Скрипт 1. Вирахування ПДВ рахунку

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Опис задачі (Що необхідно зробити, Як буде тестуватись, Очікуваний результат)** | | | | |
| Реалізувати обчислення суми ПДВ та суми без ПДВ рахунку.  Приклад:  // Вирахування суми за послугу  function onChangeInvoice() {  var VATpercentage = 0;  var attrVATAmount = EdocsApi.getAttributeValue('InvoiceVATAmount');  var attrVATpercentage = EdocsApi.getAttributeValue('InvoiceVATPercent');  var attrContractAmount = EdocsApi.getAttributeValue('AccountInvoice');  var attrAmountOutVAT = EdocsApi.getAttributeValue('InvoiceAmountOutVAT');  switch (attrVATpercentage.value) {  case '20%': // if (x === 'если сумма НДС=20%')  var VATpercentage = 1.2;  break  case '7%': // if (x === 'если сумма НДС=7%')  var VATpercentage = 1.07;  break  }  if (attrVATpercentage.value === null || attrContractAmount.value === null) {  // если нет ставки НДС и суммы, то укажем ноль в сумме НДС и без НДС  attrVATAmount.value = 0;  attrAmountOutVAT.value = 0;  } else if (VATpercentage == 0) {  attrVATAmount.value = 0;  attrAmountOutVAT.value = attrContractAmount.value;  } else {  attrAmountOutVAT.value = Math.floor(100 \* attrContractAmount.value / VATpercentage) / 100;  attrVATAmount.value = attrContractAmount.value - attrAmountOutVAT.value;  }  EdocsApi.setAttributeValue(attrVATAmount);  EdocsApi.setAttributeValue(attrAmountOutVAT);  }  function onChangeVATpercentage() {  onChangeAuctionScrapMetalSalesContractAmount();  } | | | | |
| **Опис сценарію (що потрібно реалізувати за допомогою JS. Обов’язково використовуючи «ЯКЩО» -> «ТО» -> «ІНАКШЕ» і тригером (умови(a), або дія при яких має виконатись сценарій)** | | | | |
|  | | | | |
|  | | | | |
|  | | | | |
| **Атрибути картки, які задіяні в JS** | | | | |
| **Назва** | **Код** | **Тип в SP** | **Сист.атрибут** | **Код довідника** |
| Сума рахунку | AccountInvoice | Число (1; 1,0; 100) | - | - |
| Відсоток ПДВ | InvoiceVATPercent | Вибір (меню):   * 20 * 7 * 0 | - | - |
| Сума ПДВ | InvoiceVATAmount | Число (1; 1,0; 100) | - | - |
| Сума рахунку, без ПДВ | InvoiceAmountOutVAT | Число (1; 1,0; 100) | - | - |

Скрипт 2. Передача рахунку для ознайомлення в зовнішню систему

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Опис задачі (Що необхідно зробити, Як буде тестуватись, Очікуваний результат)** | | | | | | | |
| Необхідно налаштувати передачу рахунку (тип документу - invoice) для ознайомлення при виконанні завдання SendOutDoc.  // Відправлення на підпис в зовнішній сервіс eSign договору  //-------------------------------  function setDataForESIGN() {  debugger;  var registrationDate = EdocsApi.getAttributeValue('RegDate').value;  var registrationNumber = EdocsApi.getAttributeValue('RegNumber').value;  var caseType = EdocsApi.getAttributeValue('DocType').value;  var caseKind = EdocsApi.getAttributeValue('DocKind').text;  var name = '';  if (caseKind) {  name += caseKind;  } else {  name += caseType;  }  name += ' №' + (registrationNumber ? registrationNumber : CurrentDocument.id) + (!registrationDate ? '' : (' від ' + moment(registrationDate)  .format('DD.MM.YYYY')));  doc = {  DocName: name,  extSysDocId: CurrentDocument.id,  ExtSysDocVersion: CurrentDocument.version,  docType: "invoice",  docDate: registrationDate,  docNum: registrationNumber,  "File": "",  "parties": [  {  "taskType": "ToSign",  "taskState": "Done",  "legalEntityCode": EdocsApi.getAttributeValue('OrgCode').value,  "contactPersonEmail": EdocsApi.getAttributeValue('OrgRPEmail').value,  "signatures": []  },  {  "taskType": "ToSign",  "taskState": "NotAssigned",  "legalEntityCode": EdocsApi.getAttributeValue('ContractorCode').value,  "contactPersonEmail": EdocsApi.getAttributeValue('ContractorRPEmail').value,  "expectedSignatures": []  }],  "additionalAttributes": [  {  "code": "docDate",  "type": "dateTime",  "value": registrationDate  },  {  "code": "docNum",  "type": "string",  "value": registrationNumber  }  ],  sendingSettings: {  attachFiles: 'fixed', //, можна також встановлювати 'firstOnly' - Лише файл із першої зафіксованої вкладки(Головний файл), або 'all' - всі файли, 'fixed' - усі зафіксовані  attachSignatures: 'signatureAndStamp' // -'signatureAndStamp'Типи “Підпис” або “Печатка”, можна також встановити 'all' - усі типи цифрових підписів  }  }  EdocsApi.setAttributeValue({code: 'JSON', value: JSON.stringify(doc)});  }  function onTaskExecuteSendOutDoc(routeStage) {  debugger;  if (routeStage.executionResult == 'rejected') {  return;  }  setDataForESIGN();  var idnumber = EdocsApi.getAttributeValue('DocId');  var methodData = {  extSysDocId: idnumber.value  };  routeStage.externalAPIExecutingParams = {  externalSystemCode: 'ESIGN1', // код зовнішньої системи  externalSystemMethod: 'integration/importDoc', // метод зовнішньої системи  data: methodData, // дані, що очікує зовнішня система для заданого методу  executeAsync: true // виконувати завдання асинхронно  }    }  function onTaskCommentedSendOutDoc(caseTaskComment){  debugger;  var orgCode = EdocsApi.getAttributeValue('OrgCode').value;  var orgShortName= EdocsApi.getAttributeValue('OrgShortName').value;  if(!orgCode || !orgShortName){  return;  }  var idnumber = EdocsApi.getAttributeValue('DocId');  var methodData = {  extSysDocId:idnumber.value,  eventType:"CommentAdded",  comment: caseTaskComment.comment,  partyCode: orgCode,  userTitle: CurrentUser.name,  partyName: orgShortName,  occuredAt: new Date()  };  caseTaskComment.externalAPIExecutingParams = {  externalSystemCode: 'ESIGN1', // код зовнішньої системи  externalSystemMethod: 'integration/processEvent', // метод зовнішньої системи  data: methodData, // дані, що очікує зовнішня система для заданого методу  executeAsync: true // виконувати завдання асинхронно  }  } | | | | | | | |
| **Опис сценарію (що потрібно реалізувати за допомогою JS. Обов’язково використовуючи «ЯКЩО» -> «ТО» -> «ІНАКШЕ» і тригером (умови(a), або дія при яких має виконатись сценарій)** | | | | | | | |
| ЯКЩО призначене завдання SendOutDoc, ТО документ передається на підпис в зовнішню систему. | | | | | | | |
| **Інформування користувача** | | | | | | | |
| **Передача даних в зовн.систему** | | | | | | | |
| **Код зовн.системи** | | ESIGN1 | | **Метод зовн.системи** |  | | |
|  |  |  |  | **Завдання, яке активує передачу** | SendOutDoc | |
| **Атрибути картки, які задіяні в JS** | | | | | | | |
| **Назва** | | **Код** | | **Тип в SP** | **Сист.атрибут** | **Код довідника** | |
| Номер рахунку | | RegDate | | Дата | - | - | |
| Дата рахунку | | RegNumber | | Однорядковий текст |  |  | |
| Тип документу | | DocType | | Однорядковий текст | Документ.Тип документа |  | |
| Вид документу | | DocKind | | Однорядковий текст | Документ.Вид документу |  | |
| Код ЄДРПОУ | | OrgCode | | Однорядковий текст | Домашня юр. особа.Код ЄДРПОУ |  | |
| Email контактної особи Організації | | OrgRPEmail | | Багаторядковий текст | - |  | |
| Код ЄДРПОУ | | ContractorEDRPOU | | Однорядковий текст | Контрагент.Код ЄДРПОУ |  | |
| Email контактної особи Контрагента | | ContractorRPEmail | | Однорядковий текст | - |  | |
| DocId | | DocId | | Однорядковий текст | Документ.Ідентифікатор |  | |
| Скорочене найменування | | OrgShortName | | Однорядковий текст | Домашня юр. особа.Назва |  | |
| **Завдання, які задіяні в JS** | | | | | | | |
| **Назва** | | **Код** | | **Тип виконавця (співробітник/роль/роль з розрізом/виконавець за атрибутом** |  | **Поле для розрізу або виконавця за атрибутом** | |
| Відправка Контрагенту | | SendOutDoc | | Роль – Ініціатор | - | - | |

Скрипт 3. Автоматичне визначення email ініціатора рахунку та підрозділу

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Опис задачі (Що необхідно зробити, Як буде тестуватись, Очікуваний результат)** | | | | |
| При створенні карточки автоматично має заповнюватись:  1) поле OrgRPEmail даними із орг структури Ініціатора документу, якщо такі дані є в орг структурі.  2) поле Branch даними структурного підрозділу рівня ‘1’. | | | | |
| **Опис сценарію (що потрібно реалізувати за допомогою JS. Обов’язково використовуючи «ЯКЩО» -> «ТО» -> «ІНАКШЕ» і тригером (умови(a), або дія при яких має виконатись сценарій)** | | | | |
|  | | | | |
| **Атрибути картки, які задіяні в JS** | | | | |
| **Назва** | **Код** | **Тип в SP** | **Сист.атрибут** | **Код довідника** |
| Email контактної особи Організації | OrgRPEmail | Багаторядковий текст | - |  |

Скрипт 4. Зміна властивостей атрибутів

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Опис задачі (Що необхідно зробити, Як буде тестуватись, Очікуваний результат)** | | | | |
| При призначенні завдання EnterInformation поле StatusInvoice має стати видимим та обов’язковим для заповнення.  Виконавець даного завдання береться із розрізів (можливо, це важливо) | | | | |
| **Опис сценарію (що потрібно реалізувати за допомогою JS. Обов’язково використовуючи «ЯКЩО» -> «ТО» -> «ІНАКШЕ» і тригером (умови(a), або дія при яких має виконатись сценарій)** | | | | |
|  | | | | |
| **Атрибути картки, які задіяні в JS** | | | | |
| **Назва** | **Код** | **Тип в SP** | **Сист.атрибут** | **Код довідника** |
|  |  |  |  |  |

Скрипт 5. Визначення ролі за розрізом

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Опис задачі (Що необхідно зробити, Як буде тестуватись, Очікуваний результат)** | | | | |
| Необхідно реалізувати автоматичне визначення Фінансової служби РФ (роль із розрізом) на основі даних структурного підрозділу із поля Branch. Фінансова служба РФ має визначатись для виконання завдання EnterInformation.  Додаткова інформація: В системі налаштована роль Фінансова служба РФ із розрізами (для кожної філії – окремий відповідальний). Також в карточку додане приховане поле «Розрізи» (Sections) із системним атрибутом «Розрізи». | | | | |
| **Опис сценарію (що потрібно реалізувати за допомогою JS. Обов’язково використовуючи «ЯКЩО» -> «ТО» -> «ІНАКШЕ» і тригером (умови(a), або дія при яких має виконатись сценарій)** | | | | |
|  | | | | |
| **Атрибути картки, які задіяні в JS** | | | | |
| **Назва** | **Код** | **Тип в SP** | **Сист.атрибут** | **Код довідника** |
|  |  |  |  |  |