-ENV:Envelope>

```

JSON ответ:

```

{
  "Envelope": {
    "Body": {
      "EditOperationTemplateResponse": {}
    }
  }
}

```

FindOperationTemplates

Запрос на поиск и чтение шаблонов.

Подробнее: [FindOperationTemplates Endpoint](#).

Поиск по идентификатору шаблона

Примеры

SOAP запрос:

```

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header>
    <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd" SOAP-ENV:mustUnderstand="1">
      <wsse:UsernameToken
        xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd"

```

```

wsu:Id="XWSSGID-14951903438121655666418"
xmlns:wss="http://docs.oasis-open.org/wss/2004/01/oasis-200401-
wss-wssecurity-secext-1.0.xsd">
  <wsse:Username>USERNAME</wsse:Username>
  <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/
oasis-200401-wss-username-token-profile-1.0#PasswordText">PASSWORD</wsse:Password>
  </wsse:UsernameToken>
</wsse:Security>
</SOAP-ENV:Header>
<SOAP-ENV:Body>
  <ns2:FindOperationTemplatesRequest xmlns:ns2="http://www.moneta.ru/schemas/
messages.xsd" xmlns="">
    <ns2:id>12345</ns2:id>
  </ns2:FindOperationTemplatesRequest>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

JSON запрос:

```

{
  "Envelope": {
    "Header": {
      "Security": {
        "UsernameToken": {
          "Username": "USERNAME",
          "Password": "PASSWORD"
        }
      }
    },
    "Body": {
      "FindOperationTemplatesRequest": {
        "id": 12345
      }
    }
  }
}

```

В ответе - нерегулярный шаблон

Примеры

Минимальный SOAP ответ:

```

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <ns2:FindOperationTemplatesResponse xmlns:ns2="http://www.moneta.ru/schemas/
messages.xsd">
      <ns2:operationTemplate>
        <ns2:id>12345</ns2:id>
        <ns2:unitId>11111111</ns2:unitId>
        <ns2:type>COMMON</ns2:type>
        <ns2:name>TemplateName</ns2:name>
        <ns2:payer>12345678</ns2:payer>
        <ns2:payee>87654321</ns2:payee>
        <ns2:operationTypeCategory>TRANSFER</ns2:operationTypeCategory>
        <ns2:description/>
        <ns2:favorite>>false</ns2:favorite>
        <ns2:commonParameters>
          <ns2:amount>10</ns2:amount>
          <ns2:isPayerAmount>true</ns2:isPayerAmount>
        </ns2:commonParameters>
      </ns2:operationTemplate>
    </ns2:FindOperationTemplatesResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

Минимальный JSON ответ:

```
{
  "Envelope": {
    "Body": {
      "FindOperationTemplatesResponse": {
        "operationTemplate": [
          {
            "payee": 87654321,
            "commonParameters": {
              "amount": 10,
              "isPayerAmount": true
            },
            "type": "COMMON",
            "favorite": false,
            "additionalInfo": [],
            "operationTypeCategory": "TRANSFER",
            "id": 12345,
            "payer": 12345678,
            "description": "",
            "name": "TemplateName",
            "operationInfo": [],
            "unitId": 11111111
          }
        ]
      }
    }
  }
}
```

Максимальный SOAP ответ:

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <ns2:FindOperationTemplatesResponse xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd">
      <ns2:operationTemplate>
        <ns2:id>12345</ns2:id>
        <ns2:unitId>11111111</ns2:unitId>
        <ns2:type>COMMON</ns2:type>
        <ns2:name>TemplateName</ns2:name>
        <ns2:payer>12345678</ns2:payer>
        <ns2:payee>87654321</ns2:payee>
        <ns2:operationTypeCategory>TRANSFER</ns2:operationTypeCategory>
        <ns2:description>Template description</ns2:description>
        <ns2:tags>work, John, transfer</ns2:tags>
        <ns2:favorite>true</ns2:favorite>
        <ns2:commonParameters>
          <ns2:amount>10</ns2:amount>
          <ns2:isPayerAmount>true</ns2:isPayerAmount>
        </ns2:commonParameters>
      </ns2:operationTemplate>
    </ns2:FindOperationTemplatesResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Максимальный JSON ответ:

```
{
  "Envelope": {
    "Body": {
      "FindOperationTemplatesResponse": {
        "operationTemplate": [
          {
            "id": 12345,
            "unitId": 11111111,
            "type": "COMMON",
            "name": "templateName",
            "payer": 12345678,
```

```
"payee": 87654321,  
  "operationTypeCategory": "TRANSFER",  
  "description": "Template description",  
  "tags": "work,John,transfer",  
  "favorite": true,  
  "commonParameters": {  
    "amount": 10,  
    "isPayerAmount": true  
  },  
  "operationInfo": [],  
  "additionalInfo": []  
}  
]  
}  
}  
}
```

В ответе - регулярный шаблон

Примеры

SOAP ответ (amountType = AMOUNT):

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <ns2:FindOperationTemplatesResponse xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd">
      <ns2:operationTemplate>
        <ns2:id>12345</ns2:id>
        <ns2:unitId>11111111</ns2:unitId>
        <ns2:type>REGULAR</ns2:type>
        <ns2:name>Template name</ns2:name>
        <ns2:payer>12345678</ns2:payer>
        <ns2:payee>87654321</ns2:payee>
        <ns2:operationTypeCategory>TRANSFER</ns2:operationTypeCategory>
        <ns2:description/>
        <ns2:favorite>false</ns2:favorite>
        <ns2:regularParameters>
          <ns2:amountInfo>
            <ns2:type>AMOUNT</ns2:type>
            <ns2:amount>
              <ns2:amount>10</ns2:amount>
              <ns2:isPayerAmount>true</ns2:isPayerAmount>
            </ns2:amount>
          </ns2:amountInfo>
          <ns2:timeInfo>
            <ns2:type>EVERY_DAY</ns2:type>
            <ns2:startTime>2099-09-04T13:00:00.000+03:00</
ns2:startTime>
              <ns2:endTime>2099-11-26T02:01:39.000+03:00</
ns2:endTime>
            </ns2:timeInfo>
            <ns2:reminderInfo>
              <ns2:remind>true</ns2:remind>
              <ns2:hoursBeforeExecution>25</ns2:hoursBeforeExecution>
            </ns2:reminderInfo>
          </ns2:regularParameters>
          <ns2:additionalInfo>
            <ns2:key>executionlastdatetime</ns2:key>
            <ns2:value>2099-09-05T13:00:00.000+03:00</ns2:value>
          </ns2:additionalInfo>
          <ns2:additionalInfo>
            <ns2:key>executionlastoperationid</ns2:key>
            <ns2:value>22222222</ns2:value>
          </ns2:additionalInfo>
        </ns2:operationTemplate>
      </ns2:FindOperationTemplatesResponse>
    </SOAP-ENV:Body>
  </SOAP-ENV:Envelope>
```

```
</SOAP-ENV:Envelope>
```

JSON ответ (amountType = AMOUNT):

```
{
  "Envelope": {
    "Body": {
      "FindOperationTemplatesResponse": {
        "operationTemplate": [
          {
            "payee": 87654321,
            "type": "REGULAR",
            "favorite": false,
            "additionalInfo": [],
            "operationTypeCategory": "TRANSFER",
            "id": 12345,
            "payer": 12345678,
            "description": "",
            "name": "Template name",
            "operationInfo": [],
            "unitId": 11111111,
            "regularParameters": {
              "timeInfo": {
                "type": "EVERY_DAY",
                "startDateTime": "2099-09-04T13:00:00.000+03:00",
                "endDateTime": "2099-11-26T02:01:39.000+03:00"
              },
              "amountInfo": {
                "type": "AMOUNT",
                "amount": {
                  "amount": 10,
                  "isPayerAmount": true
                }
              },
              "reminderInfo": {
                "remind": true,
                "hoursBeforeExecution": 25,
                "notification": []
              }
            },
            "additionalInfo": [
              {
                "value": "2099-09-05T13:00:00.000+03:00",
                "key": "executionlastdatetime"
              },
              {
                "value": "22222222",
                "key": "executionlastoperationid"
              }
            ]
          }
        ]
      }
    }
  }
}
```

SOAP ответ (amountType = BALANCE):

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <ns2:FindOperationTemplatesResponse xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd">
      <ns2:operationTemplate>
        <ns2:id>12345</ns2:id>
        <ns2:unitId>11111111</ns2:unitId>
        <ns2:type>REGULAR</ns2:type>
        <ns2:name>Template name</ns2:name>
        <ns2:payer>12345678</ns2:payer>
```

```

<ns2:payee>87654321</ns2:payee>
<ns2:operationTypeCategory>TRANSFER</ns2:operationTypeCategory>
<ns2:description/>
<ns2:favorite>false</ns2:favorite>
<ns2:regularParameters>
  <ns2:amountInfo>
    <ns2:type>BALANCE</ns2:type>
  </ns2:amountInfo>
  <ns2:timeInfo>
    <ns2:type>EVERY_WORKDAY</ns2:type>
    <ns2:startDateTime>2099-09-04T13:00:00.000+03:00</
ns2:startDateTime>
    <ns2:endDateTime>2099-11-26T02:01:39.000+03:00</
ns2:endDateTime>
  </ns2:timeInfo>
</ns2:regularParameters>
</ns2:operationTemplate>
</ns2:FindOperationTemplatesResponse>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

JSON ответ (amountType = BALANCE):

```

{
  "Envelope": {
    "Body": {
      "FindOperationTemplatesResponse": {
        "operationTemplate": [
          {
            "id": 12345,
            "unitId": 11111111,
            "payer": 12345678,
            "payee": 87654321,
            "type": "REGULAR",
            "favorite": false,
            "additionalInfo": [],
            "operationTypeCategory": "TRANSFER",
            "description": "",
            "name": "Template name",
            "operationInfo": [],
            "regularParameters": {
              "amountInfo": {
                "type": "BALANCE"
              },
              "timeInfo": {
                "type": "EVERY_WORKDAY",
                "startDateTime": "2099-09-04T13:00:00.000+03:00",
                "endDateTime": "2099-11-26T02:01:39.000+03:00"
              }
            }
          }
        ]
      }
    }
  }
}

```

SOAP ответ (amountType = PAYMENTS):

```

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <ns2:FindOperationTemplatesResponse xmlns:ns2="http://www.moneta.ru/schemas/
messages.xsd">
      <ns2:operationTemplate>
        <ns2:id>12345</ns2:id>
        <ns2:unitId>11111111</ns2:unitId>
        <ns2:type>REGULAR</ns2:type>
        <ns2:name>Template name</ns2:name>
        <ns2:payer>12345678</ns2:payer>

```

```

<ns2:payee>87654321</ns2:payee>
<ns2:operationTypeCategory>TRANSFER</ns2:operationTypeCategory>
<ns2:description/>
<ns2:favorite>false</ns2:favorite>
<ns2:regularParameters>
  <ns2:amountInfo>
    <ns2:type>PAYMENTS</ns2:type>
  </ns2:amountInfo>
  <ns2:timeInfo>
    <ns2:type>EVERY_MONTH</ns2:type>
    <ns2:startDateTime>2099-09-04T13:00:00.000+03:00</
ns2:startDateTime>
    <ns2:endDateTime>2099-11-26T02:01:39.000+03:00</
ns2:endDateTime>
  </ns2:timeInfo>
  <ns2:operationsReportNotifications>
    <ns2:id>222222</ns2:id>
    <ns2:type>EMAIL</ns2:type>
    <ns2:recipient>report@site.com</ns2:recipient>
    <ns2:selected>true</ns2:selected>
  </ns2:operationsReportNotifications>
</ns2:regularParameters>
</ns2:operationTemplate>
</ns2:FindOperationTemplatesResponse>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

JSON ответ (amountType = PAYMENTS):

```

{
  "Envelope": {
    "Body": {
      "FindOperationTemplatesResponse": {
        "operationTemplate": [
          {
            "payee": 87654321,
            "type": "REGULAR",
            "favorite": false,
            "additionalInfo": [],
            "operationTypeCategory": "TRANSFER",
            "id": 12345,
            "regularParameters": {
              "amountInfo": {
                "type": "PAYMENTS"
              },
              "timeInfo": {
                "type": "EVERY_MONTH",
                "startDateTime": "2099-09-04T13:00:00.000+03:00",
                "endDateTime": "2099-11-26T02:01:39.000+03:00"
              },
              "operationsReportNotifications": [
                {
                  "id": 222222,
                  "type": "EMAIL",
                  "recipient": [
                    "report@site.com"
                  ],
                  "selected": true
                }
              ]
            },
            "payer": 12345678,
            "description": "",
            "name": "Template name",
            "operationInfo": [],
            "unitId": 11111111
          }
        ]
      }
    }
  }
}

```

```
}
}
```

SOAP ответ (amountType = RANGE):

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <ns2:FindOperationTemplatesResponse xmlns:ns2="http://www.moneta.ru/schemas/
messages.xsd">
      <ns2:operationTemplate>
        <ns2:id>12345</ns2:id>
        <ns2:unitId>11111111</ns2:unitId>
        <ns2:type>REGULAR</ns2:type>
        <ns2:name>Template name</ns2:name>
        <ns2:payer>12345678</ns2:payer>
        <ns2:payee>87654321</ns2:payee>
        <ns2:operationTypeCategory>TRANSFER</ns2:operationTypeCategory>
        <ns2:description/>
        <ns2:favorite>false</ns2:favorite>
        <ns2:regularParameters>
          <ns2:amountInfo>
            <ns2:type>RANGE</ns2:type>
            <ns2:range>
              <ns2:amountMinValue>10</ns2:amountMinValue>
              <ns2:amountMaxValue>20</ns2:amountMaxValue>
            </ns2:range>
          </ns2:amountInfo>
          <ns2:timeInfo>
            <ns2:type>EVERY_LAST_DAY_OF_MONTH</ns2:type>
            <ns2:startDateTime>2099-09-30T13:00:00.000+03:00</
ns2:startDateTime>
            <ns2:endDateTime>2099-11-26T02:01:39.000+03:00</
ns2:endDateTime>
          </ns2:timeInfo>
        </ns2:regularParameters>
      </ns2:operationTemplate>
    </ns2:FindOperationTemplatesResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

JSON ответ (amountType = RANGE):

```
{
  "Envelope": {
    "Body": {
      "FindOperationTemplatesResponse": {
        "operationTemplate": [
          {
            "payee": 87654321,
            "type": "REGULAR",
            "favorite": false,
            "additionalInfo": [],
            "operationTypeCategory": "TRANSFER",
            "id": 12345,
            "regularParameters": {
              "amountInfo": {
                "type": "RANGE",
                "range": {
                  "amountMinValue": 10,
                  "amountMaxValue": 20
                }
              },
            },
            "timeInfo": {
              "type": "EVERY_LAST_DAY_OF_MONTH",
              "startDateTime": "2099-09-30T13:00:00.000+03:00",
              "endDateTime": "2099-11-26T02:01:39.000+03:00"
            }
          }
        ],
        "payer": 12345678,
```



```

        "description": "",
        "name": "Template name",
        "operationInfo": [],
        "unitId": 11111111
    }
}
]
}
}
}
}
}
}

```

SOAP ответ (amountType = REST):

```

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <ns2:FindOperationTemplatesResponse xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd">
      <ns2:operationTemplate>
        <ns2:id>12345</ns2:id>
        <ns2:unitId>11111111</ns2:unitId>
        <ns2:type>REGULAR</ns2:type>
        <ns2:name>Template name</ns2:name>
        <ns2:payer>12345678</ns2:payer>
        <ns2:payee>87654321</ns2:payee>
        <ns2:operationTypeCategory>TRANSFER</ns2:operationTypeCategory>
        <ns2:description/>
        <ns2:favorite>false</ns2:favorite>
        <ns2:regularParameters>
          <ns2:amountInfo>
            <ns2:type>REST</ns2:type>
            <ns2:rest>
              <ns2:amount>10</ns2:amount>
            </ns2:rest>
          </ns2:amountInfo>
          <ns2:timeInfo>
            <ns2:type>ONCE</ns2:type>
            <ns2:startDateTime>2099-09-04T13:00:00.000+03:00</
ns2:startDateTime>
          </ns2:timeInfo>
        </ns2:regularParameters>
      </ns2:operationTemplate>
    </ns2:FindOperationTemplatesResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

JSON ответ (amountType = REST):

```

{
  "Envelope": {
    "Body": {
      "FindOperationTemplatesResponse": {
        "operationTemplate": [
          {
            "payee": 87654321,
            "type": "REGULAR",
            "favorite": false,
            "additionalInfo": [],
            "operationTypeCategory": "TRANSFER",
            "id": 12345,
            "regularParameters": {
              "amountInfo": {
                "type": "REST",
                "rest": {
                  "amount": 10
                }
              },
            },
            "timeInfo": {
              "type": "ONCE",
              "startDateTime": "2099-09-04T13:00:00.000+03:00"
            }
          }
        ]
      }
    }
  }
}

```

```

    }
    },
    "payer": 12345678,
    "description": "",
    "name": "Template name",
    "operationInfo": [],
    "unitId": 11111111
  }
]
}
}
}
}
}

```

SOAP ответ (amountType = CREDIT):

```

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <ns2:FindOperationTemplatesResponse xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd">
      <ns2:operationTemplate>
        <ns2:id>12345</ns2:id>
        <ns2:unitId>11111111</ns2:unitId>
        <ns2:type>REGULAR</ns2:type>
        <ns2:name>Template name</ns2:name>
        <ns2:payer>12345678</ns2:payer>
        <ns2:payee>87654321</ns2:payee>
        <ns2:operationTypeCategory>TRANSFER</ns2:operationTypeCategory>
        <ns2:description/>
        <ns2:favorite>false</ns2:favorite>
        <ns2:regularParameters>
          <ns2:amountInfo>
            <ns2:type>CREDIT</ns2:type>
          </ns2:amountInfo>
          <ns2:timeInfo>
            <ns2:type>EVERY_WEEK</ns2:type>
            <ns2:startDateTime>2099-09-04T13:00:00.000+03:00</
ns2:startDateTime>
            <ns2:endDateTime>2099-11-26T02:01:39.000+03:00</
ns2:endDateTime>
          </ns2:timeInfo>
          <ns2:operationsReportNotifications>
            <ns2:id>222222</ns2:id>
            <ns2:type>EMAIL</ns2:type>
            <ns2:recipient>report@site.com</ns2:recipient>
            <ns2:selected>true</ns2:selected>
          </ns2:operationsReportNotifications>
        </ns2:regularParameters>
      </ns2:operationTemplate>
    </ns2:FindOperationTemplatesResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

JSON ответ (amountType = CREDIT):

```

{
  "Envelope": {
    "Body": {
      "FindOperationTemplatesResponse": {
        "operationTemplate": [
          {
            "payee": 87654321,
            "type": "REGULAR",
            "favorite": false,
            "additionalInfo": [],
            "operationTypeCategory": "TRANSFER",
            "id": 12345,
            "regularParameters": {
              "amountInfo": {

```

```

    "type": "CREDIT"
  },
  "timeInfo": {
    "type": "EVERY_WEEK",
    "startDateTime": "2099-09-04T13:00:00.000+03:00",
    "endDateTime": "2099-11-26T02:01:39.000+03:00"
  },
  "operationsReportNotifications": [
    {
      "id": 222222,
      "type": "EMAIL",
      "recipient": [
        "report@site.com"
      ],
      "selected": true
    }
  ]
},
"payer": 12345678,
"description": "",
"name": "Template name",
"operationInfo": [],
"unitId": 11111111
}
]
}
}
}
}
```

Поиск по фильтру

Примеры

SOAP запрос (поиск по unitId):

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header>
    <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd" SOAP-ENV:mustUnderstand="1">
      <wsse:UsernameToken
        xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd"
        wsu:Id="XWSSGID-14955238558031621056608"
        xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd">
          <wsse:Username>USERNAME</wsse:Username>
          <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-username-token-profile-1.0#PasswordText">PASSWORD</wsse:Password>
        </wsse:UsernameToken>
      </wsse:Security>
    </SOAP-ENV:Header>
    <SOAP-ENV:Body>
      <ns2:FindOperationTemplatesRequest xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd" xmlns="">
        <ns2:unitId>11111111</ns2:unitId>
        </ns2:FindOperationTemplatesRequest>
      </SOAP-ENV:Body>
    </SOAP-ENV:Envelope>
```

JSON запрос (поиск по unitId):

```
{
  "Envelope": {
    "Header": {
      "Security": {
        "UsernameToken": {
          "Username": "USERNAME",
          "Password": "PASSWORD",

```

```

    }
  },
  "Body": {
    "FindOperationTemplatesRequest": {
      "unitId": 11111111
    }
  }
}

```

SOAP запрос (поиск по фильтру):

```

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header>
    <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd" SOAP-ENV:mustUnderstand="1">
      <wsse:UsernameToken
        xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd"
        wsu:Id="XWSSGID-14955238558031621056608"
        xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd">
        <wsse:Username>USERNAME</wsse:Username>
        <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-username-token-profile-1.0#PasswordText">PASSWORD</wsse:Password>
      </wsse:UsernameToken>
    </wsse:Security>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <ns2:FindOperationTemplatesRequest xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd" xmlns="">
      <ns2:unitId>11111111</ns2:unitId>
      <ns2:operationTypeCategory>WITHDRAWAL</ns2:operationTypeCategory>
      <ns2:type>COMMON</ns2:type>
      <ns2:tag>work</ns2:tag>
    </ns2:FindOperationTemplatesRequest>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

JSON запрос (поиск по фильтру):

```

{
  "Envelope": {
    "Header": {
      "Security": {
        "UsernameToken": {
          "Username": "USERNAME",
          "Password": "PASSWORD"
        }
      }
    },
    "Body": {
      "FindOperationTemplatesRequest": {
        "unitId": 11111111,
        "operationTypeCategory": "WITHDRAWAL",
        "type": "COMMON",
        "tag": "work"
      }
    }
  }
}

```

Ответ при "поиске по фильтру"

Примеры

SOAP ответ:

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <ns2:FindOperationTemplatesResponse xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd">
      <ns2:operationTemplate>
        <ns2:id>12345</ns2:id>
        <ns2:unitId>11111111</ns2:unitId>
        <ns2:type>COMMON</ns2:type>
        <ns2:name>Template Name</ns2:name>
        <ns2:payer>12345678</ns2:payer>
        <ns2:payee>87654321</ns2:payee>
        <ns2:operationTypeCategory>WITHDRAWAL</ns2:operationTypeCategory>
        <ns2:favorite>>false</ns2:favorite>
        <ns2:commonParameters>
          <ns2:amount>10</ns2:amount>
          <ns2:isPayerAmount>>false</ns2:isPayerAmount>
        </ns2:commonParameters>
        <ns2:additionalInfo>
          <ns2:key>payer_alias</ns2:key>
          <ns2:value>Payer alias</ns2:value>
        </ns2:additionalInfo>
        <ns2:additionalInfo>
          <ns2:key>payee_alias</ns2:key>
          <ns2:value>Payee alias</ns2:value>
        </ns2:additionalInfo>
        <ns2:additionalInfo>
          <ns2:key>payer_currency</ns2:key>
          <ns2:value>RUB</ns2:value>
        </ns2:additionalInfo>
        <ns2:additionalInfo>
          <ns2:key>payee_currency</ns2:key>
          <ns2:value>RUB</ns2:value>
        </ns2:additionalInfo>
      </ns2:operationTemplate>
    </ns2:FindOperationTemplatesResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

JSON ответ:

```
{
  "Envelope": {
    "Body": {
      "FindOperationTemplatesResponse": {
        "operationTemplate": [
          {
            "id": 12345,
            "unitId": 11111111,
            "name": "Template Name",
            "type": "COMMON",
            "payer": 12345678,
            "payee": 87654321,
            "operationTypeCategory": "WITHDRAWAL",
            "favorite": false,
            "commonParameters": {
              "amount": 10,
              "isPayerAmount": false
            },
            "operationInfo": [],
            "additionalInfo": [
              {
                "value": "Payer alias",
                "key": "payer_alias"
              }
            ]
          }
        ]
      }
    }
  }
}
```

```

    },
    {
      "value": "Payee alias",
      "key": "payee_alias"
    },
    {
      "value": "RUB",
      "key": "payer_currency"
    },
    {
      "value": "RUB",
      "key": "payee_currency"
    }
  ]
}
}
}
}
}
}
}
}
}
}

```

Шаблоны операций на вывод средств

Примеры шаблонов операций на вывод средств в различные системы. Показаны примеры для заполнения поля operationInfo.

Вывод в банк

Для вывода средств в российский банк, можно указать следующие поля:

- **WIREBANKBIK** - БИК
- **WIREBANKACCOUNT** - Расчетный счет
- **WIREPAYMENTPURPOSE** - Назначение платежа
- **WIREUSERNAME** - Наименование получателя
- **WIREUSERINN** - ИНН получателя
- **WIREKBK** - КБК
- **WIREOKTMO** - ОКТМО
- **WIREKPP** - КПП

Примеры запросов

SOAP запрос на создание:

```

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header>
    <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd" SOAP-ENV:mustUnderstand="1">
      <wsse:UsernameToken
        xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd"
        wsu:Id="XWSSGID-14954390854211684520532"
        xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd">
        <wsse:Username>USERNAME</wsse:Username>
        <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-username-token-profile-1.0#PasswordText">PASSWORD</wsse:Password>
      </wsse:UsernameToken>
    </wsse:Security>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <ns2:CreateOperationTemplateRequest xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd" xmlns="">
      <ns2:unitId>11111111</ns2:unitId>
      <ns2:type>REGULAR</ns2:type>
      <ns2:name>Template Name</ns2:name>
      <ns2:payer>12345678</ns2:payer>
      <ns2:payee>5</ns2:payee>
      <ns2:regularParameters>

```

```

        <ns2:amountInfo>
            <ns2:type>PAYMENTS</ns2:type>
        </ns2:amountInfo>
        <ns2:timeInfo>
            <ns2:type>EVERY_DAY</ns2:type>
            <ns2:startDateTime>2099-09-04T13:00:00.000+03:00</
ns2:startDateTime>
        </ns2:timeInfo>
    </ns2:regularParameters>
    <ns2:operationInfo>
        <ns2:key>WIREBANKBIK</ns2:key>
        <ns2:value>044525214</ns2:value>
    </ns2:operationInfo>
    <ns2:operationInfo>
        <ns2:key>WIREBANKACCOUNT</ns2:key>
        <ns2:value>40101678901234567896</ns2:value>
    </ns2:operationInfo>
    <ns2:operationInfo>
        <ns2:key>WIREPAYMENTPURPOSE</ns2:key>
        <ns2:value></ns2:value>
    </ns2:operationInfo>
    <ns2:paymentPassword>
        <ns2:paymentPassword>11111</ns2:paymentPassword>
    </ns2:paymentPassword>
</ns2:CreateOperationTemplateRequest>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

JSON запрос на создание:

```

{
  "Envelope": {
    "Header": {
      "Security": {
        "UsernameToken": {
          "Username": "USERNAME",
          "Password": "PASSWORD"
        }
      }
    }
  },
  "Body": {
    "CreateOperationTemplateRequest": {
      "unitId": 11111111,
      "type": "REGULAR",
      "name": "Template Name",
      "payer": 12345678,
      "payee": 5,
      "regularParameters": {
        "amountInfo": {
          "type": "PAYMENTS"
        },
        "timeInfo": {
          "type": "EVERY_DAY",
          "startDateTime": "2099-09-04T13:00:00.000+03:00"
        }
      },
      "operationInfo": [
        {
          "value": "044525214",
          "key": "WIREBANKBIK"
        },
        {
          "value": "40101678901234567896",
          "key": "WIREBANKACCOUNT"
        },
        {
          "value": "",
          "key": "WIREPAYMENTPURPOSE"
        }
      ]
    }
  }
}

```

```

        "paymentPassword": {
            "paymentPassword": "11111"
        }
    }
}
}
}
}

```

SOAP ответ на поиск:

```

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <ns2:FindOperationTemplatesResponse xmlns:ns2="http://www.moneta.ru/schemas/
messages.xsd">
      <ns2:operationTemplate>
        <ns2:id>12345</ns2:id>
        <ns2:unitId>11111111</ns2:unitId>
        <ns2:type>REGULAR</ns2:type>
        <ns2:name>Template Name</ns2:name>
        <ns2:payer>12345678</ns2:payer>
        <ns2:payee>5</ns2:payee>
        <ns2:operationTypeCategory>WITHDRAWAL</ns2:operationTypeCategory>
        <ns2:description/>
        <ns2:favorite>false</ns2:favorite>
        <ns2:regularParameters>
          <ns2:amountInfo>
            <ns2:type>PAYMENTS</ns2:type>
          </ns2:amountInfo>
          <ns2:timeInfo>
            <ns2:type>EVERY_DAY</ns2:type>
            <ns2:startDateTime>2099-09-04T13:00:00.000+03:00</
ns2:startDateTime>
          </ns2:timeInfo>
        </ns2:regularParameters>
        <ns2:operationInfo>
          <ns2:key>wirebankbik</ns2:key>
          <ns2:value>044525214</ns2:value>
        </ns2:operationInfo>
        <ns2:operationInfo>
          <ns2:key>wirepaymentpurpose</ns2:key>
          <ns2:value></ns2:value>
        </ns2:operationInfo>
        <ns2:operationInfo>
          <ns2:key>wirebankaccount</ns2:key>
          <ns2:value>40101678901234567896</ns2:value>
        </ns2:operationInfo>
        <ns2:additionalInfo>
          <ns2:key>executionlastdatetime</ns2:key>
          <ns2:value>2099-09-05T13:00:00.000+03:00</ns2:value>
        </ns2:additionalInfo>
        <ns2:additionalInfo>
          <ns2:key>executionlastoperationid</ns2:key>
          <ns2:value>22222222</ns2:value>
        </ns2:additionalInfo>
      </ns2:operationTemplate>
    </ns2:FindOperationTemplatesResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

JSON ответ на поиск:

```

{
  "Envelope": {
    "Body": {
      "FindOperationTemplatesResponse": {
        "operationTemplate": [
          {
            "id": 12345,
            "unitId": 11111111,

```



```
"type": "REGULAR",
"name": "Template Name",
"payer": 12345678,
"payee": 5,
"operationTypeCategory": "WITHDRAWAL",
"description": "",
"favorite": false,
"regularParameters": {
  "timeInfo": {
    "type": "EVERY_DAY",
    "startDateTime": "2099-09-04T13:00:00.000+03:00"
  },
  "amountInfo": {
    "type": "PAYMENTS"
  }
},
"operationInfo": [
  {
    "value": "044525214",
    "key": "wirebankbik"
  },
  {
    "value": "",
    "key": "wirepaymentpurpose"
  },
  {
    "value": "40101678901234567896",
    "key": "wirebankaccount"
  }
],
"additionalInfo": [
  {
    "value": "2099-09-05T13:00:00.000+03:00",
    "key": "executionlastdatetime"
  },
  {
    "value": "22222222",
    "key": "executionlastoperationid"
  }
]
}
}
```

Вывод на карту

Примеры запросов

SOAP запрос на создание:

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header>
    <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd" SOAP-ENV:mustUnderstand="1">
      <wsse:UsernameToken
        xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd"
        wsu:Id="XWSSGID-1495544753255-363074483"
        xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd">
          <wsse:Username>USERNAME</wsse:Username>
          <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-username-token-profile-1.0#PasswordText">PASSWORD</wsse:Password>
        </wsse:UsernameToken>
      </wsse:Security>
    </SOAP-ENV:Header>
    <SOAP-ENV:Body>
```

```

<ns2:CreateOperationTemplateRequest xmlns:ns2="http://www.moneta.ru/schemas/
messages.xsd" xmlns="">
  <ns2:type>COMMON</ns2:type>
  <ns2:name>Template Name</ns2:name>
  <ns2:payer>12345678</ns2:payer>
  <ns2:payee>279</ns2:payee>
  <ns2:commonParameters>
    <ns2:amount>10.00</ns2:amount>
    <ns2:isPayerAmount>>false</ns2:isPayerAmount>
  </ns2:commonParameters>
  <ns2:operationInfo>
    <ns2:key>PAYEECARDNUMBER</ns2:key>
    <ns2:value>4444441111114444</ns2:value>
  </ns2:operationInfo>
  <ns2:operationInfo>
    <ns2:key>CARDEXPIRATION</ns2:key>
    <ns2:value>01/2020</ns2:value>
  </ns2:operationInfo>
</ns2:CreateOperationTemplateRequest>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

JSON запрос на создание:

```

{
  "Envelope": {
    "Header": {
      "Security": {
        "UsernameToken": {
          "Username": "USERNAME",
          "Password": "PASSWORD"
        }
      }
    }
  },
  "Body": {
    "CreateOperationTemplateRequest": {
      "payer": 12345678,
      "name": "Template Name",
      "payee": 279,
      "commonParameters": {
        "amount": 10.0,
        "isPayerAmount": false
      },
      "operationInfo": [
        {
          "value": "4444441111114444",
          "key": "PAYEECARDNUMBER"
        },
        {
          "value": "01\2020",
          "key": "CARDEXPIRATION"
        }
      ],
      "type": "COMMON"
    }
  }
}

```

SOAP ответ на поиск:

```

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <ns2:FindOperationTemplatesResponse xmlns:ns2="http://www.moneta.ru/schemas/
messages.xsd">
      <ns2:operationTemplate>
        <ns2:id>12345</ns2:id>
        <ns2:unitId>11111111</ns2:unitId>
        <ns2:type>COMMON</ns2:type>
      </ns2:operationTemplate>
    </ns2:FindOperationTemplatesResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

```

<ns2:name>Template Name</ns2:name>
<ns2:payer>12345678</ns2:payer>
<ns2:payee>279</ns2:payee>
<ns2:operationTypeCategory>WITHDRAWAL</ns2:operationTypeCategory>
<ns2:description/>
<ns2:favorite>false</ns2:favorite>
<ns2:commonParameters>
  <ns2:amount>10</ns2:amount>
  <ns2:isPayerAmount>false</ns2:isPayerAmount>
</ns2:commonParameters>
<ns2:operationInfo>
  <ns2:key>cardexpiration</ns2:key>
  <ns2:value>01/2020</ns2:value>
</ns2:operationInfo>
<ns2:operationInfo>
  <ns2:key>payeecardnumber</ns2:key>
  <ns2:value>444444*****4444</ns2:value>
</ns2:operationInfo>
</ns2:operationTemplate>
</ns2:FindOperationTemplatesResponse>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

JSON ответ на поиск:

```

{
  "Envelope": {
    "Body": {
      "FindOperationTemplatesResponse": {
        "operationTemplate": [
          {
            "payee": 279,
            "commonParameters": {
              "amount": 10,
              "isPayerAmount": false
            },
            "type": "COMMON",
            "favorite": false,
            "additionalInfo": [],
            "operationTypeCategory": "WITHDRAWAL",
            "id": 12345,
            "payer": 12345678,
            "description": "",
            "name": "Template Name",
            "operationInfo": [
              {
                "value": "01\2020",
                "key": "cardexpiration"
              },
              {
                "value": "444444*****4444",
                "key": "payeecardnumber"
              }
            ],
            "unitId": 11111111
          }
        ],
        "unitId": 11111111
      }
    }
  }
}

```

Вывод в QIWI

Примеры запросов

SOAP запрос на создание:

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
```

```

<SOAP-ENV:Header>
  <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-
wss-wssecurity-secext-1.0.xsd" SOAP-ENV:mustUnderstand="1">
    <wsse:UsernameToken
      xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-utility-1.0.xsd"
      wsu:Id="XWSSGID-1495544443076-462207690"
      xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-
wss-wssecurity-secext-1.0.xsd">
      <wsse:Username>USERNAME</wsse:Username>
      <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/
oasis-200401-wss-username-token-profile-1.0#PasswordText">PASSWORD</wsse:Password>
    </wsse:UsernameToken>
  </wsse:Security>
</SOAP-ENV:Header>
<SOAP-ENV:Body>
  <ns2:CreateOperationTemplateRequest xmlns:ns2="http://www.moneta.ru/schemas/
messages.xsd" xmlns="">
    <ns2:type>COMMON</ns2:type>
    <ns2:name>Template Name</ns2:name>
    <ns2:payer>12345678</ns2:payer>
    <ns2:payee>255</ns2:payee>
    <ns2:commonParameters>
      <ns2:amount>10.00</ns2:amount>
      <ns2:isPayerAmount>>false</ns2:isPayerAmount>
    </ns2:commonParameters>
    <ns2:operationInfo>
      <ns2:key>EXTERNALACCOUNTID</ns2:key>
      <ns2:value>111111111111</ns2:value>
    </ns2:operationInfo>
  </ns2:CreateOperationTemplateRequest>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

JSON запрос на создание:

```

{
  "Envelope": {
    "Header": {
      "Security": {
        "UsernameToken": {
          "Username": "USERNAME",
          "Password": "PASSWORD"
        }
      }
    },
    "Body": {
      "CreateOperationTemplateRequest": {
        "payer": 12345678,
        "name": "Template Name",
        "payee": 255,
        "commonParameters": {
          "amount": 10.0,
          "isPayerAmount": false
        },
        "operationInfo": [
          {
            "value": "111111111111",
            "key": "EXTERNALACCOUNTID"
          }
        ],
        "type": "COMMON"
      }
    }
  }
}

```

SOAP ответ на поиск:

```

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">

```

```

<SOAP-ENV:Header/>
<SOAP-ENV:Body>
  <ns2:FindOperationTemplatesResponse xmlns:ns2="http://www.moneta.ru/schemas/
messages.xsd">
    <ns2:operationTemplate>
      <ns2:id>12345</ns2:id>
      <ns2:unitId>11111111</ns2:unitId>
      <ns2:type>COMMON</ns2:type>
      <ns2:name>Template Name</ns2:name>
      <ns2:payer>12345678</ns2:payer>
      <ns2:payee>255</ns2:payee>
      <ns2:operationTypeCategory>WITHDRAWAL</ns2:operationTypeCategory>
      <ns2:description/>
      <ns2:favorite>false</ns2:favorite>
      <ns2:commonParameters>
        <ns2:amount>10</ns2:amount>
        <ns2:isPayerAmount>false</ns2:isPayerAmount>
      </ns2:commonParameters>
      <ns2:operationInfo>
        <ns2:key>externalaccountid</ns2:key>
        <ns2:value>111111111111</ns2:value>
      </ns2:operationInfo>
    </ns2:operationTemplate>
  </ns2:FindOperationTemplatesResponse>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

JSON ответ на поиск:

```

{
  "Envelope": {
    "Body": {
      "FindOperationTemplatesResponse": {
        "operationTemplate": [
          {
            "payee": 255,
            "commonParameters": {
              "amount": 10,
              "isPayerAmount": false
            },
            "type": "COMMON",
            "favorite": false,
            "additionalInfo": [],
            "operationTypeCategory": "WITHDRAWAL",
            "id": 12345,
            "payer": 12345678,
            "description": "",
            "name": "Template Name",
            "operationInfo": [
              {
                "value": "111111111111",
                "key": "externalaccountid"
              }
            ],
            "unitId": 11111111
          }
        ]
      }
    }
  }
}

```

DeleteOperationTemplate

Запрос на удаление шаблона операции.

Подробнее: [DeleteOperationTemplate Endpoint](#).

Примеры запросов

SOAP запрос:

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header>
    <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-
wss-wssecurity-secext-1.0.xsd" SOAP-ENV:mustUnderstand="1">
      <wsse:UsernameToken
        xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-utility-1.0.xsd"
        wsu:Id="XWSSGID-14951903443231747031771"
        xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-
wss-wssecurity-secext-1.0.xsd">
        <wsse:Username>USERNAME</wsse:Username>
        <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/
oasis-200401-wss-username-token-profile-1.0#PasswordText">PASSWORD</wsse:Password>
      </wsse:UsernameToken>
    </wsse:Security>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <ns2:DeleteOperationTemplateRequest xmlns:ns2="http://www.moneta.ru/schemas/
messages.xsd" xmlns="">
      <ns2:id>12345</ns2:id>
    </ns2:DeleteOperationTemplateRequest>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

SOAP ответ:

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <ns2:DeleteOperationTemplateResponse xmlns:ns2="http://www.moneta.ru/schemas/
messages.xsd"/>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

JSON запрос:

```
{
  "Envelope": {
    "Header": {
      "Security": {
        "UsernameToken": {
          "Username": "USERNAME",
          "Password": "PASSWORD"
        }
      }
    },
    "Body": {
      "DeleteOperationTemplateRequest": {
        "id": 12345
      }
    }
  }
}
```

JSON ответ:

```
{
  "Envelope": {
    "Body": {
      "DeleteOperationTemplateResponse": {}
    }
  }
}
```

Работа с профайлом пользователя

Чтение профайла

SOAP запрос:

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header>
  <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-secext-1.0.xsd" SOAP-ENV:mustUnderstand="1">
    <wsse:UsernameToken xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/
oasis-200401-wss-wssecurity-utility-1.0.xsd" wsu:Id="XWSSGID-1417525998002-1416764885"
xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
secext-1.0.xsd">
      <wsse:Username>USERNAME</wsse:Username>
      <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-
wss-username-token-profile-1.0#PasswordText">PASSWORD</wsse:Password>
    </wsse:UsernameToken>
  </wsse:Security>
</SOAP-ENV:Header>
<SOAP-ENV:Body>
  <ns2:GetProfileInfoRequest xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd"
ns2:version="VERSION_2" xmlns="">
    <ns2:unitId>10054789</ns2:unitId>
  </ns2:GetProfileInfoRequest>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

SOAP ответ:

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
  <ns2:GetProfileInfoResponse xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd">
    <ns2:attribute>
      <ns2:key>last_name</ns2:key>
      <ns2:value>Last name</ns2:value>
      <ns2:approved>>false</ns2:approved>
      <ns2:published>>false</ns2:published>
    </ns2:attribute>
    <ns2:attribute>
      <ns2:key>first_name</ns2:key>
      <ns2:value>First name</ns2:value>
      <ns2:approved>>false</ns2:approved>
      <ns2:published>>false</ns2:published>
    </ns2:attribute>
    <ns2:attribute>
      <ns2:key>date_of_birth</ns2:key>
      <ns2:value>1981-07-23</ns2:value>
      <ns2:approved>>false</ns2:approved>
      <ns2:published>>false</ns2:published>
    </ns2:attribute>
    <ns2:attribute>
      <ns2:key>profileType</ns2:key>
      <ns2:value>client</ns2:value>
      <ns2:approved>>false</ns2:approved>
      <ns2:published>>false</ns2:published>
    </ns2:attribute>
    <ns2:attribute>
      <ns2:key>cell_phone</ns2:key>
      <ns2:value>+71234567890</ns2:value>
      <ns2:approved>>false</ns2:approved>
      <ns2:published>>false</ns2:published>
    </ns2:attribute>
    <ns2:attribute>
      <ns2:key>unitid</ns2:key>
      <ns2:value>10054789</ns2:value>
    </ns2:attribute>
  </ns2:GetProfileInfoResponse>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

```

        <ns2:approved>false</ns2:approved>
        <ns2:published>false</ns2:published>
    </ns2:attribute>
</ns2:GetProfileInfoResponse>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

JSON запрос:

```

{"Envelope": {
  "Header": {
    "Security": {
      "UsernameToken": {
        "Username": "USERNAME",
        "Password": "PASSWORD"
      }
    }
  },
  "Body": {
    "GetProfileInfoRequest": {
      "version": "VERSION_2",
      "unitId": 10060344
    }
  }
}}

```

JSON ответ:

```

{"Envelope": {
  "Body": {
    "GetProfileInfoResponse": {
      "attribute": [
        {
          "approved": false,
          "value": "Last name",
          "published": false,
          "key": "last_name"
        },
        {
          "approved": false,
          "value": "First name",
          "published": false,
          "key": "first_name"
        },
        {
          "approved": false,
          "value": "1981-07-23",
          "published": false,
          "key": "date_of_birth"
        },
        {
          "approved": false,
          "value": "client",
          "published": false,
          "key": "profileType"
        },
        {
          "approved": false,
          "value": "+71234567890",
          "published": false,
          "key": "cell_phone"
        },
        {
          "approved": false,
          "value": "10060344",
          "published": false,
          "key": "unitid"
        }
      ],
      "approved": false,

```



```

        "value": "10065755",
        "published": false,
        "key": "profileid"
      }
    ]
  }
}
}
}

```

Создание профайла

SOAP запрос:

```

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header>
    <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd" SOAP-ENV:mustUnderstand="1">
      <wsse:UsernameToken xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd" wsu:Id="XWSSGID-1417525994891695150469"
        xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd">
        <wsse:Username>USERNAME</wsse:Username>
        <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-username-token-profile-1.0#PasswordText">PASSWORD</wsse:Password>
      </wsse:UsernameToken>
    </wsse:Security>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <ns2:CreateProfileRequest xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd"
      xmlns="">
      <ns2:unitId>10053304</ns2:unitId>
      <ns2:profileType>client</ns2:profileType>
      <ns2:profile>
        <ns2:attribute>
          <ns2:key>LAST_NAME</ns2:key>
          <ns2:value>Last name</ns2:value>
        </ns2:attribute>
        <ns2:attribute>
          <ns2:key>FIRST_NAME</ns2:key>
          <ns2:value>First name</ns2:value>
        </ns2:attribute>
        <ns2:attribute>
          <ns2:key>CELL_PHONE</ns2:key>
          <ns2:value>+71234567899</ns2:value>
        </ns2:attribute>
        <ns2:attribute>
          <ns2:key>DATE_OF_BIRTH</ns2:key>
          <ns2:value>1981-05-24</ns2:value>
        </ns2:attribute>
      </ns2:profile>
    </ns2:CreateProfileRequest>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

SOAP ответ:

```

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <ns2:CreateProfileResponse xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd">10054789</ns2:CreateProfileResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

JSON запрос:

```

{"Envelope": {
  "Header": {
    "Security": {

```

```

        "UsernameToken": {
            "Username": "USERNAME",
            "Password": "PASSWORD"
        }
    },
    "Body": {
        "CreateProfileRequest": {
            "unitId": 10053304,
            "profileType": "CLIENT",
            "profile": {
                "attribute": [
                    {
                        "key": "LAST_NAME",
                        "value": "Last name"
                    },
                    {
                        "key": "FIRST_NAME",
                        "value": "First name"
                    },
                    {
                        "key": "CELL_PHONE",
                        "value": "+71234567899"
                    },
                    {
                        "key": "DATE_OF_BIRTH",
                        "value": "1981-05-24"
                    }
                ]
            }
        }
    }
}
}
```

JSON ответ:

```
{
  "Envelope": {
    "Body": {
      "CreateProfileResponse": 10060344
    }
  }
}
```

Редактирование профиля

SOAP запрос:

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header>
  <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-secext-1.0.xsd" SOAP-ENV:mustUnderstand="1">
    <wsse:UsernameToken xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/
oasis-200401-wss-wssecurity-utility-1.0.xsd" wsu:Id="XWSSGID-14175259972751278772983"
  xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
secext-1.0.xsd">
      <wsse:Username>USERNAME</wsse:Username>
      <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-
wss-username-token-profile-1.0#PasswordText">PASSWORD</wsse:Password>
    </wsse:UsernameToken>
  </wsse:Security>
</SOAP-ENV:Header>
<SOAP-ENV:Body>
  <ns2:EditProfileRequest xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd"
xmlns="">
    <ns2:unitId>10054789</ns2:unitId>
    <ns2:profile>
      <ns2:attribute>
        <ns2:key>LAST_NAME</ns2:key>
        <ns2:value>Last name</ns2:value>
      </ns2:attribute>
    </ns2:profile>
  </ns2:EditProfileRequest>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

```

        <ns2:attribute>
          <ns2:key>FIRST_NAME</ns2:key>
          <ns2:value>First name</ns2:value>
        </ns2:attribute>
        <ns2:attribute>
          <ns2:key>CELL_PHONE</ns2:key>
          <ns2:value>+71234567890</ns2:value>
        </ns2:attribute>
        <ns2:attribute>
          <ns2:key>DATE_OF_BIRTH</ns2:key>
          <ns2:value>1981-07-23</ns2:value>
        </ns2:attribute>
      </ns2:profile>
    </ns2:EditProfileRequest>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

SOAP ответ:

```

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <ns2:EditProfileResponse xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd"/>
  >
    </SOAP-ENV:Body>
  </SOAP-ENV:Envelope>

```

JSON запрос:

```

{"Envelope": {
  "Header": {
    "Security": {
      "UsernameToken": {
        "Username": "USERNAME",
        "Password": "PASSWORD"
      }
    }
  },
  "Body": {
    "EditProfileRequest": {
      "unitId": 10060344,
      "profile": {
        "attribute": [
          {
            "key": "LAST_NAME",
            "value": "Last name"
          },
          {
            "key": "FIRST_NAME",
            "value": "Fist name"
          },
          {
            "key": "CELL_PHONE",
            "value": "+71234567890"
          },
          {
            "key": "DATE_OF_BIRTH",
            "value": "1981-07-23"
          }
        ]
      }
    }
  }
}
}
}
}

```

JSON ответ:

```

{"Envelope": {
  "Body": {

```

```

    "EditProfileResponse": {}
  }
}

```

Работа со счетами пользователя

Чтение списка счетов

SOAP запрос:

```

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header>
  <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-secext-1.0.xsd" SOAP-ENV:mustUnderstand="1">
    <wsse:UsernameToken xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/
oasis-200401-wss-wssecurity-utility-1.0.xsd" wsu:Id="XWSSGID-1419574025786-400683759"
xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
secext-1.0.xsd">
      <wsse:Username>USERNAME</wsse:Username>
      <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-
wss-username-token-profile-1.0#PasswordText">PASSWORD</wsse:Password>
    </wsse:UsernameToken>
  </wsse:Security>
</SOAP-ENV:Header>
<SOAP-ENV:Body>
  <ns2:FindAccountsListRequest xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd"
ns2:version="VERSION_2" xmlns="">
    <ns2:unitId>10055203</ns2:unitId>
  </ns2:FindAccountsListRequest>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

SOAP ответ:

```

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
  <ns2:FindAccountsListResponse xmlns:ns2="http://www.moneta.ru/schemas/
messages.xsd">
    <ns2:account>
      <ns2:id>26939899</ns2:id>
      <ns2:currency>RUB</ns2:currency>
      <ns2:balance>0</ns2:balance>
      <ns2:availableBalance>0</ns2:availableBalance>
      <ns2:type>2</ns2:type>
      <ns2:status>1</ns2:status>
      <ns2:alias>Alias</ns2:alias>
      <ns2:attribute>
        <ns2:key>paymentPasswordType</ns2:key>
        <ns2:value>SMS_SIMPLE</ns2:value>
      </ns2:attribute>
      <ns2:attribute>
        <ns2:key>paymentPasswordChallengeRequired</ns2:key>
        <ns2:value>true</ns2:value>
      </ns2:attribute>
      <ns2:attribute>
        <ns2:key>ALIAS</ns2:key>
        <ns2:value>Alias</ns2:value>
        <ns2:approved>false</ns2:approved>
        <ns2:published>false</ns2:published>
      </ns2:attribute>
    </ns2:account>
    <ns2:account>
      <ns2:id>86323445</ns2:id>
      <ns2:currency>RUB</ns2:currency>
      <ns2:balance>0</ns2:balance>
      <ns2:availableBalance>0</ns2:availableBalance>

```

```

        <ns2:type>2</ns2:type>
        <ns2:status>1</ns2:status>
        <ns2:alias>Alias</ns2:alias>
        <ns2:onSuccessfulDebitUrl>https://server.com/moneta_callback</
ns2:onSuccessfulDebitUrl>
        <ns2:onSuccessfulCreditUrl>https://server.com/moneta_callback</
ns2:onSuccessfulCreditUrl>
        <ns2:signature>QWERTY123</ns2:signature>
        <ns2:lowBalanceThreshold>-10000</ns2:lowBalanceThreshold>
        <ns2:highBalanceThreshold>500000</ns2:highBalanceThreshold>
        <ns2:prototypeAccountId>86360536</ns2:prototypeAccountId>
        <ns2:onCancelledDebitUrl>https://server.com/moneta_callback</
ns2:onCancelledDebitUrl>
        <ns2:onCancelledCreditUrl>https://server.com/moneta_callback</
ns2:onCancelledCreditUrl>
        <ns2:attribute>
            <ns2:key>paymentPasswordType</ns2:key>
            <ns2:value>STATIC</ns2:value>
        </ns2:attribute>
        <ns2:attribute>
            <ns2:key>ALIAS</ns2:key>
            <ns2:value>Alias</ns2:value>
            <ns2:approved>false</ns2:approved>
            <ns2:published>true</ns2:published>
        </ns2:attribute>
    </ns2:account>
</ns2:FindAccountsListResponse>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

JSON запрос:

```

{"Envelope": {
  "Header": {
    "Security": {
      "UsernameToken": {
        "Username": "USERNAME",
        "Password": "PASSWORD"
      }
    }
  },
  "Body": {
    "FindAccountsListRequest": {
      "version": "VERSION_2",
      "unitId": 10060344
    }
  }
}}

```

JSON ответ:

```

{"Envelope": {
  "Body": {
    "FindAccountsListResponse": {
      "account": [
        {
          "highBalanceThreshold": 500000,
          "lowBalanceThreshold": -10000,
          "status": 1,
          "alias": "Alias",
          "attribute": [
            {
              "key": "paymentPasswordType",
              "value": "SMS_SIMPLE"
            },
            {
              "key": "paymentPasswordChallengeRequired",
              "value": "true"
            }
          ]
        }
      ]
    }
  }
}}

```

```
        "key": "ALIAS",
        "value": "Alias",
        "approved": false,
        "published": false
    },
    {
        "availableBalance": 0,
        "type": 2,
        "currency": "RUB",
        "id": 30523002,
        "balance": 0
    },
    {
        "highBalanceThreshold": 500000,
        "lowBalanceThreshold": -10000,
        "status": 1,
        "alias": "Alias",
        "attribute": [
            {
                "key": "paymentPasswordType",
                "value": "STATIC"
            },
            {
                "key": "ALIAS",
                "value": "Alias",
                "approved": false,
                "published": true
            }
        ],
        "onSuccessfulCreditUrl": "",
        "availableBalance": 0,
        "type": 2,
        "onCancelledCreditUrl": "",
        "currency": "RUB",
        "id": 87198713,
        "balance": 0,
        "onCancelledDebitUrl": "",
        "prototypeAccountId": 86360536,
        "onSuccessfulDebitUrl": "",
        "signature": "QWERTY123"
    }
}

}
```

Чтение счета

SOAP запрос:

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header>
    <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-secext-1.0.xsd" SOAP-ENV:mustUnderstand="1">
      <wsse:UsernameToken xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/
oasis-200401-wss-wssecurity-utility-1.0.xsd" wsu:Id="XWSSGID-1419574022832875529758"
xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
secext-1.0.xsd">
        <wsse:Username>USERNAME</wsse:Username>
        <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-
wss-username-token-profile-1.0#PasswordText">PASSWORD</wsse:Password>
      </wsse:UsernameToken>
    </wsse:Security>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <ns2:FindAccountByIdRequest xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd"
ns2:version="VERSION_2" xmlns="">86323445</ns2:FindAccountByIdRequest>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

SOAP ответ:

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <ns2:FindAccountByIdResponse xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd">
      <ns2:account>
        <ns2:id>86323445</ns2:id>
        <ns2:currency>RUB</ns2:currency>
        <ns2:balance>0</ns2:balance>
        <ns2:availableBalance>0</ns2:availableBalance>
        <ns2:type>2</ns2:type>
        <ns2:status>1</ns2:status>
        <ns2:alias>Alias</ns2:alias>
        <ns2:onSuccessfulDebitUrl>https://server.com/moneta_callback</
ns2:onSuccessfulDebitUrl>
        <ns2:onSuccessfulCreditUrl>https://server.com/moneta_callback</
ns2:onSuccessfulCreditUrl>
        <ns2:signature>QWERTY123</ns2:signature>
        <ns2:lowBalanceThreshold>-10000</ns2:lowBalanceThreshold>
        <ns2:highBalanceThreshold>500000</ns2:highBalanceThreshold>
        <ns2:prototypeAccountId>86360536</ns2:prototypeAccountId>
        <ns2:onCancelledDebitUrl>https://server.com/moneta_callback</
ns2:onCancelledDebitUrl>
        <ns2:onCancelledCreditUrl>https://server.com/moneta_callback</
ns2:onCancelledCreditUrl>
        <ns2:attribute>
          <ns2:key>paymentPasswordType</ns2:key>
          <ns2:value>STATIC</ns2:value>
        </ns2:attribute>
        <ns2:attribute>
          <ns2:key>ALIAS</ns2:key>
          <ns2:value>Alias</ns2:value>
          <ns2:approved>false</ns2:approved>
          <ns2:published>true</ns2:published>
        </ns2:attribute>
      </ns2:account>
    </ns2:FindAccountByIdResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

JSON запрос:

```
{
  "Envelope": {
    "Header": {
      "Security": {
        "UsernameToken": {
          "Username": "USERNAME",
          "Password": "PASSWORD"
        }
      }
    },
    "Body": {
      "FindAccountByIdRequest": {
        "value": 87198713,
        "version": "VERSION_2"
      }
    }
  }
}
```

JSON ответ:

```
{
  "Envelope": {
    "Body": {
      "FindAccountByIdResponse": {
        "account": {
          "highBalanceThreshold": 500000,
          "lowBalanceThreshold": -10000,
          "status": 1,
          "alias": "Alias",

```

```
"attribute": [
    {
        "key": "paymentPasswordType",
        "value": "STATIC"
    },
    {
        "key": "ALIAS",
        "value": "Alias",
        "approved": false,
        "published": true
    }
],
"onSuccessfulCreditUrl": "",
"availableBalance": 0,
"type": 2,
"onCancelledCreditUrl": "",
"currency": "RUB",
"id": 87198713,
"balance": 0,
"onCancelledDebitUrl": "",
"prototypeAccountId": 86360536,
"onSuccessfulDebitUrl": "",
"signature": "QWERTY123"
}
}
```

Создание счета

SOAP запрос:

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header>
    <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd" SOAP-ENV:mustUnderstand="1">
      <wsse:UsernameToken xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd" wsu:Id="XWSSGID-14195740207011900660334"
        xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd">
          <wsse:Username>USERNAME</wsse:Username>
          <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-username-token-profile-1.0#PasswordText">PASSWORD</wsse:Password>
        </wsse:UsernameToken>
      </wsse:Security>
    </SOAP-ENV:Header>
    <SOAP-ENV:Body>
      <ns2:CreateAccountRequest xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd"
        xmlns="">
        <ns2:currency>RUB</ns2:currency>
        <ns2:alias>Alias</ns2:alias>
        <ns2:paymentPassword>12345</ns2:paymentPassword>
        <ns2:unitId>10055203</ns2:unitId>
        <ns2:onSuccessfulDebitUrl>https://server.com/moneta_callback</
ns2:onSuccessfulDebitUrl>
        <ns2:onSuccessfulCreditUrl>https://server.com/moneta_callback</
ns2:onSuccessfulCreditUrl>
        <ns2:signature>QWERTY001</ns2:signature>
        <ns2:lowBalanceThreshold>0</ns2:lowBalanceThreshold>
        <ns2:highBalanceThreshold>200000</ns2:highBalanceThreshold>
        <ns2:prototypeAccountId>86360536</ns2:prototypeAccountId>
        <ns2:onCancelledDebitUrl>https://server.com/moneta_callback</
ns2:onCancelledDebitUrl>
        <ns2:onCancelledCreditUrl>https://server.com/moneta_callback</
ns2:onCancelledCreditUrl>
        <ns2:attribute>
          <ns2:key>ALIAS</ns2:key>
          <ns2:value>Alias</ns2:value>
          <ns2:published>true</ns2:published>
        </ns2:attribute>
      </ns2:CreateAccountRequest>
    </SOAP-ENV:Body>
  </SOAP-ENV:Envelope>
```



```

    </ns2:CreateAccountRequest>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

SOAP ответ:

```

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <ns2:CreateAccountResponse xmlns:ns2="http://www.moneta.ru/schemas/
messages.xsd">86323445</ns2:CreateAccountResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

JSON запрос:

```

{"Envelope": {
  "Header": {
    "Security": {
      "UsernameToken": {
        "Username": "USERNAME",
        "Password": "PASSWORD"
      }
    }
  },
  "Body": {
    "CreateAccountRequest": {
      "currency": "RUB",
      "alias": "Alias",
      "paymentPassword": "12345",
      "unitId": 10060344,
      "onSuccessfulDebitUrl": "https://server.com/moneta_callback",
      "onSuccessfulCreditUrl": "https://server.com/moneta_callback",
      "signature": "QWERTY001",
      "lowBalanceThreshold": 50000,
      "highBalanceThreshold": 100000,
      "prototypeAccountId": 86360536,
      "onCancelledDebitUrl": "https://server.com/moneta_callback",
      "onCancelledCreditUrl": "https://server.com/moneta_callback",
      "attribute": [
        {
          "key": "ALIAS",
          "value": "Alias",
          "published": true
        }
      ]
    }
  }
}}

```

JSON ответ:

```

{"Envelope": {
  "Body": {
    "CreateAccountResponse": 87198713
  }
}}

```

Редактирование счета

SOAP запрос:

```

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header>
    <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-secext-1.0.xsd" SOAP-ENV:mustUnderstand="1">
      <wsse:UsernameToken xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/
oasis-200401-wss-wssecurity-utility-1.0.xsd" wsu:Id="XWSSGID-1419574021946-353601982"

```

```

    xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
secext-1.0.xsd">
        <wsse:Username>USERNAME</wsse:Username>
        <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-
wss-username-token-profile-1.0#PasswordText">PASSWORD</wsse:Password>
        </wsse:UsernameToken>
    </wsse:Security>
</SOAP-ENV:Header>
<SOAP-ENV:Body>
    <ns2:EditAccountRequest xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd"
xmlns="">
        <ns2:id>86323445</ns2:id>
        <ns2:alias>Alias</ns2:alias>
        <ns2:paymentPassword>11111</ns2:paymentPassword>
        <ns2:oldPaymentPassword>12345</ns2:oldPaymentPassword>
        <ns2:onSuccessfulDebitUrl>https://server.com/moneta_callback</
ns2:onSuccessfulDebitUrl>
        <ns2:onSuccessfulCreditUrl>https://server.com/moneta_callback</
ns2:onSuccessfulCreditUrl>
        <ns2:signature>QWERTY123</ns2:signature>
        <ns2:lowBalanceThreshold>10000</ns2:lowBalanceThreshold>
        <ns2:highBalanceThreshold>500000</ns2:highBalanceThreshold>
        <ns2:onCancelledDebitUrl>https://server.com/moneta_callback</
ns2:onCancelledDebitUrl>
        <ns2:onCancelledCreditUrl>https://server.com/moneta_callback</
ns2:onCancelledCreditUrl>
        <ns2:attribute>
            <ns2:key>ALIAS</ns2:key>
            <ns2:value>Alias</ns2:value>
            <ns2:published>true</ns2:published>
        </ns2:attribute>
    </ns2:EditAccountRequest>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

SOAP ответ:

```

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
    <ns2:EditAccountResponse xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd"/>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

JSON запрос:

```

{"Envelope": {
  "Header": {
    "Security": {
      "UsernameToken": {
        "Username": "USERNAME",
        "Password": "PASSWORD"
      }
    }
  },
  "Body": {
    "EditAccountRequest": {
      "id": 87198713,
      "alias": "Alias",
      "paymentPassword": "11111",
      "oldPaymentPassword": "12345",
      "onSuccessfulDebitUrl": "https://server.com/moneta_callback",
      "onSuccessfulCreditUrl": "https://server.com/moneta_callback",
      "signature": "QWERTY123",
      "lowBalanceThreshold": 10000,
      "highBalanceThreshold": 500000,
      "onCancelledDebitUrl": "https://server.com/moneta_callback",
      "onCancelledCreditUrl": "https://server.com/moneta_callback",
      "attribute": [
        {

```

```

    "key": "ALIAS",
    "value": "Alias",
    "published": true
  }
]
}
}
}}
```

JSON ответ:

```
{
  "Envelope": {
    "Body": {
      "EditAccountResponse": {}
    }
  }
}
```

Создание счета с SMS платежным паролем

SOAP запрос:

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header>
  <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-secext-1.0.xsd" SOAP-ENV:mustUnderstand="1">
    <wsse:UsernameToken xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/
oasis-200401-wss-wssecurity-utility-1.0.xsd" wsu:Id="XWSSGID-1419574023021-692096092"
  xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
secext-1.0.xsd">
      <wsse:Username>USERNAME</wsse:Username>
      <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-
wss-username-token-profile-1.0#PasswordText">PASSWORD</wsse:Password>
    </wsse:UsernameToken>
  </wsse:Security>
</SOAP-ENV:Header>
<SOAP-ENV:Body>
  <ns2:CreateAccountRequest xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd"
xmlns="">
    <ns2:currency>RUB</ns2:currency>
    <ns2:alias>Alias</ns2:alias>
    <ns2:paymentPasswordType>SMS_SIMPLE</ns2:paymentPasswordType>
    <ns2:unitId>10055203</ns2:unitId>
  </ns2:CreateAccountRequest>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

SOAP ответ:

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
  <ns2:CreateAccountResponse xmlns:ns2="http://www.moneta.ru/schemas/
messages.xsd">26939899</ns2:CreateAccountResponse>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

JSON запрос:

```
{
  "Envelope": {
    "Header": {
      "Security": {
        "UsernameToken": {
          "Username": "USERNAME",
          "Password": "PASSWORD"
        }
      }
    },
    "Body": {
```

```

        "CreateAccountRequest": {
            "currency": "RUB",
            "alias": "Alias",
            "paymentPasswordType": "SMS_SIMPLE",
            "unitId": 10060344
        }
    }
}

```

JSON ответ:

```

{"Envelope": {
  "Body": {
    "CreateAccountResponse": 30523002
  }
}}

```

Отсылка SMS пароля пользователю

SOAP запрос:

```

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header>
  <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-secext-1.0.xsd" SOAP-ENV:mustUnderstand="1">
    <wsse:UsernameToken xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/
oasis-200401-wss-wssecurity-utility-1.0.xsd" wsu:Id="XWSSGID-14195740239361078121633"
    xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
secext-1.0.xsd">
      <wsse:Username>USERNAME</wsse:Username>
      <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-
wss-username-token-profile-1.0#PasswordText">PASSWORD</wsse:Password>
    </wsse:UsernameToken>
  </wsse:Security>
</SOAP-ENV:Header>
<SOAP-ENV:Body>
  <ns2:GetAccountPaymentPasswordChallengeRequest xmlns:ns2="http://www.moneta.ru/
schemas/messages.xsd" xmlns="">
    <ns2:accountId>26939899</ns2:accountId>
  </ns2:GetAccountPaymentPasswordChallengeRequest>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

SOAP ответ:

```

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
  <ns2:GetAccountPaymentPasswordChallengeResponse xmlns:ns2="http://www.moneta.ru/
schemas/messages.xsd">
    <ns2:paymentPasswordChallenge>SMS</ns2:paymentPasswordChallenge>
  </ns2:GetAccountPaymentPasswordChallengeResponse>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

JSON запрос:

```

{"Envelope": {
  "Header": {
    "Security": {
      "UsernameToken": {
        "Username": "USERNAME",
        "Password": "PASSWORD"
      }
    }
  },
  "Body": {
    "GetAccountPaymentPasswordChallengeRequest": {

```

```

        "accountId": 30523002
      }
    }
  }
}

```

JSON ответ:

```

{
  "Envelope": {
    "Body": {
      "GetAccountPaymentPasswordChallengeResponse": {
        "paymentPasswordChallenge": "SMS"
      }
    }
  }
}

```

Создание счета

SOAP запрос:

```

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header>
    <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd" SOAP-ENV:mustUnderstand="1">
      <wsse:UsernameToken xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd" wsu:Id="XWSSGID-14195740207011900660334"
        xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd">
        <wsse:Username>USERNAME</wsse:Username>
        <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-username-token-profile-1.0#PasswordText">PASSWORD</wsse:Password>
      </wsse:UsernameToken>
    </wsse:Security>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <ns2:CreateAccountRequest xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd"
      xmlns="">
      <ns2:currency>RUB</ns2:currency>
      <ns2:alias>Alias</ns2:alias>
      <ns2:paymentPassword>12345</ns2:paymentPassword>
      <ns2:unitId>10055203</ns2:unitId>
      <ns2:onSuccessfulDebitUrl>https://server.com/moneta_callback</
    ns2:onSuccessfulDebitUrl>
      <ns2:onSuccessfulCreditUrl>https://server.com/moneta_callback</
    ns2:onSuccessfulCreditUrl>
      <ns2:signature>QWERTY001</ns2:signature>
      <ns2:lowBalanceThreshold>0</ns2:lowBalanceThreshold>
      <ns2:highBalanceThreshold>200000</ns2:highBalanceThreshold>
      <ns2:prototypeAccountId>86360536</ns2:prototypeAccountId>
      <ns2:onCancelledDebitUrl>https://server.com/moneta_callback</
    ns2:onCancelledDebitUrl>
      <ns2:onCancelledCreditUrl>https://server.com/moneta_callback</
    ns2:onCancelledCreditUrl>
      <ns2:attribute>
        <ns2:key>ALIAS</ns2:key>
        <ns2:value>Alias</ns2:value>
        <ns2:published>true</ns2:published>
      </ns2:attribute>
    </ns2:CreateAccountRequest>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

SOAP ответ:

```

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <ns2:CreateAccountResponse xmlns:ns2="http://www.moneta.ru/schemas/
    messages.xsd">86323445</ns2:CreateAccountResponse>
  </SOAP-ENV:Body>

```

```
</SOAP-ENV:Envelope>
```

JSON запрос:

```
{
  "Envelope": {
    "Header": {
      "Security": {
        "UsernameToken": {
          "Username": "USERNAME",
          "Password": "PASSWORD"
        }
      }
    },
    "Body": {
      "CreateAccountRequest": {
        "currency": "RUB",
        "alias": "Alias",
        "paymentPassword": "12345",
        "unitId": 10060344,
        "onSuccessfulDebitUrl": "https://server.com/moneta_callback",
        "onSuccessfulCreditUrl": "https://server.com/moneta_callback",
        "signature": "QWERTY001",
        "lowBalanceThreshold": 50000,
        "highBalanceThreshold": 100000,
        "prototypeAccountId": 86360536,
        "onCancelledDebitUrl": "https://server.com/moneta_callback",
        "onCancelledCreditUrl": "https://server.com/moneta_callback",
        "attribute": [
          {
            "key": "ALIAS",
            "value": "Alias",
            "published": true
          }
        ]
      }
    }
  }
}
```

JSON ответ:

```
{
  "Envelope": {
    "Body": {
      "CreateAccountResponse": 87198713
    }
  }
}
```

Редактирование счета

SOAP запрос:

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header>
    <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd" SOAP-ENV:mustUnderstand="1">
      <wsse:UsernameToken xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd" wsu:Id="XWSSGID-1419574021946-353601982"
        xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd">
        <wsse:Username>USERNAME</wsse:Username>
        <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-username-token-profile-1.0#PasswordText">PASSWORD</wsse:Password>
      </wsse:UsernameToken>
    </wsse:Security>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <ns2:EditAccountRequest xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd"
      xmlns="">
      <ns2:id>86323445</ns2:id>
      <ns2:alias>Alias</ns2:alias>
    </ns2:EditAccountRequest>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

```

        <ns2:paymentPassword>11111</ns2:paymentPassword>
        <ns2:oldPaymentPassword>12345</ns2:oldPaymentPassword>
        <ns2:onSuccessfulDebitUrl>https://server.com/moneta_callback</
ns2:onSuccessfulDebitUrl>
        <ns2:onSuccessfulCreditUrl>https://server.com/moneta_callback</
ns2:onSuccessfulCreditUrl>
        <ns2:signature>QWERTY123</ns2:signature>
        <ns2:lowBalanceThreshold>10000</ns2:lowBalanceThreshold>
        <ns2:highBalanceThreshold>500000</ns2:highBalanceThreshold>
        <ns2:onCancelledDebitUrl>https://server.com/moneta_callback</
ns2:onCancelledDebitUrl>
        <ns2:onCancelledCreditUrl>https://server.com/moneta_callback</
ns2:onCancelledCreditUrl>
        <ns2:attribute>
            <ns2:key>ALIAS</ns2:key>
            <ns2:value>Alias</ns2:value>
            <ns2:published>true</ns2:published>
        </ns2:attribute>
    </ns2:EditAccountRequest>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

SOAP ответ:

```

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
    <ns2:EditAccountResponse xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd"/>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

JSON запрос:

```

{"Envelope": {
  "Header": {
    "Security": {
      "UsernameToken": {
        "Username": "USERNAME",
        "Password": "PASSWORD"
      }
    }
  },
  "Body": {
    "EditAccountRequest": {
      "id": 87198713,
      "alias": "Alias",
      "paymentPassword": "11111",
      "oldPaymentPassWord": "12345",
      "onSuccessfulDebitUrl": "https://server.com/moneta_callback",
      "onSuccessfulCreditUrl": "https://server.com/moneta_callback",
      "signature": "QWERTY123",
      "lowBalanceThreshold": 10000,
      "highBalanceThreshold": 500000,
      "onCancelledDebitUrl": "https://server.com/moneta_callback",
      "onCancelledCreditUrl": "https://server.com/moneta_callback",
      "attribute": [
        {
          "key": "ALIAS",
          "value": "Alias",
          "published": true
        }
      ]
    }
  }
}}

```

JSON ответ:

```

{"Envelope": {

```

```

    "Body": {
      "EditAccountResponse": {}
    }
  }
}

```

Чтение счета

Чтение списка счетов

SOAP запрос:

```

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header>
    <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd" SOAP-ENV:mustUnderstand="1">
      <wsse:UsernameToken xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd" wsu:Id="XWSSGID-1419574025786-400683759"
        xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd">
        <wsse:Username>USERNAME</wsse:Username>
        <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-username-token-profile-1.0#PasswordText">PASSWORD</wsse:Password>
      </wsse:UsernameToken>
    </wsse:Security>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <ns2:FindAccountsListRequest xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd"
      ns2:version="VERSION_2" xmlns="">
      <ns2:unitId>10055203</ns2:unitId>
    </ns2:FindAccountsListRequest>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

SOAP ответ:

```

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <ns2:FindAccountsListResponse xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd">
      <ns2:account>
        <ns2:id>26939899</ns2:id>
        <ns2:currency>RUB</ns2:currency>
        <ns2:balance>0</ns2:balance>
        <ns2:availableBalance>0</ns2:availableBalance>
        <ns2:type>2</ns2:type>
        <ns2:status>1</ns2:status>
        <ns2:alias>Alias</ns2:alias>
        <ns2:attribute>
          <ns2:key>paymentPasswordType</ns2:key>
          <ns2:value>SMS_SIMPLE</ns2:value>
        </ns2:attribute>
        <ns2:attribute>
          <ns2:key>paymentPasswordChallengeRequired</ns2:key>
          <ns2:value>true</ns2:value>
        </ns2:attribute>
        <ns2:attribute>
          <ns2:key>ALIAS</ns2:key>
          <ns2:value>Alias</ns2:value>
          <ns2:approved>false</ns2:approved>
          <ns2:published>false</ns2:published>
        </ns2:attribute>
      </ns2:account>
      <ns2:account>
        <ns2:id>86323445</ns2:id>
        <ns2:currency>RUB</ns2:currency>
        <ns2:balance>0</ns2:balance>
        <ns2:availableBalance>0</ns2:availableBalance>

```



```

        <ns2:type>2</ns2:type>
        <ns2:status>1</ns2:status>
        <ns2:alias>Alias</ns2:alias>
        <ns2:onSuccessfulDebitUrl>https://server.com/moneta_callback</
ns2:onSuccessfulDebitUrl>
        <ns2:onSuccessfulCreditUrl>https://server.com/moneta_callback</
ns2:onSuccessfulCreditUrl>
        <ns2:signature>QWERTY123</ns2:signature>
        <ns2:lowBalanceThreshold>-10000</ns2:lowBalanceThreshold>
        <ns2:highBalanceThreshold>500000</ns2:highBalanceThreshold>
        <ns2:prototypeAccountId>86360536</ns2:prototypeAccountId>
        <ns2:onCancelledDebitUrl>https://server.com/moneta_callback</
ns2:onCancelledDebitUrl>
        <ns2:onCancelledCreditUrl>https://server.com/moneta_callback</
ns2:onCancelledCreditUrl>
        <ns2:attribute>
            <ns2:key>paymentPasswordType</ns2:key>
            <ns2:value>STATIC</ns2:value>
        </ns2:attribute>
        <ns2:attribute>
            <ns2:key>ALIAS</ns2:key>
            <ns2:value>Alias</ns2:value>
            <ns2:approved>false</ns2:approved>
            <ns2:published>true</ns2:published>
        </ns2:attribute>
    </ns2:account>
</ns2:FindAccountsListResponse>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

JSON запрос:

```

{"Envelope": {
  "Header": {
    "Security": {
      "UsernameToken": {
        "Username": "USERNAME",
        "Password": "PASSWORD"
      }
    }
  },
  "Body": {
    "FindAccountsListRequest": {
      "version": "VERSION_2",
      "unitId": 10060344
    }
  }
}}

```

JSON ответ:

```

{"Envelope": {
  "Body": {
    "FindAccountsListResponse": {
      "account": [
        {
          "highBalanceThreshold": 500000,
          "lowBalanceThreshold": -10000,
          "status": 1,
          "alias": "Alias",
          "attribute": [
            {
              "key": "paymentPasswordType",
              "value": "SMS_SIMPLE"
            },
            {
              "key": "paymentPasswordChallengeRequired",
              "value": "true"
            }
          ]
        }
      ]
    }
  }
}}

```

```
        "key": "ALIAS",
        "value": "Alias",
        "approved": false,
        "published": false
    },
    {
        "availableBalance": 0,
        "type": 2,
        "currency": "RUB",
        "id": 30523002,
        "balance": 0
    },
    {
        "highBalanceThreshold": 500000,
        "lowBalanceThreshold": -10000,
        "status": 1,
        "alias": "Alias",
        "attribute": [
            {
                "key": "paymentPasswordType",
                "value": "STATIC"
            },
            {
                "key": "ALIAS",
                "value": "Alias",
                "approved": false,
                "published": true
            }
        ],
        "onSuccessfulCreditUrl": "",
        "availableBalance": 0,
        "type": 2,
        "onCancelledCreditUrl": "",
        "currency": "RUB",
        "id": 87198713,
        "balance": 0,
        "onCancelledDebitUrl": "",
        "prototypeAccountId": 86360536,
        "onSuccessfulDebitUrl": "",
        "signature": "QWERTY123"
    }
}
}
```

Чтение счета

SOAP запрос:

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header>
    <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-secext-1.0.xsd" SOAP-ENV:mustUnderstand="1">
      <wsse:UsernameToken xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/
oasis-200401-wss-wssecurity-utility-1.0.xsd" wsu:Id="XWSSGID-1419574022832875529758"
xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
secext-1.0.xsd">
        <wsse:Username>USERNAME</wsse:Username>
        <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-
wss-username-token-profile-1.0#PasswordText">PASSWORD</wsse:Password>
      </wsse:UsernameToken>
    </wsse:Security>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <ns2:FindAccountByIdRequest xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd"
ns2:version="VERSION_2" xmlns="">86323445</ns2:FindAccountByIdRequest>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

SOAP ответ:

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <ns2:FindAccountByIdResponse xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd">
      <ns2:account>
        <ns2:id>86323445</ns2:id>
        <ns2:currency>RUB</ns2:currency>
        <ns2:balance>0</ns2:balance>
        <ns2:availableBalance>0</ns2:availableBalance>
        <ns2:type>2</ns2:type>
        <ns2:status>1</ns2:status>
        <ns2:alias>Alias</ns2:alias>
        <ns2:onSuccessfulDebitUrl>https://server.com/moneta_callback</
ns2:onSuccessfulDebitUrl>
        <ns2:onSuccessfulCreditUrl>https://server.com/moneta_callback</
ns2:onSuccessfulCreditUrl>
        <ns2:signature>QWERTY123</ns2:signature>
        <ns2:lowBalanceThreshold>-10000</ns2:lowBalanceThreshold>
        <ns2:highBalanceThreshold>500000</ns2:highBalanceThreshold>
        <ns2:prototypeAccountId>86360536</ns2:prototypeAccountId>
        <ns2:onCancelledDebitUrl>https://server.com/moneta_callback</
ns2:onCancelledDebitUrl>
        <ns2:onCancelledCreditUrl>https://server.com/moneta_callback</
ns2:onCancelledCreditUrl>
        <ns2:attribute>
          <ns2:key>paymentPasswordType</ns2:key>
          <ns2:value>STATIC</ns2:value>
        </ns2:attribute>
        <ns2:attribute>
          <ns2:key>ALIAS</ns2:key>
          <ns2:value>Alias</ns2:value>
          <ns2:approved>false</ns2:approved>
          <ns2:published>true</ns2:published>
        </ns2:attribute>
      </ns2:account>
    </ns2:FindAccountByIdResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

JSON запрос:

```
{
  "Envelope": {
    "Header": {
      "Security": {
        "UsernameToken": {
          "Username": "USERNAME",
          "Password": "PASSWORD"
        }
      }
    },
    "Body": {
      "FindAccountByIdRequest": {
        "value": 87198713,
        "version": "VERSION_2"
      }
    }
  }
}
```

JSON ответ:

```
{
  "Envelope": {
    "Body": {
      "FindAccountByIdResponse": {
        "account": {
          "highBalanceThreshold": 500000,
          "lowBalanceThreshold": -10000,
          "status": 1,
          "alias": "Alias",

```

```
"attribute": [
    {
        "key": "paymentPasswordType",
        "value": "STATIC"
    },
    {
        "key": "ALIAS",
        "value": "Alias",
        "approved": false,
        "published": true
    }
],
"onSuccessfulCreditUrl": "",
"availableBalance": 0,
"type": 2,
"onCancelledCreditUrl": "",
"currency": "RUB",
"id": 87198713,
"balance": 0,
"onCancelledDebitUrl": "",
"prototypeAccountId": 86360536,
"onSuccessfulDebitUrl": "",
"signature": "QWERTY123"
}
}
```

Счет с SMS платежным паролем

Создание счета с SMS платежным паролем

SOAP запрос:

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header>
  <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-secext-1.0.xsd" SOAP-ENV:mustUnderstand="1">
    <wsse:UsernameToken xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/
oasis-200401-wss-wssecurity-utility-1.0.xsd" wsu:Id="XWSSGID-1419574023021-692096092"
  xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
secext-1.0.xsd">
    <wsse:Username>USERNAME</wsse:Username>
    <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-
wss-username-token-profile-1.0#PasswordText">PASSWORD</wsse:Password>
  </wsse:UsernameToken>
</wsse:Security>
</SOAP-ENV:Header>
<SOAP-ENV:Body>
  <ns2:CreateAccountRequest xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd"
xmlns="">
    <ns2:currency>RUB</ns2:currency>
    <ns2:alias>Alias</ns2:alias>
    <ns2:paymentPasswordType>SMS_SIMPLE</ns2:paymentPasswordType>
    <ns2:unitId>10055203</ns2:unitId>
  </ns2:CreateAccountRequest>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

SOAP ответ:

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
    <ns2:CreateAccountResponse xmlns:ns2="http://www.moneta.ru/schemas/
messages.xsd">26939899</ns2:CreateAccountResponse>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

JSON запрос:

```
{
  "Envelope": {
    "Header": {
      "Security": {
        "UsernameToken": {
          "Username": "USERNAME",
          "Password": "PASSWORD"
        }
      }
    },
    "Body": {
      "CreateAccountRequest": {
        "currency": "RUB",
        "alias": "Alias",
        "paymentPasswordType": "SMS_SIMPLE",
        "unitId": 10060344
      }
    }
  }
}
```

JSON ответ:

```
{
  "Envelope": {
    "Body": {
      "CreateAccountResponse": 30523002
    }
  }
}
```

Отсылка SMS пароля пользователю

SOAP запрос:

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header>
    <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd" SOAP-ENV:mustUnderstand="1">
      <wsse:UsernameToken xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd" wsu:Id="XWSSGID-14195740239361078121633"
        xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd">
        <wsse:Username>USERNAME</wsse:Username>
        <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-username-token-profile-1.0#PasswordText">PASSWORD</wsse:Password>
      </wsse:UsernameToken>
    </wsse:Security>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <ns2:GetAccountPaymentPasswordChallengeRequest xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd" xmlns="">
      <ns2:accountId>26939899</ns2:accountId>
    </ns2:GetAccountPaymentPasswordChallengeRequest>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

SOAP ответ:

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <ns2:GetAccountPaymentPasswordChallengeResponse xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd">
      <ns2:paymentPasswordChallenge>SMS</ns2:paymentPasswordChallenge>
    </ns2:GetAccountPaymentPasswordChallengeResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

JSON запрос:

```
{
  "Envelope": {
    "Header": {
      "Security": {
        "UsernameToken": {
          "Username": "USERNAME",
          "Password": "PASSWORD"
        }
      }
    },
    "Body": {
      "GetAccountPaymentPasswordChallengeRequest": {
        "accountId": 30523002
      }
    }
  }
}
```

JSON ответ:

```
{
  "Envelope": {
    "Body": {
      "GetAccountPaymentPasswordChallengeResponse": {
        "paymentPasswordChallenge": "SMS"
      }
    }
  }
}
```

Блокировка/разблокирование счета

Блокировка счета

SOAP запрос:

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header>
    <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd" SOAP-ENV:mustUnderstand="1">
      <wsse:UsernameToken
        xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd"
        wsu:Id="XWSSGID-15058958299241023459384"
        xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd">
        <wsse:Username>USERNAME</wsse:Username>
        <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-username-token-profile-1.0#PasswordText">PASSWORD</wsse:Password>
      </wsse:UsernameToken>
    </wsse:Security>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <ns2:BlockAccountRequest xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd"
      xmlns="">
      <ns2:id>12345678</ns2:id>
    </ns2:BlockAccountRequest>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

SOAP ответ:

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <ns2:BlockAccountResponse xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd"/>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

JSON запрос:

```
{
  "Envelope": {
    "Header": {
      "Security": {
        "UsernameToken": {
          "Username": "USERNAME",
          "Password": "PASSWORD"
        }
      }
    },
    "Body": {
      "BlockAccountRequest": {
        "id": 12345678
      }
    }
  }
}
```

JSON ответ:

```
{
  "Envelope": {
    "Body": {
      "BlockAccountResponse": {}
    }
  }
}
```

Разблокирование счета

SOAP запрос:

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header>
    <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd" SOAP-ENV:mustUnderstand="1">
      <wsse:UsernameToken
        xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd"
        wsu:Id="XWSSGID-15058958315831594550495"
        xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd">
        <wsse:Username>USERNAME</wsse:Username>
        <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-username-token-profile-1.0#PasswordText">PASSWORD</wsse:Password>
      </wsse:UsernameToken>
    </wsse:Security>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <ns2:UnblockAccountRequest xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd" xmlns="">
      <ns2:id>12345678</ns2:id>
      <ns2:secretAnswer>ANSWER</ns2:secretAnswer>
      <ns2:paymentPassword>
        <ns2:paymentPassword>12345</ns2:paymentPassword>
      </ns2:paymentPassword>
    </ns2:UnblockAccountRequest>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

SOAP ответ:

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
```

```

<ns2:UnblockAccountResponse xmlns:ns2="http://www.moneta.ru/schemas/
messages.xsd"/>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

JSON запрос:

```

{
  "Envelope": {
    "Header": {
      "Security": {
        "UsernameToken": {
          "Username": "USERNAME",
          "Password": "PASSWORD"
        }
      }
    },
    "Body": {
      "UnblockAccountRequest": {
        "id": 12345678,
        "secretAnswer": "ANSWER",
        "paymentPassword": {
          "paymentPassword": "12345"
        }
      }
    }
  }
}

```

JSON ответ:

```

{
  "Envelope": {
    "Body": {
      "UnblockAccountResponse": {}
    }
  }
}

```

Информация о доступных платежных системах

SOAP запрос:

```

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header>
    <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-
wss-wssecurity-secext-1.0.xsd" SOAP-ENV:mustUnderstand="1">
      <wsse:UsernameToken
        xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-utility-1.0.xsd"
        wsu:Id="XWSSGID-1504855068128-1992070225"
        xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-
wss-wssecurity-secext-1.0.xsd">
        <wsse:Username>USERNAME</wsse:Username>
        <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/
oasis-200401-wss-username-token-profile-1.0#PasswordText">PASSWORD</wsse:Password>
      </wsse:UsernameToken>
    </wsse:Security>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <ns2:PaymentSystemInfoRequest xmlns:ns2="http://www.moneta.ru/schemas/
messages.xsd" xmlns="">
      <ns2:accountId>12345678</ns2:accountId>
      <ns2:amount>100</ns2:amount>
    </ns2:PaymentSystemInfoRequest>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```


SOAP ответ:

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <ns2:PaymentSystemInfoResponse xmlns:ns2="http://www.moneta.ru/schemas/
messages.xsd">
      <ns2:items>
        <ns2:unitId>11111111</ns2:unitId>
        <ns2:name>VISA, MasterCard</ns2:name>
        <ns2:icon>visa_mc_mir.jpg</ns2:icon>
        <ns2:logicalGroup>Cards</ns2:logicalGroup>
        <ns2:infoUrl>
          <ns2:href></ns2:href>
          <ns2:text></ns2:text>
        </ns2:infoUrl>
        <ns2:referenceData>
          <ns2:period>1 min.</ns2:period>
          <ns2:percentage>0,0%</ns2:percentage>
          <ns2:sourceFee>0</ns2:sourceFee>
          <ns2:sourceFeeExt>0</ns2:sourceFeeExt>
          <ns2:targetFee>0.01</ns2:targetFee>
          <ns2:targetFeeExt>0</ns2:targetFeeExt>
        </ns2:referenceData>
        <ns2:infoTariff>
          <ns2:targetRangeCurrency>RUB</ns2:targetRangeCurrency>
          <ns2:targetAmountMax>10000000</ns2:targetAmountMax>
        </ns2:infoTariff>
        <ns2:currencies>RUB</ns2:currencies>
        <ns2:psAccountIds>317</ns2:psAccountIds>
      </ns2:items>
    </ns2:PaymentSystemInfoResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

JSON запрос:

```
{
  "Header": {
    "Security": {
      "UsernameToken": {
        "Username": "USERNAME",
        "Password": "PASSWORD"
      }
    }
  },
  "Envelope": {
    "Body": {
      "PaymentSystemInfoRequest": {
        "amount": 100,
        "accountId": 12345678
      }
    }
  }
}
```

JSON ответ:

```
{
  "Envelope": {
    "Body": {
      "PaymentSystemInfoResponse": {
        "items": [
          {
            "icon": "visa_mc_mir.jpg",
            "currencies": "RUB",
            "logicalGroup": "Cards",
            "name": "VISA, MasterCard",
            "infoTariff": {
              "targetAmountMax": 10000000,

```

```

        "targetRangeCurrency": "RUB"
      },
      "referenceData": {
        "targetFeeExt": 0,
        "percentage": "0,0%",
        "sourceFeeExt": 0,
        "period": "1 min.",
        "targetFee": 0.01,
        "sourceFee": 0
      },
      "infoUrl": {
        "text": "",
        "href": ""
      },
      "psAccountIds": "317",
      "unitId": 11111111
    }
  ]
}
}
}
}
}

```

Работа с документами пользователя

Чтение документов

SOAP запрос:

```

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header>
    <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-secext-1.0.xsd" SOAP-ENV:mustUnderstand="1">
      <wsse:UsernameToken xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/
oasis-200401-wss-wssecurity-utility-1.0.xsd" wsu:Id="XWSSGID-141752599958782043320"
xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
secext-1.0.xsd">
        <wsse:Username>USERNAME</wsse:Username>
        <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-
wss-username-token-profile-1.0#PasswordText">PASSWORD</wsse:Password>
      </wsse:UsernameToken>
    </wsse:Security>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <ns2:FindProfileDocumentsRequest xmlns:ns2="http://www.moneta.ru/schemas/
messages.xsd" xmlns="">
      <ns2:unitId>10054789</ns2:unitId>
    </ns2:FindProfileDocumentsRequest>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

SOAP ответ:

```

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <ns2:FindProfileDocumentsResponse xmlns:ns2="http://www.moneta.ru/schemas/
messages.xsd">
      <ns2:document>
        <ns2:id>9996</ns2:id>
        <ns2:type>PASSPORT</ns2:type>
        <ns2:attribute>
          <ns2:key>series</ns2:key>
          <ns2:value>8804</ns2:value>
          <ns2:approved>false</ns2:approved>
        </ns2:attribute>
        <ns2:attribute>

```

```

        <ns2:key>issuer</ns2:key>
        <ns2:value>issuer</ns2:value>
        <ns2:approved>>false</ns2:approved>
    </ns2:attribute>
    <ns2:attribute>
        <ns2:key>department</ns2:key>
        <ns2:value>123-001</ns2:value>
        <ns2:approved>>false</ns2:approved>
    </ns2:attribute>
    <ns2:attribute>
        <ns2:key>issued</ns2:key>
        <ns2:value>2004-07-19</ns2:value>
        <ns2:approved>>false</ns2:approved>
    </ns2:attribute>
    <ns2:attribute>
        <ns2:key>number</ns2:key>
        <ns2:value>424000</ns2:value>
        <ns2:approved>>false</ns2:approved>
    </ns2:attribute>
    <ns2:attribute>
        <ns2:key>modificationdate</ns2:key>
        <ns2:value>2014-12-02T16:12:48.000+03:00</ns2:value>
        <ns2:approved>>true</ns2:approved>
    </ns2:attribute>
    <ns2:hasAttachedFiles>>false</ns2:hasAttachedFiles>
</ns2:document>
</ns2:FindProfileDocumentsResponse>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

JSON запрос:

```

{"Envelope": {
  "Header": {
    "Security": {
      "UsernameToken": {
        "Username": "USERNAME",
        "Password": "PASSWORD"
      }
    }
  },
  "Body": {
    "FindProfileDocumentsRequest": {
      "unitId": 10060344
    }
  }
}}

```

JSON ответ:

```

{"Envelope": {
  "Body": {
    "FindProfileDocumentsResponse": {
      "document": [
        {
          "id": 11334,
          "hasAttachedFiles": false,
          "attribute": [
            {
              "approved": false,
              "value": "8804",
              "key": "series"
            },
            {
              "approved": false,
              "value": "issuer",
              "key": "issuer"
            },
            {
              "approved": false,

```

```

    "value": "123-001",
    "key": "department"
  },
  {
    "approved": false,
    "value": "2004-07-19",
    "key": "issued"
  },
  {
    "approved": false,
    "value": "424000",
    "key": "number"
  },
  {
    "approved": true,
    "value": "2015-06-02T15:48:48.000+03:00",
    "key": "modificationdate"
  }
],
"type": "PASSPORT"
}
]
}
}
}
```

Создание документа

SOAP запрос:

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header>
  <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-secext-1.0.xsd" SOAP-ENV:mustUnderstand="1">
    <wsse:UsernameToken xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/
oasis-200401-wss-wssecurity-utility-1.0.xsd" wsu:Id="XWSSGID-1417525998490-750226304"
  xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
secext-1.0.xsd">
      <wsse:Username>USERNAME</wsse:Username>
      <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-
wss-username-token-profile-1.0#PasswordText">PASSWORD</wsse:Password>
    </wsse:UsernameToken>
  </wsse:Security>
</SOAP-ENV:Header>
<SOAP-ENV:Body>
  <ns2:CreateProfileDocumentRequest xmlns:ns2="http://www.moneta.ru/schemas/
messages.xsd" xmlns="">
    <ns2:type>PASSPORT</ns2:type>
    <ns2:attribute>
      <ns2:key>SERIES</ns2:key>
      <ns2:value>8803</ns2:value>
    </ns2:attribute>
    <ns2:attribute>
      <ns2:key>NUMBER</ns2:key>
      <ns2:value>424001</ns2:value>
    </ns2:attribute>
    <ns2:attribute>
      <ns2:key>ISSUER</ns2:key>
      <ns2:value>issuer</ns2:value>
    </ns2:attribute>
    <ns2:attribute>
      <ns2:key>ISSUED</ns2:key>
      <ns2:value>2004-05-18</ns2:value>
    </ns2:attribute>
    <ns2:attribute>
      <ns2:key>DEPARTMENT</ns2:key>
      <ns2:value>123-002</ns2:value>
    </ns2:attribute>
    <ns2:unitId>10054789</ns2:unitId>
  </ns2:CreateProfileDocumentRequest>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

```
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

SOAP ответ:

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
  <ns2:CreateProfileDocumentResponse xmlns:ns2="http://www.moneta.ru/schemas/
messages.xsd">
    <ns2:id>9996</ns2:id>
  </ns2:CreateProfileDocumentResponse>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

JSON запрос:

```
{ "Envelope": {
  "Header": {
    "Security": {
      "UsernameToken": {
        "Username": "USERNAME",
        "Password": "PASSWORD"
      }
    }
  },
  "Body": {
    "CreateProfileDocumentRequest": {
      "type": "PASSPORT",
      "attribute": [
        {
          "key": "SERIES",
          "value": "8803"
        },
        {
          "key": "NUMBER",
          "value": "424001"
        },
        {
          "key": "ISSUER",
          "value": "issuer"
        },
        {
          "key": "ISSUED",
          "value": "2004-05-18"
        },
        {
          "key": "DEPARTMENT",
          "value": "123-002"
        }
      ],
      "unitId": 10060344
    }
  }
}
```

JSON ответ:

```
{ "Envelope": {
  "Body": {
    "CreateProfileDocumentResponse": {
      "id": 11334
    }
  }
}
```

Редактирование документа

SOAP запрос:

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header>
  <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-secext-1.0.xsd" SOAP-ENV:mustUnderstand="1">
    <wsse:UsernameToken xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/
oasis-200401-wss-wssecurity-utility-1.0.xsd" wsu:Id="XWSSGID-14175259990301283414556"
  xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
secext-1.0.xsd">
      <wsse:Username>USERNAME</wsse:Username>
      <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-
wss-username-token-profile-1.0#PasswordText">PASSWORD</wsse:Password>
    </wsse:UsernameToken>
  </wsse:Security>
</SOAP-ENV:Header>
<SOAP-ENV:Body>
  <ns2:EditProfileDocumentRequest xmlns:ns2="http://www.moneta.ru/schemas/
messages.xsd" xmlns="">
    <ns2:id>9996</ns2:id>
    <ns2:attribute>
      <ns2:key>SERIES</ns2:key>
      <ns2:value>8804</ns2:value>
    </ns2:attribute>
    <ns2:attribute>
      <ns2:key>NUMBER</ns2:key>
      <ns2:value>424000</ns2:value>
    </ns2:attribute>
    <ns2:attribute>
      <ns2:key>ISSUER</ns2:key>
      <ns2:value>issuer</ns2:value>
    </ns2:attribute>
    <ns2:attribute>
      <ns2:key>ISSUED</ns2:key>
      <ns2:value>2004-07-19</ns2:value>
    </ns2:attribute>
    <ns2:attribute>
      <ns2:key>DEPARTMENT</ns2:key>
      <ns2:value>123-001</ns2:value>
    </ns2:attribute>
    <ns2:unitId>10054789</ns2:unitId>
  </ns2:EditProfileDocumentRequest>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

SOAP ответ:

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
  <ns2:EditProfileDocumentResponse xmlns:ns2="http://www.moneta.ru/schemas/
messages.xsd"/>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

JSON запрос:

```
{ "Envelope": {
  "Header": {
    "Security": {
      "UsernameToken": {
        "Username": "USERNAME",
        "Password": "PASSWORD"
      }
    }
  },
  "Body": {
    "EditProfileDocumentRequest": {
```

```

        "id": 11334,
        "attribute": [
            {
                "key": "SERIES",
                "value": "8804"
            },
            {
                "key": "NUMBER",
                "value": "424000"
            },
            {
                "key": "ISSUER",
                "value": "issuer"
            },
            {
                "key": "ISSUED",
                "value": "2004-07-19"
            },
            {
                "key": "DEPARTMENT",
                "value": "123-001"
            }
        ],
        "unitId": 10060344
    }
}
}
}

```

JSON ответ:

```

{"Envelope": {
  "Body": {
    "EditProfileDocumentResponse": {}
  }
}}

```

Работа с банковскими реквизитами

Чтение банковских реквизитов

SOAP запрос:

```

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header>
  <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-secext-1.0.xsd" SOAP-ENV:mustUnderstand="1">
    <wsse:UsernameToken xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/
oasis-200401-wss-wssecurity-utility-1.0.xsd" wsu:Id="XWSSGID-14175260021691023607342"
    xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
secext-1.0.xsd">
      <wsse:Username>USERNAME</wsse:Username>
      <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-
wss-username-token-profile-1.0#PasswordText">PASSWORD</wsse:Password>
    </wsse:UsernameToken>
  </wsse:Security>
</SOAP-ENV:Header>
<SOAP-ENV:Body>
  <ns2:FindBankAccountsRequest xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd"
  ns2:version="VERSION_2" xmlns="">
    <ns2:unitId>10054789</ns2:unitId>
  </ns2:FindBankAccountsRequest>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

SOAP ответ:

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
  <ns2:FindBankAccountsResponse xmlns:ns2="http://www.moneta.ru/schemas/
messages.xsd">
    <ns2:bankAccount>
      <ns2:id>10000914</ns2:id>
      <ns2:attribute>
        <ns2:key>bank</ns2:key>
        <ns2:value>Bank name</ns2:value>
        <ns2:approved>false</ns2:approved>
        <ns2:published>false</ns2:published>
      </ns2:attribute>
      <ns2:attribute>
        <ns2:key>account</ns2:key>
        <ns2:value>12345678901234567890</ns2:value>
        <ns2:approved>false</ns2:approved>
        <ns2:published>false</ns2:published>
      </ns2:attribute>
      <ns2:attribute>
        <ns2:key>bik</ns2:key>
        <ns2:value>040037469</ns2:value>
        <ns2:approved>false</ns2:approved>
        <ns2:published>false</ns2:published>
      </ns2:attribute>
      <ns2:attribute>
        <ns2:key>is_international</ns2:key>
        <ns2:value>false</ns2:value>
        <ns2:approved>false</ns2:approved>
        <ns2:published>false</ns2:published>
      </ns2:attribute>
      <ns2:attribute>
        <ns2:key>corr_account</ns2:key>
        <ns2:value>301018109000000000469</ns2:value>
        <ns2:approved>false</ns2:approved>
        <ns2:published>false</ns2:published>
      </ns2:attribute>
    </ns2:bankAccount>
  </ns2:FindBankAccountsResponse>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

JSON запрос:

```
{
  "Envelope": {
    "Header": {
      "Security": {
        "UsernameToken": {
          "Username": "USERNAME",
          "Password": "PASSWORD"
        }
      }
    },
    "Body": {
      "FindBankAccountsRequest": {
        "version": "VERSION_2",
        "unitId": 10060344
      }
    }
  }
}
```

JSON ответ:

```
{
  "Envelope": {
    "Body": {
      "FindBankAccountsResponse": {
        "bankAccount": [
          {

```



```
"id": 10001185,  
  "attribute": [  
    {  
      "approved": false,  
      "value": "Bank name",  
      "published": false,  
      "key": "bank"  
    },  
    {  
      "approved": false,  
      "value": "12345678901234567897",  
      "published": false,  
      "key": "account"  
    },  
    {  
      "approved": false,  
      "value": "044585214",  
      "published": false,  
      "key": "bik"  
    },  
    {  
      "approved": false,  
      "value": "30101810800000000214",  
      "published": false,  
      "key": "corr_account"  
    },  
    {  
      "approved": false,  
      "value": "false",  
      "published": false,  
      "key": "is_international"  
    }  
  ]  
}  
  
}  
  
}
```

Создание банковских реквизитов

SOAP запрос:

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header>
  <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-secext-1.0.xsd" SOAP-ENV:mustUnderstand="1">
    <wsse:UsernameToken xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/
oasis-200401-wss-wssecurity-utility-1.0.xsd" wsu:Id="XWSSGID-1417526001372-735973564"
xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
secext-1.0.xsd">
      <wsse:Username>USERNAME</wsse:Username>
      <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-
wss-username-token-profile-1.0#PasswordText">PASSWORD</wsse:Password>
    </wsse:UsernameToken>
  </wsse:Security>
</SOAP-ENV:Header>
<SOAP-ENV:Body>
  <ns2:CreateBankAccountRequest xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd"
xmlns="">
    <ns2:attribute>
      <ns2:key>BIK</ns2:key>
      <ns2:value>040037469</ns2:value>
    </ns2:attribute>
    <ns2:attribute>
      <ns2:key>ACCOUNT</ns2:key>
      <ns2:value>12345678901234567890</ns2:value>
    </ns2:attribute>
    <ns2:unitId>10054789</ns2:unitId>
  </ns2:CreateBankAccountRequest>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

```
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

SOAP ответ:

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
  <ns2:CreateBankAccountResponse xmlns:ns2="http://www.moneta.ru/schemas/
messages.xsd">
    <ns2:id>10000914</ns2:id>
  </ns2:CreateBankAccountResponse>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

JSON запрос:

```
{
  "Envelope": {
    "Header": {
      "Security": {
        "UsernameToken": {
          "Username": "USERNAME",
          "Password": "PASSWORD"
        }
      }
    },
    "Body": {
      "CreateBankAccountRequest": {
        "attribute": [
          {
            "key": "BIK",
            "value": "044585214"
          },
          {
            "key": "ACCOUNT",
            "value": "12345678901234567897"
          }
        ],
        "unitId": 10060344
      }
    }
  }
}
```

JSON ответ:

```
{
  "Envelope": {
    "Body": {
      "CreateBankAccountResponse": {
        "id": 10001185
      }
    }
  }
}
```

Редактирование банковских реквизитов

SOAP запрос:

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header>
  <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-secext-1.0.xsd" SOAP-ENV:mustUnderstand="1">
    <wsse:UsernameToken xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/
oasis-200401-wss-wssecurity-utility-1.0.xsd" wsu:Id="XWSSGID-1417526001803-917345112"
      xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-secext-1.0.xsd">
      <wsse:Username>USERNAME</wsse:Username>
      <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-
wss-username-token-profile-1.0#PasswordText">PASSWORD</wsse:Password>
```

```

        </wsse:UsernameToken>
    </wsse:Security>
</SOAP-ENV:Header>
<SOAP-ENV:Body>
    <ns2:EditBankAccountRequest xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd"
    xmlns="">
        <ns2:id>10000914</ns2:id>
        <ns2:attribute>
            <ns2:key>BIK</ns2:key>
            <ns2:value>040037469</ns2:value>
        </ns2:attribute>
        <ns2:attribute>
            <ns2:key>ACCOUNT</ns2:key>
            <ns2:value>12345678901234567890</ns2:value>
        </ns2:attribute>
        <ns2:unitId>10054789</ns2:unitId>
    </ns2:EditBankAccountRequest>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

SOAP ответ:

```

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
    <ns2:EditBankAccountResponse xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd"/>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

JSON запрос:

```

{"Envelope": {
  "Header": {
    "Security": {
      "UsernameToken": {
        "Username": "USERNAME",
        "Password": "PASSWORD"
      }
    }
  },
  "Body": {
    "EditBankAccountRequest": {
      "id": 10001185,
      "attribute": [
        {
          "key": "BIK",
          "value": "044585214"
        },
        {
          "key": "ACCOUNT",
          "value": "12345678901234567897"
        }
      ],
      "unitId": 10060344
    }
  }
}}

```

JSON ответ:

```

{"Envelope": {
  "Body": {
    "EditBankAccountResponse": {}
  }
}}

```

Работа с отчетами

Чтение списка отчетов

SOAP запрос:

```
<soapenv:Envelope xmlns:mes="http://www.moneta.ru/schemas/messages.xsd"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Header>
    <wsse:Security soapenv:mustUnderstand="1" xmlns:wsse="http://docs.oasis-
open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd">
      <wsse:UsernameToken wsu:Id="UsernameToken-1" xmlns:wsu="http://docs.oasis-
open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd">
        <wsse:Username>USERNAME</wsse:Username>
        <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/
oasis-200401-wss-username-token-profile-1.0#PasswordText">PASSWORD</wsse:Password>
      </wsse:UsernameToken>
    </wsse:Security>
  </soapenv:Header>
  <soapenv:Body>
    <mes:FindReportsRequest>
      <mes:unitId>12345678</mes:unitId>
      <mes:year>2016</mes:year>
    </mes:FindReportsRequest>
  </soapenv:Body>
</soapenv:Envelope>
```

SOAP ответ:

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <ns2:FindReportsResponse xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd">
      <ns2:report>
        <ns2:id>1084</ns2:id>
        <ns2:name>Act of acceptance</ns2:name>
        <ns2:unitId>12345678</ns2:unitId>
        <ns2:typeId>3</ns2:typeId>
        <ns2:reportInstance>
          <ns2:id>2010</ns2:id>
          <ns2:reportId>1084</ns2:reportId>
          <ns2:year>2016</ns2:year>
          <ns2:month>4</ns2:month>
          <ns2:url>https://moneta.ru/downloadReport.htm?
reportInstanceId=2010&date=2016-06-29_12-21&
publicId=82739741-3968-4068-b620-328cdf4ce351&
signature=4f93d845fe70d23eafd062b941d1f547be29a6ee6c778ea2fcdbc53ac7db5ff
          </ns2:url>
          <ns2:urlExpirationDate>2016-06-29T12:21:00.000+03:00</
ns2:urlExpirationDate>
        </ns2:reportInstance>
      </ns2:report>
    </ns2:FindReportsResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

JSON запрос:

```
{
  "Envelope": {
    "Header": {
      "Security": {
        "UsernameToken": {
          "Username": "USERNAME",
          "Password": "PASSWORD"
        }
      }
    }
  }
}
```

```

    },
    "Body": {
      "FindReportsRequest": {
        "unitId": 12345678,
        "year": 2016
      }
    }
  }
}

```

JSON ответ:

```

{
  "Envelope": {
    "Body": {
      "FindReportsResponse": {
        "report": [
          {
            "id": 1084,
            "reportInstance": [
              {
                "id": 2010,
                "month": 4,
                "year": 2016,
                "reportId": 1084,
                "url": "https://moneta.ru/downloadReport.htm?
reportInstanceId=2010&date=2016-06-29_12-04&
publicId=82739741-3968-4068-b620-328cdf4ce351&
signature=2757ef81cedb2b37ae14136a2f05b698541b6084f2b7be09ccce4de2ad83d726",
                "urlExpirationDate": "2016-06-29T12:04:00.000+03:00",
                "attribute": []
              }
            ],
            "name": "Act of acceptance",
            "attribute": [],
            "typeId": 3,
            "unitId": 12345678
          }
        ]
      }
    }
  }
}

```

Подтверждение сотового телефона

Запрос на получение кода подтверждения

SOAP запрос:

```

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header>
  <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-secext-1.0.xsd" SOAP-ENV:mustUnderstand="1">
    <wsse:UsernameToken xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/
oasis-200401-wss-wssecurity-utility-1.0.xsd" wsu:Id="XWSSGID-1417526002418-988211293"
xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
secext-1.0.xsd">
      <wsse:Username>USERNAME</wsse:Username>
      <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-
wss-username-token-profile-1.0#PasswordText">PASSWORD</wsse:Password>
    </wsse:UsernameToken>
  </wsse:Security>
</SOAP-ENV:Header>
<SOAP-ENV:Body>
  <ns2:ApprovePhoneSendConfirmationRequest xmlns:ns2="http://www.moneta.ru/schemas/
messages.xsd" xmlns="">

```

```

        <ns2:unitId>10054789</ns2:unitId>
        <ns2:text>{CODE}</ns2:text>
    </ns2:ApprovePhoneSendConfirmationRequest>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

SOAP ответ:

```

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
    <ns2:ApprovePhoneSendConfirmationResponse xmlns:ns2="http://www.moneta.ru/schemas/
messages.xsd">
        <ns2:phoneNumber>+71234567890</ns2:phoneNumber>
    </ns2:ApprovePhoneSendConfirmationResponse>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

JSON запрос:

```

{"Envelope": {
  "Header": {
    "Security": {
      "UsernameToken": {
        "Username": "USERNAME",
        "Password": "PASSWORD"
      }
    }
  },
  "Body": {
    "ApprovePhoneSendConfirmationRequest": {
      "unitId": 10060344,
      "text": "{CODE}"
    }
  }
}}

```

JSON ответ:

```

{"Envelope": {
  "Body": {
    "ApprovePhoneSendConfirmationResponse": {
      "phoneNumber": "+71234567890"
    }
  }
}}

```

Подтверждение сотового телефона

SOAP запрос:

```

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header>
    <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-secext-1.0.xsd" SOAP-ENV:mustUnderstand="1">
        <wsse:UsernameToken xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/
oasis-200401-wss-wssecurity-utility-1.0.xsd" wsu:Id="XWSSGID-1417526003898-834549441"
xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
secext-1.0.xsd">
            <wsse:Username>USERNAME</wsse:Username>
            <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-
wss-username-token-profile-1.0#PasswordText">PASSWORD</wsse:Password>
        </wsse:UsernameToken>
    </wsse:Security>
</SOAP-ENV:Header>
<SOAP-ENV:Body>
    <ns2:ApprovePhoneApplyCodeRequest xmlns:ns2="http://www.moneta.ru/schemas/
messages.xsd" xmlns="">

```

```

        <ns2:unitId>10054789</ns2:unitId>
        <ns2:confirmationCode>999739</ns2:confirmationCode>
    </ns2:ApprovePhoneApplyCodeRequest>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

SOAP ответ:

```

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
    <ns2:ApprovePhoneApplyCodeResponse xmlns:ns2="http://www.moneta.ru/schemas/
messages.xsd"/>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

JSON запрос:

```

{"Envelope": {
  "Header": {
    "Security": {
      "UsernameToken": {
        "Username": "USERNAME",
        "Password": "PASSWORD"
      }
    }
  },
  "Body": {
    "ApprovePhoneApplyCodeRequest": {
      "unitId": 10060344,
      "confirmationCode": "27924"
    }
  }
}}

```

JSON ответ:

```

{"Envelope": {
  "Body": {
    "ApprovePhoneApplyCodeResponse": {}
  }
}}

```

SimplifiedIdentificationRequest (AsyncRequest) - упрощённая идентификация

Запрос на проведение Упрощённой идентификации пользователя.

Подробнее: [SimplifiedIdentification Endpoint](#).

Примеры запросов

SOAP запрос (начинаем запрос):

```

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header>
    <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-
wss-wssecurity-secext-1.0.xsd" SOAP-ENV:mustUnderstand="1">
      <wsse:UsernameToken
        xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-utility-1.0.xsd"
        wsu:Id="XwSSGID-1503645719981488841737"
        xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-
wss-wssecurity-secext-1.0.xsd">
          <wsse:Username>USERNAME</wsse:Username>

```

```

        <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/
oasis-200401-wss-username-token-profile-1.0#PasswordText">PASSWORD</wsse:Password>
    </wsse:UsernameToken>
</wsse:Security>
</SOAP-ENV:Header>
<SOAP-ENV:Body>
    <ns2:AsyncRequest xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd"
xmlns="">
        <ns2:SimplifiedIdentificationRequest>
            <ns2:unitId>11111</ns2:unitId>
            <ns2:personalInformation>
                <ns2:profile>
                    <ns2:attribute>
                        <ns2:key>LAST_NAME</ns2:key>
                        <ns2:value>Last name</ns2:value>
                    </ns2:attribute>
                    <ns2:attribute>
                        <ns2:key>FIRST_NAME</ns2:key>
                        <ns2:value>First name</ns2:value>
                    </ns2:attribute>
                    <ns2:attribute>
                        <ns2:key>MIDDLE_INITIAL_NAME</ns2:key>
                        <ns2:value>Middle initial</ns2:value>
                    </ns2:attribute>
                    <ns2:attribute>
                        <ns2:key>CELL_PHONE</ns2:key>
                        <ns2:value>79001234567</ns2:value>
                    </ns2:attribute>
                    <ns2:attribute>
                        <ns2:key>SNILS</ns2:key>
                        <ns2:value>000-000-000 00</ns2:value>
                    </ns2:attribute>
                    <ns2:attribute>
                        <ns2:key>INN</ns2:key>
                        <ns2:value>00000000000000</ns2:value>
                    </ns2:attribute>
                </ns2:profile>
                <ns2:document>
                    <ns2:id>123</ns2:id>
                    <ns2:type>PASSPORT</ns2:type>
                    <ns2:attribute>
                        <ns2:key>SERIES</ns2:key>
                        <ns2:value>0000</ns2:value>
                    </ns2:attribute>
                    <ns2:attribute>
                        <ns2:key>NUMBER</ns2:key>
                        <ns2:value>000000</ns2:value>
                    </ns2:attribute>
                    <ns2:attribute>
                        <ns2:key>ISSUER</ns2:key>
                        <ns2:value>Issuer name</ns2:value>
                    </ns2:attribute>
                    <ns2:attribute>
                        <ns2:key>ISSUED</ns2:key>
                        <ns2:value>2017-08-15</ns2:value>
                    </ns2:attribute>
                    <ns2:attribute>
                        <ns2:key>DEPARTMENT</ns2:key>
                        <ns2:value>000-000</ns2:value>
                    </ns2:attribute>
                </ns2:document>
            </ns2:personalInformation>
        </ns2:SimplifiedIdentificationRequest>
    </ns2:AsyncRequest>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

JSON запрос (начинаем запрос):

```
{
```



```

"Envelope": {
  "Header": {
    "Security": {
      "UsernameToken": {
        "Username": "USERNAME",
        "Password": "PASSWORD"
      }
    }
  },
  "Body": {
    "AsyncRequest": {
      "SimplifiedIdentificationRequest": {
        "unitId": 11111,
        "personalInformation": {
          "profile": {
            "attribute": [
              {
                "value": "Last name",
                "key": "LAST_NAME"
              },
              {
                "value": "First name",
                "key": "FIRST_NAME"
              },
              {
                "value": "Middle initial",
                "key": "MIDDLE_INITIAL_NAME"
              },
              {
                "value": "79001234567",
                "key": "CELL_PHONE"
              },
              {
                "value": "000-000-000 00",
                "key": "SNILS"
              },
              {
                "value": "000000000000",
                "key": "INN"
              }
            ]
          }
        },
        "document": {
          "id": 123,
          "type": "PASSPORT",
          "attribute": [
            {
              "value": "0000",
              "key": "SERIES"
            },
            {
              "value": "000000",
              "key": "NUMBER"
            },
            {
              "value": "Issuer name",
              "key": "ISSUER"
            },
            {
              "value": "2017-08-15",
              "key": "ISSUED"
            },
            {
              "value": "000-000",
              "key": "DEPARTMENT"
            }
          ]
        }
      }
    }
  }
}

```

```

    }
  }
}

```

SOAP запрос (получение результата по asyncId):

```

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header>
    <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd" SOAP-ENV:mustUnderstand="1">
      <wsse:UsernameToken
        xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd"
        wsu:Id="XWSSGID-15035778598551166901355"
        xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd">
          <wsse:Username>USERNAME</wsse:Username>
          <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-username-token-profile-1.0#PasswordText">PASSWORD</wsse:Password>
        </wsse:UsernameToken>
      </wsse:Security>
    </SOAP-ENV:Header>
    <SOAP-ENV:Body>
      <ns2:AsyncRequest xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd"
        xmlns="">
        <ns2:asyncId>333333</ns2:asyncId>
      </ns2:AsyncRequest>
    </SOAP-ENV:Body>
  </SOAP-ENV:Envelope>

```

SOAP ответ (success=true):

```

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <ns2:AsyncResponse xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd">
      <ns2:SimplifiedIdentificationResponse>
        <ns2:success>true</ns2:success>
        <ns2:personalInformation>
          <ns2:profile>
            <ns2:attribute>
              <ns2:key>last_name</ns2:key>
              <ns2:value>Last name</ns2:value>
              <ns2:approved>true</ns2:approved>
              <ns2:published>false</ns2:published>
            </ns2:attribute>
            <ns2:attribute>
              <ns2:key>snils</ns2:key>
              <ns2:value>000-000-000 00</ns2:value>
              <ns2:approved>true</ns2:approved>
              <ns2:published>false</ns2:published>
            </ns2:attribute>
            <ns2:attribute>
              <ns2:key>first_name</ns2:key>
              <ns2:value>First name</ns2:value>
              <ns2:approved>true</ns2:approved>
              <ns2:published>false</ns2:published>
            </ns2:attribute>
            <ns2:attribute>
              <ns2:key>inn</ns2:key>
              <ns2:value>0000000000000</ns2:value>
              <ns2:approved>true</ns2:approved>
              <ns2:published>false</ns2:published>
            </ns2:attribute>
            <ns2:attribute>
              <ns2:key>cell_phone</ns2:key>
              <ns2:value>79001234567</ns2:value>
              <ns2:approved>true</ns2:approved>
              <ns2:published>false</ns2:published>
            </ns2:attribute>
          </ns2:profile>
        </ns2:personalInformation>
      </ns2:SimplifiedIdentificationResponse>
    </ns2:AsyncResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

```

        <ns2:attribute>
          <ns2:key>middle_initial_name</ns2:key>
          <ns2:value>Middle initial</ns2:value>
          <ns2:approved>true</ns2:approved>
          <ns2:published>>false</ns2:published>
        </ns2:attribute>
        <ns2:attribute>
          <ns2:key>unitid</ns2:key>
          <ns2:value>11111</ns2:value>
          <ns2:approved>>false</ns2:approved>
          <ns2:published>>false</ns2:published>
        </ns2:attribute>
        <ns2:attribute>
          <ns2:key>profileid</ns2:key>
          <ns2:value>22222</ns2:value>
          <ns2:approved>>false</ns2:approved>
          <ns2:published>>false</ns2:published>
        </ns2:attribute>
      </ns2:profile>
    <ns2:document>
      <ns2:id>123</ns2:id>
      <ns2:type>PASSPORT</ns2:type>
      <ns2:attribute>
        <ns2:key>series</ns2:key>
        <ns2:value>0000</ns2:value>
        <ns2:approved>true</ns2:approved>
        <ns2:published>>false</ns2:published>
      </ns2:attribute>
      <ns2:attribute>
        <ns2:key>issuer</ns2:key>
        <ns2:value>Issuer name</ns2:value>
        <ns2:approved>>false</ns2:approved>
        <ns2:published>>false</ns2:published>
      </ns2:attribute>
      <ns2:attribute>
        <ns2:key>department</ns2:key>
        <ns2:value>000-000</ns2:value>
        <ns2:approved>>false</ns2:approved>
        <ns2:published>>false</ns2:published>
      </ns2:attribute>
      <ns2:attribute>
        <ns2:key>issued</ns2:key>
        <ns2:value>2017-08-15</ns2:value>
        <ns2:approved>>false</ns2:approved>
        <ns2:published>>false</ns2:published>
      </ns2:attribute>
      <ns2:attribute>
        <ns2:key>number</ns2:key>
        <ns2:value>000000</ns2:value>
        <ns2:approved>true</ns2:approved>
        <ns2:published>>false</ns2:published>
      </ns2:attribute>
      <ns2:attribute>
        <ns2:key>modificationdate</ns2:key>
        <ns2:value>2017-08-08T13:53:22.000+03:00</ns2:value>
        <ns2:approved>true</ns2:approved>
      </ns2:attribute>
      <ns2:hasAttachedFiles>>false</ns2:hasAttachedFiles>
    </ns2:document>
  </ns2:personalInformation>
</ns2:SimplifiedIdentificationResponse>
</ns2:AsyncResponse>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

SOAP ответ (success=false):

```

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>

```

```

<ns2:AsyncResponse xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd">
  <ns2:SimplifiedIdentificationResponse>
    <ns2:success>false</ns2:success>
    <ns2:error>Customer data did not pass ESIA check.</ns2:error>
    <ns2:errorCode>500.7.3</ns2:errorCode>
    <ns2:personalInformation>
      <ns2:profile>
        <ns2:attribute>
          <ns2:key>last_name</ns2:key>
          <ns2:value>Last name</ns2:value>
          <ns2:approved>false</ns2:approved>
          <ns2:published>false</ns2:published>
        </ns2:attribute>
        <ns2:attribute>
          <ns2:key>snils</ns2:key>
          <ns2:value>000-000-000 00</ns2:value>
          <ns2:approved>false</ns2:approved>
          <ns2:published>false</ns2:published>
        </ns2:attribute>
        <ns2:attribute>
          <ns2:key>first_name</ns2:key>
          <ns2:value>First name</ns2:value>
          <ns2:approved>false</ns2:approved>
          <ns2:published>false</ns2:published>
        </ns2:attribute>
        <ns2:attribute>
          <ns2:key>inn</ns2:key>
          <ns2:value>000000000000</ns2:value>
          <ns2:approved>false</ns2:approved>
          <ns2:published>false</ns2:published>
        </ns2:attribute>
        <ns2:attribute>
          <ns2:key>cell_phone</ns2:key>
          <ns2:value>79001234567</ns2:value>
          <ns2:approved>true</ns2:approved>
          <ns2:published>false</ns2:published>
        </ns2:attribute>
        <ns2:attribute>
          <ns2:key>middle_initial_name</ns2:key>
          <ns2:value>Middle initial</ns2:value>
          <ns2:approved>false</ns2:approved>
          <ns2:published>false</ns2:published>
        </ns2:attribute>
        <ns2:attribute>
          <ns2:key>unitid</ns2:key>
          <ns2:value>11111</ns2:value>
          <ns2:approved>false</ns2:approved>
          <ns2:published>false</ns2:published>
        </ns2:attribute>
        <ns2:attribute>
          <ns2:key>profileid</ns2:key>
          <ns2:value>22222</ns2:value>
          <ns2:approved>false</ns2:approved>
          <ns2:published>false</ns2:published>
        </ns2:attribute>
      </ns2:profile>
    </ns2:personalInformation>
    <ns2:document>
      <ns2:id>123</ns2:id>
      <ns2:type>PASSPORT</ns2:type>
      <ns2:attribute>
        <ns2:key>series</ns2:key>
        <ns2:value>0000</ns2:value>
        <ns2:approved>false</ns2:approved>
        <ns2:published>false</ns2:published>
      </ns2:attribute>
      <ns2:attribute>
        <ns2:key>issuer</ns2:key>
        <ns2:value>Issuer name</ns2:value>
        <ns2:approved>false</ns2:approved>
        <ns2:published>false</ns2:published>
      </ns2:attribute>
    </ns2:document>
  </ns2:SimplifiedIdentificationResponse>
</ns2:AsyncResponse>

```

```

        <ns2:attribute>
          <ns2:key>department</ns2:key>
          <ns2:value>000-000</ns2:value>
          <ns2:approved>false</ns2:approved>
          <ns2:published>false</ns2:published>
        </ns2:attribute>
        <ns2:attribute>
          <ns2:key>issued</ns2:key>
          <ns2:value>22017-08-15</ns2:value>
          <ns2:approved>false</ns2:approved>
          <ns2:published>false</ns2:published>
        </ns2:attribute>
        <ns2:attribute>
          <ns2:key>number</ns2:key>
          <ns2:value>000000</ns2:value>
          <ns2:approved>false</ns2:approved>
          <ns2:published>false</ns2:published>
        </ns2:attribute>
        <ns2:attribute>
          <ns2:key>modificationdate</ns2:key>
          <ns2:value>2017-08-17T10:23:15.000+03:00</ns2:value>
          <ns2:approved>true</ns2:approved>
        </ns2:attribute>
        <ns2:hasAttachedFiles>false</ns2:hasAttachedFiles>
      </ns2:document>
    </ns2:personalInformation>
  </ns2:SimplifiedIdentificationResponse>
</ns2:AsyncResponse>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

JSON запрос (получение результата по asyncId):

```

{
  "Envelope": {
    "Header": {
      "Security": {
        "UsernameToken": {
          "Username": "USERNAME",
          "Password": "PASSWORD"
        }
      }
    },
    "Body": {
      "AsyncRequest": {
        "asyncId": 333333
      }
    }
  }
}

```

JSON ответ (success=true):

```

{
  "Envelope": {
    "Body": {
      "AsyncResponse": {
        "SimplifiedIdentificationResponse": {
          "success": true,
          "personalInformation": {
            "profile": {
              "attribute": [
                {
                  "approved": true,
                  "value": "Last name",
                  "published": false,
                  "key": "last_name"
                },
                {
                  "approved": true,

```

```

        "value": "000-000-000 00",
        "published": false,
        "key": "snils"
    },
    {
        "approved": true,
        "value": "First name",
        "published": false,
        "key": "first_name"
    },
    {
        "approved": true,
        "value": "000000000000",
        "published": false,
        "key": "inn"
    },
    {
        "approved": true,
        "value": "79001234567",
        "published": false,
        "key": "cell_phone"
    },
    {
        "approved": true,
        "value": "Middle initial",
        "published": false,
        "key": "middle_initial_name"
    },
    {
        "approved": false,
        "value": "11111",
        "published": false,
        "key": "unitid"
    },
    {
        "approved": false,
        "value": "22222",
        "published": false,
        "key": "profileid"
    }
]
},
"document": {
    "id": 123,
    "type": "PASSPORT",
    "hasAttachedFiles": false,
    "attribute": [
        {
            "approved": true,
            "value": "0000",
            "published": false,
            "key": "series"
        },
        {
            "approved": false,
            "value": "Issuer name",
            "published": false,
            "key": "issuer"
        },
        {
            "approved": false,
            "value": "000-000",
            "published": false,
            "key": "department"
        },
        {
            "approved": false,
            "value": "2017-08-15",
            "published": false,
            "key": "issued"
        }
    ]
}

```

```
{
  {
    "approved": true,
    "value": "000000",
    "published": false,
    "key": "number"
  },
  {
    "approved": true,
    "value": "2017-08-08T13:53:22.000+03:00",
    "key": "modificationdate"
  }
]
}
}
}
}
}
}
```

JSON ответ (success=false):

```
{
  "Envelope": {
    "Body": {
      "AsyncResponse": {
        "SimplifiedIdentificationResponse": {
          "success": false,
          "error": "Customer data did not pass ESIA check.",
          "errorCode": "500.7.3",
          "personalInformation": {
            "document": {
              "id": 123,
              "hasAttachedFiles": false,
              "attribute": [
                {
                  "approved": false,
                  "value": "0000",
                  "published": false,
                  "key": "series"
                },
                {
                  "approved": false,
                  "value": "Issuer name",
                  "published": false,
                  "key": "issuer"
                },
                {
                  "approved": false,
                  "value": "000-000",
                  "published": false,
                  "key": "department"
                },
                {
                  "approved": false,
                  "value": "2017-08-15",
                  "published": false,
                  "key": "issued"
                },
                {
                  "approved": false,
                  "value": "000000",
                  "published": false,
                  "key": "number"
                },
                {
                  "approved": true,
                  "value": "2017-08-17T10:23:15.000+03:00",
                  "key": "modificationdate"
                }
              ]
            }
          }
        }
      }
    }
  }
}
```

```
    ],  
    "type": "PASSPORT"  
  },  
  "profile": {  
    "attribute": [  
      {  
        "approved": false,  
        "value": "Last name",  
        "published": false,  
        "key": "last_name"  
      },  
      {  
        "approved": false,  
        "value": "000-000-000 00",  
        "published": false,  
        "key": "snils"  
      },  
      {  
        "approved": false,  
        "value": "First name",  
        "published": false,  
        "key": "first_name"  
      },  
      {  
        "approved": false,  
        "value": "00000000000000",  
        "published": false,  
        "key": "inn"  
      },  
      {  
        "approved": true,  
        "value": "79001234567",  
        "published": false,  
        "key": "cell_phone"  
      },  
      {  
        "approved": false,  
        "value": "Middle initial",  
        "published": false,  
        "key": "middle_initial_name"  
      },  
      {  
        "approved": false,  
        "value": "11111",  
        "published": false,  
        "key": "unitid"  
      },  
      {  
        "approved": false,  
        "value": "22222",  
        "published": false,  
        "key": "profileid"  
      }  
    ]  
  }  
}  
  
}  
  
}  
  
}  
  
}
```

Глава

6

Примеры кода

- [SOAP запрос с помощью CURL](#)
- [JSON запрос с помощью CURL](#)
- [Пример Java API для XML Web Services \(JAX-WS\)](#)
- [Пример Java \(SAAJ\) для SOAP with Attachments](#)
- [SOAP пример на C#](#)
- [SOAP пример на PHP](#)
- [SOAP пример на Python](#)

SOAP запрос с помощью CURL

Для тестирования MONETA.MerchantAPI Вы можете использовать программу cURL.

1. Сохраните текст XML запроса в файле request.xml.

- Если Вы планируете использовать идентификатор пользователя и пароль для авторизации, то в запрос следует добавить заголовок.

Следующий пример показывает, как получить информацию о пользователе по номеру счета:

```
<?xml version="1.0" encoding="UTF-8"?>
<soap:Envelope
  xmlns="http://www.moneta.ru/schemas/messages.xsd"
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Header>
    <wsse:Security soap:mustUnderstand="1"
      xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-
        wss-wssecurity-secext-1.0.xsd">
      <wsse:UsernameToken wsu:Id="UsernameToken-31877484"
        xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-
          wss-wssecurity-utility-1.0.xsd">
        <wsse:Username>user_name</wsse:Username>
        <wsse:Password
          Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-username-
            token-profile-1.0#PasswordText">password</wsse:Password>
        </wsse:UsernameToken>
      </wsse:Security>
    </soap:Header>
    <soap:Body>
      <FindProfileInfoByAccountIdRequest>12345678</
        FindProfileInfoByAccountIdRequest>
    </soap:Body>
  </soap:Envelope>
```

- Если для авторизации Вы будете использовать клиентский сертификат, то дополнительные заголовки в запросе не нужны.

Тот же самый запрос с использованием клиентского сертификата:

```
<?xml version="1.0" encoding="UTF-8"?>
<soap:Envelope
  xmlns="http://www.moneta.ru/schemas/messages.xsd"
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <FindProfileInfoByAccountIdRequest>12345678</
FindProfileInfoByAccountIdRequest>
  </soap:Body>
</soap:Envelope>
```

2. Сделайте запрос к MONETA.MerchantAPI. В зависимости от Вашего метода авторизации используйте нужную cURL команду.

- Если Вы используете идентификатор пользователя и пароль для авторизации, то выполните команду:

```
curl --verbose --http1.0 --show-error --data-binary "@request.xml" --output
"response.xml" --header "Content-type: text/xml; charset=UTF-8" --header
"SOAPAction: \"\"\" --insecure https://www.moneta.ru/services
```

- Если Вы используете клиентский сертификат, то выполните команду:

```
curl --verbose --http1.0 --show-error --data-binary "@request.xml" --output
"response.xml" --cert "certificate.pem" --key "private.key" --header "Content-
type: text/xml; charset=UTF-8" --header "SOAPAction: \"\"\" --insecure https://
www.moneta.ru:8443/services/x509
```

Эти команды сохраняют ответ в файл response.xml.

Пример ответа:

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <ns2:FindProfileInfoByAccountIdResponse
      xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd">
      <ns2:accountId>12345678</ns2:accountId>
      <ns2:currency>RUB</ns2:currency>
      <ns2:profile>
        <ns2:attribute>
          <ns2:key>last_name</ns2:key>
          <ns2:value>moneta.ru</ns2:value>
          <ns2:approved>true</ns2:approved>
        </ns2:attribute>
        <ns2:attribute>
          <ns2:key>first_name</ns2:key>
          <ns2:value>support</ns2:value>
          <ns2:approved>true</ns2:approved>
        </ns2:attribute>
        <ns2:attribute>
          <ns2:key>unitid</ns2:key>
          <ns2:value>1256</ns2:value>
        </ns2:attribute>
      </ns2:profile>
    </ns2:FindProfileInfoByAccountIdResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

JSON запрос с помощью CURL

Для тестирования MONETA.MerchantAPI Вы можете использовать программу cURL.

1. Сохраните текст JSON запроса в файле request.json.

Если Вы планируете использовать идентификатор пользователя и пароль для авторизации, то в запрос следует добавить заголовок.

Следующий пример показывает, как получить информацию о пользователе по номеру счета:

```
{
  "Envelope": {
    "Header": {
      "Security": {
        "UsernameToken": {
          "Username": "USERNAME",
          "Password": "PASSWORD"
        }
      }
    },
    "Body": {
      "FindProfileInfoByAccountIdRequest": {
        "value": 12345678,
        "version": "VERSION_2"
      }
    }
  }
}
```

2. Сделайте запрос к MONETA.MerchantAPI.

Если Вы используете идентификатор пользователя и пароль для авторизации, то выполните команду:

```
curl --verbose --http1.0 --show-error --data-binary "@request.json" --output
  "response.json" --header "Content-type: application/json;charset=UTF-8" --insecure
  https://www.moneta.ru/services
```

Ответ сохранен в файл response.json.

Пример ответа:

```
{
  "Envelope": {
    "Body": {
      "FindProfileInfoByAccountIdResponse": {
        "accountId": 12345678,
        "profile": {
          "attribute": [
            {
              "approved": true,
              "value": "Last name",
              "published": true,
              "key": "last_name"
            },
            {
              "approved": true,
              "value": "First name",
              "published": true,
              "key": "first_name"
            },
            {
              "value": "1000",
              "key": "unitid"
            },
            {
              "value": "1008",
              "key": "profileid"
            }
          ]
        },
        "currency": "RUB"
      }
    }
  }
}
```

Пример Java API для XML Web Services (JAX-WS)

Данный пример показывает, как создать Java классы и получить баланс счета по номеру счета.

Создание Java классов

Запустите команду для генерации Java классов по wsdl файлу:

```
wsimport -clientjar moneta.jar https://service.moneta.ru:8443/services/x509.wsdl
```

Данная команда создаст файл moneta.jar с Java классами, которые можно использовать в Вашем приложении.

Код на Java

После генерации классов Вы сможете использовать их для создания SOAP запросов и обработки ответов.

1. Убедитесь, что созданный файл moneta.jar находится в classpath Вашего приложения.
2. Сделайте импорт классов из файла moneta.jar:

```
import ru.moneta.schemas.messages.*;
import ru.moneta.services.*;
```

3. Создайте экземпляр FindAccountByIdRequest класса. Укажите номер счета, например 12345678:

```
FindAccountByIdRequest msg = new FindAccountByIdRequest();
msg.setValue(12345678L);
```

4. Создайте экземпляр MessagesService класса, чтобы послать запрос к MONETA.MerchantAPI:

```
MessagesService service = new MessagesService();
```

5. Вызовите метод getMessagesSoap11:

```
Messages port = service.getMessagesSoap11();
```

6. Вызовите метод findAccountById и передайте экземпляр запроса в качестве параметра. Затем получите ответ и вызовите метод getBalance:

```
FindAccountByIdResponse result = port.findAccountById(msg);
Number balance = result.getAccount().getBalance();
```

Полный текст примера:

```
package ru.moneta.soap.wsimport;

import ru.moneta.schemas.messages.*;
import ru.moneta.services.*;

public class Main {

    public static void main(String[] args) {

        FindAccountByIdRequest msg = new FindAccountByIdRequest();
        msg.setValue(12345678L);

        MessagesService service = new MessagesService();
        Messages port = service.getMessagesSoap11();
        FindAccountByIdResponse result = port.findAccountById(msg);
        Number balance = result.getAccount().getBalance();
        System.out.print(balance);
    }
}
```

Авторизация клиентским сертификатом

Для того чтобы авторизоваться клиентским сертификатом Вы должны получить сертификат в Moneta.Ru.

1. Получите клиентский сертификат и добавьте его в хранилище ключей PKCS12. Подробнее смотрите [Получение клиентского сертификата](#).
2. Укажите хранилище ключей при запуске Java. Используйте следующие аргументы:

```
-Djavax.net.ssl.keyStore=keystore.p12 -Djavax.net.ssl.keyStoreType=pkcs12 -
Djavax.net.ssl.keyStorePassword=keystore_password
```

Пример Java (SAAJ) для SOAP with Attachments

1. Добавьте импорт необходимых классов:

```
import javax.xml.namespace.QName;
import javax.xml.soap.*;

import javax.xml.transform.TransformerFactory;
import javax.xml.transform.Transformer;
import javax.xml.transform.Source;

import javax.xml.transform.stream.StreamResult;
```

2. Создайте экземпляр класса SOAPConnection:

```
SOAPConnectionFactory soapConnFactory = SOAPConnectionFactory.newInstance();
SOAPConnection connection = soapConnFactory.createConnection();
```

3. Создайте SOAP запрос.

- a. Создайте экземпляр класса SOAPMessage:

```
MessageFactory messageFactory = MessageFactory.newInstance();
SOAPMessage message = messageFactory.createMessage();
```

- b. Создайте envelope, body, и header Вашего сообщения:

```
SOAPPart soapPart = message.getSOAPPart();
SOAPEnvelope envelope = soapPart.getEnvelope();
SOAPBody body = envelope.getBody();
SOAPHeader header = envelope.getHeader();
```

- c. В данном примере для авторизации мы используем идентификатор пользователя и пароль. Укажем в заголовке *username* и *password*:

```
SOAPElement security = header.addChildElement("Security", "wsse",
    "http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
    secext-1.0.xsd");

SOAPElement usernameToken = security.addChildElement("UsernameToken", "wsse");
usernameToken.addAttribute(new QName("xmlns:wsu"), "http://docs.oasis-open.org/
wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd");

SOAPElement username = usernameToken.addChildElement("Username", "wsse");
username.addTextNode("username");

SOAPElement password = usernameToken.addChildElement("Password", "wsse");
password.setAttribute("Type", "http://docs.oasis-open.org/wss/2004/01/
oasis-200401-wss-username-token-profile-1.0#PasswordText");
password.addTextNode("password");
```

- d. Создайте тело сообщения. Следующий пример создает запрос FindProfileInfoByAccountRequest с номером счета 12345678:

```
SOAPElement bodyElement =
    body.addChildElement(envelope.createName("FindProfileInfoByAccountIdRequest",
        "mes", "http://www.moneta.ru/schemas/messages.xsd"));
bodyElement.addTextNode("12345678");
message.saveChanges();
```

Запрос будет выглядеть так:

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header>
    <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/
oasis-200401-wss-wssecurity-secext-1.0.xsd">
      <wsse:UsernameToken xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/
oasis-200401-wss-wssecurity-utility-1.0.xsd">
        <wsse:Username>user_name</wsse:Username>
        <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
username-token-profile-1.0#PasswordText">password</wsse:Password>
      </wsse:UsernameToken>
    </wsse:Security>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <mes:FindProfileInfoByAccountIdRequest xmlns:mes="http://www.moneta.ru/
schemas/messages.xsd">12345678</mes:FindProfileInfoByAccountIdRequest>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

4. Используйте объект connection для отправки запроса и получения ответа:

```
String destination = "https://demo.moneta.ru/services";
SOAPMessage reply = connection.call(message, destination);
```

Примечание: Данный пример делает запрос к DEMO серверу Moneta.Ru. Рабочий адрес для MONETA.MerchantAPI:

```
https://service.moneta.ru/services
```

5. Обрабатываем ответ.

- Создайте экземпляр класса Transformer и используйте его, чтобы распечатать ответ:

```
TransformerFactory transformerFactory = TransformerFactory.newInstance();
Transformer transformer = transformerFactory.newTransformer();
Source sourceContent = reply.getSOAPPart().getContent();
StreamResult result = new StreamResult(System.out);
transformer.transform(sourceContent, result);
System.out.println();
```

Полный текст ответа:

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <ns2:FindProfileInfoByAccountIdResponse
      xmlns:ns2="http://www.moneta.ru/schemas/messages.xsd">
      <ns2:accountId>12345678</ns2:accountId>
      <ns2:currency>RUB</ns2:currency>
      <ns2:profile>
        <ns2:attribute>
          <ns2:key>unitid</ns2:key>
          <ns2:value>55555</ns2:value>
        </ns2:attribute>
      </ns2:profile>
    </ns2:FindProfileInfoByAccountIdResponse>
  </SOAP-ENV:Body>
```

```
</SOAP-ENV:Envelope>
```

- Получите данные из XML ответа:

```
String accountId = reply.getSOAPBody().
    getElementsByTagName("ns2:accountId").item(0).
    getFirstChild().getNodeValue();
System.out.println("Account ID: "+accountId);
```

6. Закройте соединение:

```
connection.close();
```

Полный текст примера:

```
package ru.moneta;

import javax.xml.namespace.QName;
import javax.xml.soap.*;

import javax.xml.transform.TransformerFactory;
import javax.xml.transform.Transformer;
import javax.xml.transform.Source;

import javax.xml.transform.stream.StreamResult;

public class Main {

    public static void main(String args[]) {
        try {
            SOAPConnectionFactory soapConnFactory =
                SOAPConnectionFactory.newInstance();
            SOAPConnection connection = soapConnFactory.createConnection();

            MessageFactory messageFactory = MessageFactory.newInstance();
            SOAPMessage message = messageFactory.createMessage();

            SOAPPart soapPart = message.getSOAPPart();
            SOAPEnvelope envelope = soapPart.getEnvelope();
            SOAPBody body = envelope.getBody();
            SOAPHeader header = envelope.getHeader();

            SOAPElement security = header.addChildElement("Security", "wsse", "http://
docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd");

            SOAPElement usernameToken = security.addChildElement("UsernameToken",
"wsse");
            usernameToken.addAttribute(new QName("xmlns:wsu"),
                "http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-utility-1.0.xsd");

            SOAPElement username = usernameToken.addChildElement("Username", "wsse");
            username.addTextNode("user_name");

            SOAPElement password = usernameToken.addChildElement("Password", "wsse");
            password.setAttribute("Type",
                "http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-username-
token-profile-1.0#PasswordText");
            password.addTextNode("password");

            SOAPElement bodyElement =

            body.addChildElement(envelope.createName("FindProfileInfoByAccountIdRequest" ,
                "mes",
                "http://www.moneta.ru/schemas/messages.xsd"));

            bodyElement.addTextNode("12345678");
```

```

        message.saveChanges();

        String destination = "https://demo.moneta.ru/services";
        SOAPMessage reply = connection.call(message, destination);

        String accountId = reply.getSOAPBody().
            getElementsByTagName("ns2:accountId").item(0)
            .getFirstChild().getNodeValue();
        System.out.println("Account ID: "+accountId);

        connection.close();
    } catch (Exception e) {
        System.out.println(e.getMessage());
    }
}
}

```

SOAP пример на C#

WSE3

Для начала работы следует установить [WSE3](#).

Пакет Microsoft.Web.Services3 должен быть виден и доступен в Вашем проекте.

Ошибка при генерации классов из wsdl

Если Вы сгенерировали классы для работы с MONETA.MerchantAPI по wsdl схеме, то во время компиляции у Вас может возникнуть ошибка, которая говорит о несовпадении типов.

Проблема в том, что Visual Studio при генерации классов допускает ошибку. Следует удостовериться в правильности сгенерированного кода и поправить вручную при необходимости:

```

1) public partial class FindProfileInfoResponse : object,
   System.ComponentModel.INotifyPropertyChanged {

        private long pageSizeField;

        private long pageNumberField;

        private long pagesCountField;

        private long sizeField;

        private long totalSizeField;

        private KeyValueApprovedAttribute[] profileField;

2)     [System.Xml.Serialization.XmlArrayAttribute(Order=5)]
        [System.Xml.Serialization.XmlArrayItemAttribute("attribute",
        typeof(KeyValueApprovedAttribute), IsNullable=false)]
        public KeyValueApprovedAttribute[] profile {
            get {
                return this.profileField;
            }
            set {
                this.profileField = value;
                this.RaisePropertyChanged("profile");
            }
        }
}

```


SSL соединение и HTTP версия

```
protected override WebRequest GetWebRequest(Uri uri)
{
    HttpWebRequest request = (HttpWebRequest) base.GetWebRequest(uri);
    request.KeepAlive = false;
    request.ProtocolVersion = new Version(1, 0); //this was the big deal...
    return request;
}

ServicePointManager.ServerCertificateValidationCallback = new
    RemoteCertificateValidationCallback(
        delegate(object sender2, X509Certificate certificate, X509Chain chain,
            SslPolicyErrors sslPolicyErrors)
        {
            return true;
        }
    );
```

Пример вызова

```
class Program {
    public static bool ValidateServerCertificate(
        object sender,
        X509Certificate certificate,
        X509Chain chain,
        SslPolicyErrors sslPolicyErrors)
    {
        // Do not allow this client to communicate with unauthenticated servers.
        return true;
    }
    static void doWork() {
// installing custom server certificate validator
        ServicePointManager.ServerCertificateValidationCallback = new
        RemoteCertificateValidationCallback(ValidateServerCertificate);
        ru.moneta.www.MessagesServiceWse ms = new
        ru.moneta.www.MessagesServiceWse();
// using client certificate for authentication
// X509Certificate cert = new X509Certificate("c:\\your-client-
certificate.p12", "");
// ms.ClientCertificates.Add(cert);
// using login and password for authentication
        ms.RequestSoapContext.Security.Tokens.Add(new UsernameToken(V_LOGIN,
        V_PASSWORD, PasswordOption.SendPlainText));
        ru.moneta.www.FindProfileInfoByAccountIdResponse o =
        ms.FindProfileInfoByAccountId(12345678);
        System.Console.Out.WriteLine(o.currency);
    }
    static void Main(string[] args) {
        doWork();
    }
}
```

SOAP пример на PHP

Следующий пример использует библиотеку MonetaWebService для получения баланса счета.

1. Сделайте импорт библиотеки MonetaWebService:

```
require_once 'MonetaWebService.php';
```

- Создайте экземпляр класса `MonetaWebService`. Этот класс требует три параметра: url WSDL файла, идентификатор пользователя и пароль:

```
$service = new MonetaWebService("https://demo.moneta.ru/services.wsdl", "user_name", "password");
```

Примечание: Данный пример делает запрос к DEMO серверу Moneta.Ru. Рабочий адрес для MONETA.MerchantAPI:

```
https://service.moneta.ru/services.wsdl
```

- Вызовите метод `FindAccountById`. Этот метод посылает запрос и получает ответ:

```
$response = $service->FindAccountById($account_ID);
```

- Обработаем ответ:

```
echo "balance: " . $response->account->availableBalance;
echo "currency: " . $response->account->currency;
```

Полный текст примера:

```
<?php

require_once 'MonetaWebService.php';

$service = new MonetaWebService("https://demo.moneta.ru/services.wsdl",
    "merchant_email", "merchant_password");

try
{
    $response = $service->FindAccountById(12345678);
    echo "balance: " . $response->account->availableBalance;
    echo "currency: " . $response->account->currency;
}
catch (Exception $e)
{
    echo $e->getMessage();
}

?>
```

SOAP пример на Python

Пример показывает, как получить информацию о профайле пользователя используя библиотеку `Suds`.

- Добавьте импорт необходимых классов:

```
from suds.client import Client
from suds.wsse import *
```

- Вы можете сконфигурировать библиотеку, чтобы она записывала в лог запросы и ответы:

```
import logging
logging.basicConfig(level=logging.INFO)
logging.getLogger('suds.client').setLevel(logging.DEBUG)
```

- Создайте экземпляр класса `Suds.Client`. Используйте его, чтобы посылать запросы и получать ответы:

```
wsdl='https://demo.moneta.ru/services.wsdl'
client = Client(wsdl, cache=None)
```

Примечание: Данный пример делает запрос к DEMO серверу Moneta.Ru. Рабочий адрес для MONETA.MerchantAPI:

```
https://service.moneta.ru/services.wsdl
```

4. В данном примере для авторизации мы используем идентификатор пользователя и пароль. Укажем в заголовке *username* и *password*:

```
security = Security()
token = UsernameToken('username', 'password')
security.tokens.append(token)
client.set_options(wsse=security)
```

Подсказка: Вы можете использовать экземпляр класса Client, чтобы распечатать все доступные методы:

```
print client
```

Следующий фрагмент показывает, что для вызова метода GetProfileInfo требуется передать параметр unitId:

```
GetProfileInfo(xs:long unitId)
```

5. Используйте метод GetProfileInfo, чтобы послать запрос и получить ответ:

```
response = client.service['MessagesSoap11']['GetProfileInfo'](unitId=40488)
```

6. Обработайте ответ. Следующий код распечатывает пары "ключ-значение", которые пришли в ответе:

```
for attr in response:
    print attr['key']+": "+attr['value']
```

Полный текст примера:

```
#!/usr/bin/python2.7

from suds.client import Client
from suds.wsse import *

import logging
logging.basicConfig(level=logging.INFO)
logging.getLogger('suds.client').setLevel(logging.DEBUG)

wsdl='https://demo.moneta.ru/services.wsdl'
client = Client(wsdl,cache=None)

security = Security()
token = UsernameToken('username', 'password')
security.tokens.append(token)
client.set_options(wsse=security)

response = client.service['MessagesSoap11']['GetProfileInfo'](unitId=40488)

for attr in response:
    print attr['key']+": "+attr['value']
```