

Правила применимости: productApplicabilityRule, applicabilityRule

Оглавление

- 1 Примеры скриптов правил применимости
- 2 Примеры получения значений полей по данным заявки для использования в правилах применимости

Примеры правил применимости, которые можно использовать в правиле применимости продукта в целом или для пакетов документов в составе продукта.

Вы можете по аналогии написать собственные правила.

 Примеры сформированных тестовых заявок можно посмотреть и скачать в документе [I. Подготовительные работы по интеграции с применением API](#)

1 Примеры скриптов правил применимости

Описание правила	Пример правила применимости, script
для ИП (пример синтаксиса действует с версии 1.9.0)	<pre>function isAvailableForDealApplicationData(dealApplicationData) { var clientCompany = dealApplicationData.getFetchField ('Company', dealApplicationData.result.client.id); if (clientCompany == null clientCompany.legalType == null) { return null; } return clientCompany.legalType.id == 'IP'; }</pre> где для указания значения <code>clientCompany.legalType.id</code> используется справочник Формы предпринимательской деятельности: legal_types
ЮЛ с суммой гарантии больше 10 млн (пример синтаксиса действует с версии 1.9.0)	<pre>function isAvailableForDealApplicationData(dealApplicationData) { var clientCompany = dealApplicationData.getFetchField ('Company', dealApplicationData.result.client.id); if (clientCompany == null clientCompany.legalType == null dealApplicationData.result.productDealState.amount == null) { return null; } return clientCompany.legalType.id == 'ORGANIZATION' && dealApplicationData.result.productDealState.amount > 10000000; }</pre>
ЮЛ с фактической ценой контракта до 100 тысяч (пример синтаксиса действует с версии 1.9.0)	<pre>function isAvailableForDealApplicationData(dealApplicationData) { var clientCompany = dealApplicationData.getFetchField ('Company', dealApplicationData.result.client.id); if (clientCompany == null clientCompany.legalType == null dealApplicationData.result.productDealState.actualContractPrice == null) { return null; } return clientCompany.legalType.id == 'ORGANIZATION' && dealApplicationData.result.productDealState.actualContractPrice < 100000; }</pre>

ЮЛ со сроком гарантии меньше 30 дней (пример синтаксиса действует с версии 1.9.0)	<pre>function isAvailableForDealApplicationData(dealApplicationData) { var clientCompany = dealApplicationData.getFetchField('Company', dealApplicationData.result.client.id); if (clientCompany == null clientCompany.legalType == null dealApplicationData.result.productDealState.termDays == null) { return null; } return clientCompany.legalType.id == 'ORGANIZATION' && dealApplicationData.result.productDealState.termDays < 30; }</pre>
ЮЛ, финансовая (FIN) или некоммерческая организации (UNCOM) (пример синтаксиса действует с версии 1.9.0)	<pre>function isAvailableForDealApplicationData(dealApplicationData) { var clientCompany = dealApplicationData.getFetchField('Company', dealApplicationData.result.client.id); var clientOrgTypeId = clientCompany.orgType.id; var orgTypes = ['FIN', 'UNCOM']; if (clientCompany == null clientOrgTypeId == null clientCompany.legalType == null) { return null; } return orgTypes.indexOf(clientOrgTypeId) !== -1 && clientCompany.legalType.id == 'ORGANIZATION'; }</pre> где orgTypes может быть 'COM' - коммерческая организация, 'UNCOM' - некоммерческая организация, 'FIN' - финансовая организация
ЮЛ, регион заказчика Камчатский край или Чукотский АО (пример синтаксиса действует с версии 1.9.0)	<pre>function isAvailableForDealApplicationData(dealApplicationData) { var clientCompany = dealApplicationData.getFetchField('Company', dealApplicationData.result.client.id); var customerRegion = dealApplicationData.result.productDealState.customer.region.id; var customerRegions = [' ', ' ']; if (clientCompany == null clientCompany.legalType == null customerRegion == null) { return null; } return clientCompany.legalType.id == 'ORGANIZATION' && customerRegions.indexOf(customerRegion) !== -1; }</pre>
ИП с типом налогообложения УСН, Сельхоз, ЕНВД (пример синтаксиса действует с версии 1.9.0)	<pre>function isAvailableForDealApplicationData(dealApplicationData) { var clientCompany = dealApplicationData.getFetchField('Company', dealApplicationData.result.client.id); if (clientCompany == null) { return null; } var taxSystemDto = dealApplicationData.getFetchField('TaxSystemCatalog', clientCompany.taxSystem); if (clientCompany.legalType == null taxSystemDto == null) { return null; } var taxSystems = ['USN', 'ENVD', 'ESHN']; return taxSystems.indexOf(taxSystemDto.code) !== -1 && clientCompany.legalType.id == 'IP'; }</pre>

ЮЛ, дата государственной регистрации после 2003-06-05 (пример синтаксиса действует с версии 1.9.0)	<pre>function isAvailableForDealApplicationData(dealApplicationData) { var clientCompany = dealApplicationData.getFetchField('Company', dealApplicationData.result.client.id); if (clientCompany == null clientCompany.legalType == null clientCompany.regDate == null) { return null; } return clientCompany.legalType.id == 'ORGANIZATION' && clientCompany.regDate > '2003-06-05'; }</pre>
ЮЛ, Сумма БГ до 2 млн.руб. - срок деятельности клиента от 180 дней. Сумма БГ от 2 млн.руб. срок деятельности клиента - от 365 дней. (пример синтаксиса действует с версии 1.9.0)	<pre>function isAvailableForDealApplicationData(dealApplicationData) { var clientCompany = dealApplicationData.getFetchField('Company', dealApplicationData.result.client.id); if (clientCompany == null clientCompany.legalType == null clientCompany.regDate == null dealApplicationData.result.productDealState.amount == null) { return null; } now = new Date(); rD = new Date(clientCompany.regDate); termClient = (now - rD)/(60 * 60 * 24 * 1000); return clientCompany.legalType.id == 'ORGANIZATION' && ((termClient >= 180 && dealApplicationData.result.productDealState.amount <= 2000000) (termClient >= 365 && dealApplicationData.result.productDealState.amount > 2000000)); }</pre>
ЮЛ, базовый ОКВЭД 86.23 - Стоматологическая практика или 86.90.9 - Деятельность в области медицины прочая, не включенная в другие группировки (пример синтаксиса действует с версии 1.9.0)	<pre>function isAvailableForDealApplicationData(dealApplicationData) { var clientCompany = dealApplicationData.getFetchField('Company', dealApplicationData.result.client.id); if (clientCompany == null) { return null; } var clientBaseOkved = dealApplicationData.getFetchField('OkvedCatalog', clientCompany.baseOkved); if (clientCompany.legalType == null clientBaseOkved == null) { return null; } var baseOkveds = ['86.23', '86.90.9']; return baseOkveds.indexOf(clientBaseOkved.code) !== -1 && clientCompany.legalType.id == 'ORGANIZATION'; }</pre>
ЮЛ с организационно-правовыми формами АО (код 12200), ПАО (код 12247), НАО (код 12267) (пример синтаксиса действует с версии 1.9.0)	<pre>function isAvailableForDealApplicationData(dealApplicationData) { var clientCompany = dealApplicationData.getFetchField('Company', dealApplicationData.result.client.id); if (clientCompany == null) { return null; } var okopfDto = dealApplicationData.getFetchField('OkopfCatalog', clientCompany.okopf); if(okopfDto == null clientCompany.legalType == null) { return null; } var okopfTypes = ['12200', '12247', '12267']; return clientCompany.legalType.id == 'ORGANIZATION' && okopfTypes.indexOf(okopfDto.code) !== -1; }</pre>

ЮЛ, 223-ФЗ или 44-ФЗ (пример синтаксиса действует с версии 1.9.0)	<pre>function isAvailableForDealApplicationData(dealApplicationData) { var clientCompany = dealApplicationData.getFetchField('Company', dealApplicationData.result.client.id); var federalLaw = dealApplicationData.result.productDealState.tender.federalLaw.id; if (clientCompany == null clientCompany.legalType == null federalLaw == null) { return null; } var federalLaws = ['223FL', '44FL']; return federalLaws.indexOf(federalLaw) !== -1 && clientCompany.legalType.id == 'ORGANIZATION'; }</pre> где для указания значения federalLaws используется справочник Федеральные законы: federalLaw_types
Продукт применим только для головных компаний (ЮЛ), филиалы исключаются (пример синтаксиса действует с версии 1.9.0)	<pre>function isAvailableForDealApplicationData(dealApplicationData) { var clientCompany = dealApplicationData.getFetchField('Company', dealApplicationData.result.client.id); if(clientCompany == null) { return null; } var okopfDto = dealApplicationData.getFetchField('OkopfCatalog', clientCompany.okopf); if(okopfDto == null clientCompany.legalType == null) { return null; } var sourceInfo = clientCompany.sourceInfo; var okopfTypes = ['30000', '30001', '30002', '30003', '30004', '30005', '30006', '30008']; return clientCompany.legalType.id == 'ORGANIZATION' && okopfTypes.indexOf(okopfDto.code) === -1 && sourceInfo.headCompany === true && sourceInfo.hasBranches === false; }</pre>
ЮЛ, регион фактического местоположения компании Камчатский край или Тульская область (пример синтаксиса действует с версии 1.14.0)	<pre>function isAvailableForDealApplicationData(dealApplicationData) { var clientCompany = dealApplicationData.getFetchField('Company', dealApplicationData.result.client.id); if (clientCompany == null) { return null; } var factAddressDto = dealApplicationData.getFetchField('CompanyAddress', clientCompany.factAddress); if (clientCompany.legalType == null factAddressDto == null) { return null; } var companyRegions = [' ', ' ']; return companyRegions.indexOf(factAddressDto.region) !== -1 && clientCompany.legalType.id == 'ORGANIZATION'; }</pre>

ЮЛ, стоп-регионы по юридическому адресу: Адыгея, Республика Ингушетия, Чеченская республика, Республика Дагестан, республика Крым, Севастополь (пример синтаксиса действует с версии 1.14.0)	<pre>function isAvailableForDealApplicationData(dealApplicationData) { var clientCompany = dealApplicationData.getFetchField('Company', dealApplicationData.result.client.id); var LegalAddress = dealApplicationData.getFetchField('CompanyAddress', clientCompany.legalAddress); if (clientCompany == null clientCompany.legalType == null LegalAddress.region == null) { return null; } var regex = new RegExp(' ', 'i'); var clientLegalReg = regex.test(LegalAddress.region); return !clientLegalReg && clientCompany.legalType.id == 'ORGANIZATION'; }</pre>
ЮЛ, среди собственников есть юр. лица (пример синтаксиса действует с версии 1.14.0)	<pre>function isAvailableForDealApplicationData(dealApplicationData) { var clientCompany = dealApplicationData.getFetchField('Company', dealApplicationData.result.client.id); if (clientCompany == null clientCompany.legalType == null) { return null; } var legalFounders = dealApplicationData.getFetchFieldAsMap('LegalFounder'); var keyArray = legalFounders.keySet().toArray(); if(keyArray.length == 0) { return false; } return clientCompany.legalType.id == 'ORGANIZATION'; }</pre>
ЮЛ, среди собственников только физ. лица (пример синтаксиса действует с версии 1.14.0)	<pre>function isAvailableForDealApplicationData(dealApplicationData) { var clientCompany = dealApplicationData.getFetchField('Company', dealApplicationData.result.client.id); if (clientCompany == null clientCompany.legalType == null) { return null; } var legalFounders = dealApplicationData.getFetchFieldAsMap('LegalFounder'); var keyArray = legalFounders.keySet().toArray(); if(keyArray.length != 0) { return false; } return clientCompany.legalType.id == 'ORGANIZATION'; }</pre>

ЮЛ, запрет тендеров по 275-ФЗ (пример синтаксиса действует с версии 1.16.0)	<pre>function isAvailableForDealApplicationData(dealApplicationData) { var clientCompany = dealApplicationData.getFetchField('Company', dealApplicationData.result.client.id); var tenderClosedLaw = dealApplicationData.result.productDealState.tender.closedLaw; if (clientCompany == null clientCompany.legalType == null) { return null; } if (tenderClosedLaw === true) { return false; } return clientCompany.legalType.id == 'ORGANIZATION'; }</pre>
ЮЛ, валюта контракта - рубли (пример синтаксиса действует с версии 1.9.0)	<pre>function isAvailableForDealApplicationData(dealApplicationData) { var clientCompany = dealApplicationData.getFetchField('Company', dealApplicationData.result.client.id); var tenderCurrencyType = dealApplicationData.result.productDealState.tender.currencyType; var urrencyTypeDto = dealApplicationData.getFetchField('CurrencyTypeCatalog', tenderCurrencyType); if (clientCompany == null clientCompany.legalType == null) { return null; } var currencyTypes = ['RUB']; return clientCompany.legalType.id == 'ORGANIZATION' && currencyTypes.indexOf(urrencyTypeDto.code) !== -1; }</pre>
ИП, БГ(У), где сумма БГ на исполнение не превышает 10 млн. (пример синтаксиса действует с версии 1.16.0)	<pre>function isAvailableForDealApplicationData(dealApplicationData) { var clientCompany = dealApplicationData.getFetchField('Company', dealApplicationData.result.client.id); var isNeedPerformanceBgDeal = dealApplicationData.result.productDealState.isNeedPerformanceBg; if (isNeedPerformanceBgDeal === true) { var amountBG = dealApplicationData.result.productDealState.performanceBg.amount; return amountBG <= 100000000; } if (clientCompany == null clientCompany.legalType == null) { return null; } return clientCompany.legalType.id == 'IP'; }</pre>
Не заполнены фин. показатели. (пример синтаксиса действует с версии 1.20.0) Ниже приведен аналог проверки заполнения фин. показателей для версии 1.19.0	<pre>function isAvailableForDealApplicationData(dealApplicationData) { return !isFinanceIndicatorFilledForAllPeriods(dealApplicationData, 'FORM_1', '1600') !isFinanceIndicatorFilledForAllPeriods(dealApplicationData, 'FORM_1', '1700') !isFinanceIndicatorFilledForAllPeriods(dealApplicationData, 'FORM_2', '2110'); }</pre>

Сумма БГ меньше 1млн И не
заполнены фин. показатели.

(пример синтаксиса действует с
версии 1.19.0)

```
function isAvailableForDealApplicationData(dealApplicationData) {
    if (dealApplicationData.result.productDealState.amount > 1000000) {
        return false;
    }
    if (dealApplicationData.getFetchFieldAsMap('FinanceIndicator') ==
null) {
        return true;
    }
    var allIndicators = dealApplicationData.getFetchFieldAsMap
('FinanceIndicator').values();
    var idOf1600 = allIndicators.stream()
        .filter(function(e) {
            return '1600' == e.code
        })
        .findFirst()
        .map(function(e){ return e.id})
        .orElse(null);
    var idOf1700 = allIndicators.stream()
        .filter(function(e) {
            return '1700' == e.code
        })
        .findFirst()
        .map(function(e){ return e.id})
        .orElse(null);
    var idOf2110 = allIndicators.stream()
        .filter(function(e) {
            return '2110' == e.code
        })
        .findFirst()
        .map(function(e){ return e.id})
        .orElse(null);
    if (idOf1600 == null || idOf1700 == null || idOf2110 == null) {
        return true;
    }
    var form1Values = dealApplicationData.result.financeForms.stream()
        .filter(function(e) {
            return '1b010932-3e5b-4992-861e-b3648f83a03d' == e.formId
        })
        .findFirst()
        .get();
    var form2Values = dealApplicationData.result.financeForms.stream()
        .filter(function(e) {
            return 'a48142d6-ded8-41c2-bfbf-57035a4fc773' == e.formId
        })
        .findFirst()
        .get();
    if (form1Values == null || form2Values == null) {
        return true;
    }
    for (var j = 0; j < form1Values.periods.size(); j++) {
        if (form1Values.values[form1Values.getIndicators().indexOf
(idOf1600)][j] == null ||
            form1Values.values[form1Values.getIndicators().indexOf
(idOf1700)][j] == null) {
            return true;
        }
    }
    for (var j = 0; j < form2Values.periods.size(); j++) {
        if (form2Values.values[form2Values.getIndicators().indexOf
(idOf2110)][j] == null) {
            return true;
        }
    }
    return false;
}
```

ЮЛ, если все физ. лица компании с документом "Паспорт гражданина РФ"

(пример синтаксиса действует с версии 1.19.0)

```
function isAvailableForDealApplicationData(dealApplicationData) {
    var clientCompany = dealApplicationData.getFetchField('Company',
dealApplicationData.result.client.id);
    var allPersonsHasPassport = true;
    var personsMap = dealApplicationData.getFetchFieldAsMap('Person');
    if (personsMap !== null) {
        var persons = personsMap.values();
        allPersonsHasPassport = persons.stream().allMatch(function
(person) {
            return person.docType != null && person.docType.id ==
'PASSPORT';
        });
    }
    return clientCompany.legalType.id == 'ORGANIZATION' &&
allPersonsHasPassport;
}
```

ЮЛ, если прикреплены документы: "Бухгалтерская (финансовая) отчетность на последнюю календарную квартальную дату" И "Годовая бухгалтерская (финансовая) отчетность"

(пример синтаксиса действует с версии 1.19.0)

```
function isAvailableForDealApplicationData(dealApplicationData) {
    var clientCompany = dealApplicationData.getFetchField('Company',
dealApplicationData.result.client.id);
    if (clientCompany == null) {
        return null;
    }
    //      :
    // 1. ,
    //-----
    var financeReportExists = false;
    //-----
    var analyticalBalanceExists = false;
    var documentsMap = dealApplicationData.getFetchFieldAsMap
('ApplicationDocument');
    if (documentsMap !== null) {
        var documents = documentsMap.values();
        // 2. ( ) ; (true/false)
        //-----
        financeReportExists = documents.stream().anyMatch(function(doc) {
            return doc.docType != null && doc.docType.id ==
'FINANCE_REPORT_LAST_QUARTER' && doc.docState != null && doc.docState.id ==
'ACTUAL';
        });
        //-----
        analyticalBalanceExists = documents.stream().anyMatch(function(doc)
{
            return doc.docType != null && doc.docType.id ==
'ANALYTICAL_BALANCE_LAST_REPORTING_YEAR' && doc.docState != null && doc.
docState.id == 'ACTUAL';
        });
    }
    // 3. , return "! (___1 && ... && ___N)"
    //      , return "! (___1 || ... || ___N)"
    //-----
    return clientCompany.legalType.id == 'ORGANIZATION' && !
(financeReportExists && analyticalBalanceExists);
    //-----
}
```

ЮЛ, стоп по ИНН заказчика: 1660049283, 1653014843. (пример синтаксиса действует с версии 1.6.0)	<pre>function isAvailableForDealApplicationData(dealApplicationData) { var clientCompany = dealApplicationData.getFetchField('Company', dealApplicationData.result.client.id); var customerInn = dealApplicationData.result.productDealState.customer. inn; if (clientCompany == null clientCompany.legalType == null) { return null; } var stopCustomerInn = ['1660049283', '1653014843']; return clientCompany.legalType.id == 'ORGANIZATION' && stopCustomerInn. indexOf(customerInn) == -1; }</pre>
ЮЛ, стоп по ОКПД (пример синтаксиса действует с версии 1.19.0)	<pre>function isAvailableForDealApplicationData(dealApplicationData) { var clientCompany = dealApplicationData.getFetchField ('Company', dealApplicationData.result.client.id); var okpdMap = dealApplicationData.getFetchFieldAsMap ('OkpdCatalog'); var okpdIsValid = true; if (okpdMap !== null) { var okpds = okpdMap.values(); okpdIsValid = okpds.stream().allMatch(function(okpd) { return okpd.code !== null && okpd.code !== '42.11.20.000'; }); } return clientCompany.legalType.id == 'ORGANIZATION' && okpdIsValid; }</pre>
ИП, стоп по группам ОКПД (пример синтаксиса действует с версии 1.19.0)	<pre>function isAvailableForDealApplicationData(dealApplicationData) { var clientCompany = dealApplicationData.getFetchField ('Company', dealApplicationData.result.client.id); var okpdMap = dealApplicationData.getFetchFieldAsMap ('OkpdCatalog'); var okpdIsValid = true; var regexOkpd = new RegExp('^(06 19 41.2 42 43. 99 64 65 66 68 99)', 'i'); if (okpdMap !== null) { var okpds = okpdMap.values(); okpdIsValid = okpds.stream().allMatch(function(okpd){ var legalOkpd = regexOkpd.test(okpd.code); return okpd.code !== null && !legalOkpd; }); } return clientCompany.legalType.id == 'IP' && okpdIsValid; }</pre>

СТОП-Регионы по юридическому адресу принципала с исключениями отдельных населённых пунктов из регионов:

- Республика Адыгея (за исключением г. Майкоп, станица Ханская (административно подчинена Майкопу), п. Яблоновский, г. Энем, станица Гиагинская, г. Адыгейск и село Красногвардейское)
- Республика Алтай
- Чукотский АО
- Республика Калмыкия
- Ненецкий АО
- Республика Тува
- Чеченская республика
- Республика Дагестан
- Республика Ингушетия
- Карачаево-Черкесская Республика
- Кабардино-Балкарская республика (за исключением г. Нальчик, г. Прохладный)
- Республика Северная Осетия-Алания (за исключением г. Владикавказ, г. Беслан)

(пример синтаксиса действует с версии 1.25.16)

```
function isAvailableForDealApplicationData(dealApplicationData)
{
    var LegalAddress = dealApplicationData.getFetchField
('CompanyAddress', clientCompany.legalAddress);
    if (LegalAddress == null || LegalAddress.region ==
null) {
        return null;
    }
    var regex1 = new RegExp('|', 'i');
    var regex1_1 = new RegExp('|', 'i');

    var regex2 = new RegExp('', 'i');
    var regex2_1 = new RegExp('|', 'i');

    var regex3 = new RegExp('', 'i');
    var regex3_1 = new RegExp('|||||', 'i');

    var alwaysDenyRegionsRegex = new RegExp('|||||||', 'i');
    if (alwaysDenyRegionsRegex.test(LegalAddress.region))
    {
        return false;
    }
    var reg1check = regex1.test(LegalAddress.region) && !regex1_1.
test(LegalAddress.city);
    var reg2check = regex2.test(LegalAddress.region) && !regex2_1.
test(LegalAddress.city);
    var reg3check = regex3.test(LegalAddress.region) && !regex3_1.
test(LegalAddress.city);
    if (reg1check || reg2check || reg3check) {
        return false;
    }
}
```

Исключения в `var regex1 = new RegExp('осетия|алания', 'i');`

`var regex1_1 = new RegExp('владикавказ|беслан', 'i');`

Исключения в `var regex2 = new RegExp('кабарди', 'i');`

`var regex2_1 = new RegExp('нальчик|прохладный', 'i');`

Исключения в `var regex3 = new RegExp('адыгея', 'i');`

`var regex3_1 = new RegExp('майкоп|яблоновский|энем|адыгейск|красногвардейское|гиагинская|ханская', 'i');`

Регионы без исключений `var alwaysDenyRegionsRegex = new RegExp('карачае|ингушет|чечен|дагестан|алтай|чукотск|калмык|ненецкий|тыва|тува', 'i');`
`if (alwaysDenyRegionsRegex.test(LegalAddress.region))`

44-ФЗ, 223-ФЗ - без ограничений по ИНН заказчика. 185-ФЗ/615ПП - с ограничением по ИНН заказчика, для указанных ИНН заказчиков доступна отправка заявки в сервис: (пример синтаксиса действует с версии 1.25.16)	<pre>function isAvailableForDealApplicationData(dealApplicationData) { var AvailableFederalLaws = ['223FL', '44FL']; var federalLaw = dealApplicationData.result.productDealState.tender.federalLaw.id; var commercialCustomerInn = ['1644003838','1651044095','1651000041','1644040195','5009053687','1644057262','1631050005', '1644024517','1651031650','1651024807','1651049488','7606040300','1651000027','1651049262','1651061372','1655428783','7703476220']; return ((federalLaw == 'COMMERCIAL_BIDDING' && commercialCustomerInn.indexOf(customerInn) !== -1) AvailableFederalLaws.indexOf(federalLaw) !== -1) }</pre>
PKO, проверка по городам (пример синтаксиса действует с версии 1.25.16)	<pre>function isAvailableForDealApplicationData(dealApplicationData) { var clientCity = dealApplicationData.result.productDealState.city; var allowedCity = ['А', 'Б', 'В', 'Г', 'Д', 'Е', 'Ж', 'З', 'И', 'Й', 'К', 'Л', 'М', 'Н', 'О', 'П', 'Р', 'С', 'Т', 'У', 'Ф', 'Х', 'Ц', 'Ч', 'Ш', 'Щ', 'Ъ', 'Ы', 'Ь', 'Э', 'Ю', 'Я', 'а', 'б', 'в', 'г', 'д', 'е', 'ж', 'з', 'и', 'й', 'к', 'л', 'м', 'н', 'о', 'п', 'р', 'с', 'т', 'у', 'ф', 'х', 'ц', 'ч', 'ш', 'щ', 'ъ', 'ы', 'ь', 'э', 'ю', 'я', '0', '1', '2', '3', '4', '5', '6', '7', '8', '9', '!', '@', '#', '\$', '%', '&', '*', '(', ')', '-', '=', '+', '^', '&#x2013;', '&#x2014;', '&#x2018;', '&#x2019;', '&#x201a;', '&#x201b;', '&#x201c;', '&#x201d;', '&#x201e;', '&#x201f;', '&#x2020;', '&#x2021;', '&#x2022;', '&#x2023;', '&#x2024;', '&#x2025;', '&#x2026;', '&#x2027;', '&#x2028;', '&#x2029;', '&#x2030;', '&#x2031;', '&#x2032;', '&#x2033;', '&#x2034;', '&#x2035;', '&#x2036;', '&#x2037;', '&#x2038;', '&#x2039;', '&#x203a;', '&#x203b;', '&#x203c;', '&#x203d;', '&#x203e;', '&#x203f;', '&#x2040;', '&#x2041;', '&#x2042;', '&#x2043;', '&#x2044;', '&#x2045;', '&#x2046;', '&#x2047;', '&#x2048;', '&#x2049;', '&#x204a;', '&#x204b;', '&#x204c;', '&#x204d;', '&#x204e;', '&#x204f;', '&#x2050;', '&#x2051;', '&#x2052;', '&#x2053;', '&#x2054;', '&#x2055;', '&#x2056;', '&#x2057;', '&#x2058;', '&#x2059;', '&#x205a;', '&#x205b;', '&#x205c;', '&#x205d;', '&#x205e;', '&#x205f;', '&#x2060;', '&#x2061;', '&#x2062;', '&#x2063;', '&#x2064;', '&#x2065;', '&#x2066;', '&#x2067;', '&#x2068;', '&#x2069;', '&#x206a;', '&#x206b;', '&#x206c;', '&#x206d;', '&#x206e;', '&#x206f;', '&#x2070;', '&#x2071;', '&#x2072;', '&#x2073;', '&#x2074;', '&#x2075;', '&#x2076;', '&#x2077;', '&#x2078;', '&#x2079;', '&#x207a;', '&#x207b;', '&#x207c;', '&#x207d;', '&#x207e;', '&#x207f;', '&#x2080;', '&#x2081;', '&#x2082;', '&#x2083;', '&#x2084;', '&#x2085;', '&#x2086;', '&#x2087;', '&#x2088;', '&#x2089;', '&#x208a;', '&#x208b;', '&#x208c;', '&#x208d;', '&#x208e;', '&#x208f;', '&#x2090;', '&#x2091;', '&#x2092;', '&#x2093;', '&#x2094;', '&#x2095;', '&#x2096;', '&#x2097;', '&#x2098;', '&#x2099;', '&#x209a;', '&#x209b;', '&#x209c;', '&#x209d;', '&#x209e;', '&#x209f;', '&#x20a0;', '&#x20a1;', '&#x20a2;', '&#x20a3;', '&#x20a4;', '&#x20a5;', '&#x20a6;', '&#x20a7;', '&#x20a8;', '&#x20a9;', '&#x20aa;', '&#x20ab;', '&#x20ac;', '&#x20ad;', '&#x20ae;', '&#x20af;', '&#x20b0;', '&#x20b1;', '&#x20b2;', '&#x20b3;', '&#x20b4;', '&#x20b5;', '&#x20b6;', '&#x20b7;', '&#x20b8;', '&#x20b9;', '&#x20ba;', '&#x20bb;', '&#x20bc;', '&#x20bd;', '&#x20be;', '&#x20bf;', '&#x20c0;', '&#x20c1;', '&#x20c2;', '&#x20c3;', '&#x20c4;', '&#x20c5;', '&#x20c6;', '&#x20c7;', '&#x20c8;', '&#x20c9;', '&#x20ca;', '&#x20cb;', '&#x20cc;', '&#x20cd;', '&#x20ce;', '&#x20cf;', '&#x20d0;', '&#x20d1;', '&#x20d2;', '&#x20d3;', '&#x20d4;', '&#x20d5;', '&#x20d6;', '&#x20d7;', '&#x20d8;', '&#x20d9;', '&#x20da;', '&#x20db;', '&#x20dc;', '&#x20dd;', '&#x20de;', '&#x20df;', '&#x20e0;', '&#x20e1;', '&#x20e2;', '&#x20e3;', '&#x20e4;', '&#x20e5;', '&#x20e6;', '&#x20e7;', '&#x20e8;', '&#x20e9;', '&#x20ea;', '&#x20eb;', '&#x20ec;', '&#x20ed;', '&#x20ee;', '&#x20ef;', '&#x20f0;', '&#x20f1;', '&#x20f2;', '&#x20f3;', '&#x20f4;', '&#x20f5;', '&#x20f6;', '&#x20f7;', '&#x20f8;', '&#x20f9;', '&#x20fa;', '&#x20fb;', '&#x20fc;', '&#x20fd;', '&#x20fe;', '&#x20ff;', '&#x2100;', '&#x2101;', '&#x2102;', '&#x2103;', '&#x2104;', '&#x2105;', '&#x2106;', '&#x2107;', '&#x2108;', '&#x2109;', '&#x210a;', '&#x210b;', '&#x210c;', '&#x210d;', '&#x210e;', '&#x210f;', '&#x2110;', '&#x2111;', '&#x2112;', '&#x2113;', '&#x2114;', '&#x2115;', '&#x2116;', '&#x2117;', '&#x2118;', '&#x2119;', '&#x211a;', '&#x211b;', '&#x211c;', '&#x211d;', '&#x211e;', '&#x211f;', '&#x2120;', '&#x2121;', '&#x2122;', '&#x2123;', '&#x2124;', '&#x2125;', '&#x2126;', '&#x2127;', '&#x2128;', '&#x2129;', '&#x212a;', '&#x212b;', '&#x212c;', '&#x212d;', '&#x212e;', '&#x212f;', '&#x2130;', '&#x2131;', '&#x2132;', '&#x2133;', '&#x2134;', '&#x2135;', '&#x2136;', '&#x2137;', '&#x2138;', '&#x2139;', '&#x213a;', '&#x213b;', '&#x213c;', '&#x213d;', '&#x213e;', '&#x213f;', '&#x2140;', '&#x2141;', '&#x2142;', '&#x2143;', '&#x2144;', '&#x2145;', '&#x2146;', '&#x2147;', '&#x2148;', '&#x2149;', '&#x214a;', '&#x214b;', '&#x214c;', '&#x214d;', '&#x214e;', '&#x214f;', '&#x2150;', '&#x2151;', '&#x2152;', '&#x2153;', '&#x2154;', '&#x2155;', '&#x2156;', '&#x2157;', '&#x2158;', '&#x2159;', '&#x215a;', '&#x215b;', '&#x215c;', '&#x215d;', '&#x215e;', '&#x215f;', '&#x2160;', '&#x2161;', '&#x2162;', '&#x2163;', '&#x2164;', '&#x2165;', '&#x21</pre>

2 Примеры получения значений полей по данным заявки для использования в правилах применимости

Поле согласно структуре заявки с учетом вложенности	Описание	Как объявить и получить значение в java
result.client	Объект компания клиента/ИП	<pre>var clientCompany = dealApplicationData.getFetchField('Company', dealApplicationData.result.client.id);</pre>
result.Company.legalType	Категория клиента (организация, ИП, физ. лицо)	<pre>var clientCompany = dealApplicationData.getFetchField('Company', dealApplicationData.result.client.id); var companyLegalType = clientCompany.legalType.id;</pre>

result.productDealState. amount	Сумма продукта	<pre>var amountProductDealState = dealApplicationData.result. productDealState.amount;</pre>
result.productDealState. actualContractPrice	Сумма ФЦК продукта	<pre>var priceProductDealState = dealApplicationData.result. productDealState.actualContractPrice;</pre>
result.productDealState. termDays	Срок гарантии	<pre>var termDaysProductDealState = dealApplicationData.result. productDealState.termDays;</pre>
result.Company.orgType	Тип организации	<pre>var clientCompany = dealApplicationData.getFetchField ('Company', dealApplicationData.result.client.id); var clientOrgTypeId = clientCompany.orgType.id;</pre>
result.productDealState. customer.region	Местонахождения (регион) заказчика	<pre>var customerRegion = dealApplicationData.result. productDealState.customer.region.id;</pre>
result.Company.taxSystem	Система налогообложения	<pre>var clientCompany = dealApplicationData.getFetchField ('Company', dealApplicationData.result.client.id); var taxSystemDto = dealApplicationData.getFetchField ('TaxSystemCatalog', clientCompany.taxSystem);</pre>
result.Company.regDate	Дата государственной регистрации	<pre>var clientCompany = dealApplicationData.getFetchField ('Company', dealApplicationData.result.client.id); var companyRegDate = clientCompany.regDate;</pre>
result.Company.baseOkved	Базовый ОКВЭД	<pre>var clientCompany = dealApplicationData.getFetchField ('Company', dealApplicationData.result.client.id); var clientBaseOkved = dealApplicationData.getFetchField ('OkvedCatalog', clientCompany.baseOkved);</pre>
result.Company.okopf	Организационно- правовая форма	<pre>var clientCompany = dealApplicationData.getFetchField ('Company', dealApplicationData.result.client.id); var okopfDto = dealApplicationData.getFetchField ('OkopfCatalog', clientCompany.okopf);</pre>
result.productDealState. tender.federalLaw	Федеральный закон	<pre>var federalLaw = dealApplicationData.result.productDealState. tender.federalLaw.id;</pre>
fetchFields. CompanySourceInfo	Головная компания	<pre>var clientCompany = dealApplicationData.getFetchField ('Company', dealApplicationData.result.client.id); var sourceInfo = clientCompany.sourceInfo;</pre>

fetchFields.CompanyAddress.region	Регион фактического местоположения компании	<pre>var clientCompany = dealApplicationData.getFetchField('Company', dealApplicationData.result.client.id); var factAddressDto = dealApplicationData.getFetchField('CompanyAddress', clientCompany.factAddress);</pre>
fetchFields.CompanyAddress.region	Регион юридического адреса компании	<pre>var clientCompany = dealApplicationData.getFetchField('Company', dealApplicationData.result.client.id); var LegalAddress = dealApplicationData.getFetchField('CompanyAddress', clientCompany.legalAddress);</pre>
fetchFields.LegalFounder	Собственники (юр лицо)	<pre>var legalFounders = dealApplicationData.getFetchFieldAsMap('LegalFounder'); var keyArray = legalFounders.keySet().toArray();</pre>
result.Company.regDate	Срок деятельности клиента	<pre>var clientCompany = dealApplicationData.getFetchField('Company', dealApplicationData.result.client.id); now = new Date(); rD = new Date(clientCompany.regDate); termClient = (now - rD)/(60 * 60 * 24 * 1000);</pre>
result.productDealState.tender.closedLaw	Тендер по 275-ФЗ	<pre>var tenderClosedLaw = dealApplicationData.result.productDealState.tender.closedLaw;</pre>
result.productDealState.tender.currencyType	Валюта контракта	<pre>var tenderCurrencyType = dealApplicationData.result.productDealState.tender.currencyType; var urrencyTypeDto = dealApplicationData.getFetchField('CurrencyTypeCatalog', tenderCurrencyType);</pre>
result.productDealState.performanceBg.amount	Сумма БГ на исполнение вне типа продукта БГ(И)	<pre>var amountBG = dealApplicationData.result.productDealState.performanceBg.amount;</pre>
result.productDealState.isNeedPerformanceBg	Потребуется БГ на исполнение контракта?	<pre>var isNeedPerformanceBgDeal = dealApplicationData.result.productDealState.isNeedPerformanceBg;</pre>
result.productDealState.customer.inn	ИНН заказчика	<pre>var customerInn = dealApplicationData.result.productDealState.customer.inn;</pre>
result.productDealState.city	Город клиента	<pre>var clientCity = dealApplicationData.result.productDealState.city;</pre>
result.responsible.id	Проверка ответственного по заявке	<pre>var responsible = dealApplicationData.getFetchField('CompanyContactParticipant', dealApplicationData.result.responsible.id);</pre>

