Краткие алгоритмы на этой странице  
  
  
1) ждем полной загрузки страницы  
2) если есть поп ап нажимаем на продолжить с USD  
3) заполняем таможеную информацию  
4) подверждаем номер карты если есть  
5) проверяем целостность позиций товаров   
6) нажимаем на кнопку оформить заказ.   
  
  
1) после того как на предыдущей странице мы нажали на кнопку оформить заказ мы попадаем на страницу предварительного просмотра и заполнения таможни, его адрес такой: <https://checkout12.iherb.com/scd>  
однако по URL не переходим, переход будет после окончания работы модуля cartPage, то есть работу начинаем на активной странице.   
сначала подождем полной загрузки страницы.

2) делаем проверку, если появился поп ап <div class="Modal\_\_ModalContainer-sc-af2llh-0 fsmbOd"><div class="ui\_\_Div-sc-1osfsge-3 cQXkrr"><div class="Modal\_\_ModalTitle-sc-af2llh-2 hHlRHm">Подтверждение валюты <svg width="24" height="24" viewBox="0 0 24 24" fill="none" xmlns="http://www.w3.org/2000/svg" aria-hidden="true" role="img" id="modal-close" class="Modal\_\_ModalLinedCloseLIcon-sc-af2llh-5 kZpApI"><path fill-rule="evenodd" clip-rule="evenodd" d="m12 10.587 7.293-7.294a1 1 0 1 1 1.414 1.414l-7.292 7.294 7.292 7.292a1 1 0 0 1-1.414 1.414L12 13.415l-7.294 7.296a1 1 0 1 1-1.414-1.414L10.587 12 3.293 4.707a1 1 0 0 1 1.414-1.414l7.294 7.294Z" fill="currentColor"></path></svg></div><div class="Modal\_\_ModalContent-sc-af2llh-3 jQWqjw"><div class="ConnectedCurrencyConfirmationModal\_\_Text-sc-1ct4ad4-0 CsdCa">Ваш текущий выбор валюты, USD, не соответствует вашей местной валюте KZT. Подтвердите, какую валюту вы хотите использовать для оформления заказа.</div><div class="Modal\_\_ModalButtonWrapper-sc-af2llh-4 jIeAzT"><button class="sc-dkzDqf ibpgzt MuiButtonBase-root MuiButton-root MuiButton-outlined MuiButton-outlinedInfo MuiButton-sizeMedium MuiButton-outlinedSizeMedium MuiButton-disableElevation sc-hKMtZM hDgTIA" tabindex="0" type="button" id="continue-ccl-button">Продолжить с USD<span class="sc-bczRLJ jdQlYe MuiTouchRipple-root"></span></button><button class="sc-dkzDqf ibpgzt MuiButtonBase-root MuiButton-root MuiButton-contained MuiButton-containedPrimary MuiButton-sizeMedium MuiButton-containedSizeMedium MuiButton-disableElevation sc-hKMtZM gDhTLx" tabindex="0" type="button" id="switch-ccl-button">Изменить на KZT<span class="sc-bczRLJ jdQlYe MuiTouchRipple-root"></span></button></div></div></div></div>  
нажимаем на кнопку button#continue-ccl-button

3) заполняем таможеную информацию, для этого сначала нужно нажать на кнопку изменить #customs-panel-toggle-button > #collapse-label  
затем отрисуются инпуты что я написал ниже, нужно убедится что инпуты доступны для заполнения.   
берем того же user а что брали при установке адресной книги, то есть из config.js поле defaultAdress.  
например (это просто пример)   defaultAdress: "Саудагерова Айзат",   
он должен взять Фамилию и имя и найти этого пользователя в adressList.JSON.   
например, этот объект подходит:  
 {

    "FullName": "Саудагерова Айзат Толешовна",

    "Name": "Айзат",

    "MiddleName": "Толешовна",

    "Surname": "Саудагерова",

    "Pasport": 850507401671,

    "Street": "Ак Жайык 113А Уй",

    "City": "Каскелен",

    "Region": "Алматинская Область",

    "Post": "040904",

    "Number": 77083036267

  },

Далее он должен заполнить отсюда номер паспорта, телефона и другие данные из этого объекта   
  
заполняем из adressList.JSON  
input[name="identificationNumber"] – Pasport   
input[id="firstName"] – Name   
input[name="midName"] – MiddleName   
input[name="lastName"] – Surname   
input#mobileNumber – Number

Пример файла: adressList.JSON (это только пример, не надо захардкоживать никакие даннные, повторяю хардкод запрещен)  
[

  {

    "FullName": "Толешов Бауыржан Айдосулы",

    "Name": "Бауыржан",

    "MiddleName": "Айдосулы",

    "Surname": "Толешов",

    "Pasport": 960722301895,

    "Street": "Бакинский Бурылысы 1Г Уй",

    "City": "Каскелен",

    "Region": "Алматинская Область",

    "Post": "040901",

    "Number": 77077751996

  },

Когда удостоверимся что все инпуты были полностью заполнены имитацией чел ввода и что ни одна буква не была пропущена, нажимаем на кнопку продолжить: button.MuiButton-disableElevation:nth-child(2)  
  
затем проверяем что все информация соовтетвует по апи  
<https://checkout12.iherb.com/api/customs/v1/EMEXDelivery?countryCode=KZ>  
URL запроса:

https://checkout12.iherb.com/api/customs/v1/EMEXDelivery?countryCode=KZ

Метод запроса:

GET

Код статуса:

200 OK

Удаленный адрес:

104.18.38.11:443

Правило для URL перехода:

strict-origin-when-cross-origin

ответ:  
{

"customsData": {

"IdentificationType": "ID",

"idValidationType": "ID",

"PassportCountryCode": "KZ",

"IndividualIDNumber": " 960722301895",

"FirstName": " Бауыржан",

"PatronymicName": " Айдосулы",

"LastName": " Толешов",

"TelNumber": "+ 77077751996",

"Save": "true"

},

"active": "true",

"countryCode": "KZ",

"serviceName": "EMEXDelivery"

}

Затем ждем окончания подгрузки спиннера: <div class="ConnectedLoading\_\_SpinnerWrapper-sc-1245kmi-0 dFApI"><div class="Spinner\_\_Wrapper-sc-1f2iq4l-0 lpsYbk"><svg width="24" height="24" viewBox="0 0 24 24" fill="none" xmlns="http://www.w3.org/2000/svg" aria-hidden="true" role="img" class="Spinner\_\_StyledLeaf-sc-1f2iq4l-1 hbxNqB"><path fill-rule="evenodd" clip-rule="evenodd" d="M8.84 16.64c-1.697-.887-4.573-6.99-.11-10.128 5.51-3.87 12.11-3.237 12.295-3.219h.004s.791 6.431-.754 10.579c-2.447 6.57-8.387 4.905-10.567 3.439.224-.214 2.119-2.86 2.119-2.86s5.527 1.602 5.593.146c.06-1.35-5.374-.597-5.374-.597l1.683-2.037s4.153.765 4.153-.187c0-1.464-3.878-.428-3.878-.428l1.807-2.118s2.35.518 2.588-.357c.28-1.025-2.247-.12-2.247-.12s3.001-2.334 2.326-3.042c-.815-.855-2.643 2.657-2.643 2.657s.21-2.195-.613-2.145c-1.079.065.075 2.629.075 2.629l-1.651 2.091s.507-4.039-1.064-3.153c-.41.232-.093 1.011.225 1.794.28.688.56 1.38.349 1.704-.453.694-1.72 2.232-1.72 2.232s.12-1.13-.091-2.155c-.21-1.025-.912-2.262-1.357-1.887-.428.36-.067 1.118.335 1.96.439.921.926 1.943.48 2.655-.382.61-.924 1.27-1.397 1.845-.209.255-.405.493-.567.703Zm-6.042 2.267c-.12-.648.144-1.224.829-1.297.622-.065.821.17 1.04.43.223.262.465.547 1.187.569 1.219.035 2.452-1.276 2.984-1.967.137.124.664.532.87.67-1.332 2.134-2.81 3.472-4.629 3.263-1.606-.184-2.131-.857-2.281-1.668Z" fill="#458500"></path></svg><svg viewBox="25 25 50 50" class="SvgIcon-sc-1jecnkl-0 edERTo Spinner\_\_StyledCircular-sc-1f2iq4l-2 cMiaLi" color="inherit" height="24" width="24"><circle cx="50" cy="50" r="20" fill="none" stroke-width="2" stroke-miterlimit="10"></circle></svg></div></div>

4) проверяем просит ли сайт повторно ввести номер карты или можно продолжить без этого.  
Если он просит ввести номер карты заново, то выйдет такое предупреждение:  
<div data-testid="panel-error" class="SavedPaymentMethod\_\_ErrorWrapper-sc-1s95kl0-1 cnKBSR panel-error"><div type="ERROR" class="ui\_\_Div-sc-1osfsge-3 Alert\_\_Container-sc-1a3q635-0 cQXkrr evZrcE"><svg width="24" height="24" viewBox="0 0 24 24" fill="none" xmlns="http://www.w3.org/2000/svg" aria-hidden="true" role="img" class="Alert\_\_ErrorIcon-sc-1a3q635-1 bOMmRv"><path fill-rule="evenodd" clip-rule="evenodd" d="M12 2C6.477 2 2 6.477 2 12s4.477 10 10 10 10-4.477 10-10S17.523 2 12 2Zm0 2a8 8 0 1 1 0 16 8 8 0 0 1 0-16Zm0 10a1 1 0 0 0 1-1V7a1 1 0 1 0-2 0v6a1 1 0 0 0 1 1Zm0 2a1 1 0 1 0 0 2 1 1 0 0 0 0-2Z" fill="currentColor"></path></svg><div class="InnerHtml\_\_Content-sc-9fr40b-0">Необходимо повторно ввести номер банковской карты для дополнительной проверки на безопасность при отправлении заказа по этому адресу в первый раз.</div></div></div>

вводим номер карты в этот инпут: .js-iframe |> input#encryptedCardNumber  
сначало кликаем по нему, затем вводим номер карты имитируя чел ввод puppeteer  
обязательно нужно использовать методы ввода puppeteer потому что стандартные методы ввода JS не работает на этой странице, по этому все методы ввода, клика и так далее должны быть использованы методы Puppeteer/   
  
  
номер карты которую нужно брать сейчас напишу.   
Сначало что бы знать какой номер нам нужно брать , смотрим на сайт, сайт показывает 4 последних цирфы карты которую мы должны ввести:  
  
 вот сюда <div class="CreditCardPayment\_\_CardDigitsExpiry-sc-lcaa6f-2 gDLHRb">\*\*\*\* 1261 Exp. 7/2027</div>  
  
где 1261 является последней 4 цифрой нужной нам карты.  
Естественно там могут быть любые 4 цифры, 1261 я взял только для примера, и вообще почти все данные у меня только для примера, кроме селектором на странице или апи, по этому запрещаю тебе хардкод  
 Нам нужно просто ввести номер карты, номер карты берем отсюда:  
В файле config.js есть список номеров моих карт:  
 cards: [

      '5191702005971261',

      '4628188880881898'

    ],

Здесь в данном случае берем карту '5191702005971261' потому что 4 последние цифры совпадают с тем что просит у нас айхерб/  
Тут тоже не надо делать хардкод, в файле config.js могут быть любые карты, если искомый номер карты не нашли, то пишем что карта не найден, ждем ручной ввод карты и дожидаемся ввод пользователя, сигналом окончания будет то что он нажмет кнопку продолжить сам, далее программа должен работать сам.  
  
после того как ввели номер карты и удостоверились что все нормально ввелось и не пропустили ни единой цифры, нажимаем на кнопку продолжить после того как ввели номер карты, button#credit-card-continue-button используя исключительно метод puppeteer/

После того как нажали ждем спиннера подгрузки, ждем точечно по классу или просто ждем пару секунд и проверяем доступность страницы.   
  
  
5) для этого нам нужно получить сумму корзины из предыдущей страницы cartPage/   
его нам нужно что бы передал cartPage.js или checkSumm.js  
смысл заключается в том что бы проверить что бы сумма заказа из этой страницы совпадала с суммой корзины .   
Сумма корзины находится здесь: <div class="LineItem-sc-wlrx-1 bzMOFj">$155.71</div>  
Но, это не полная сумма, мы также должны добавить сюда сумму вознаграждений как на странице корзины:  
<div color="#BD3C37" class="LineItem-sc-wlrx-1 bpbjEF">-$35.00</div>  
  
6) Если все суммы совпадают, то нажимаем на кнопку и все элементы и кнопку доступны, нету никаких спиннеров, нажимаем на кнопку разместить заказ: button#place-order-button  
  
  
7) если все прошло нормально, то мы попадаем на страницу инвойса, мы должны подождать загрузку страницы с помощью утилиты  
<https://checkout12.iherb.com/scd/order-receipt?on=534001026&sca=1>  
страница не всегда будет иметь такой адрес, поскольку он содержит номер заказа

И вот здесь нам нужно получить а также записать в JSON номер заказа, сумму корзины, адрес заказа, имя клиента, дату.  
во время перехода на страницу получаем апи:  
URL запроса:

https://checkout12.iherb.com/api/checkout/order?OrderNumber=534001258

Метод запроса:

GET

Код статуса:

200 OK

Удаленный адрес:

172.64.149.245:443

Правило для URL перехода:

strict-origin-when-cross-origin

В ответе получаем объект, где нам нужно записать данные:  
{

"orderNumber": "534001258",

"smsOptedIn": false,

"paidRewardTotal": 35.0000,

"formattedPaidRewardTotal": "-$35.00",  
…….

"emailAddress": "zunaika2011@gmail.com",

"usedRewardsCredit": "-$35.00",

"orderTotal": "$40.33",

"shippingAddress": {

"addressID": 1723794116,

"firstName": "Абдразаков Қайрат",

"fullName": "Абдразаков Қайрат ",

orderNumber – номер заказа  
formattedPaidRewardTotal – использованные вознаграждения   
orderTotal – сумма корзины

fullName – адрес заказа (имя, обязательно должны быть внутри shippingAddress)  
emailAddress – аккаунт заказа  
  
  
он должен записать данные только после успешного заказа.   
записать должен здесь …CRM\data\OrdersData\ordersData.json  
  
Структура файла такая:  
{

  "orders": [

    {

      "orderNumber": "532110706",

      "trackCode": "722500351310",

      "date": "2024-12-01T10:00:00Z",

      "price": {

        "rub": 10000,

        "commission": 10400,

        "usd": 100

      },

      "orderId": 10,

      "client": "Иван",

      "paymentMethod": "мобильный перевод",

      "paymentStatus": "не оплачен",

      "orderAccount": "sony2020kz@proton.me",

      "cartLink": "https://kz.iherb.com/tr/cb?pcodes=CDL-10100qty1\_NAP-29928qty1&v=2&rcode=HIY3376",

      "orderSystemStatus": "заказ обработан",

      "iherbStatus": "выполняется обработка",

      "trackCodeStatus": "Готово к отправке",

      "deliveryAddress": "Сагимбаева Гульнара Назарбаева",

      "promoCodeUsed": "Gold120",

      "referralCodeUsed": "DUA9670",

      "cardUsed": "1238",

      "rewardsUsed": 0

    },

Если есть доп вопросы, или нужен код каких то файлов или апи, спрашивай, отвечу на любые вопросы .   
  
Также для работы с этой функцией создаем файл preOrderPage.js внутри папки preOrderPage  
preOrderPage будет просто главной функцией для управления остальными функциями, раздели фнукцию на несколько файлов внутри той же папки preOrderPage согласно по отвественностям и логике.