```
/**
* @fileOverview Checkout JavaScript Function Library.
* @author <a href="https://github.com/richardhenyash">Richard Ash</a>
* @version 1.1.1
*/
/*jshint esversion: 6 */
/* globals $, Stripe, buildInformationModal */

// On click event handler added to delivery information button to build modal dialog
$("#deliveryInformationBtn").click(function() {
    (buildInformationModal("#deliveryInformationBtn", "information-modal-title", null, "#i
});

// Accepting a payment in Stripe: https://stripe.com/docs/payments/accept-a-payment
// Stripe.js and Stripe elements: https://stripe.com/docs/stripe-js
// Using google fonts with stripe: https://stackoverflow.com/questions/43824382/custom-fon

let stripePublicKey = $('#id_stripe_public_key').text().slice(1, -1);
let clientSecret = $('#id_client_secret').text().slice(1, -1);
// Instantiate stripe
let stripe = Stripe(stripePublicKey);
// Set stripe elements
let elements = stripe.elements({
  fonts: [
    {
      // integrate Google Fonts Montserrat into stripe
      cssSrc: 'https://fonts.googleapis.com/css?family=Montserrat:400,500,600',
    }
  ]
});
// Set style
let style = {
    base: {
        color: '#FAFAFA',
        fontFamily: 'Montserrat, sans-serif',  // set integrated font family
        fontSmoothing: 'antialiased',
        fontWeight: '500',
        fontSize: '14px',
        '::placeholder': {
          color: '#CCCCCC'
        }
    },
    invalid: {
        color: '#FFAD99',
        iconColor: '#FFAD99'
    }
};
// Create card element
let card = elements.create('card', {style: style, hidePostalCode: true});
// Mount card
card.mount('#card-element');

// Handle realtime validation errors on the card element
card.addEventListener('change', function (event) {
    let errorDiv = document.getElementById('card-errors');
    // If there is an error
    if (event.error) {
        let html = `
            <p><span class="icon" role="alert">
                <i class="fas fa-times"></i>
            </span>
            <span>${event.error.message}</span></p>
        `;
        // Update errorDiv with error
        $(errorDiv).html(html);
    } else {
        // Update errorDiv with blank string
        errorDiv.textContent = '';
    }
});

// Handle form submit
```

CONFIGURE

Metrics

There are 6 functions in this file.

Function with the largest signature take 1 arguments, while the median is 0.5.

Largest function has 11 statements in it, while the median is 3.

The most complex function has a cyclomatic complexity value of 3 while the median is 1.

version 2.13.1 (https://github.com/jshint/j

About (/about)

Documentation (/docs)

Install (/install)

Contribute (/contribute)

Blog (/blog)

```
let form = document.getElementById('payment-form');
form.addEventListener('submit', function (ev) {
    ev.preventDefault();
    card.update({
        'disabled': true
    });
    $('#submit-button').attr('disabled', true);
    //Fade out payment form
    $('#payment-form').fadeToggle(100);
    // Fade in loader overlay
    $('#loader-overlay').fadeToggle(100);

    // Get save_info checkbox value
    let saveInfo = Boolean($('#id-save-info').attr('checked'));
    // Get csrf token from using {% csrf_token %} in the form
    let csrfToken = $('input[name="csrfmiddlewaretoken"]').val();
    // Get discount
    let discount = $('#discount').data('discount');
    // Set post data
    let postData = {
        'csrfmiddlewaretoken': csrfToken,
        'client_secret': clientSecret,
        'save_info': saveInfo,
        'discount': discount,
    };
    // Set url
    let url = '/checkout/cache_checkout_data/';

    $.post(url, postData).done(function () {
        // Confirm card payment
        stripe.confirmCardPayment(clientSecret, {
            // Set payment method
            payment_method: {
                card: card,
                // Set billing details
                billing_details: {
                    name: $.trim(form.full_name.value),
                    phone: $.trim(form.phone_number.value),
                    email: $.trim(form.email.value),
                    address: {
                        line1: $.trim(form.address_1.value),
                        line2: $.trim(form.address_2.value),
                        city: $.trim(form.town_or_city.value),
                        state: $.trim(form.county.value),
                        country: $.trim(form.country.value),
                    }
                }
            },
            // Set shipping details
            shipping: {
                name: $.trim(form.full_name.value),
                phone: $.trim(form.phone_number.value),
                address: {
                    line1: $.trim(form.address_1.value),
                    line2: $.trim(form.address_2.value),
                    city: $.trim(form.town_or_city.value),
                    state: $.trim(form.county.value),
                    country: $.trim(form.country.value),
                    postal_code: $.trim(form.postcode.value),
                }
            },
        }).then(function (result) {
            if (result.error) {
                // Display any card errors
                let errorDiv = document.getElementById('card-errors');
                let html = `
                <p><span class="icon" role="alert">
                <i class="fas fa-times"></i>
                </span>
                <span>${result.error.message}</span></p>`;
                $(errorDiv).html(html);
                // Fade in payment form
                $('#payment-form').fadeToggle(100);
                // Fade out loader overlay
```

CONFIGURE

Metrics

There are 6 functions in this file.

Function with the largest signature take 1 arguments, while the median is 0.5.

Largest function has 11 statements in it, while the median is 3.

The most complex function has a cyclomatic complexity value of 3 while the median is 1.

JSHint

version 2.13.1 (https://github.com/jshint/j

- About (/about)
- Documentation (/docs)
- Install (/install)
- Contribute (/contribute)
- Blog (/blog)

```
                $('#loader-overlay').fadeToggle(100);
                card.update({
                    'disabled': false
                });
                // Enable submit button
                $('#submit-button').attr('disabled', false);
            } else {
                // Submit form if payment intent is succesfull
                if (result.paymentIntent.status === 'succeeded') {
                    form.submit();
                }
            }
        });
    }).fail(function () {
        // reload the page - the error will be in django messages
        location.reload();
    });
});
```

CONFIGURE

Metrics

There are 6 functions in this file.

Function with the largest signature take 1 arguments, while the median is 0.5.

Largest function has 11 statements in it, while the median is 3.

The most complex function has a cyclomatic complexity value of 3 while the median is 1.

version 2.13.1
(https://github.com/jshint/js

- About (/about)
- Documentation (/docs)
- Install (/install)
- Contribute (/contribute)
- Blog (/blog)