body {

background-color: #645846;

font-size: .857em;

-webkit-font-smoothing: subpixel-antialiased; }

body:before, body:after {

content: "";

display: none;

position: fixed;

top: 0;

left: 0;

width: 100%;

height: 1em;

background-color: rgba(0, 0, 0, 0.05);

z-index: 99999; }

body:after {

top: auto;

bottom: 0; }

body.admin-bar:before {

top: 32px; }

html:before, html:after {

content: "";

display: none;

position: fixed;

top: 0;

left: 0;

height: 100%;

width: 1em;

background-color: rgba(0, 0, 0, 0.05); }

html:after {

left: auto;

right: 0; }

body,

button,

input,

select,

textarea {

font-family: "Anonymous Pro", monospace;

font-weight: 400; }

h1,

h2,

h3,

h4,

h5,

h6 {

font-weight: 400;

letter-spacing: 0 !important;

font-family: "Copse", sans-serif; }

h1 a,

h1 label,

h2 a,

h2 label,

h3 a,

h3 label,

h4 a,

h4 label,

h5 a,

h5 label,

h6 a,

h6 label {

font-weight: 400; }

h1,

h2,

h3 {

font-weight: 400; }

.widget h1,

.widget h2,

.widget h3,

.widget h4,

.widget h5,

.widget h6 {

font-weight: 400; }

.site-header,

.site-footer {

background-color: transparent !important; }

a {

font-weight: 400; }

button,

input[type="button"],

input[type="reset"],

input[type="submit"],

.button,

.added\_to\_cart,

.widget-area .widget a.button,

.site-header-cart .widget\_shopping\_cart a.button {

box-shadow: none;

border-radius: 3px;

font-weight: 400;

-webkit-font-smoothing: antialiased;

font-family: "Oswald", sans-serif; }

input[type="text"],

input[type="email"],

input[type="url"],

input[type="password"],

input[type="search"],

.input-text {

border-radius: 0;

border: 2px dashed rgba(0, 0, 0, 0.15);

padding-left: 1em;

background: #f6f6f6;

color: #777;

box-shadow: none; }

textarea.input-text {

border-radius: 3px; }

img.avatar {

padding: 3px;

background: #fff; }

.site-branding .site-title {

letter-spacing: 0;

font-weight: 700;

font-family: "Copse", sans-serif; }

.site-branding .site-title a {

font-weight: 400; }

button.menu-toggle {

margin-bottom: 0; }

.storefront-primary-navigation {

background-color: transparent; }

.deli-primary-navigation {

-webkit-font-smoothing: antialiased;

margin-bottom: 1.618em;

text-shadow: none;

font-family: "Oswald", sans-serif; }

.deli-primary-navigation::after {

clear: both;

content: "";

display: table; }

.deli-primary-navigation a {

font-weight: 400; }

.woocommerce-breadcrumb {

border-bottom: 0;

color: rgba(255, 255, 255, 0.7);

text-shadow: 0 1px 1px rgba(0, 0, 0, 0.2);

margin-bottom: 2.618em;

position: relative;

margin-left: 0;

margin-right: 0;

padding-left: 0;

padding-right: 0; }

.woocommerce-breadcrumb a {

font-weight: 400;

color: rgba(255, 255, 255, 0.9); }

.woocommerce-breadcrumb a:hover {

text-decoration: underline; }

.no-wc-breadcrumb .site-header {

margin-bottom: auto; }

table th,

#payment .payment\_methods > li {

background-color: #fff !important; }

table tbody td,

fieldset,

fieldset legend {

background-color: #fafafa !important; }

table tbody tr:nth-child(2n) td,

#payment .payment\_methods > li .payment\_box, #payment .place-order {

background-color: #f6f6f6 !important; }

.woocommerce-tabs {

border-style: dotted;

border-width: 2px 0;

border-color: rgba(0, 0, 0, 0.1); }

.col-full,

.header-widget-region {

box-sizing: content-box; }

.home.blog .site-header,

.home.page:not(.page-template-template-homepage) .site-header {

margin-bottom: 0; }

.header-widget-region {

margin-bottom: 1.618em; }

.header-widget-region .col-full {

padding: 0 1em; }

.header-widget-region .widget {

font-size: 1em; }

.header-widget-region .widget h1,

.header-widget-region .widget h2,

.header-widget-region .widget h3,

.header-widget-region .widget h4,

.header-widget-region .widget h5,

.header-widget-region .widget h6 {

border: 0;

padding-bottom: 0; }

.col-full {

padding: 0; }

.deli-content-wrapper::after {

clear: both;

content: "";

display: table; }

.storefront-product-section {

border-bottom-width: 2px;

border-bottom-style: dotted;

border-bottom-color: rgba(0, 0, 0, 0.1);

margin-bottom: 2.618em;

padding-top: 1em; }

.storefront-product-section h2.section-title:before, .storefront-product-section h2.section-title:after {

display: inline-block;

content: " - ";

font-size: .618em;

margin: 0 1em;

opacity: 0.5;

font-weight: 700; }

.storefront-product-section ul.products {

margin-bottom: 0; }

.hentry .entry-header {

border: 0;

position: relative; }

.hentry .entry-header h1 {

border-bottom-style: double;

border-bottom-width: 4px;

border-bottom-color: #ddd;

font-size: 2.618em; }

.hentry .entry-header:after, .hentry .entry-header:before {

content: "";

height: 4px;

width: 4px;

line-height: 1;

position: absolute;

bottom: 0px;

left: 0;

background: #ddd; }

.hentry .entry-header:after {

left: auto;

right: 0; }

.content-area {

background: #f9f9f9;

position: relative;

margin-bottom: 3.631em !important;

padding: 1em; }

.content-area:after, .content-area:before {

content: "";

display: block;

border: 2.618em solid transparent;

position: absolute;

top: 0;

right: 0;

border-right-color: #eee;

border-top-width: 0; }

.content-area:before {

border-right-color: transparent;

border-bottom-color: #fff;

border-left-width: 0; }

.content-area .site-main:before {

content: "";

display: block;

position: absolute;

bottom: -9px;

left: 0;

width: 100%;

height: 9px;

background: -webkit-linear-gradient(transparent 0%, transparent 0%), -webkit-linear-gradient(135deg, #f9f9f9 33.33%, transparent 33.33%) 0 0%, transparent -webkit-linear-gradient(45deg, #f9f9f9 33.33%, transparent 33.33%) 0 0%;

background-size: 0px 100%, 9px 9px, 9px 9px;

-webkit-transform: rotate(180deg);

-moz-transform: rotate(180deg);

-ms-transform: rotate(180deg);

-o-transform: rotate(180deg);

transform: rotate(180deg); }

.site-header {

text-shadow: 0 1px 1px rgba(0, 0, 0, 0.2);

padding: 1.618em;

padding-bottom: 0;

border-bottom: 0; }

.site-search .widget\_product\_search input[type=search],

.site-search .widget\_product\_search input[type=text],

.header-widget-region .widget\_product\_search input[type=search],

.header-widget-region .widget\_product\_search input[type=text] {

background: rgba(0, 0, 0, 0.65);

color: #fff;

text-shadow: 0 1px 0 rgba(0, 0, 0, 0.5);

box-shadow: inset 0 1px 3px rgba(0, 0, 0, 0.3), 0 1px 1px rgba(255, 255, 255, 0.2); }

.site-footer {

padding: 0; }

.site-footer .col-full {

padding: 1.618em;

background: transparent !important; }

.site-footer a:hover {

text-decoration: underline; }

.sprh-hero .col-full,

.sph-hero .col-full {

padding: 1.618em;

-webkit-font-smoothing: antialiased; }

.page-template-template-homepage .site-main .sph-hero:first-child,

.page-template-template-homepage .site-main .sprh-hero:first-child {

margin-top: 0; }

#comments .comment-list .comment-content .comment-text {

background-color: rgba(0, 0, 0, 0.0125); }

.comment-text .description {

padding: 1em;

background: #f5f2e8;

border: 0;

box-shadow: 0 1px 2px rgba(0, 0, 0, 0.2);

border-top: 1.618em solid rgba(0, 0, 0, 0.025);

font-family: "Kalam"; }

#reviews .commentlist li {

border: 0; }

.site-main ul.products li.product h3 {

font-size: 1.2em; }

.storefront-sorting {

border-style: dotted;

border-width: 2px 0;

border-color: rgba(0, 0, 0, 0.1);

padding-top: 1em;

padding-bottom: 1em; }

.woocommerce-breadcrumb {

margin-top: 0;

background-color: transparent !important; }

.site-header-cart {

margin-bottom: 0; }

.single-product div.product .summary .price {

font-weight: 400;

background-color: #0e7784;

color: #fff;

display: inline-block;

padding: .53em 2.244em .53em 1em;

border-bottom-right-radius: 2em;

border-top-right-radius: 2em;

position: relative; }

.single-product div.product .summary .price:after {

display: block;

position: absolute;

content: "";

top: 50%;

right: 1em;

height: 6px;

width: 6px;

margin-top: -3px;

background-color: #fff;

border-radius: 100%; }

table.cart td.actions {

border: 0; }

.widget-area .widget {

padding: 2.618em 1em 1em 1em;

position: relative;

background: rgba(255, 255, 255, 0.85);

font-family: "Kalam"; }

.widget-area .widget:after {

content: "";

display: block;

position: absolute;

bottom: -7px;

left: 0;

width: 100%;

height: 7px;

background: -webkit-linear-gradient(transparent 0%, transparent 0%), -webkit-linear-gradient(135deg, rgba(255, 255, 255, 0.85) 33.33%, transparent 33.33%) 0 0%, transparent -webkit-linear-gradient(45deg, rgba(255, 255, 255, 0.85) 33.33%, transparent 33.33%) 0 0%;

background-size: 0px 100%, 9px 9px, 9px 9px;

-webkit-transform: rotate(180deg);

-moz-transform: rotate(180deg);

-ms-transform: rotate(180deg);

-o-transform: rotate(180deg);

transform: rotate(180deg); }

.widget-area .widget:before {

content: "";

display: block;

height: 2.618em;

width: 30%;

position: absolute;

top: -1.309em;

left: 35%;

background: url(images/paper.png) rgba(255, 255, 255, 0.3);

-webkit-transform: rotate(1deg);

-moz-transform: rotate(1deg);

-ms-transform: rotate(1deg);

-o-transform: rotate(1deg);

transform: rotate(1deg); }

.widget-area .widget:nth-child(2n):before {

-webkit-transform: rotate(2deg);

-moz-transform: rotate(2deg);

-ms-transform: rotate(2deg);

-o-transform: rotate(2deg);

transform: rotate(2deg); }

.widget-area .widget:nth-child(3n):before {

-webkit-transform: rotate(-3deg);

-moz-transform: rotate(-3deg);

-ms-transform: rotate(-3deg);

-o-transform: rotate(-3deg);

transform: rotate(-3deg);

left: 30%; }

.widget-area .widget:nth-child(5n):before {

left: 40%; }

.widget-area .widget h3.widget-title {

padding-bottom: .618em;

border-bottom-width: 2px;

border-bottom-style: dotted;

border-bottom-color: rgba(0, 0, 0, 0.2);

text-align: center; }

.widget-area .widget h3.widget-title:before, .widget-area .widget h3.widget-title:after {

display: inline-block;

content: " - ";

font-size: .618em;

margin: 0 1em;

opacity: 0.5;

font-weight: 700; }

.widget\_product\_categories ul li:before,

.widget\_categories ul li:before {

content: "\f07b" !important; }

.widget\_product\_categories ul li.current-cat:before,

.widget\_categories ul li.current-cat:before {

content: "\f07c" !important; }

.widget\_recent\_comments ul li:before {

content: "\f075" !important; }

.widget\_archive ul li:before {

content: "\f07c" !important; }

.widget\_recent\_entries ul li:before, .widget\_pages ul li:before {

content: "\f15c" !important; }

body:not(.page-template-template-homepage) .site-main {

padding: 2.618em; }

.page-template-template-homepage:not(.has-post-thumbnail) .site-main,

.page-template-template-homepage .site-main {

padding-top: 0; }

.page-template-template-homepage .site-main > .type-page:first-child {

margin-left: 0 !important;

width: auto !important; }

.woocommerce-checkout.scc-distraction-free-checkout .deli-primary-navigation {

display: none; }

.woocommerce-checkout.scc-distraction-free-checkout .site-header {

margin-bottom: 0; }

#order\_review {

background-color: #f9f9f9; }

.sph-hero,

.sprh-hero {

font-size: 1.1em;

font-weight: 700; }

.sph-hero.full,

.sprh-hero.full {

margin-left: -5.8823529412% !important;

margin-right: -5.8823529412% !important; }

.sph-hero .col-full,

.sprh-hero .col-full {

background: transparent !important;

box-shadow: none; }

.page-template-template-fullwidth-php .sph-hero.full,

.page-template-template-fullwidth-php .sprh-hero.full,

.page-template-template-homepage-php .sph-hero.full,

.page-template-template-homepage-php .sprh-hero.full,

.storefront-full-width-content .sph-hero.full,

.storefront-full-width-content .sprh-hero.full {

margin-left: -4.347826087% !important;

margin-right: -4.347826087% !important; }

.sd-header-sticky .site-header {

padding-top: 5.236em; }

.sd-scale-larger {

font-size: .9em; }

.sd-scale-largest {

font-size: 1em; }

.sd-scale-smaller {

font-size: .8em; }

.sd-scale-smallest {

font-size: .75em; }

.sd-typography-lora .deli-primary-navigation {

font-family: "Lora", serif; }

.sd-typography-roboto-slab .deli-primary-navigation {

font-family: "Roboto Slab", sans-serif; }

.sd-typography-courier .deli-primary-navigation {

font-family: "Courier", sans-serif; }

.swc-product-gallery-stacked.single-product div.product .images, .swc-product-gallery-stacked.storefront-full-width-content.single-product div.product .images {

margin: 0; }

.swc-product-gallery-stacked.single-product div.product .images > a, .swc-product-gallery-stacked.storefront-full-width-content.single-product div.product .images > a {

width: 100%; }

.swc-product-gallery-stacked.single-product div.product .images .thumbnails, .swc-product-gallery-stacked.storefront-full-width-content.single-product div.product .images .thumbnails {

width: 100%;

margin-bottom: 3.631em; }

.swc-product-gallery-stacked.single-product div.product .images .thumbnails a.zoom, .swc-product-gallery-stacked.storefront-full-width-content.single-product div.product .images .thumbnails a.zoom {

width: 5.8823529412%;

float: left;

margin-right: 5.8823529412%; }

.swc-product-gallery-stacked.single-product div.product .images .thumbnails a.zoom.first, .swc-product-gallery-stacked.storefront-full-width-content.single-product div.product .images .thumbnails a.zoom.first {

clear: none; }

.swc-product-gallery-stacked.storefront-full-width-content.single-product div.product .images .thumbnails a.zoom {

width: 13.0434782609%;

float: left;

margin-right: 4.347826087%; }

.woocommerce-pagination .page-numbers li .page-numbers.current {

background: #fff; }

@media screen and (min-width: 768px) {

body:before, body:after,

html:before,

html:after {

display: block; }

.site {

background-attachment: fixed;

padding: 2.618em; }

.home.blog .deli-content-wrapper,

.home.page:not(.page-template-template-homepage) .deli-content-wrapper {

padding-top: 3.631em; }

.col-full {

max-width: 85.999em; }

.site-content > .col-full {

margin-bottom: 2.618em; }

.site-content .col-full {

padding: 0; }

.content-area {

padding: 0; }

.deli-primary-navigation {

border-top: 3px double rgba(0, 0, 0, 0.2);

-webkit-font-smoothing: antialiased;

width: 108.695652174%;

float: left;

margin-left: 0;

margin-right: 0;

margin: -4.347826087%;

padding: 4.347826087%;

margin-top: 0;

margin-bottom: 2.618em;

padding-top: 0;

padding-bottom: 0;

box-shadow: 0 0.327em 0 rgba(0, 0, 0, 0.075);

position: relative;

clear: both;

background: #b64902; }

.deli-primary-navigation .smm-mega-menu {

margin-left: 0;

margin-right: 0;

padding: 4.347826087%;

box-shadow: inset 0 0.327em 0 0 rgba(0, 0, 0, 0.015); }

.site-header {

padding: 0;

padding-top: 2.618em; }

.site-header .site-branding,

.site-header .site-search,

.site-header .site-header-cart,

.site-header .site-logo-anchor,

.site-header .site-logo-link,

.site-header .custom-logo-link {

margin-bottom: 1.618em; }

.home .site-header {

margin-bottom: 2.618em; }

.main-navigation ul.menu > li > a, .main-navigation ul.nav-menu > li > a,

.site-header-cart .cart-contents {

padding-bottom: 1em !important; }

.header-widget-region .col-full {

padding-top: 0;

padding-bottom: 0; }

.site-footer {

text-shadow: 0 1px 1px rgba(0, 0, 0, 0.3); }

.site-footer .footer-widgets {

padding-top: 0; }

.main-navigation ul.menu,

.main-navigation ul.nav-menu {

float: left; }

.main-navigation ul.menu::after,

.main-navigation ul.nav-menu::after {

clear: both;

content: "";

display: table; }

.main-navigation ul.menu > li,

.main-navigation ul.nav-menu > li {

border-right: 1px solid rgba(255, 255, 255, 0.1);

float: left; }

.main-navigation ul.menu > li > a,

.main-navigation ul.nav-menu > li > a {

text-transform: uppercase; }

.main-navigation ul.menu > li.current-menu-item > a,

.main-navigation ul.nav-menu > li.current-menu-item > a {

background-color: rgba(0, 0, 0, 0.1); }

.main-navigation ul.menu > li:first-child,

.main-navigation ul.nav-menu > li:first-child {

margin-left: 0; }

.main-navigation ul.menu ul.sub-menu,

.main-navigation ul.nav-menu ul.sub-menu {

box-shadow: inset 0 0.327em 0 0 rgba(0, 0, 0, 0.015); }

.site-header-cart {

border-left: 1px solid rgba(255, 255, 255, 0.1); }

.site-header-cart .cart-contents {

font-weight: 400;

padding-left: 1em;

padding-right: 1em; }

.site-header-cart .widget\_shopping\_cart {

box-shadow: inset 0 0.46em 0 0 rgba(0, 0, 0, 0.015); }

.site-header-cart .widget\_shopping\_cart img {

padding: .327em;

background: #fff; }

.site-header-cart .widget\_shopping\_cart a.remove:before {

color: #fff; }

.woocommerce-active .site-header .site-header-cart,

.main-navigation {

padding-top: 0; }

.woocommerce-active .site-header .site-header-cart > li > a,

.woocommerce-active .site-header .site-header-cart > li .cart-contents,

.woocommerce-active .site-header .site-header-cart ul.menu > li > a,

.woocommerce-active .site-header .site-header-cart ul.menu > li .cart-contents,

.woocommerce-active .site-header .site-header-cart ul.nav-menu > li > a,

.woocommerce-active .site-header .site-header-cart ul.nav-menu > li .cart-contents,

.main-navigation > li > a,

.main-navigation > li .cart-contents,

.main-navigation ul.menu > li > a,

.main-navigation ul.menu > li .cart-contents,

.main-navigation ul.nav-menu > li > a,

.main-navigation ul.nav-menu > li .cart-contents {

padding-top: 1em; }

.site-header-cart .cart-contents:after {

top: 1em; }

.woocommerce-pagination a {

font-weight: 700; }

.page-template-template-homepage .content-area .sph-hero,

.page-template-template-homepage .content-area .sprh-hero {

margin-top: 0; }

.page-template-template-homepage .content-area .sprh-full-height .overlay {

padding-left: 4.236em !important;

padding-right: 4.236em !important; }

.widget-area .widget .sph-hero,

.widget-area .widget .sprh-hero {

text-shadow: none; }

.widget-area .widget .sph-hero:last-child,

.widget-area .widget .sprh-hero:last-child {

margin: 0; }

.sph-hero .col-full,

.sprh-hero .col-full {

background-color: transparent !important;

padding: 0; }

.site-main .sph-hero .overlay,

.site-main .sprh-hero .overlay {

padding: 4.236em; }

.post.type-post .entry-content,

.page:not(.woocommerce-page) .type-page .entry-content {

font-size: 1.2em; }

.site-main ul.products li.product {

width: 20.5882352941%;

float: left;

margin-right: 5.8823529412%;

overflow: visible; }

.site-main ul.products li.product h3 {

font-weight: 700; }

.site-main ul.products li.product .onsale {

position: absolute;

top: .618em;

right: .618em; }

.site-main ul.products li.product a {

display: block; }

.site-main ul.products li.product .button,

.site-main ul.products li.product .added\_to\_cart {

display: inline-block; }

.site-main .columns-2 ul.products li.product {

width: 47.0588235294%;

float: left;

margin-right: 5.8823529412%; }

.site-main .columns-3 ul.products li.product {

width: 29.4117647059%;

float: left;

margin-right: 5.8823529412%; }

.site-main .columns-4 ul.products li.product {

width: 20.5882352941%;

float: left;

margin-right: 5.8823529412%; }

.site-main .columns-5 ul.products li.product {

width: 15.2941176471%;

float: left;

margin-right: 5.8823529412%; }

.onsale {

background: #fff;

border: 0; }

.page-template-template-fullwidth-php .site-main ul.products li.product,

.page-template-template-homepage-php .site-main ul.products li.product,

.storefront-full-width-content .site-main ul.products li.product {

width: 21.7391304348%;

float: left;

margin-right: 4.347826087%; }

.page-template-template-fullwidth-php .site-main .columns-2 ul.products li.product,

.page-template-template-homepage-php .site-main .columns-2 ul.products li.product,

.storefront-full-width-content .site-main .columns-2 ul.products li.product {

width: 47.8260869565%;

float: left;

margin-right: 4.347826087%; }

.page-template-template-fullwidth-php .site-main .columns-3 ul.products li.product,

.page-template-template-homepage-php .site-main .columns-3 ul.products li.product,

.storefront-full-width-content .site-main .columns-3 ul.products li.product {

width: 30.4347826087%;

float: left;

margin-right: 4.347826087%; }

.page-template-template-fullwidth-php .site-main .columns-4 ul.products li.product,

.page-template-template-homepage-php .site-main .columns-4 ul.products li.product,

.storefront-full-width-content .site-main .columns-4 ul.products li.product {

width: 21.7391304348%;

float: left;

margin-right: 4.347826087%; }

.page-template-template-fullwidth-php .site-main .columns-5 ul.products li.product,

.page-template-template-homepage-php .site-main .columns-5 ul.products li.product,

.storefront-full-width-content .site-main .columns-5 ul.products li.product {

width: 16.5217391304%;

float: left;

margin-right: 4.347826087%; }

.single-product div.product {

overflow: visible; }

.single-product div.product > .onsale {

position: absolute;

top: 1em;

left: 1em; }

.single-product div.product .images {

width: 52.9411764706%;

float: left;

margin-right: 5.8823529412%; }

.single-product div.product .images > a:not(.woocommerce-product-gallery\_\_trigger) {

display: block;

width: 77.7777777778%;

float: left;

margin-right: 11.1111111111%; }

.single-product div.product .images .thumbnails {

width: 11.1111111111%;

float: right;

margin-right: 0; }

.single-product div.product .images .thumbnails a.zoom {

width: 100%;

float: none; }

.single-product div.product .summary {

width: 41.1764705882%;

float: right;

margin-right: 0; }

#reviews .commentlist li .description {

padding: 1.618em; }

.storefront-full-width-content.single-product div.product .images {

width: 47.8260869565%;

float: left;

margin-right: 4.347826087%; }

.storefront-full-width-content.single-product div.product .images > a {

width: 81.8181818182%;

float: left;

margin-right: 9.0909090909%; }

.storefront-full-width-content.single-product div.product .images .thumbnails {

width: 9.0909090909%;

float: right;

margin-right: 0; }

.storefront-full-width-content.single-product div.product .images .thumbnails a.zoom {

width: 100%;

float: none; }

.storefront-full-width-content.single-product div.product .summary {

width: 30.4347826087%;

float: right;

margin-right: 0;

margin-right: 8.6956521739%;

padding-top: 1.618em; }

.storefront-full-width-content.single-product .related ul.products li.product {

width: 21.7391304348%;

float: left;

margin-right: 4.347826087%; }

.right-sidebar .content-area {

width: 82.6086956522%;

float: left;

margin-right: 4%;

margin: -4.347826087%;

padding: 4.347826087%;

margin-top: 0; }

.right-sidebar .widget-area {

width: 21.7391304348%;

float: right;

margin-right: 0;

position: relative; }

.right-sidebar .widget-area .widget {

box-shadow: inset 0.53em 0 0 rgba(0, 0, 0, 0.015); }

.left-sidebar .content-area {

width: 82.6086956522%;

float: right;

margin-right: 0;

margin: -4.347826087%;

padding: 4.347826087%;

margin-top: 0; }

.left-sidebar .widget-area {

width: 21.7391304348%;

float: left;

margin-right: 4.347826087%;

position: relative; }

.left-sidebar .widget-area .widget {

box-shadow: inset -0.53em 0 0 rgba(0, 0, 0, 0.015); }

.widget-area .widget {

padding: 2.618em 2.244em 2.244em 2.244em; }

.footer-widgets {

border-bottom: 2px dotted rgba(0, 0, 0, 0.1); }

/\*\*

\* Full width & homepage template

\*/

.page-template-template-fullwidth-php .content-area,

.page-template-template-homepage-php .content-area,

.storefront-full-width-content .content-area {

width: 108.695652174%;

float: left;

margin-left: 0;

margin-right: 0;

margin: -4.347826087%;

padding: 4.347826087%;

margin-top: 0; } }

@media screen and (max-width: 65.141em) {

.col-full {

margin-left: 0;

margin-right: 0; } }

/\*\*

\* Storefront 2.3

\*/

.storefront-2-3 .widget\_product\_categories ul li:before,

.storefront-2-3 .widget\_categories ul li:before {

font-family: 'Font Awesome 5 Free';

font-weight: 900;

content: "\f07b" !important; }

.storefront-2-3 .widget\_product\_categories ul li.current-cat:before,

.storefront-2-3 .widget\_categories ul li.current-cat:before {

font-family: 'Font Awesome 5 Free';

font-weight: 900;

content: "\f07b" !important; }

.storefront-2-3 .widget\_recent\_comments ul li:before {

font-family: 'Font Awesome 5 Free';

font-weight: 900;

content: "\f075" !important; }

.storefront-2-3 .widget\_archive ul li:before {

font-family: 'Font Awesome 5 Free';

font-weight: 900;

content: "\f07b" !important; }

.storefront-2-3 .widget\_recent\_entries ul li:before, .storefront-2-3 .widget\_pages ul li:before {

font-family: 'Font Awesome 5 Free';

font-weight: 900;

content: "\f15c" !important; }

<?php

/\*\*

\* Confirms that the activation key that is sent in an email after a user signs

\* up for a new site matches the key for that user and then displays confirmation.

\*

\* @package WordPress

\*/

define( 'WP\_INSTALLING', true );

/\*\* Sets up the WordPress Environment. \*/

require \_\_DIR\_\_ . '/wp-load.php';

require \_\_DIR\_\_ . '/wp-blog-header.php';

if ( ! is\_multisite() ) {

wp\_redirect( wp\_registration\_url() );

die();

}

$valid\_error\_codes = array( 'already\_active', 'blog\_taken' );

list( $activate\_path ) = explode( '?', wp\_unslash( $\_SERVER['REQUEST\_URI'] ) );

$activate\_cookie = 'wp-activate-' . COOKIEHASH;

$key = '';

$result = null;

if ( isset( $\_GET['key'] ) && isset( $\_POST['key'] ) && $\_GET['key'] !== $\_POST['key'] ) {

wp\_die( \_\_( 'A key value mismatch has been detected. Please follow the link provided in your activation email.' ), \_\_( 'An error occurred during the activation' ), 400 );

} elseif ( ! empty( $\_GET['key'] ) ) {

$key = $\_GET['key'];

} elseif ( ! empty( $\_POST['key'] ) ) {

$key = $\_POST['key'];

}

if ( $key ) {

$redirect\_url = remove\_query\_arg( 'key' );

if ( remove\_query\_arg( false ) !== $redirect\_url ) {

setcookie( $activate\_cookie, $key, 0, $activate\_path, COOKIE\_DOMAIN, is\_ssl(), true );

wp\_safe\_redirect( $redirect\_url );

exit;

} else {

$result = wpmu\_activate\_signup( $key );

}

}

if ( null === $result && isset( $\_COOKIE[ $activate\_cookie ] ) ) {

$key = $\_COOKIE[ $activate\_cookie ];

$result = wpmu\_activate\_signup( $key );

setcookie( $activate\_cookie, ' ', time() - YEAR\_IN\_SECONDS, $activate\_path, COOKIE\_DOMAIN, is\_ssl(), true );

}

if ( null === $result || ( is\_wp\_error( $result ) && 'invalid\_key' === $result->get\_error\_code() ) ) {

status\_header( 404 );

} elseif ( is\_wp\_error( $result ) ) {

$error\_code = $result->get\_error\_code();

if ( ! in\_array( $error\_code, $valid\_error\_codes, true ) ) {

status\_header( 400 );

}

}

nocache\_headers();

if ( is\_object( $wp\_object\_cache ) ) {

$wp\_object\_cache->cache\_enabled = false;

}

// Fix for page title.

$wp\_query->is\_404 = false;

/\*\*

\* Fires before the Site Activation page is loaded.

\*

\* @since 3.0.0

\*/

do\_action( 'activate\_header' );

/\*\*

\* Adds an action hook specific to this page.

\*

\* Fires on {@see 'wp\_head'}.

\*

\* @since MU (3.0.0)

\*/

function do\_activate\_header() {

/\*\*

\* Fires before the Site Activation page is loaded.

\*

\* Fires on the {@see 'wp\_head'} action.

\*

\* @since 3.0.0

\*/

do\_action( 'activate\_wp\_head' );

}

add\_action( 'wp\_head', 'do\_activate\_header' );

/\*\*

\* Loads styles specific to this page.

\*

\* @since MU (3.0.0)

\*/

function wpmu\_activate\_stylesheet() {

?>

<style type="text/css">

form { margin-top: 2em; }

#submit, #key { width: 90%; font-size: 24px; }

#language { margin-top: .5em; }

.error { background: #f66; }

span.h3 { padding: 0 8px; font-size: 1.3em; font-weight: 600; }

</style>

<?php

}

add\_action( 'wp\_head', 'wpmu\_activate\_stylesheet' );

add\_action( 'wp\_head', 'wp\_strict\_cross\_origin\_referrer' );

add\_filter( 'wp\_robots', 'wp\_robots\_sensitive\_page' );

get\_header( 'wp-activate' );

$blog\_details = get\_blog\_details();

?>

<div id="signup-content" class="widecolumn">

<div class="wp-activate-container">

<?php if ( ! $key ) { ?>

<h2><?php \_e( 'Activation Key Required' ); ?></h2>

<form name="activateform" id="activateform" method="post" action="<?php echo network\_site\_url( $blog\_details->path . 'wp-activate.php' ); ?>">

<p>

<label for="key"><?php \_e( 'Activation Key:' ); ?></label>

<br /><input type="text" name="key" id="key" value="" size="50" />

</p>

<p class="submit">

<input id="submit" type="submit" name="Submit" class="submit" value="<?php esc\_attr\_e( 'Activate' ); ?>" />

</p>

</form>

<?php

} else {

if ( is\_wp\_error( $result ) && in\_array( $result->get\_error\_code(), $valid\_error\_codes, true ) ) {

$signup = $result->get\_error\_data();

?>

<h2><?php \_e( 'Your account is now active!' ); ?></h2>

<?php

echo '<p class="lead-in">';

if ( '' === $signup->domain . $signup->path ) {

printf(

/\* translators: 1: Login URL, 2: Username, 3: User email address, 4: Lost password URL. \*/

\_\_( 'Your account has been activated. You may now <a href="%1$s">log in</a> to the site using your chosen username of &#8220;%2$s&#8221;. Please check your email inbox at %3$s for your password and login instructions. If you do not receive an email, please check your junk or spam folder. If you still do not receive an email within an hour, you can <a href="%4$s">reset your password</a>.' ),

network\_site\_url( $blog\_details->path . 'wp-login.php', 'login' ),

$signup->user\_login,

$signup->user\_email,

wp\_lostpassword\_url()

);

} else {

printf(

/\* translators: 1: Site URL, 2: Username, 3: User email address, 4: Lost password URL. \*/

\_\_( 'Your site at %1$s is active. You may now log in to your site using your chosen username of &#8220;%2$s&#8221;. Please check your email inbox at %3$s for your password and login instructions. If you do not receive an email, please check your junk or spam folder. If you still do not receive an email within an hour, you can <a href="%4$s">reset your password</a>.' ),

sprintf( '<a href="http://%1$s%2$s">%1$s%2$s</a>', $signup->domain, $blog\_details->path ),

$signup->user\_login,

$signup->user\_email,

wp\_lostpassword\_url()

);

}

echo '</p>';

} elseif ( null === $result || is\_wp\_error( $result ) ) {

?>

<h2><?php \_e( 'An error occurred during the activation' ); ?></h2>

<?php if ( is\_wp\_error( $result ) ) : ?>

<p><?php echo $result->get\_error\_message(); ?></p>

<?php endif; ?>

<?php

} else {

$url = isset( $result['blog\_id'] ) ? get\_home\_url( (int) $result['blog\_id'] ) : '';

$user = get\_userdata( (int) $result['user\_id'] );

?>

<h2><?php \_e( 'Your account is now active!' ); ?></h2>

<div id="signup-welcome">

<p><span class="h3"><?php \_e( 'Username:' ); ?></span> <?php echo $user->user\_login; ?></p>

<p><span class="h3"><?php \_e( 'Password:' ); ?></span> <?php echo $result['password']; ?></p>

</div>

<?php

if ( $url && network\_home\_url( '', 'http' ) !== $url ) :

switch\_to\_blog( (int) $result['blog\_id'] );

$login\_url = wp\_login\_url();

restore\_current\_blog();

?>

<p class="view">

<?php

/\* translators: 1: Site URL, 2: Login URL. \*/

printf( \_\_( 'Your account is now activated. <a href="%1$s">View your site</a> or <a href="%2$s">Log in</a>' ), $url, esc\_url( $login\_url ) );

?>

</p>

<?php else : ?>

<p class="view">

<?php

printf(

/\* translators: 1: Login URL, 2: Network home URL. \*/

\_\_( 'Your account is now activated. <a href="%1$s">Log in</a> or go back to the <a href="%2$s">homepage</a>.' ),

network\_site\_url( $blog\_details->path . 'wp-login.php', 'login' ),

network\_home\_url( $blog\_details->path )

);

?>

</p>

<?php

endif;

}

}

?>

</div>

</div>

<script type="text/javascript">

var key\_input = document.getElementById('key');

key\_input && key\_input.focus();

</script>

<?php

get\_footer( 'wp-activate' );

Login

<?php

/\*\*

\* WordPress User Page

\*

\* Handles authentication, registering, resetting passwords, forgot password,

\* and other user handling.

\*

\* @package WordPress

\*/

/\*\* Make sure that the WordPress bootstrap has run before continuing. \*/

require \_\_DIR\_\_ . '/wp-load.php';

// Redirect to HTTPS login if forced to use SSL.

if ( force\_ssl\_admin() && ! is\_ssl() ) {

if ( 0 === strpos( $\_SERVER['REQUEST\_URI'], 'http' ) ) {

wp\_safe\_redirect( set\_url\_scheme( $\_SERVER['REQUEST\_URI'], 'https' ) );

exit;

} else {

wp\_safe\_redirect( 'https://' . $\_SERVER['HTTP\_HOST'] . $\_SERVER['REQUEST\_URI'] );

exit;

}

}

/\*\*

\* Output the login page header.

\*

\* @since 2.1.0

\*

\* @global string $error Login error message set by deprecated pluggable wp\_login() function

\* or plugins replacing it.

\* @global bool|string $interim\_login Whether interim login modal is being displayed. String 'success'

\* upon successful login.

\* @global string $action The action that brought the visitor to the login page.

\*

\* @param string $title Optional. WordPress login Page title to display in the `<title>` element.

\* Default 'Log In'.

\* @param string $message Optional. Message to display in header. Default empty.

\* @param WP\_Error $wp\_error Optional. The error to pass. Default is a WP\_Error instance.

\*/

function login\_header( $title = 'Log In', $message = '', $wp\_error = null ) {

global $error, $interim\_login, $action;

// Don't index any of these forms.

add\_filter( 'wp\_robots', 'wp\_robots\_sensitive\_page' );

add\_action( 'login\_head', 'wp\_strict\_cross\_origin\_referrer' );

add\_action( 'login\_head', 'wp\_login\_viewport\_meta' );

if ( ! is\_wp\_error( $wp\_error ) ) {

$wp\_error = new WP\_Error();

}

// Shake it!

$shake\_error\_codes = array( 'empty\_password', 'empty\_email', 'invalid\_email', 'invalidcombo', 'empty\_username', 'invalid\_username', 'incorrect\_password', 'retrieve\_password\_email\_failure' );

/\*\*

\* Filters the error codes array for shaking the login form.

\*

\* @since 3.0.0

\*

\* @param array $shake\_error\_codes Error codes that shake the login form.

\*/

$shake\_error\_codes = apply\_filters( 'shake\_error\_codes', $shake\_error\_codes );

if ( $shake\_error\_codes && $wp\_error->has\_errors() && in\_array( $wp\_error->get\_error\_code(), $shake\_error\_codes, true ) ) {

add\_action( 'login\_footer', 'wp\_shake\_js', 12 );

}

$login\_title = get\_bloginfo( 'name', 'display' );

/\* translators: Login screen title. 1: Login screen name, 2: Network or site name. \*/

$login\_title = sprintf( \_\_( '%1$s &lsaquo; %2$s &#8212; WordPress' ), $title, $login\_title );

if ( wp\_is\_recovery\_mode() ) {

/\* translators: %s: Login screen title. \*/

$login\_title = sprintf( \_\_( 'Recovery Mode &#8212; %s' ), $login\_title );

}

/\*\*

\* Filters the title tag content for login page.

\*

\* @since 4.9.0

\*

\* @param string $login\_title The page title, with extra context added.

\* @param string $title The original page title.

\*/

$login\_title = apply\_filters( 'login\_title', $login\_title, $title );

?><!DOCTYPE html>

<html <?php language\_attributes(); ?>>

<head>

<meta http-equiv="Content-Type" content="<?php bloginfo( 'html\_type' ); ?>; charset=<?php bloginfo( 'charset' ); ?>" />

<title><?php echo $login\_title; ?></title>

<?php

wp\_enqueue\_style( 'login' );

/\*

\* Remove all stored post data on logging out.

\* This could be added by add\_action('login\_head'...) like wp\_shake\_js(),

\* but maybe better if it's not removable by plugins.

\*/

if ( 'loggedout' === $wp\_error->get\_error\_code() ) {

?>

<script>if("sessionStorage" in window){try{for(var key in sessionStorage){if(key.indexOf("wp-autosave-")!=-1){sessionStorage.removeItem(key)}}}catch(e){}};</script>

<?php

}

/\*\*

\* Enqueue scripts and styles for the login page.

\*

\* @since 3.1.0

\*/

do\_action( 'login\_enqueue\_scripts' );

/\*\*

\* Fires in the login page header after scripts are enqueued.

\*

\* @since 2.1.0

\*/

do\_action( 'login\_head' );

$login\_header\_url = \_\_( 'https://wordpress.org/' );

/\*\*

\* Filters link URL of the header logo above login form.

\*

\* @since 2.1.0

\*

\* @param string $login\_header\_url Login header logo URL.

\*/

$login\_header\_url = apply\_filters( 'login\_headerurl', $login\_header\_url );

$login\_header\_title = '';

/\*\*

\* Filters the title attribute of the header logo above login form.

\*

\* @since 2.1.0

\* @deprecated 5.2.0 Use {@see 'login\_headertext'} instead.

\*

\* @param string $login\_header\_title Login header logo title attribute.

\*/

$login\_header\_title = apply\_filters\_deprecated(

'login\_headertitle',

array( $login\_header\_title ),

'5.2.0',

'login\_headertext',

\_\_( 'Usage of the title attribute on the login logo is not recommended for accessibility reasons. Use the link text instead.' )

);

$login\_header\_text = empty( $login\_header\_title ) ? \_\_( 'Powered by WordPress' ) : $login\_header\_title;

/\*\*

\* Filters the link text of the header logo above the login form.

\*

\* @since 5.2.0

\*

\* @param string $login\_header\_text The login header logo link text.

\*/

$login\_header\_text = apply\_filters( 'login\_headertext', $login\_header\_text );

$classes = array( 'login-action-' . $action, 'wp-core-ui' );

if ( is\_rtl() ) {

$classes[] = 'rtl';

}

if ( $interim\_login ) {

$classes[] = 'interim-login';

?>

<style type="text/css">html{background-color: transparent;}</style>

<?php

if ( 'success' === $interim\_login ) {

$classes[] = 'interim-login-success';

}

}

$classes[] = ' locale-' . sanitize\_html\_class( strtolower( str\_replace( '\_', '-', get\_locale() ) ) );

/\*\*

\* Filters the login page body classes.

\*

\* @since 3.5.0

\*

\* @param array $classes An array of body classes.

\* @param string $action The action that brought the visitor to the login page.

\*/

$classes = apply\_filters( 'login\_body\_class', $classes, $action );

?>

</head>

<body class="login no-js <?php echo esc\_attr( implode( ' ', $classes ) ); ?>">

<script type="text/javascript">

document.body.className = document.body.className.replace('no-js','js');

</script>

<?php

/\*\*

\* Fires in the login page header after the body tag is opened.

\*

\* @since 4.6.0

\*/

do\_action( 'login\_header' );

?>

<div id="login">

<h1><a href="<?php echo esc\_url( $login\_header\_url ); ?>"><?php echo $login\_header\_text; ?></a></h1>

<?php

/\*\*

\* Filters the message to display above the login form.

\*

\* @since 2.1.0

\*

\* @param string $message Login message text.

\*/

$message = apply\_filters( 'login\_message', $message );

if ( ! empty( $message ) ) {

echo $message . "\n";

}

// In case a plugin uses $error rather than the $wp\_errors object.

if ( ! empty( $error ) ) {

$wp\_error->add( 'error', $error );

unset( $error );

}

if ( $wp\_error->has\_errors() ) {

$errors = '';

$messages = '';

foreach ( $wp\_error->get\_error\_codes() as $code ) {

$severity = $wp\_error->get\_error\_data( $code );

foreach ( $wp\_error->get\_error\_messages( $code ) as $error\_message ) {

if ( 'message' === $severity ) {

$messages .= ' ' . $error\_message . "<br />\n";

} else {

$errors .= ' ' . $error\_message . "<br />\n";

}

}

}

if ( ! empty( $errors ) ) {

/\*\*

\* Filters the error messages displayed above the login form.

\*

\* @since 2.1.0

\*

\* @param string $errors Login error message.

\*/

echo '<div id="login\_error">' . apply\_filters( 'login\_errors', $errors ) . "</div>\n";

}

if ( ! empty( $messages ) ) {

/\*\*

\* Filters instructional messages displayed above the login form.

\*

\* @since 2.5.0

\*

\* @param string $messages Login messages.

\*/

echo '<p class="message">' . apply\_filters( 'login\_messages', $messages ) . "</p>\n";

}

}

} // End of login\_header().

/\*\*

\* Outputs the footer for the login page.

\*

\* @since 3.1.0

\*

\* @global bool|string $interim\_login Whether interim login modal is being displayed. String 'success'

\* upon successful login.

\*

\* @param string $input\_id Which input to auto-focus.

\*/

function login\_footer( $input\_id = '' ) {

global $interim\_login;

// Don't allow interim logins to navigate away from the page.

if ( ! $interim\_login ) {

?>

<p id="backtoblog">

<?php

$html\_link = sprintf(

'<a href="%s">%s</a>',

esc\_url( home\_url( '/' ) ),

sprintf(

/\* translators: %s: Site title. \*/

\_x( '&larr; Go to %s', 'site' ),

get\_bloginfo( 'title', 'display' )

)

);

/\*\*

\* Filter the "Go to site" link displayed in the login page footer.

\*

\* @since 5.7.0

\*

\* @param string $link HTML link to the home URL of the current site.

\*/

echo apply\_filters( 'login\_site\_html\_link', $html\_link );

?>

</p>

<?php

the\_privacy\_policy\_link( '<div class="privacy-policy-page-link">', '</div>' );

}

?>

</div><?php // End of <div id="login">. ?>

<?php

if ( ! empty( $input\_id ) ) {

?>

<script type="text/javascript">

try{document.getElementById('<?php echo $input\_id; ?>').focus();}catch(e){}

if(typeof wpOnload==='function')wpOnload();

</script>

<?php

}

/\*\*

\* Fires in the login page footer.

\*

\* @since 3.1.0

\*/

do\_action( 'login\_footer' );

?>

<div class="clear"></div>

</body>

</html>

<?php

}

/\*\*

\* Outputs the JavaScript to handle the form shaking on the login page.

\*

\* @since 3.0.0

\*/

function wp\_shake\_js() {

?>

<script type="text/javascript">

document.querySelector('form').classList.add('shake');

</script>

<?php

}

/\*\*

\* Outputs the viewport meta tag for the login page.

\*

\* @since 3.7.0

\*/

function wp\_login\_viewport\_meta() {

?>

<meta name="viewport" content="width=device-width" />

<?php

}

//

// Main.

//

$action = isset( $\_REQUEST['action'] ) ? $\_REQUEST['action'] : 'login';

$errors = new WP\_Error();

if ( isset( $\_GET['key'] ) ) {

$action = 'resetpass';

}

if ( isset( $\_GET['checkemail'] ) ) {

$action = 'checkemail';

}

$default\_actions = array(

'confirm\_admin\_email',

'postpass',

'logout',

'lostpassword',

'retrievepassword',

'resetpass',

'rp',

'register',

'checkemail',

'confirmaction',

'login',

WP\_Recovery\_Mode\_Link\_Service::LOGIN\_ACTION\_ENTERED,

);

// Validate action so as to default to the login screen.

if ( ! in\_array( $action, $default\_actions, true ) && false === has\_filter( 'login\_form\_' . $action ) ) {

$action = 'login';

}

nocache\_headers();

header( 'Content-Type: ' . get\_bloginfo( 'html\_type' ) . '; charset=' . get\_bloginfo( 'charset' ) );

if ( defined( 'RELOCATE' ) && RELOCATE ) { // Move flag is set.

if ( isset( $\_SERVER['PATH\_INFO'] ) && ( $\_SERVER['PATH\_INFO'] !== $\_SERVER['PHP\_SELF'] ) ) {

$\_SERVER['PHP\_SELF'] = str\_replace( $\_SERVER['PATH\_INFO'], '', $\_SERVER['PHP\_SELF'] );

}

$url = dirname( set\_url\_scheme( 'http://' . $\_SERVER['HTTP\_HOST'] . $\_SERVER['PHP\_SELF'] ) );

if ( get\_option( 'siteurl' ) !== $url ) {

update\_option( 'siteurl', $url );

}

}

// Set a cookie now to see if they are supported by the browser.

$secure = ( 'https' === parse\_url( wp\_login\_url(), PHP\_URL\_SCHEME ) );

setcookie( TEST\_COOKIE, 'WP Cookie check', 0, COOKIEPATH, COOKIE\_DOMAIN, $secure );

if ( SITECOOKIEPATH !== COOKIEPATH ) {

setcookie( TEST\_COOKIE, 'WP Cookie check', 0, SITECOOKIEPATH, COOKIE\_DOMAIN, $secure );

}

/\*\*

\* Fires when the login form is initialized.

\*

\* @since 3.2.0

\*/

do\_action( 'login\_init' );

/\*\*

\* Fires before a specified login form action.

\*

\* The dynamic portion of the hook name, `$action`, refers to the action

\* that brought the visitor to the login form.

\*

\* Possible hook names include:

\*

\* - 'login\_form\_checkemail'

\* - 'login\_form\_confirm\_admin\_email'

\* - 'login\_form\_confirmaction'

\* - 'login\_form\_entered\_recovery\_mode'

\* - 'login\_form\_login'

\* - 'login\_form\_logout'

\* - 'login\_form\_lostpassword'

\* - 'login\_form\_postpass'

\* - 'login\_form\_register'

\* - 'login\_form\_resetpass'

\* - 'login\_form\_retrievepassword'

\* - 'login\_form\_rp'

\*

\* @since 2.8.0

\*/

do\_action( "login\_form\_{$action}" );

$http\_post = ( 'POST' === $\_SERVER['REQUEST\_METHOD'] );

$interim\_login = isset( $\_REQUEST['interim-login'] );

/\*\*

\* Filters the separator used between login form navigation links.

\*

\* @since 4.9.0

\*

\* @param string $login\_link\_separator The separator used between login form navigation links.

\*/

$login\_link\_separator = apply\_filters( 'login\_link\_separator', ' | ' );

switch ( $action ) {

case 'confirm\_admin\_email':

/\*

\* Note that `is\_user\_logged\_in()` will return false immediately after logging in

\* as the current user is not set, see wp-includes/pluggable.php.

\* However this action runs on a redirect after logging in.

\*/

if ( ! is\_user\_logged\_in() ) {

wp\_safe\_redirect( wp\_login\_url() );

exit;

}

if ( ! empty( $\_REQUEST['redirect\_to'] ) ) {

$redirect\_to = $\_REQUEST['redirect\_to'];

} else {

$redirect\_to = admin\_url();

}

if ( current\_user\_can( 'manage\_options' ) ) {

$admin\_email = get\_option( 'admin\_email' );

} else {

wp\_safe\_redirect( $redirect\_to );

exit;

}

/\*\*

\* Filters the interval for dismissing the admin email confirmation screen.

\*

\* If `0` (zero) is returned, the "Remind me later" link will not be displayed.

\*

\* @since 5.3.1

\*

\* @param int $interval Interval time (in seconds). Default is 3 days.

\*/

$remind\_interval = (int) apply\_filters( 'admin\_email\_remind\_interval', 3 \* DAY\_IN\_SECONDS );

if ( ! empty( $\_GET['remind\_me\_later'] ) ) {

if ( ! wp\_verify\_nonce( $\_GET['remind\_me\_later'], 'remind\_me\_later\_nonce' ) ) {

wp\_safe\_redirect( wp\_login\_url() );

exit;

}

if ( $remind\_interval > 0 ) {

update\_option( 'admin\_email\_lifespan', time() + $remind\_interval );

}

$redirect\_to = add\_query\_arg( 'admin\_email\_remind\_later', 1, $redirect\_to );

wp\_safe\_redirect( $redirect\_to );

exit;

}

if ( ! empty( $\_POST['correct-admin-email'] ) ) {

if ( ! check\_admin\_referer( 'confirm\_admin\_email', 'confirm\_admin\_email\_nonce' ) ) {

wp\_safe\_redirect( wp\_login\_url() );

exit;

}

/\*\*

\* Filters the interval for redirecting the user to the admin email confirmation screen.

\*

\* If `0` (zero) is returned, the user will not be redirected.

\*

\* @since 5.3.0

\*

\* @param int $interval Interval time (in seconds). Default is 6 months.

\*/

$admin\_email\_check\_interval = (int) apply\_filters( 'admin\_email\_check\_interval', 6 \* MONTH\_IN\_SECONDS );

if ( $admin\_email\_check\_interval > 0 ) {

update\_option( 'admin\_email\_lifespan', time() + $admin\_email\_check\_interval );

}

wp\_safe\_redirect( $redirect\_to );

exit;

}

login\_header( \_\_( 'Confirm your administration email' ), '', $errors );

/\*\*

\* Fires before the admin email confirm form.

\*

\* @since 5.3.0

\*

\* @param WP\_Error $errors A `WP\_Error` object containing any errors generated by using invalid

\* credentials. Note that the error object may not contain any errors.

\*/

do\_action( 'admin\_email\_confirm', $errors );

?>

<form class="admin-email-confirm-form" name="admin-email-confirm-form" action="<?php echo esc\_url( site\_url( 'wp-login.php?action=confirm\_admin\_email', 'login\_post' ) ); ?>" method="post">

<?php

/\*\*

\* Fires inside the admin-email-confirm-form form tags, before the hidden fields.

\*

\* @since 5.3.0

\*/

do\_action( 'admin\_email\_confirm\_form' );

wp\_nonce\_field( 'confirm\_admin\_email', 'confirm\_admin\_email\_nonce' );

?>

<input type="hidden" name="redirect\_to" value="<?php echo esc\_attr( $redirect\_to ); ?>" />

<h1 class="admin-email\_\_heading">

<?php \_e( 'Administration email verification' ); ?>

</h1>

<p class="admin-email\_\_details">

<?php \_e( 'Please verify that the <strong>administration email</strong> for this website is still correct.' ); ?>

<?php

/\* translators: URL to the WordPress help section about admin email. \*/

$admin\_email\_help\_url = \_\_( 'https://wordpress.org/support/article/settings-general-screen/#email-address' );

/\* translators: Accessibility text. \*/

$accessibility\_text = sprintf( '<span class="screen-reader-text"> %s</span>', \_\_( '(opens in a new tab)' ) );

printf(

'<a href="%s" rel="noopener" target="\_blank">%s%s</a>',

esc\_url( $admin\_email\_help\_url ),

\_\_( 'Why is this important?' ),

$accessibility\_text

);

?>

</p>

<p class="admin-email\_\_details">

<?php

printf(

/\* translators: %s: Admin email address. \*/

\_\_( 'Current administration email: %s' ),

'<strong>' . esc\_html( $admin\_email ) . '</strong>'

);

?>

</p>

<p class="admin-email\_\_details">

<?php \_e( 'This email may be different from your personal email address.' ); ?>

</p>

<div class="admin-email\_\_actions">

<div class="admin-email\_\_actions-primary">

<?php

$change\_link = admin\_url( 'options-general.php' );

$change\_link = add\_query\_arg( 'highlight', 'confirm\_admin\_email', $change\_link );

?>

<a class="button button-large" href="<?php echo esc\_url( $change\_link ); ?>"><?php \_e( 'Update' ); ?></a>

<input type="submit" name="correct-admin-email" id="correct-admin-email" class="button button-primary button-large" value="<?php esc\_attr\_e( 'The email is correct' ); ?>" />

</div>

<?php if ( $remind\_interval > 0 ) : ?>

<div class="admin-email\_\_actions-secondary">

<?php

$remind\_me\_link = wp\_login\_url( $redirect\_to );

$remind\_me\_link = add\_query\_arg(

array(

'action' => 'confirm\_admin\_email',

'remind\_me\_later' => wp\_create\_nonce( 'remind\_me\_later\_nonce' ),

),

$remind\_me\_link

);

?>

<a href="<?php echo esc\_url( $remind\_me\_link ); ?>"><?php \_e( 'Remind me later' ); ?></a>

</div>

<?php endif; ?>

</div>

</form>

<?php

login\_footer();

break;

case 'postpass':

if ( ! array\_key\_exists( 'post\_password', $\_POST ) ) {

wp\_safe\_redirect( wp\_get\_referer() );

exit;

}

require\_once ABSPATH . WPINC . '/class-phpass.php';

$hasher = new PasswordHash( 8, true );

/\*\*

\* Filters the life span of the post password cookie.

\*

\* By default, the cookie expires 10 days from creation. To turn this

\* into a session cookie, return 0.

\*

\* @since 3.7.0

\*

\* @param int $expires The expiry time, as passed to setcookie().

\*/

$expire = apply\_filters( 'post\_password\_expires', time() + 10 \* DAY\_IN\_SECONDS );

$referer = wp\_get\_referer();

if ( $referer ) {

$secure = ( 'https' === parse\_url( $referer, PHP\_URL\_SCHEME ) );

} else {

$secure = false;

}

setcookie( 'wp-postpass\_' . COOKIEHASH, $hasher->HashPassword( wp\_unslash( $\_POST['post\_password'] ) ), $expire, COOKIEPATH, COOKIE\_DOMAIN, $secure );

wp\_safe\_redirect( wp\_get\_referer() );

exit;

case 'logout':

check\_admin\_referer( 'log-out' );

$user = wp\_get\_current\_user();

wp\_logout();

if ( ! empty( $\_REQUEST['redirect\_to'] ) ) {

$redirect\_to = $\_REQUEST['redirect\_to'];

$requested\_redirect\_to = $redirect\_to;

} else {

$redirect\_to = add\_query\_arg(

array(

'loggedout' => 'true',

'wp\_lang' => get\_user\_locale( $user ),

),

wp\_login\_url()

);

$requested\_redirect\_to = '';

}

/\*\*

\* Filters the log out redirect URL.

\*

\* @since 4.2.0

\*

\* @param string $redirect\_to The redirect destination URL.

\* @param string $requested\_redirect\_to The requested redirect destination URL passed as a parameter.

\* @param WP\_User $user The WP\_User object for the user that's logging out.

\*/

$redirect\_to = apply\_filters( 'logout\_redirect', $redirect\_to, $requested\_redirect\_to, $user );

wp\_safe\_redirect( $redirect\_to );

exit;

case 'lostpassword':

case 'retrievepassword':

if ( $http\_post ) {

$errors = retrieve\_password();

if ( ! is\_wp\_error( $errors ) ) {

$redirect\_to = ! empty( $\_REQUEST['redirect\_to'] ) ? $\_REQUEST['redirect\_to'] : 'wp-login.php?checkemail=confirm';

wp\_safe\_redirect( $redirect\_to );

exit;

}

}

if ( isset( $\_GET['error'] ) ) {

if ( 'invalidkey' === $\_GET['error'] ) {

$errors->add( 'invalidkey', \_\_( '<strong>Error</strong>: Your password reset link appears to be invalid. Please request a new link below.' ) );

} elseif ( 'expiredkey' === $\_GET['error'] ) {

$errors->add( 'expiredkey', \_\_( '<strong>Error</strong>: Your password reset link has expired. Please request a new link below.' ) );

}

}

$lostpassword\_redirect = ! empty( $\_REQUEST['redirect\_to'] ) ? $\_REQUEST['redirect\_to'] : '';

/\*\*

\* Filters the URL redirected to after submitting the lostpassword/retrievepassword form.

\*

\* @since 3.0.0

\*

\* @param string $lostpassword\_redirect The redirect destination URL.

\*/

$redirect\_to = apply\_filters( 'lostpassword\_redirect', $lostpassword\_redirect );

/\*\*

\* Fires before the lost password form.

\*

\* @since 1.5.1

\* @since 5.1.0 Added the `$errors` parameter.

\*

\* @param WP\_Error $errors A `WP\_Error` object containing any errors generated by using invalid

\* credentials. Note that the error object may not contain any errors.

\*/

do\_action( 'lost\_password', $errors );

login\_header( \_\_( 'Lost Password' ), '<p class="message">' . \_\_( 'Please enter your username or email address. You will receive an email message with instructions on how to reset your password.' ) . '</p>', $errors );

$user\_login = '';

if ( isset( $\_POST['user\_login'] ) && is\_string( $\_POST['user\_login'] ) ) {

$user\_login = wp\_unslash( $\_POST['user\_login'] );

}

?>

<form name="lostpasswordform" id="lostpasswordform" action="<?php echo esc\_url( network\_site\_url( 'wp-login.php?action=lostpassword', 'login\_post' ) ); ?>" method="post">

<p>

<label for="user\_login"><?php \_e( 'Username or Email Address' ); ?></label>

<input type="text" name="user\_login" id="user\_login" class="input" value="<?php echo esc\_attr( $user\_login ); ?>" size="20" autocapitalize="off" />

</p>

<?php

/\*\*

\* Fires inside the lostpassword form tags, before the hidden fields.

\*

\* @since 2.1.0

\*/

do\_action( 'lostpassword\_form' );

?>

<input type="hidden" name="redirect\_to" value="<?php echo esc\_attr( $redirect\_to ); ?>" />

<p class="submit">

<input type="submit" name="wp-submit" id="wp-submit" class="button button-primary button-large" value="<?php esc\_attr\_e( 'Get New Password' ); ?>" />

</p>

</form>

<p id="nav">

<a href="<?php echo esc\_url( wp\_login\_url() ); ?>"><?php \_e( 'Log in' ); ?></a>

<?php

if ( get\_option( 'users\_can\_register' ) ) {

$registration\_url = sprintf( '<a href="%s">%s</a>', esc\_url( wp\_registration\_url() ), \_\_( 'Register' ) );

echo esc\_html( $login\_link\_separator );

/\*\* This filter is documented in wp-includes/general-template.php \*/

echo apply\_filters( 'register', $registration\_url );

}

?>

</p>

<?php

login\_footer( 'user\_login' );

break;

case 'resetpass':

case 'rp':

list( $rp\_path ) = explode( '?', wp\_unslash( $\_SERVER['REQUEST\_URI'] ) );

$rp\_cookie = 'wp-resetpass-' . COOKIEHASH;

if ( isset( $\_GET['key'] ) && isset( $\_GET['login'] ) ) {

$value = sprintf( '%s:%s', wp\_unslash( $\_GET['login'] ), wp\_unslash( $\_GET['key'] ) );

setcookie( $rp\_cookie, $value, 0, $rp\_path, COOKIE\_DOMAIN, is\_ssl(), true );

wp\_safe\_redirect( remove\_query\_arg( array( 'key', 'login' ) ) );

exit;

}

if ( isset( $\_COOKIE[ $rp\_cookie ] ) && 0 < strpos( $\_COOKIE[ $rp\_cookie ], ':' ) ) {

list( $rp\_login, $rp\_key ) = explode( ':', wp\_unslash( $\_COOKIE[ $rp\_cookie ] ), 2 );

$user = check\_password\_reset\_key( $rp\_key, $rp\_login );

if ( isset( $\_POST['pass1'] ) && ! hash\_equals( $rp\_key, $\_POST['rp\_key'] ) ) {

$user = false;

}

} else {

$user = false;

}

if ( ! $user || is\_wp\_error( $user ) ) {

setcookie( $rp\_cookie, ' ', time() - YEAR\_IN\_SECONDS, $rp\_path, COOKIE\_DOMAIN, is\_ssl(), true );

if ( $user && $user->get\_error\_code() === 'expired\_key' ) {

wp\_redirect( site\_url( 'wp-login.php?action=lostpassword&error=expiredkey' ) );

} else {

wp\_redirect( site\_url( 'wp-login.php?action=lostpassword&error=invalidkey' ) );

}

exit;

}

$errors = new WP\_Error();

if ( isset( $\_POST['pass1'] ) && $\_POST['pass1'] !== $\_POST['pass2'] ) {

$errors->add( 'password\_reset\_mismatch', \_\_( '<strong>Error</strong>: The passwords do not match.' ) );

}

/\*\*

\* Fires before the password reset procedure is validated.

\*

\* @since 3.5.0

\*

\* @param WP\_Error $errors WP Error object.

\* @param WP\_User|WP\_Error $user WP\_User object if the login and reset key match. WP\_Error object otherwise.

\*/

do\_action( 'validate\_password\_reset', $errors, $user );

if ( ( ! $errors->has\_errors() ) && isset( $\_POST['pass1'] ) && ! empty( $\_POST['pass1'] ) ) {

reset\_password( $user, $\_POST['pass1'] );

setcookie( $rp\_cookie, ' ', time() - YEAR\_IN\_SECONDS, $rp\_path, COOKIE\_DOMAIN, is\_ssl(), true );

login\_header( \_\_( 'Password Reset' ), '<p class="message reset-pass">' . \_\_( 'Your password has been reset.' ) . ' <a href="' . esc\_url( wp\_login\_url() ) . '">' . \_\_( 'Log in' ) . '</a></p>' );

login\_footer();

exit;

}

wp\_enqueue\_script( 'utils' );

wp\_enqueue\_script( 'user-profile' );

login\_header( \_\_( 'Reset Password' ), '<p class="message reset-pass">' . \_\_( 'Enter your new password below or generate one.' ) . '</p>', $errors );

?>

<form name="resetpassform" id="resetpassform" action="<?php echo esc\_url( network\_site\_url( 'wp-login.php?action=resetpass', 'login\_post' ) ); ?>" method="post" autocomplete="off">

<input type="hidden" id="user\_login" value="<?php echo esc\_attr( $rp\_login ); ?>" autocomplete="off" />

<div class="user-pass1-wrap">

<p>

<label for="pass1"><?php \_e( 'New password' ); ?></label>

</p>

<div class="wp-pwd">

<input type="password" data-reveal="1" data-pw="<?php echo esc\_attr( wp\_generate\_password( 16 ) ); ?>" name="pass1" id="pass1" class="input password-input" size="24" value="" autocomplete="off" aria-describedby="pass-strength-result" />

<button type="button" class="button button-secondary wp-hide-pw hide-if-no-js" data-toggle="0" aria-label="<?php esc\_attr\_e( 'Hide password' ); ?>">

<span class="dashicons dashicons-hidden" aria-hidden="true"></span>

</button>

<div id="pass-strength-result" class="hide-if-no-js" aria-live="polite"><?php \_e( 'Strength indicator' ); ?></div>

</div>

<div class="pw-weak">

<input type="checkbox" name="pw\_weak" id="pw-weak" class="pw-checkbox" />

<label for="pw-weak"><?php \_e( 'Confirm use of weak password' ); ?></label>

</div>

</div>

<p class="user-pass2-wrap">

<label for="pass2"><?php \_e( 'Confirm new password' ); ?></label>

<input type="password" name="pass2" id="pass2" class="input" size="20" value="" autocomplete="off" />

</p>

<p class="description indicator-hint"><?php echo wp\_get\_password\_hint(); ?></p>

<br class="clear" />

<?php

/\*\*

\* Fires following the 'Strength indicator' meter in the user password reset form.

\*

\* @since 3.9.0

\*

\* @param WP\_User $user User object of the user whose password is being reset.

\*/

do\_action( 'resetpass\_form', $user );

?>

<input type="hidden" name="rp\_key" value="<?php echo esc\_attr( $rp\_key ); ?>" />

<p class="submit reset-pass-submit">

<button type="button" class="button wp-generate-pw hide-if-no-js" aria-expanded="true"><?php \_e( 'Generate Password' ); ?></button>

<input type="submit" name="wp-submit" id="wp-submit" class="button button-primary button-large" value="<?php esc\_attr\_e( 'Save Password' ); ?>" />

</p>

</form>

<p id="nav">

<a href="<?php echo esc\_url( wp\_login\_url() ); ?>"><?php \_e( 'Log in' ); ?></a>

<?php

if ( get\_option( 'users\_can\_register' ) ) {

$registration\_url = sprintf( '<a href="%s">%s</a>', esc\_url( wp\_registration\_url() ), \_\_( 'Register' ) );

echo esc\_html( $login\_link\_separator );

/\*\* This filter is documented in wp-includes/general-template.php \*/

echo apply\_filters( 'register', $registration\_url );

}

?>

</p>

<?php

login\_footer( 'user\_pass' );

break;

case 'register':

if ( is\_multisite() ) {

/\*\*

\* Filters the Multisite sign up URL.

\*

\* @since 3.0.0

\*

\* @param string $sign\_up\_url The sign up URL.

\*/

wp\_redirect( apply\_filters( 'wp\_signup\_location', network\_site\_url( 'wp-signup.php' ) ) );

exit;

}

if ( ! get\_option( 'users\_can\_register' ) ) {

wp\_redirect( site\_url( 'wp-login.php?registration=disabled' ) );

exit;

}

$user\_login = '';

$user\_email = '';

if ( $http\_post ) {

if ( isset( $\_POST['user\_login'] ) && is\_string( $\_POST['user\_login'] ) ) {

$user\_login = wp\_unslash( $\_POST['user\_login'] );

}

if ( isset( $\_POST['user\_email'] ) && is\_string( $\_POST['user\_email'] ) ) {

$user\_email = wp\_unslash( $\_POST['user\_email'] );

}

$errors = register\_new\_user( $user\_login, $user\_email );

if ( ! is\_wp\_error( $errors ) ) {

$redirect\_to = ! empty( $\_POST['redirect\_to'] ) ? $\_POST['redirect\_to'] : 'wp-login.php?checkemail=registered';

wp\_safe\_redirect( $redirect\_to );

exit;

}

}

$registration\_redirect = ! empty( $\_REQUEST['redirect\_to'] ) ? $\_REQUEST['redirect\_to'] : '';

/\*\*

\* Filters the registration redirect URL.

\*

\* @since 3.0.0

\*

\* @param string $registration\_redirect The redirect destination URL.

\*/

$redirect\_to = apply\_filters( 'registration\_redirect', $registration\_redirect );

login\_header( \_\_( 'Registration Form' ), '<p class="message register">' . \_\_( 'Register For This Site' ) . '</p>', $errors );

?>

<form name="registerform" id="registerform" action="<?php echo esc\_url( site\_url( 'wp-login.php?action=register', 'login\_post' ) ); ?>" method="post" novalidate="novalidate">

<p>

<label for="user\_login"><?php \_e( 'Username' ); ?></label>

<input type="text" name="user\_login" id="user\_login" class="input" value="<?php echo esc\_attr( wp\_unslash( $user\_login ) ); ?>" size="20" autocapitalize="off" />

</p>

<p>

<label for="user\_email"><?php \_e( 'Email' ); ?></label>

<input type="email" name="user\_email" id="user\_email" class="input" value="<?php echo esc\_attr( wp\_unslash( $user\_email ) ); ?>" size="25" />

</p>

<?php

/\*\*

\* Fires following the 'Email' field in the user registration form.

\*

\* @since 2.1.0

\*/

do\_action( 'register\_form' );

?>

<p id="reg\_passmail">

<?php \_e( 'Registration confirmation will be emailed to you.' ); ?>

</p>

<br class="clear" />

<input type="hidden" name="redirect\_to" value="<?php echo esc\_attr( $redirect\_to ); ?>" />

<p class="submit">

<input type="submit" name="wp-submit" id="wp-submit" class="button button-primary button-large" value="<?php esc\_attr\_e( 'Register' ); ?>" />

</p>

</form>

<p id="nav">

<a href="<?php echo esc\_url( wp\_login\_url() ); ?>"><?php \_e( 'Log in' ); ?></a>

<?php echo esc\_html( $login\_link\_separator ); ?>

<a href="<?php echo esc\_url( wp\_lostpassword\_url() ); ?>"><?php \_e( 'Lost your password?' ); ?></a>

</p>

<?php

login\_footer( 'user\_login' );

break;

case 'checkemail':

$redirect\_to = admin\_url();

$errors = new WP\_Error();

if ( 'confirm' === $\_GET['checkemail'] ) {

$errors->add(

'confirm',

sprintf(

/\* translators: %s: Link to the login page. \*/

\_\_( 'Check your email for the confirmation link, then visit the <a href="%s">login page</a>.' ),

wp\_login\_url()

),

'message'

);

} elseif ( 'registered' === $\_GET['checkemail'] ) {

$errors->add(

'registered',

sprintf(

/\* translators: %s: Link to the login page. \*/

\_\_( 'Registration complete. Please check your email, then visit the <a href="%s">login page</a>.' ),

wp\_login\_url()

),

'message'

);

}

/\*\* This action is documented in wp-login.php \*/

$errors = apply\_filters( 'wp\_login\_errors', $errors, $redirect\_to );

login\_header( \_\_( 'Check your email' ), '', $errors );

login\_footer();

break;

case 'confirmaction':

if ( ! isset( $\_GET['request\_id'] ) ) {

wp\_die( \_\_( 'Missing request ID.' ) );

}

if ( ! isset( $\_GET['confirm\_key'] ) ) {

wp\_die( \_\_( 'Missing confirm key.' ) );

}

$request\_id = (int) $\_GET['request\_id'];

$key = sanitize\_text\_field( wp\_unslash( $\_GET['confirm\_key'] ) );

$result = wp\_validate\_user\_request\_key( $request\_id, $key );

if ( is\_wp\_error( $result ) ) {

wp\_die( $result );

}

/\*\*

\* Fires an action hook when the account action has been confirmed by the user.

\*

\* Using this you can assume the user has agreed to perform the action by

\* clicking on the link in the confirmation email.

\*

\* After firing this action hook the page will redirect to wp-login a callback

\* redirects or exits first.

\*

\* @since 4.9.6

\*

\* @param int $request\_id Request ID.

\*/

do\_action( 'user\_request\_action\_confirmed', $request\_id );

$message = \_wp\_privacy\_account\_request\_confirmed\_message( $request\_id );

login\_header( \_\_( 'User action confirmed.' ), $message );

login\_footer();

exit;

case 'login':

default:

$secure\_cookie = '';

$customize\_login = isset( $\_REQUEST['customize-login'] );

if ( $customize\_login ) {

wp\_enqueue\_script( 'customize-base' );

}

// If the user wants SSL but the session is not SSL, force a secure cookie.

if ( ! empty( $\_POST['log'] ) && ! force\_ssl\_admin() ) {

$user\_name = sanitize\_user( wp\_unslash( $\_POST['log'] ) );

$user = get\_user\_by( 'login', $user\_name );

if ( ! $user && strpos( $user\_name, '@' ) ) {

$user = get\_user\_by( 'email', $user\_name );

}

if ( $user ) {

if ( get\_user\_option( 'use\_ssl', $user->ID ) ) {

$secure\_cookie = true;

force\_ssl\_admin( true );

}

}

}

if ( isset( $\_REQUEST['redirect\_to'] ) ) {

$redirect\_to = $\_REQUEST['redirect\_to'];

// Redirect to HTTPS if user wants SSL.

if ( $secure\_cookie && false !== strpos( $redirect\_to, 'wp-admin' ) ) {

$redirect\_to = preg\_replace( '|^http://|', 'https://', $redirect\_to );

}

} else {

$redirect\_to = admin\_url();

}

$reauth = empty( $\_REQUEST['reauth'] ) ? false : true;

$user = wp\_signon( array(), $secure\_cookie );

if ( empty( $\_COOKIE[ LOGGED\_IN\_COOKIE ] ) ) {

if ( headers\_sent() ) {

$user = new WP\_Error(

'test\_cookie',

sprintf(

/\* translators: 1: Browser cookie documentation URL, 2: Support forums URL. \*/

\_\_( '<strong>Error</strong>: Cookies are blocked due to unexpected output. For help, please see <a href="%1$s">this documentation</a> or try the <a href="%2$s">support forums</a>.' ),

\_\_( 'https://wordpress.org/support/article/cookies/' ),

\_\_( 'https://wordpress.org/support/forums/' )

)

);

} elseif ( isset( $\_POST['testcookie'] ) && empty( $\_COOKIE[ TEST\_COOKIE ] ) ) {

// If cookies are disabled, we can't log in even with a valid user and password.

$user = new WP\_Error(

'test\_cookie',

sprintf(

/\* translators: %s: Browser cookie documentation URL. \*/

\_\_( '<strong>Error</strong>: Cookies are blocked or not supported by your browser. You must <a href="%s">enable cookies</a> to use WordPress.' ),

\_\_( 'https://wordpress.org/support/article/cookies/#enable-cookies-in-your-browser' )

)

);

}

}

$requested\_redirect\_to = isset( $\_REQUEST['redirect\_to'] ) ? $\_REQUEST['redirect\_to'] : '';

/\*\*

\* Filters the login redirect URL.

\*

\* @since 3.0.0

\*

\* @param string $redirect\_to The redirect destination URL.

\* @param string $requested\_redirect\_to The requested redirect destination URL passed as a parameter.

\* @param WP\_User|WP\_Error $user WP\_User object if login was successful, WP\_Error object otherwise.

\*/

$redirect\_to = apply\_filters( 'login\_redirect', $redirect\_to, $requested\_redirect\_to, $user );

if ( ! is\_wp\_error( $user ) && ! $reauth ) {

if ( $interim\_login ) {

$message = '<p class="message">' . \_\_( 'You have logged in successfully.' ) . '</p>';

$interim\_login = 'success';

login\_header( '', $message );

?>

</div>

<?php

/\*\* This action is documented in wp-login.php \*/

do\_action( 'login\_footer' );

if ( $customize\_login ) {

?>

<script type="text/javascript">setTimeout( function(){ new wp.customize.Messenger({ url: '<?php echo wp\_customize\_url(); ?>', channel: 'login' }).send('login') }, 1000 );</script>

<?php

}

?>

</body></html>

<?php

exit;

}

// Check if it is time to add a redirect to the admin email confirmation screen.

if ( is\_a( $user, 'WP\_User' ) && $user->exists() && $user->has\_cap( 'manage\_options' ) ) {

$admin\_email\_lifespan = (int) get\_option( 'admin\_email\_lifespan' );

// If `0` (or anything "falsey" as it is cast to int) is returned, the user will not be redirected

// to the admin email confirmation screen.

/\*\* This filter is documented in wp-login.php \*/

$admin\_email\_check\_interval = (int) apply\_filters( 'admin\_email\_check\_interval', 6 \* MONTH\_IN\_SECONDS );

if ( $admin\_email\_check\_interval > 0 && time() > $admin\_email\_lifespan ) {

$redirect\_to = add\_query\_arg(

array(

'action' => 'confirm\_admin\_email',

'wp\_lang' => get\_user\_locale( $user ),

),

wp\_login\_url( $redirect\_to )

);

}

}

if ( ( empty( $redirect\_to ) || 'wp-admin/' === $redirect\_to || admin\_url() === $redirect\_to ) ) {

// If the user doesn't belong to a blog, send them to user admin. If the user can't edit posts, send them to their profile.

if ( is\_multisite() && ! get\_active\_blog\_for\_user( $user->ID ) && ! is\_super\_admin( $user->ID ) ) {

$redirect\_to = user\_admin\_url();

} elseif ( is\_multisite() && ! $user->has\_cap( 'read' ) ) {

$redirect\_to = get\_dashboard\_url( $user->ID );

} elseif ( ! $user->has\_cap( 'edit\_posts' ) ) {

$redirect\_to = $user->has\_cap( 'read' ) ? admin\_url( 'profile.php' ) : home\_url();

}

wp\_redirect( $redirect\_to );

exit;

}

wp\_safe\_redirect( $redirect\_to );

exit;

}

$errors = $user;

// Clear errors if loggedout is set.

if ( ! empty( $\_GET['loggedout'] ) || $reauth ) {

$errors = new WP\_Error();

}

if ( empty( $\_POST ) && $errors->get\_error\_codes() === array( 'empty\_username', 'empty\_password' ) ) {

$errors = new WP\_Error( '', '' );

}

if ( $interim\_login ) {

if ( ! $errors->has\_errors() ) {

$errors->add( 'expired', \_\_( 'Your session has expired. Please log in to continue where you left off.' ), 'message' );

}

} else {

// Some parts of this script use the main login form to display a message.

if ( isset( $\_GET['loggedout'] ) && $\_GET['loggedout'] ) {

$errors->add( 'loggedout', \_\_( 'You are now logged out.' ), 'message' );

} elseif ( isset( $\_GET['registration'] ) && 'disabled' === $\_GET['registration'] ) {

$errors->add( 'registerdisabled', \_\_( '<strong>Error</strong>: User registration is currently not allowed.' ) );

} elseif ( strpos( $redirect\_to, 'about.php?updated' ) ) {

$errors->add( 'updated', \_\_( '<strong>You have successfully updated WordPress!</strong> Please log back in to see what&#8217;s new.' ), 'message' );

} elseif ( WP\_Recovery\_Mode\_Link\_Service::LOGIN\_ACTION\_ENTERED === $action ) {

$errors->add( 'enter\_recovery\_mode', \_\_( 'Recovery Mode Initialized. Please log in to continue.' ), 'message' );

} elseif ( isset( $\_GET['redirect\_to'] ) && false !== strpos( $\_GET['redirect\_to'], 'wp-admin/authorize-application.php' ) ) {

$query\_component = wp\_parse\_url( $\_GET['redirect\_to'], PHP\_URL\_QUERY );

parse\_str( $query\_component, $query );

if ( ! empty( $query['app\_name'] ) ) {

/\* translators: 1: Website name, 2: Application name. \*/

$message = sprintf( 'Please log in to %1$s to authorize %2$s to connect to your account.', get\_bloginfo( 'name', 'display' ), '<strong>' . esc\_html( $query['app\_name'] ) . '</strong>' );

} else {

/\* translators: %s: Website name. \*/

$message = sprintf( 'Please log in to %s to proceed with authorization.', get\_bloginfo( 'name', 'display' ) );

}

$errors->add( 'authorize\_application', $message, 'message' );

}

}

/\*\*

\* Filters the login page errors.

\*

\* @since 3.6.0

\*

\* @param WP\_Error $errors WP Error object.

\* @param string $redirect\_to Redirect destination URL.

\*/

$errors = apply\_filters( 'wp\_login\_errors', $errors, $redirect\_to );

// Clear any stale cookies.

if ( $reauth ) {

wp\_clear\_auth\_cookie();

}

login\_header( \_\_( 'Log In' ), '', $errors );

if ( isset( $\_POST['log'] ) ) {

$user\_login = ( 'incorrect\_password' === $errors->get\_error\_code() || 'empty\_password' === $errors->get\_error\_code() ) ? esc\_attr( wp\_unslash( $\_POST['log'] ) ) : '';

}

$rememberme = ! empty( $\_POST['rememberme'] );

if ( $errors->has\_errors() ) {

$aria\_describedby\_error = ' aria-describedby="login\_error"';

} else {

$aria\_describedby\_error = '';

}

wp\_enqueue\_script( 'user-profile' );

?>

<form name="loginform" id="loginform" action="<?php echo esc\_url( site\_url( 'wp-login.php', 'login\_post' ) ); ?>" method="post">

<p>

<label for="user\_login"><?php \_e( 'Username or Email Address' ); ?></label>

<input type="text" name="log" id="user\_login"<?php echo $aria\_describedby\_error; ?> class="input" value="<?php echo esc\_attr( $user\_login ); ?>" size="20" autocapitalize="off" />

</p>

<div class="user-pass-wrap">

<label for="user\_pass"><?php \_e( 'Password' ); ?></label>

<div class="wp-pwd">

<input type="password" name="pwd" id="user\_pass"<?php echo $aria\_describedby\_error; ?> class="input password-input" value="" size="20" />

<button type="button" class="button button-secondary wp-hide-pw hide-if-no-js" data-toggle="0" aria-label="<?php esc\_attr\_e( 'Show password' ); ?>">

<span class="dashicons dashicons-visibility" aria-hidden="true"></span>

</button>

</div>

</div>

<?php

/\*\*

\* Fires following the 'Password' field in the login form.

\*

\* @since 2.1.0

\*/

do\_action( 'login\_form' );

?>

<p class="forgetmenot"><input name="rememberme" type="checkbox" id="rememberme" value="forever" <?php checked( $rememberme ); ?> /> <label for="rememberme"><?php esc\_html\_e( 'Remember Me' ); ?></label></p>

<p class="submit">

<input type="submit" name="wp-submit" id="wp-submit" class="button button-primary button-large" value="<?php esc\_attr\_e( 'Log In' ); ?>" />

<?php

if ( $interim\_login ) {

?>

<input type="hidden" name="interim-login" value="1" />

<?php

} else {

?>

<input type="hidden" name="redirect\_to" value="<?php echo esc\_attr( $redirect\_to ); ?>" />

<?php

}

if ( $customize\_login ) {

?>

<input type="hidden" name="customize-login" value="1" />

<?php

}

?>

<input type="hidden" name="testcookie" value="1" />

</p>

</form>

<?php

if ( ! $interim\_login ) {

?>

<p id="nav">

<?php

if ( get\_option( 'users\_can\_register' ) ) {

$registration\_url = sprintf( '<a href="%s">%s</a>', esc\_url( wp\_registration\_url() ), \_\_( 'Register' ) );

/\*\* This filter is documented in wp-includes/general-template.php \*/

echo apply\_filters( 'register', $registration\_url );

echo esc\_html( $login\_link\_separator );

}

?>

<a href="<?php echo esc\_url( wp\_lostpassword\_url() ); ?>"><?php \_e( 'Lost your password?' ); ?></a>

</p>

<?php

}

$login\_script = 'function wp\_attempt\_focus() {';

$login\_script .= 'setTimeout( function() {';

$login\_script .= 'try {';

if ( $user\_login ) {

$login\_script .= 'd = document.getElementById( "user\_pass" ); d.value = "";';

} else {

$login\_script .= 'd = document.getElementById( "user\_login" );';

if ( $errors->get\_error\_code() === 'invalid\_username' ) {

$login\_script .= 'd.value = "";';

}

}

$login\_script .= 'd.focus(); d.select();';

$login\_script .= '} catch( er ) {}';

$login\_script .= '}, 200);';

$login\_script .= "}\n"; // End of wp\_attempt\_focus().

/\*\*

\* Filters whether to print the call to `wp\_attempt\_focus()` on the login screen.

\*

\* @since 4.8.0

\*

\* @param bool $print Whether to print the function call. Default true.

\*/

if ( apply\_filters( 'enable\_login\_autofocus', true ) && ! $error ) {

$login\_script .= "wp\_attempt\_focus();\n";

}

// Run `wpOnload()` if defined.

$login\_script .= "if ( typeof wpOnload === 'function' ) { wpOnload() }";

?>

<script type="text/javascript">

<?php echo $login\_script; ?>

</script>

<?php

if ( $interim\_login ) {

?>

<script type="text/javascript">

( function() {

try {

var i, links = document.getElementsByTagName( 'a' );

for ( i in links ) {

if ( links[i].href ) {

links[i].target = '\_blank';

links[i].rel = 'noopener';

}

}

} catch( er ) {}

}());

</script>

<?php

}

login\_footer();

break;

} // End action switch.

