!function(root, factory) {

"object" == typeof exports && "object" == typeof module ? module.exports = factory(require("babel-runtime/core-js/json/stringify"), require("babel-runtime/helpers/slicedToArray"), require("prop-types"), require("react"), require("react-modal")) : "function" == typeof define && define.amd ? define([ "babel-runtime/core-js/json/stringify", "babel-runtime/helpers/slicedToArray", "prop-types", "react", "react-modal" ], factory) : "object" == typeof exports ? exports.ReactImageLightbox = factory(require("babel-runtime/core-js/json/stringify"), require("babel-runtime/helpers/slicedToArray"), require("prop-types"), require("react"), require("react-modal")) : root.ReactImageLightbox = factory(root["babel-runtime/core-js/json/stringify"], root["babel-runtime/helpers/slicedToArray"], root["prop-types"], root.react, root["react-modal"]);

}(this, function(\_\_WEBPACK\_EXTERNAL\_MODULE\_8\_\_, \_\_WEBPACK\_EXTERNAL\_MODULE\_9\_\_, \_\_WEBPACK\_EXTERNAL\_MODULE\_10\_\_, \_\_WEBPACK\_EXTERNAL\_MODULE\_11\_\_, \_\_WEBPACK\_EXTERNAL\_MODULE\_12\_\_) {

/\*\*\*\*\*\*/

return function(modules) {

/\*\*\*\*\*\*/

/\*\*\*\*\*\*/

// The require function

/\*\*\*\*\*\*/

function \_\_webpack\_require\_\_(moduleId) {

/\*\*\*\*\*\*/

/\*\*\*\*\*\*/

// Check if module is in cache

/\*\*\*\*\*\*/

if (installedModules[moduleId]) /\*\*\*\*\*\*/

return installedModules[moduleId].exports;

/\*\*\*\*\*\*/

/\*\*\*\*\*\*/

// Create a new module (and put it into the cache)

/\*\*\*\*\*\*/

var module = installedModules[moduleId] = {

/\*\*\*\*\*\*/

exports: {},

/\*\*\*\*\*\*/

id: moduleId,

/\*\*\*\*\*\*/

loaded: !1

};

/\*\*\*\*\*\*/

/\*\*\*\*\*\*/

// Return the exports of the module

/\*\*\*\*\*\*/

/\*\*\*\*\*\*/

/\*\*\*\*\*\*/

// Execute the module function

/\*\*\*\*\*\*/

/\*\*\*\*\*\*/

/\*\*\*\*\*\*/

// Flag the module as loaded

/\*\*\*\*\*\*/

return modules[moduleId].call(module.exports, module, module.exports, \_\_webpack\_require\_\_),

module.loaded = !0, module.exports;

}

// webpackBootstrap

/\*\*\*\*\*\*/

// The module cache

/\*\*\*\*\*\*/

var installedModules = {};

/\*\*\*\*\*\*/

/\*\*\*\*\*\*/

// Load entry module and return exports

/\*\*\*\*\*\*/

/\*\*\*\*\*\*/

/\*\*\*\*\*\*/

/\*\*\*\*\*\*/

// expose the modules object (\_\_webpack\_modules\_\_)

/\*\*\*\*\*\*/

/\*\*\*\*\*\*/

/\*\*\*\*\*\*/

// expose the module cache

/\*\*\*\*\*\*/

/\*\*\*\*\*\*/

/\*\*\*\*\*\*/

// \_\_webpack\_public\_path\_\_

/\*\*\*\*\*\*/

return \_\_webpack\_require\_\_.m = modules, \_\_webpack\_require\_\_.c = installedModules,

\_\_webpack\_require\_\_.p = "", \_\_webpack\_require\_\_(0);

}([ /\* 0 \*/

/\*\*\*/

function(module, exports, \_\_webpack\_require\_\_) {

"use strict";

module.exports = \_\_webpack\_require\_\_(2).default;

}, /\* 1 \*/

/\*\*\*/

function(module, exports) {

"use strict";

Object.defineProperty(exports, "\_\_esModule", {

value: !0

});

// Min image zoom level

exports.MIN\_ZOOM\_LEVEL = 0, exports.MAX\_ZOOM\_LEVEL = 300, exports.ZOOM\_RATIO = 1.007,

exports.ZOOM\_BUTTON\_INCREMENT\_SIZE = 100, exports.WHEEL\_MOVE\_X\_THRESHOLD = 200,

exports.WHEEL\_MOVE\_Y\_THRESHOLD = 1, exports.KEYS = {

ESC: 27,

LEFT\_ARROW: 37,

RIGHT\_ARROW: 39

}, exports.ACTION\_NONE = 0, exports.ACTION\_MOVE = 1, exports.ACTION\_SWIPE = 2, exports.ACTION\_PINCH = 3,

exports.ACTION\_ROTATE = 4, exports.SOURCE\_ANY = 0, exports.SOURCE\_MOUSE = 1, exports.SOURCE\_TOUCH = 2,

exports.SOURCE\_POINTER = 3, exports.MIN\_SWIPE\_DISTANCE = 200;

}, /\* 2 \*/

/\*\*\*/

function(module, exports, \_\_webpack\_require\_\_) {

"use strict";

function \_interopRequireDefault(obj) {

return obj && obj.\_\_esModule ? obj : {

default: obj

};

}

function \_defineProperty(obj, key, value) {

return key in obj ? Object.defineProperty(obj, key, {

value: value,

enumerable: !0,

configurable: !0,

writable: !0

}) : obj[key] = value, obj;

}

function \_classCallCheck(instance, Constructor) {

if (!(instance instanceof Constructor)) throw new TypeError("Cannot call a class as a function");

}

function \_possibleConstructorReturn(self, call) {

if (!self) throw new ReferenceError("this hasn't been initialised - super() hasn't been called");

return !call || "object" != typeof call && "function" != typeof call ? self : call;

}

function \_inherits(subClass, superClass) {

if ("function" != typeof superClass && null !== superClass) throw new TypeError("Super expression must either be null or a function, not " + typeof superClass);

subClass.prototype = Object.create(superClass && superClass.prototype, {

constructor: {

value: subClass,

enumerable: !1,

writable: !0,

configurable: !0

}

}), superClass && (Object.setPrototypeOf ? Object.setPrototypeOf(subClass, superClass) : subClass.\_\_proto\_\_ = superClass);

}

Object.defineProperty(exports, "\_\_esModule", {

value: !0

});

var \_typeof = "function" == typeof Symbol && "symbol" == typeof Symbol.iterator ? function(obj) {

return typeof obj;

} : function(obj) {

return obj && "function" == typeof Symbol && obj.constructor === Symbol && obj !== Symbol.prototype ? "symbol" : typeof obj;

}, \_slicedToArray = function() {

function sliceIterator(arr, i) {

var \_arr = [], \_n = !0, \_d = !1, \_e = void 0;

try {

for (var \_s, \_i = arr[Symbol.iterator](); !(\_n = (\_s = \_i.next()).done) && (\_arr.push(\_s.value),

!i || \_arr.length !== i); \_n = !0) ;

} catch (err) {

\_d = !0, \_e = err;

} finally {

try {

!\_n && \_i.return && \_i.return();

} finally {

if (\_d) throw \_e;

}

}

return \_arr;

}

return function(arr, i) {

if (Array.isArray(arr)) return arr;

if (Symbol.iterator in Object(arr)) return sliceIterator(arr, i);

throw new TypeError("Invalid attempt to destructure non-iterable instance");

};

}(), \_createClass = function() {

function defineProperties(target, props) {

for (var i = 0; i < props.length; i++) {

var descriptor = props[i];

descriptor.enumerable = descriptor.enumerable || !1, descriptor.configurable = !0,

"value" in descriptor && (descriptor.writable = !0), Object.defineProperty(target, descriptor.key, descriptor);

}

}

return function(Constructor, protoProps, staticProps) {

return protoProps && defineProperties(Constructor.prototype, protoProps), staticProps && defineProperties(Constructor, staticProps),

Constructor;

};

}(), \_extends = Object.assign || function(target) {

for (var i = 1; i < arguments.length; i++) {

var source = arguments[i];

for (var key in source) Object.prototype.hasOwnProperty.call(source, key) && (target[key] = source[key]);

}

return target;

}, \_react = \_\_webpack\_require\_\_(11), \_react2 = \_interopRequireDefault(\_react), \_propTypes = \_\_webpack\_require\_\_(10), \_propTypes2 = \_interopRequireDefault(\_propTypes), \_reactModal = \_\_webpack\_require\_\_(12), \_reactModal2 = \_interopRequireDefault(\_reactModal), \_util = \_\_webpack\_require\_\_(3), \_constant = \_\_webpack\_require\_\_(1), \_style = \_\_webpack\_require\_\_(6), \_style2 = \_interopRequireDefault(\_style), styles = \_style2.default, \_ieVersion = (0,

\_util.getIEVersion)();

\_ieVersion < 10 && (styles = \_extends({}, styles, {

toolbarSide: styles.toolbarSide + " " + styles.toolbarSideNoFlex,

toolbarLeftSide: styles.toolbarLeftSide + " " + styles.toolbarLeftSideNoFlex,

toolbarRightSide: styles.toolbarRightSide + " " + styles.toolbarRightSideNoFlex

}));

var ReactImageLightbox = function(\_Component) {

function ReactImageLightbox(props) {

\_classCallCheck(this, ReactImageLightbox);

var \_this = \_possibleConstructorReturn(this, (ReactImageLightbox.\_\_proto\_\_ || Object.getPrototypeOf(ReactImageLightbox)).call(this, props));

return \_this.state = {

//-----------------------------

// Animation

//-----------------------------

// Lightbox is closing

// When Lightbox is mounted, if animation is enabled it will open with the reverse of the closing animation

isClosing: !props.animationDisabled,

// Component parts should animate (e.g., when images are moving, or image is being zoomed)

shouldAnimate: !1,

//-----------------------------

// Zoom settings

//-----------------------------

// Zoom level of image

zoomLevel: \_constant.MIN\_ZOOM\_LEVEL,

//-----------------------------

// Image position settings

//-----------------------------

// Horizontal offset from center

offsetX: 0,

// Vertical offset from center

offsetY: 0

}, \_this.closeIfClickInner = \_this.closeIfClickInner.bind(\_this), \_this.handleImageDoubleClick = \_this.handleImageDoubleClick.bind(\_this),

\_this.handleImageMouseWheel = \_this.handleImageMouseWheel.bind(\_this), \_this.handleKeyInput = \_this.handleKeyInput.bind(\_this),

\_this.handleMouseUp = \_this.handleMouseUp.bind(\_this), \_this.handleMouseDown = \_this.handleMouseDown.bind(\_this),

\_this.handleMouseMove = \_this.handleMouseMove.bind(\_this), \_this.handleOuterMousewheel = \_this.handleOuterMousewheel.bind(\_this),

\_this.handleTouchStart = \_this.handleTouchStart.bind(\_this), \_this.handleTouchMove = \_this.handleTouchMove.bind(\_this),

\_this.handleTouchEnd = \_this.handleTouchEnd.bind(\_this), \_this.handlePointerEvent = \_this.handlePointerEvent.bind(\_this),

\_this.handleCaptionMousewheel = \_this.handleCaptionMousewheel.bind(\_this), \_this.handleWindowResize = \_this.handleWindowResize.bind(\_this),

\_this.handleZoomInButtonClick = \_this.handleZoomInButtonClick.bind(\_this), \_this.handleZoomOutButtonClick = \_this.handleZoomOutButtonClick.bind(\_this),

\_this.requestClose = \_this.requestClose.bind(\_this), \_this.requestMoveNext = \_this.requestMoveNext.bind(\_this),

\_this.requestMovePrev = \_this.requestMovePrev.bind(\_this), \_this;

}

return \_inherits(ReactImageLightbox, \_Component), \_createClass(ReactImageLightbox, [ {

key: "componentWillMount",

value: function() {

// Timeouts - always clear it before umount

this.timeouts = [], // Current action

this.currentAction = \_constant.ACTION\_NONE, // Events source

this.eventsSource = \_constant.SOURCE\_ANY, // Empty pointers list

this.pointerList = [], // Prevent inner close

this.preventInnerClose = !1, this.preventInnerCloseTimeout = null, // Whether event listeners for keyboard and mouse input have been attached or not

this.listenersAttached = !1, // Used to disable animation when changing props.mainSrc|nextSrc|prevSrc

this.keyPressed = !1, // Used to store load state / dimensions of images

this.imageCache = {}, // Time the last keydown event was called (used in keyboard action rate limiting)

this.lastKeyDownTime = 0, // Used for debouncing window resize event

this.resizeTimeout = null, // Used to determine when actions are triggered by the scroll wheel

this.wheelActionTimeout = null, this.resetScrollTimeout = null, this.scrollX = 0,

this.scrollY = 0, // Used in panning zoomed images

this.moveStartX = 0, this.moveStartY = 0, this.moveStartOffsetX = 0, this.moveStartOffsetY = 0,

// Used to swipe

this.swipeStartX = 0, this.swipeStartY = 0, this.swipeEndX = 0, this.swipeEndY = 0,

// Used to pinch

this.pinchTouchList = null, this.pinchDistance = 0, // Used to differentiate between images with identical src

this.keyCounter = 0, // Used to detect a move when all src's remain unchanged (four or more of the same image in a row)

this.moveRequested = !1, this.props.animationDisabled || // Make opening animation play

this.setState({

isClosing: !1

});

}

}, {

key: "componentDidMount",

value: function() {

this.mounted = !0, ReactImageLightbox.loadStyles(), this.attachListeners(), this.loadAllImages();

}

}, {

key: "shouldComponentUpdate",

value: function() {

// Wait for move...

return !this.moveRequested;

}

}, {

key: "componentWillReceiveProps",

value: function(nextProps) {

var \_this2 = this, sourcesChanged = !1, prevSrcDict = {}, nextSrcDict = {};

this.getSrcTypes().forEach(function(srcType) {

\_this2.props[srcType.name] !== nextProps[srcType.name] && (sourcesChanged = !0,

prevSrcDict[\_this2.props[srcType.name]] = !0, nextSrcDict[nextProps[srcType.name]] = !0);

}), (sourcesChanged || this.moveRequested) && (// Reset the loaded state for images not rendered next

Object.keys(prevSrcDict).forEach(function(prevSrc) {

!(prevSrc in nextSrcDict) && prevSrc in \_this2.imageCache && (\_this2.imageCache[prevSrc].loaded = !1);

}), this.moveRequested = !1, // Load any new images

this.loadAllImages(nextProps));

}

}, {

key: "componentWillUnmount",

value: function() {

this.mounted = !1, this.detachListeners(), this.timeouts.forEach(function(tid) {

return clearTimeout(tid);

});

}

}, {

key: "setTimeout",

value: function(\_setTimeout) {

function setTimeout(\_x, \_x2) {

return \_setTimeout.apply(this, arguments);

}

return setTimeout.toString = function() {

return \_setTimeout.toString();

}, setTimeout;

}(function(func, time) {

var \_this3 = this, id = setTimeout(function() {

\_this3.timeouts = \_this3.timeouts.filter(function(tid) {

return tid !== id;

}), func();

}, time);

return this.timeouts.push(id), id;

})

}, {

key: "clearTimeout",

value: function(\_clearTimeout) {

function clearTimeout(\_x3) {

return \_clearTimeout.apply(this, arguments);

}

return clearTimeout.toString = function() {

return \_clearTimeout.toString();

}, clearTimeout;

}(function(id) {

this.timeouts = this.timeouts.filter(function(tid) {

return tid !== id;

}), clearTimeout(id);

})

}, {

key: "attachListeners",

value: function() {

this.listenersAttached || "undefined" == typeof window || (window.addEventListener("resize", this.handleWindowResize),

window.addEventListener("mouseup", this.handleMouseUp), window.addEventListener("touchend", this.handleTouchEnd),

window.addEventListener("touchcancel", this.handleTouchEnd), window.addEventListener("pointerdown", this.handlePointerEvent),

window.addEventListener("pointermove", this.handlePointerEvent), window.addEventListener("pointerup", this.handlePointerEvent),

window.addEventListener("pointercancel", this.handlePointerEvent), // Have to add an extra mouseup handler to catch mouseup events outside of the window

// if the page containing the lightbox is displayed in an iframe

(0, \_util.isInSameOriginIframe)() && (window.top.addEventListener("mouseup", this.handleMouseUp),

window.top.addEventListener("touchend", this.handleTouchEnd), window.top.addEventListener("touchcancel", this.handleTouchEnd),

window.top.addEventListener("pointerdown", this.handlePointerEvent), window.top.addEventListener("pointermove", this.handlePointerEvent),

window.top.addEventListener("pointerup", this.handlePointerEvent), window.top.addEventListener("pointercancel", this.handlePointerEvent)),

this.listenersAttached = !0);

}

}, {

key: "changeZoom",

value: function(zoomLevel, clientX, clientY) {

// Ignore if zoom disabled

if (this.props.enableZoom) {

// Constrain zoom level to the set bounds

var nextZoomLevel = Math.max(\_constant.MIN\_ZOOM\_LEVEL, Math.min(\_constant.MAX\_ZOOM\_LEVEL, zoomLevel));

// Ignore requests that don't change the zoom level

if (nextZoomLevel !== this.state.zoomLevel) {

if (nextZoomLevel === \_constant.MIN\_ZOOM\_LEVEL) // Snap back to center if zoomed all the way out

return this.setState({

zoomLevel: nextZoomLevel,

offsetX: 0,

offsetY: 0

});

var imageBaseSize = this.getBestImageForType("mainSrc");

if (null !== imageBaseSize) {

var currentZoomMultiplier = this.getZoomMultiplier(), nextZoomMultiplier = this.getZoomMultiplier(nextZoomLevel), boxRect = this.getLightboxRect(), pointerX = "undefined" != typeof clientX ? clientX - boxRect.left : boxRect.width / 2, pointerY = "undefined" != typeof clientY ? clientY - boxRect.top : boxRect.height / 2, currentImageOffsetX = (boxRect.width - imageBaseSize.width \* currentZoomMultiplier) / 2, currentImageOffsetY = (boxRect.height - imageBaseSize.height \* currentZoomMultiplier) / 2, currentImageRealOffsetX = currentImageOffsetX - this.state.offsetX, currentImageRealOffsetY = currentImageOffsetY - this.state.offsetY, currentPointerXRelativeToImage = (pointerX - currentImageRealOffsetX) / currentZoomMultiplier, currentPointerYRelativeToImage = (pointerY - currentImageRealOffsetY) / currentZoomMultiplier, nextImageRealOffsetX = pointerX - currentPointerXRelativeToImage \* nextZoomMultiplier, nextImageRealOffsetY = pointerY - currentPointerYRelativeToImage \* nextZoomMultiplier, nextImageOffsetX = (boxRect.width - imageBaseSize.width \* nextZoomMultiplier) / 2, nextImageOffsetY = (boxRect.height - imageBaseSize.height \* nextZoomMultiplier) / 2, nextOffsetX = nextImageOffsetX - nextImageRealOffsetX, nextOffsetY = nextImageOffsetY - nextImageRealOffsetY;

// When zooming out, limit the offset so things don't get left askew

if (this.currentAction !== \_constant.ACTION\_PINCH) {

var maxOffsets = this.getMaxOffsets();

this.state.zoomLevel > nextZoomLevel && (nextOffsetX = Math.max(maxOffsets.minX, Math.min(maxOffsets.maxX, nextOffsetX)),

nextOffsetY = Math.max(maxOffsets.minY, Math.min(maxOffsets.maxY, nextOffsetY)));

}

this.setState({

zoomLevel: nextZoomLevel,

offsetX: nextOffsetX,

offsetY: nextOffsetY

});

}

}

}

}

}, {

key: "closeIfClickInner",

value: function(event) {

!this.preventInnerClose && event.target.className.search(/\bril-inner\b/) > -1 && this.requestClose(event);

}

}, {

key: "setPreventInnerClose",

value: function() {

var \_this4 = this;

this.preventInnerCloseTimeout && this.clearTimeout(this.preventInnerCloseTimeout),

this.preventInnerClose = !0, this.preventInnerCloseTimeout = this.setTimeout(function() {

\_this4.preventInnerClose = !1, \_this4.preventInnerCloseTimeout = null;

}, 100);

}

}, {

key: "detachListeners",

value: function() {

this.listenersAttached && (window.removeEventListener("resize", this.handleWindowResize),

window.removeEventListener("mouseup", this.handleMouseUp), window.removeEventListener("touchend", this.handleTouchEnd),

window.removeEventListener("touchcancel", this.handleTouchEnd), window.removeEventListener("pointerdown", this.handlePointerEvent),

window.removeEventListener("pointermove", this.handlePointerEvent), window.removeEventListener("pointerup", this.handlePointerEvent),

window.removeEventListener("pointercancel", this.handlePointerEvent), (0, \_util.isInSameOriginIframe)() && (window.top.removeEventListener("mouseup", this.handleMouseUp),

window.top.removeEventListener("touchend", this.handleTouchEnd), window.top.removeEventListener("touchcancel", this.handleTouchEnd),

window.top.removeEventListener("pointerdown", this.handlePointerEvent), window.top.removeEventListener("pointermove", this.handlePointerEvent),

window.top.removeEventListener("pointerup", this.handlePointerEvent), window.top.removeEventListener("pointercancel", this.handlePointerEvent)),

this.listenersAttached = !1);

}

}, {

key: "getBestImageForType",

value: function(srcType) {

var imageSrc = this.props[srcType], fitSizes = {};

if (this.isImageLoaded(imageSrc)) // Use full-size image if available

fitSizes = this.getFitSizes(this.imageCache[imageSrc].width, this.imageCache[imageSrc].height); else {

if (!this.isImageLoaded(this.props[srcType + "Thumbnail"])) return null;

// Fall back to using thumbnail if the image has not been loaded

imageSrc = this.props[srcType + "Thumbnail"], fitSizes = this.getFitSizes(this.imageCache[imageSrc].width, this.imageCache[imageSrc].height, !0);

}

return {

src: imageSrc,

height: fitSizes.height,

width: fitSizes.width

};

}

}, {

key: "getFitSizes",

value: function(width, height, stretch) {

var boxSize = this.getLightboxRect(), maxHeight = boxSize.height - 2 \* this.props.imagePadding, maxWidth = boxSize.width - 2 \* this.props.imagePadding;

stretch || (maxHeight = Math.min(maxHeight, height), maxWidth = Math.min(maxWidth, width));

var maxRatio = maxWidth / maxHeight, srcRatio = width / height;

return maxRatio > srcRatio ? {

width: width \* maxHeight / height,

height: maxHeight

} : {

width: maxWidth,

height: height \* maxWidth / width

};

}

}, {

key: "getMaxOffsets",

value: function() {

var zoomLevel = arguments.length > 0 && void 0 !== arguments[0] ? arguments[0] : this.state.zoomLevel, currentImageInfo = this.getBestImageForType("mainSrc");

if (null === currentImageInfo) return {

maxX: 0,

minX: 0,

maxY: 0,

minY: 0

};

var boxSize = this.getLightboxRect(), zoomMultiplier = this.getZoomMultiplier(zoomLevel), maxX = 0;

// if there is still blank space in the X dimension, don't limit except to the opposite edge

maxX = zoomMultiplier \* currentImageInfo.width - boxSize.width < 0 ? (boxSize.width - zoomMultiplier \* currentImageInfo.width) / 2 : (zoomMultiplier \* currentImageInfo.width - boxSize.width) / 2;

var maxY = 0;

// if there is still blank space in the Y dimension, don't limit except to the opposite edge

return maxY = zoomMultiplier \* currentImageInfo.height - boxSize.height < 0 ? (boxSize.height - zoomMultiplier \* currentImageInfo.height) / 2 : (zoomMultiplier \* currentImageInfo.height - boxSize.height) / 2,

{

maxX: maxX,

maxY: maxY,

minX: -1 \* maxX,

minY: -1 \* maxY

};

}

}, {

key: "getSrcTypes",

value: function() {

return [ {

name: "mainSrc",

keyEnding: "i" + this.keyCounter

}, {

name: "mainSrcThumbnail",

keyEnding: "t" + this.keyCounter

}, {

name: "nextSrc",

keyEnding: "i" + (this.keyCounter + 1)

}, {

name: "nextSrcThumbnail",

keyEnding: "t" + (this.keyCounter + 1)

}, {

name: "prevSrc",

keyEnding: "i" + (this.keyCounter - 1)

}, {

name: "prevSrcThumbnail",

keyEnding: "t" + (this.keyCounter - 1)

} ];

}

}, {

key: "getZoomMultiplier",

value: function() {

var zoomLevel = arguments.length > 0 && void 0 !== arguments[0] ? arguments[0] : this.state.zoomLevel;

return Math.pow(\_constant.ZOOM\_RATIO, zoomLevel);

}

}, {

key: "getLightboxRect",

value: function() {

return this.outerEl ? this.outerEl.getBoundingClientRect() : {

width: (0, \_util.getWindowWidth)(),

height: (0, \_util.getWindowHeight)(),

top: 0,

right: 0,

bottom: 0,

left: 0

};

}

}, {

key: "handleKeyInput",

value: function(event) {

// Ignore key input during animations

if (event.stopPropagation(), !this.isAnimating()) {

// Allow slightly faster navigation through the images when user presses keys repeatedly

if ("keyup" === event.type) return void (this.lastKeyDownTime -= this.props.keyRepeatKeyupBonus);

var keyCode = event.which || event.keyCode, currentTime = new Date();

if (!(currentTime.getTime() - this.lastKeyDownTime < this.props.keyRepeatLimit && keyCode !== \_constant.KEYS.ESC)) switch (this.lastKeyDownTime = currentTime.getTime(),

keyCode) {

// ESC key closes the lightbox

case \_constant.KEYS.ESC:

event.preventDefault(), this.requestClose(event);

break;

// Left arrow key moves to previous image

case \_constant.KEYS.LEFT\_ARROW:

if (!this.props.prevSrc) return;

event.preventDefault(), this.keyPressed = !0, this.requestMovePrev(event);

break;

// Right arrow key moves to next image

case \_constant.KEYS.RIGHT\_ARROW:

if (!this.props.nextSrc) return;

event.preventDefault(), this.keyPressed = !0, this.requestMoveNext(event);

}

}

}

}, {

key: "handleOuterMousewheel",

value: function(event) {

var \_this5 = this;

// Prevent scrolling of the background

event.preventDefault(), event.stopPropagation();

var xThreshold = \_constant.WHEEL\_MOVE\_X\_THRESHOLD, actionDelay = 0, imageMoveDelay = 500;

// Prevent rapid-fire zoom behavior

if (this.clearTimeout(this.resetScrollTimeout), this.resetScrollTimeout = this.setTimeout(function() {

\_this5.scrollX = 0, \_this5.scrollY = 0;

}, 300), null === this.wheelActionTimeout && !this.isAnimating()) {

if (Math.abs(event.deltaY) < Math.abs(event.deltaX)) {

// handle horizontal scrolls with image moves

this.scrollY = 0, this.scrollX += event.deltaX;

var bigLeapX = xThreshold / 2;

// If the scroll amount has accumulated sufficiently, or a large leap was taken

this.scrollX >= xThreshold || event.deltaX >= bigLeapX ? (// Scroll right moves to next

this.requestMoveNext(event), actionDelay = imageMoveDelay, this.scrollX = 0) : (this.scrollX <= -1 \* xThreshold || event.deltaX <= -1 \* bigLeapX) && (// Scroll left moves to previous

this.requestMovePrev(event), actionDelay = imageMoveDelay, this.scrollX = 0);

}

// Allow successive actions after the set delay

0 !== actionDelay && (this.wheelActionTimeout = this.setTimeout(function() {

\_this5.wheelActionTimeout = null;

}, actionDelay));

}

}

}, {

key: "handleImageMouseWheel",

value: function(event) {

event.preventDefault();

var yThreshold = \_constant.WHEEL\_MOVE\_Y\_THRESHOLD;

if (Math.abs(event.deltaY) >= Math.abs(event.deltaX)) {

// If the vertical scroll amount was large enough, perform a zoom

if (event.stopPropagation(), Math.abs(event.deltaY) < yThreshold) return;

this.scrollX = 0, this.scrollY += event.deltaY, this.changeZoom(this.state.zoomLevel - event.deltaY, event.clientX, event.clientY);

}

}

}, {

key: "handleImageDoubleClick",

value: function(event) {

this.state.zoomLevel > \_constant.MIN\_ZOOM\_LEVEL ? // A double click when zoomed in zooms all the way out

this.changeZoom(\_constant.MIN\_ZOOM\_LEVEL, event.clientX, event.clientY) : // A double click when zoomed all the way out zooms in

this.changeZoom(this.state.zoomLevel + \_constant.ZOOM\_BUTTON\_INCREMENT\_SIZE, event.clientX, event.clientY);

}

}, {

key: "shouldHandleEvent",

value: function(source) {

if (this.eventsSource === source) return !0;

if (this.eventsSource === \_constant.SOURCE\_ANY) return this.eventsSource = source,

!0;

switch (source) {

case \_constant.SOURCE\_MOUSE:

return !1;

case \_constant.SOURCE\_TOUCH:

return this.eventsSource = \_constant.SOURCE\_TOUCH, this.filterPointersBySource(),

!0;

case \_constant.SOURCE\_POINTER:

return this.eventsSource === \_constant.SOURCE\_MOUSE && (this.eventsSource = \_constant.SOURCE\_POINTER,

this.filterPointersBySource(), !0);

default:

return !1;

}

}

}, {

key: "addPointer",

value: function(pointer) {

this.pointerList.push(pointer);

}

}, {

key: "removePointer",

value: function(pointer) {

this.pointerList = this.pointerList.filter(function(\_ref) {

var id = \_ref.id;

return id !== pointer.id;

});

}

}, {

key: "filterPointersBySource",

value: function() {

var \_this6 = this;

this.pointerList = this.pointerList.filter(function(\_ref2) {

var source = \_ref2.source;

return source === \_this6.eventsSource;

});

}

}, {

key: "handleMouseDown",

value: function(event) {

this.shouldHandleEvent(\_constant.SOURCE\_MOUSE) && ReactImageLightbox.isTargetMatchImage(event.target) && (this.addPointer(ReactImageLightbox.parseMouseEvent(event)),

this.multiPointerStart(event));

}

}, {

key: "handleMouseMove",

value: function(event) {

this.shouldHandleEvent(\_constant.SOURCE\_MOUSE) && this.multiPointerMove(event, [ ReactImageLightbox.parseMouseEvent(event) ]);

}

}, {

key: "handleMouseUp",

value: function(event) {

this.shouldHandleEvent(\_constant.SOURCE\_MOUSE) && (this.removePointer(ReactImageLightbox.parseMouseEvent(event)),

this.multiPointerEnd(event));

}

}, {

key: "handlePointerEvent",

value: function(event) {

if (this.shouldHandleEvent(\_constant.SOURCE\_POINTER)) switch (event.type) {

case "pointerdown":

ReactImageLightbox.isTargetMatchImage(event.target) && (this.addPointer(ReactImageLightbox.parsePointerEvent(event)),

this.multiPointerStart(event));

break;

case "pointermove":

this.multiPointerMove(event, [ ReactImageLightbox.parsePointerEvent(event) ]);

break;

case "pointerup":

case "pointercancel":

this.removePointer(ReactImageLightbox.parsePointerEvent(event)), this.multiPointerEnd(event);

}

}

}, {

key: "handleTouchStart",

value: function(event) {

var \_this7 = this;

this.shouldHandleEvent(\_constant.SOURCE\_TOUCH) && ReactImageLightbox.isTargetMatchImage(event.target) && ([].forEach.call(event.changedTouches, function(eventTouch) {

return \_this7.addPointer(ReactImageLightbox.parseTouchPointer(eventTouch));

}), this.multiPointerStart(event));

}

}, {

key: "handleTouchMove",

value: function(event) {

this.shouldHandleEvent(\_constant.SOURCE\_TOUCH) && this.multiPointerMove(event, [].map.call(event.changedTouches, function(eventTouch) {

return ReactImageLightbox.parseTouchPointer(eventTouch);

}));

}

}, {

key: "handleTouchEnd",

value: function(event) {

var \_this8 = this;

this.shouldHandleEvent(\_constant.SOURCE\_TOUCH) && ([].map.call(event.changedTouches, function(touch) {

return \_this8.removePointer(ReactImageLightbox.parseTouchPointer(touch));

}), this.multiPointerEnd(event));

}

}, {

key: "decideMoveOrSwipe",

value: function(pointer) {

this.state.zoomLevel <= \_constant.MIN\_ZOOM\_LEVEL ? this.handleSwipeStart(pointer) : this.handleMoveStart(pointer);

}

}, {

key: "multiPointerStart",

value: function(event) {

switch (this.handleEnd(null), this.pointerList.length) {

case 1:

event.preventDefault(), this.decideMoveOrSwipe(this.pointerList[0]);

break;

case 2:

event.preventDefault(), this.handlePinchStart(this.pointerList);

}

}

}, {

key: "multiPointerMove",

value: function(event, pointerList) {

switch (this.currentAction) {

case \_constant.ACTION\_MOVE:

event.preventDefault(), this.handleMove(pointerList[0]);

break;

case \_constant.ACTION\_SWIPE:

event.preventDefault(), this.handleSwipe(pointerList[0]);

break;

case \_constant.ACTION\_PINCH:

event.preventDefault(), this.handlePinch(pointerList);

}

}

}, {

key: "multiPointerEnd",

value: function(event) {

switch (this.currentAction !== \_constant.ACTION\_NONE && (this.setPreventInnerClose(),

this.handleEnd(event)), this.pointerList.length) {

case 0:

this.eventsSource = \_constant.SOURCE\_ANY;

break;

case 1:

event.preventDefault(), this.decideMoveOrSwipe(this.pointerList[0]);

break;

case 2:

event.preventDefault(), this.handlePinchStart(this.pointerList);

}

}

}, {

key: "handleEnd",

value: function(event) {

switch (this.currentAction) {

case \_constant.ACTION\_MOVE:

this.handleMoveEnd(event);

break;

case \_constant.ACTION\_SWIPE:

this.handleSwipeEnd(event);

break;

case \_constant.ACTION\_PINCH:

this.handlePinchEnd(event);

}

}

}, {

key: "handleMoveStart",

value: function(\_ref3) {

var clientX = \_ref3.x, clientY = \_ref3.y;

this.props.enableZoom && (this.currentAction = \_constant.ACTION\_MOVE, this.moveStartX = clientX,

this.moveStartY = clientY, this.moveStartOffsetX = this.state.offsetX, this.moveStartOffsetY = this.state.offsetY);

}

}, {

key: "handleMove",

value: function(\_ref4) {

var clientX = \_ref4.x, clientY = \_ref4.y, newOffsetX = this.moveStartX - clientX + this.moveStartOffsetX, newOffsetY = this.moveStartY - clientY + this.moveStartOffsetY;

this.state.offsetX === newOffsetX && this.state.offsetY === newOffsetY || this.setState({

offsetX: newOffsetX,

offsetY: newOffsetY

});

}

}, {

key: "handleMoveEnd",

value: function() {

var \_this9 = this;

this.currentAction = \_constant.ACTION\_NONE, this.moveStartX = 0, this.moveStartY = 0,

this.moveStartOffsetX = 0, this.moveStartOffsetY = 0;

// Snap image back into frame if outside max offset range

var maxOffsets = this.getMaxOffsets(), nextOffsetX = Math.max(maxOffsets.minX, Math.min(maxOffsets.maxX, this.state.offsetX)), nextOffsetY = Math.max(maxOffsets.minY, Math.min(maxOffsets.maxY, this.state.offsetY));

nextOffsetX === this.state.offsetX && nextOffsetY === this.state.offsetY || (this.setState({

offsetX: nextOffsetX,

offsetY: nextOffsetY,

shouldAnimate: !0

}), this.setTimeout(function() {

\_this9.setState({

shouldAnimate: !1

});

}, this.props.animationDuration));

}

}, {

key: "handleSwipeStart",

value: function(\_ref5) {

var clientX = \_ref5.x, clientY = \_ref5.y;

this.currentAction = \_constant.ACTION\_SWIPE, this.swipeStartX = clientX, this.swipeStartY = clientY,

this.swipeEndX = clientX, this.swipeEndY = clientY;

}

}, {

key: "handleSwipe",

value: function(\_ref6) {

var clientX = \_ref6.x, clientY = \_ref6.y;

this.swipeEndX = clientX, this.swipeEndY = clientY;

}

}, {

key: "handleSwipeEnd",

value: function(event) {

var xDiff = this.swipeEndX - this.swipeStartX, xDiffAbs = Math.abs(xDiff), yDiffAbs = Math.abs(this.swipeEndY - this.swipeStartY);

if (this.currentAction = \_constant.ACTION\_NONE, this.swipeStartX = 0, this.swipeStartY = 0,

this.swipeEndX = 0, this.swipeEndY = 0, !(!event || this.isAnimating() || xDiffAbs < 1.5 \* yDiffAbs)) {

if (xDiffAbs < \_constant.MIN\_SWIPE\_DISTANCE) {

var boxRect = this.getLightboxRect();

if (xDiffAbs < boxRect.width / 4) return;

}

xDiff > 0 && this.props.prevSrc ? (event.preventDefault(), this.requestMovePrev()) : xDiff < 0 && this.props.nextSrc && (event.preventDefault(),

this.requestMoveNext());

}

}

}, {

key: "calculatePinchDistance",

value: function() {

var \_ref7 = arguments.length > 0 && void 0 !== arguments[0] ? arguments[0] : this.pinchTouchList, \_ref8 = \_slicedToArray(\_ref7, 2), a = \_ref8[0], b = \_ref8[1];

return Math.sqrt(Math.pow(a.x - b.x, 2) + Math.pow(a.y - b.y, 2));

}

}, {

key: "calculatePinchCenter",

value: function() {

var \_ref9 = arguments.length > 0 && void 0 !== arguments[0] ? arguments[0] : this.pinchTouchList, \_ref10 = \_slicedToArray(\_ref9, 2), a = \_ref10[0], b = \_ref10[1];

return {

x: a.x - (a.x - b.x) / 2,

y: a.y - (a.y - b.y) / 2

};

}

}, {

key: "handlePinchStart",

value: function(pointerList) {

this.props.enableZoom && (this.currentAction = \_constant.ACTION\_PINCH, this.pinchTouchList = pointerList.map(function(\_ref11) {

var id = \_ref11.id, x = \_ref11.x, y = \_ref11.y;

return {

id: id,

x: x,

y: y

};

}), this.pinchDistance = this.calculatePinchDistance());

}

}, {

key: "handlePinch",

value: function(pointerList) {

this.pinchTouchList = this.pinchTouchList.map(function(oldPointer) {

for (var i = 0; i < pointerList.length; i++) if (pointerList[i].id === oldPointer.id) return pointerList[i];

return oldPointer;

});

var newDistance = this.calculatePinchDistance(), zoomLevel = this.state.zoomLevel + newDistance - this.pinchDistance;

this.pinchDistance = newDistance;

var \_calculatePinchCenter = this.calculatePinchCenter(this.pinchTouchList), clientX = \_calculatePinchCenter.x, clientY = \_calculatePinchCenter.y;

this.changeZoom(zoomLevel, clientX, clientY);

}

}, {

key: "handlePinchEnd",

value: function() {

this.currentAction = \_constant.ACTION\_NONE, this.pinchTouchList = null, this.pinchDistance = 0;

}

}, {

key: "handleWindowResize",

value: function() {

this.clearTimeout(this.resizeTimeout), this.resizeTimeout = this.setTimeout(this.forceUpdate.bind(this), 100);

}

}, {

key: "handleZoomInButtonClick",

value: function() {

this.changeZoom(this.state.zoomLevel + \_constant.ZOOM\_BUTTON\_INCREMENT\_SIZE);

}

}, {

key: "handleZoomOutButtonClick",

value: function() {

this.changeZoom(this.state.zoomLevel - \_constant.ZOOM\_BUTTON\_INCREMENT\_SIZE);

}

}, {

key: "handleCaptionMousewheel",

value: function(event) {

if (event.stopPropagation(), this.caption) {

var height = this.caption.getBoundingClientRect().height, scrollHeight = this.caption.scrollHeight, scrollTop = this.caption.scrollTop;

(event.deltaY > 0 && height + scrollTop >= scrollHeight || event.deltaY < 0 && scrollTop <= 0) && event.preventDefault();

}

}

}, {

key: "isAnimating",

value: function() {

return this.state.shouldAnimate || this.state.isClosing;

}

}, {

key: "isImageLoaded",

value: function(imageSrc) {

return imageSrc && imageSrc in this.imageCache && this.imageCache[imageSrc].loaded;

}

}, {

key: "loadImage",

value: function(srcType, imageSrc, done) {

var \_this10 = this;

// Return the image info if it is already cached

if (this.isImageLoaded(imageSrc)) return void this.setTimeout(function() {

done();

}, 1);

var that = this, inMemoryImage = new Image();

inMemoryImage.onerror = function(errorEvent) {

\_this10.props.onImageLoadError(imageSrc, srcType, errorEvent), done(errorEvent);

}, inMemoryImage.onload = function() {

that.imageCache[imageSrc] = {

loaded: !0,

width: this.width,

height: this.height

}, done();

}, inMemoryImage.src = imageSrc;

}

}, {

key: "loadAllImages",

value: function() {

var \_this11 = this, props = arguments.length > 0 && void 0 !== arguments[0] ? arguments[0] : this.props, generateLoadDoneCallback = function(srcType, imageSrc) {

return function(err) {

// Give up showing image on error

err || // Don't rerender if the src is not the same as when the load started

// or if the component has unmounted

\_this11.props[srcType] === imageSrc && \_this11.mounted && // Force rerender with the new image

\_this11.forceUpdate();

};

};

// Load the images

this.getSrcTypes().forEach(function(srcType) {

var type = srcType.name;

// Load unloaded images

props[type] && !\_this11.isImageLoaded(props[type]) && \_this11.loadImage(type, props[type], generateLoadDoneCallback(type, props[type]));

});

}

}, {

key: "requestClose",

value: function(event) {

var \_this12 = this, closeLightbox = function() {

return \_this12.props.onCloseRequest(event);

};

// With animation

// Start closing animation

// Perform the actual closing at the end of the animation

return this.props.animationDisabled || "keydown" === event.type && !this.props.animationOnKeyInput ? closeLightbox() : (this.setState({

isClosing: !0

}), void this.setTimeout(closeLightbox, this.props.animationDuration));

}

}, {

key: "requestMove",

value: function(direction, event) {

var \_this13 = this, nextState = {

zoomLevel: \_constant.MIN\_ZOOM\_LEVEL,

offsetX: 0,

offsetY: 0

};

// Enable animated states

this.props.animationDisabled || this.keyPressed && !this.props.animationOnKeyInput || (nextState.shouldAnimate = !0,

this.setTimeout(function() {

return \_this13.setState({

shouldAnimate: !1

});

}, this.props.animationDuration)), this.keyPressed = !1, this.moveRequested = !0,

"prev" === direction ? (this.keyCounter--, this.setState(nextState), this.props.onMovePrevRequest(event)) : (this.keyCounter++,

this.setState(nextState), this.props.onMoveNextRequest(event));

}

}, {

key: "requestMoveNext",

value: function(event) {

this.requestMove("next", event);

}

}, {

key: "requestMovePrev",

value: function(event) {

this.requestMove("prev", event);

}

}, {

key: "render",

value: function() {

var \_this14 = this, \_props = this.props, animationDisabled = \_props.animationDisabled, animationDuration = \_props.animationDuration, clickOutsideToClose = \_props.clickOutsideToClose, discourageDownloads = \_props.discourageDownloads, enableZoom = \_props.enableZoom, imageTitle = \_props.imageTitle, nextSrc = \_props.nextSrc, prevSrc = \_props.prevSrc, toolbarButtons = \_props.toolbarButtons, reactModalStyle = \_props.reactModalStyle, \_onAfterOpen = \_props.onAfterOpen, \_state = this.state, zoomLevel = \_state.zoomLevel, offsetX = \_state.offsetX, offsetY = \_state.offsetY, isClosing = \_state.isClosing, boxSize = this.getLightboxRect(), transitionStyle = {};

// Transition settings for sliding animations

!animationDisabled && this.isAnimating() && (transitionStyle = \_extends({}, transitionStyle, {

transition: "transform " + animationDuration + "ms"

}));

// Key endings to differentiate between images with the same src

var keyEndings = {};

this.getSrcTypes().forEach(function(\_ref12) {

var name = \_ref12.name, keyEnding = \_ref12.keyEnding;

keyEndings[name] = keyEnding;

});

// Images to be displayed

var images = [], addImage = function(srcType, imageClass) {

var baseStyle = arguments.length > 2 && void 0 !== arguments[2] ? arguments[2] : {};

// Ignore types that have no source defined for their full size image

if (\_this14.props[srcType]) {

var imageStyle = \_extends({}, baseStyle, transitionStyle);

zoomLevel > \_constant.MIN\_ZOOM\_LEVEL && (imageStyle.cursor = "move");

var bestImageInfo = \_this14.getBestImageForType(srcType);

if (null === bestImageInfo) {

var loadingIcon = void 0;

// Fall back to loading icon if the thumbnail has not been loaded

return loadingIcon = \_ieVersion < 10 ? \_react2.default.createElement("div", {

className: styles.loadingContainer\_\_icon

}, (0, \_util.translate)("Loading...")) : \_react2.default.createElement("div", {

className: "ril-loading-circle " + styles.loadingCircle + " " + styles.loadingContainer\_\_icon

}, \_react2.default.createElement("div", {

className: "ril-loading-circle-point " + styles.loadingCirclePoint

}), \_react2.default.createElement("div", {

className: "ril-loading-circle-point " + styles.loadingCirclePoint

}), \_react2.default.createElement("div", {

className: "ril-loading-circle-point " + styles.loadingCirclePoint

}), \_react2.default.createElement("div", {

className: "ril-loading-circle-point " + styles.loadingCirclePoint

}), \_react2.default.createElement("div", {

className: "ril-loading-circle-point " + styles.loadingCirclePoint

}), \_react2.default.createElement("div", {

className: "ril-loading-circle-point " + styles.loadingCirclePoint

}), \_react2.default.createElement("div", {

className: "ril-loading-circle-point " + styles.loadingCirclePoint

}), \_react2.default.createElement("div", {

className: "ril-loading-circle-point " + styles.loadingCirclePoint

}), \_react2.default.createElement("div", {

className: "ril-loading-circle-point " + styles.loadingCirclePoint

}), \_react2.default.createElement("div", {

className: "ril-loading-circle-point " + styles.loadingCirclePoint

}), \_react2.default.createElement("div", {

className: "ril-loading-circle-point " + styles.loadingCirclePoint

}), \_react2.default.createElement("div", {

className: "ril-loading-circle-point " + styles.loadingCirclePoint

})), void images.push(\_react2.default.createElement("div", {

className: imageClass + " " + styles.image + " ril-not-loaded",

style: imageStyle,

key: \_this14.props[srcType] + keyEndings[srcType]

}, \_react2.default.createElement("div", {

className: styles.loadingContainer

}, loadingIcon)));

}

imageStyle.width = isNaN(bestImageInfo.width) ? null : bestImageInfo.width, imageStyle.height = isNaN(bestImageInfo.height) ? null : bestImageInfo.height;

var imageSrc = bestImageInfo.src;

discourageDownloads ? (imageStyle.backgroundImage = "url('" + imageSrc + "')", images.push(\_react2.default.createElement("div", {

className: imageClass + " " + styles.image + " " + styles.imageDiscourager,

onDoubleClick: \_this14.handleImageDoubleClick,

onWheel: \_this14.handleImageMouseWheel,

style: imageStyle,

key: imageSrc + keyEndings[srcType]

}, \_react2.default.createElement("div", {

className: "ril-download-blocker " + styles.downloadBlocker

})))) : images.push(\_react2.default.createElement("img", {

className: imageClass + " " + styles.image,

onDoubleClick: \_this14.handleImageDoubleClick,

onWheel: \_this14.handleImageMouseWheel,

onDragStart: function(e) {

return e.preventDefault();

},

style: imageStyle,

src: imageSrc,

key: imageSrc + keyEndings[srcType],

alt: "string" == typeof imageTitle ? imageTitle : (0, \_util.translate)("Image"),

draggable: !1

}));

}

}, zoomMultiplier = this.getZoomMultiplier();

// Next Image (displayed on the right)

addImage("nextSrc", "ril-image-next " + styles.imageNext, ReactImageLightbox.getTransform({

x: boxSize.width

})), // Main Image

addImage("mainSrc", "ril-image-current", ReactImageLightbox.getTransform({

x: -1 \* offsetX,

y: -1 \* offsetY,

zoom: zoomMultiplier

})), // Previous Image (displayed on the left)

addImage("prevSrc", "ril-image-prev " + styles.imagePrev, ReactImageLightbox.getTransform({

x: -1 \* boxSize.width

}));

var noop = function() {}, zoomInButtonClasses = [ styles.toolbarItemChild, styles.builtinButton, styles.zoomInButton ], zoomOutButtonClasses = [ styles.toolbarItemChild, styles.builtinButton, styles.zoomOutButton ], zoomInButtonHandler = this.handleZoomInButtonClick, zoomOutButtonHandler = this.handleZoomOutButtonClick;

// Disable zooming in when zoomed all the way in

zoomLevel === \_constant.MAX\_ZOOM\_LEVEL && (zoomInButtonClasses.push(styles.builtinButtonDisabled),

zoomInButtonHandler = noop), // Disable zooming out when zoomed all the way out

zoomLevel === \_constant.MIN\_ZOOM\_LEVEL && (zoomOutButtonClasses.push(styles.builtinButtonDisabled),

zoomOutButtonHandler = noop), // Ignore clicks during animation

this.isAnimating() && (zoomInButtonHandler = noop, zoomOutButtonHandler = noop);

var modalStyle = {

overlay: \_extends({

zIndex: 1e3,

backgroundColor: "transparent"

}, reactModalStyle.overlay),

content: \_extends({

backgroundColor: "transparent",

overflow: "hidden",

// Needed, otherwise keyboard shortcuts scroll the page

border: "none",

borderRadius: 0,

padding: 0,

top: 0,

left: 0,

right: 0,

bottom: 0

}, reactModalStyle.content)

};

return \_react2.default.createElement(\_reactModal2.default, {

isOpen: !0,

onRequestClose: clickOutsideToClose ? this.requestClose : noop,

onAfterOpen: function() {

// Focus on the div with key handlers

\_this14.outerEl && \_this14.outerEl.focus(), \_onAfterOpen();

},

style: modalStyle,

contentLabel: (0, \_util.translate)("Lightbox")

}, \_react2.default.createElement("div", {

// eslint-disable-line jsx-a11y/no-static-element-interactions

// Floating modal with closing animations

className: "ril-outer " + styles.outer + " " + styles.outerAnimating + " " + this.props.wrapperClassName + " " + (isClosing ? " ril-closing " + styles.outerClosing : ""),

style: {

transition: "opacity " + animationDuration + "ms",

animationDuration: animationDuration + "ms",

animationDirection: isClosing ? "normal" : "reverse"

},

ref: function(el) {

\_this14.outerEl = el;

},

onWheel: this.handleOuterMousewheel,

onMouseMove: this.handleMouseMove,

onMouseDown: this.handleMouseDown,

onTouchStart: this.handleTouchStart,

onTouchMove: this.handleTouchMove,

tabIndex: "-1",

onKeyDown: this.handleKeyInput,

onKeyUp: this.handleKeyInput

}, \_react2.default.createElement("div", {

// eslint-disable-line jsx-a11y/no-static-element-interactions

// Image holder

className: "ril-inner " + styles.inner,

onClick: clickOutsideToClose ? this.closeIfClickInner : noop

}, images), prevSrc && \_react2.default.createElement("button", {

// Move to previous image button

type: "button",

className: "ril-prev-button " + styles.navButtons + " " + styles.navButtonPrev,

key: "prev",

"aria-label": this.props.prevLabel,

onClick: this.isAnimating() ? noop : this.requestMovePrev

}), nextSrc && \_react2.default.createElement("button", {

// Move to next image button

type: "button",

className: "ril-next-button " + styles.navButtons + " " + styles.navButtonNext,

key: "next",

"aria-label": this.props.nextLabel,

onClick: this.isAnimating() ? noop : this.requestMoveNext

}), \_react2.default.createElement("div", {

// Lightbox toolbar

className: "ril-toolbar " + styles.toolbar

}, \_react2.default.createElement("ul", {

className: "ril-toolbar-left " + styles.toolbarSide + " " + styles.toolbarLeftSide

}, \_react2.default.createElement("li", {

className: "ril-toolbar\_\_item " + styles.toolbarItem

}, \_react2.default.createElement("span", {

className: "ril-toolbar\_\_item\_\_child " + styles.toolbarItemChild

}, imageTitle))), \_react2.default.createElement("ul", {

className: [ "ril-toolbar-right", styles.toolbarSide, styles.toolbarRightSide ].join(" ")

}, toolbarButtons ? toolbarButtons.map(function(button, i) {

return \_react2.default.createElement("li", {

key: i,

className: "ril-toolbar\_\_item " + styles.toolbarItem

}, button);

}) : "", enableZoom && \_react2.default.createElement("li", {

className: "ril-toolbar\_\_item " + styles.toolbarItem

}, \_react2.default.createElement("button", {

// Lightbox zoom in button

type: "button",

key: "zoom-in",

"aria-label": this.props.zoomInLabel,

className: "ril-zoom-in " + zoomInButtonClasses.join(" "),

onClick: zoomInButtonHandler

})), enableZoom && \_react2.default.createElement("li", {

className: "ril-toolbar\_\_item " + styles.toolbarItem

}, \_react2.default.createElement("button", {

// Lightbox zoom out button

type: "button",

key: "zoom-out",

"aria-label": this.props.zoomOutLabel,

className: "ril-zoom-out " + zoomOutButtonClasses.join(" "),

onClick: zoomOutButtonHandler

})), \_react2.default.createElement("li", {

className: "ril-toolbar\_\_item " + styles.toolbarItem

}, \_react2.default.createElement("button", {

// Lightbox close button

type: "button",

key: "close",

"aria-label": this.props.closeLabel,

className: "ril-close ril-toolbar\_\_item\_\_child" + (" " + styles.toolbarItemChild + " " + styles.builtinButton + " " + styles.closeButton),

onClick: this.isAnimating() ? noop : this.requestClose

})))), this.props.imageCaption && \_react2.default.createElement("div", {

// Image caption

onWheel: this.handleCaptionMousewheel,

onMouseDown: function(event) {

return event.stopPropagation();

},

className: "ril-caption " + styles.caption,

ref: function(el) {

\_this14.caption = el;

}

}, \_react2.default.createElement("div", {

className: "ril-caption-content " + styles.captionContent

}, this.props.imageCaption))));

}

} ], [ {

key: "isTargetMatchImage",

value: function(target) {

return target && /ril-image-current/.test(target.className);

}

}, {

key: "parseMouseEvent",

value: function(mouseEvent) {

return {

id: "mouse",

source: \_constant.SOURCE\_MOUSE,

x: parseInt(mouseEvent.clientX, 10),

y: parseInt(mouseEvent.clientY, 10)

};

}

}, {

key: "parseTouchPointer",

value: function(touchPointer) {

return {

id: touchPointer.identifier,

source: \_constant.SOURCE\_TOUCH,

x: parseInt(touchPointer.clientX, 10),

y: parseInt(touchPointer.clientY, 10)

};

}

}, {

key: "parsePointerEvent",

value: function(pointerEvent) {

return {

id: pointerEvent.pointerId,

source: \_constant.SOURCE\_POINTER,

x: parseInt(pointerEvent.clientX, 10),

y: parseInt(pointerEvent.clientY, 10)

};

}

}, {

key: "getTransform",

value: function(\_ref13) {

var \_ref13$x = \_ref13.x, x = void 0 === \_ref13$x ? null : \_ref13$x, \_ref13$y = \_ref13.y, y = void 0 === \_ref13$y ? null : \_ref13$y, \_ref13$zoom = \_ref13.zoom, zoom = void 0 === \_ref13$zoom ? null : \_ref13$zoom, isOldIE = \_ieVersion < 10, transforms = [];

return null === x && null === y || transforms.push(isOldIE ? "translate(" + (x || 0) + "px," + (y || 0) + "px)" : "translate3d(" + (x || 0) + "px," + (y || 0) + "px,0)"),

null !== zoom && transforms.push(isOldIE ? "scale(" + zoom + ")" : "scale3d(" + zoom + "," + zoom + ",1)"),

\_defineProperty({}, isOldIE ? "msTransform" : "transform", 0 === transforms.length ? "none" : transforms.join(" "));

}

}, {

key: "loadStyles",

value: function() {

// Insert component styles

"object" === ("undefined" == typeof window ? "undefined" : \_typeof(window)) && styles.\_insertCss();

}

} ]), ReactImageLightbox;

}(\_react.Component);

ReactImageLightbox.propTypes = {

//-----------------------------

// Image sources

//-----------------------------

// Main display image url

mainSrc: \_propTypes2.default.string.isRequired,

// eslint-disable-line react/no-unused-prop-types

// Previous display image url (displayed to the left)

// If left undefined, movePrev actions will not be performed, and the button not displayed

prevSrc: \_propTypes2.default.string,

// Next display image url (displayed to the right)

// If left undefined, moveNext actions will not be performed, and the button not displayed

nextSrc: \_propTypes2.default.string,

//-----------------------------

// Image thumbnail sources

//-----------------------------

// Thumbnail image url corresponding to props.mainSrc

mainSrcThumbnail: \_propTypes2.default.string,

// eslint-disable-line react/no-unused-prop-types

// Thumbnail image url corresponding to props.prevSrc

prevSrcThumbnail: \_propTypes2.default.string,

// eslint-disable-line react/no-unused-prop-types

// Thumbnail image url corresponding to props.nextSrc

nextSrcThumbnail: \_propTypes2.default.string,

// eslint-disable-line react/no-unused-prop-types

//-----------------------------

// Event Handlers

//-----------------------------

// Close window event

// Should change the parent state such that the lightbox is not rendered

onCloseRequest: \_propTypes2.default.func.isRequired,

// Move to previous image event

// Should change the parent state such that props.prevSrc becomes props.mainSrc,

// props.mainSrc becomes props.nextSrc, etc.

onMovePrevRequest: \_propTypes2.default.func,

// Move to next image event

// Should change the parent state such that props.nextSrc becomes props.mainSrc,

// props.mainSrc becomes props.prevSrc, etc.

onMoveNextRequest: \_propTypes2.default.func,

// Called when an image fails to load

// (imageSrc: string, srcType: string, errorEvent: object): void

onImageLoadError: \_propTypes2.default.func,

// Open window event

onAfterOpen: \_propTypes2.default.func,

//-----------------------------

// Download discouragement settings

//-----------------------------

// Enable download discouragement (prevents [right-click -> Save Image As...])

discourageDownloads: \_propTypes2.default.bool,

//-----------------------------

// Animation settings

//-----------------------------

// Disable all animation

animationDisabled: \_propTypes2.default.bool,

// Disable animation on actions performed with keyboard shortcuts

animationOnKeyInput: \_propTypes2.default.bool,

// Animation duration (ms)

animationDuration: \_propTypes2.default.number,

//-----------------------------

// Keyboard shortcut settings

//-----------------------------

// Required interval of time (ms) between key actions

// (prevents excessively fast navigation of images)

keyRepeatLimit: \_propTypes2.default.number,

// Amount of time (ms) restored after each keyup

// (makes rapid key presses slightly faster than holding down the key to navigate images)

keyRepeatKeyupBonus: \_propTypes2.default.number,

//-----------------------------

// Image info

//-----------------------------

// Image title

imageTitle: \_propTypes2.default.node,

// Image caption

imageCaption: \_propTypes2.default.node,

//-----------------------------

// Lightbox style

//-----------------------------

// Set z-index style, etc., for the parent react-modal (format: https://github.com/reactjs/react-modal#styles )

reactModalStyle: \_propTypes2.default.object,

// Padding (px) between the edge of the window and the lightbox

imagePadding: \_propTypes2.default.number,

wrapperClassName: \_propTypes2.default.string,

//-----------------------------

// Other

//-----------------------------

// Array of custom toolbar buttons

toolbarButtons: \_propTypes2.default.arrayOf(\_propTypes2.default.node),

// When true, clicks outside of the image close the lightbox

clickOutsideToClose: \_propTypes2.default.bool,

// Set to false to disable zoom functionality and hide zoom buttons

enableZoom: \_propTypes2.default.bool,

// Aria-labels

nextLabel: \_propTypes2.default.string,

prevLabel: \_propTypes2.default.string,

zoomInLabel: \_propTypes2.default.string,

zoomOutLabel: \_propTypes2.default.string,

closeLabel: \_propTypes2.default.string

}, ReactImageLightbox.defaultProps = {

onMovePrevRequest: function() {},

onMoveNextRequest: function() {},

onImageLoadError: function() {},

onAfterOpen: function() {},

discourageDownloads: !1,

animationDisabled: !1,

animationOnKeyInput: !1,

animationDuration: 300,

keyRepeatLimit: 180,

keyRepeatKeyupBonus: 40,

reactModalStyle: {},

imagePadding: 10,

clickOutsideToClose: !0,

enableZoom: !0,

wrapperClassName: "",

nextLabel: "Next image",

prevLabel: "Previous image",

zoomInLabel: "Zoom in",

zoomOutLabel: "Zoom out",

closeLabel: "Close lightbox"

}, exports.default = ReactImageLightbox;

}, /\* 3 \*/

/\*\*\*/

function(module, exports) {

"use strict";

/\*\*

\* Get the version of Internet Explorer in use, or undefined

\*

\* @return {?number} ieVersion - IE version as an integer, or undefined if not IE

\*/

function getIEVersion() {

if ("undefined" != typeof window) {

var match = window.navigator.userAgent.match(/(?:MSIE |Trident\/.\*; rv:)(\d+)/);

return match ? parseInt(match[1], 10) : void 0;

}

}

/\*\*

\* Placeholder for future translate functionality

\*/

function translate(str) {

var replaceStrings = arguments.length > 1 && void 0 !== arguments[1] ? arguments[1] : null;

if (!str) return "";

var translated = str;

return replaceStrings && Object.keys(replaceStrings).forEach(function(placeholder) {

translated = translated.replace(placeholder, replaceStrings[placeholder]);

}), translated;

}

function getWindowWidth() {

return "undefined" == typeof window ? 0 : window.innerWidth || document.documentElement.clientWidth || document.body.clientWidth;

}

function getWindowHeight() {

return "undefined" == typeof window ? 0 : window.innerHeight || document.documentElement.clientHeight || document.body.clientHeight;

}

// Returns true if this window is rendered as an iframe inside another window

// with the same origin.

function isInSameOriginIframe() {

try {

return window.self !== window.top && window.top.document;

} catch (e) {

return !1;

}

}

Object.defineProperty(exports, "\_\_esModule", {

value: !0

}), exports.getIEVersion = getIEVersion, exports.translate = translate, exports.getWindowWidth = getWindowWidth,

exports.getWindowHeight = getWindowHeight, exports.isInSameOriginIframe = isInSameOriginIframe;

}, /\* 4 \*/

/\*\*\*/

function(module, exports, \_\_webpack\_require\_\_) {

exports = module.exports = \_\_webpack\_require\_\_(5)(), // imports

// module

exports.push([ module.id, '@-webkit-keyframes closeWindow\_\_\_2Hlon{0%{opacity:1}to{opacity:0}}@keyframes closeWindow\_\_\_2Hlon{0%{opacity:1}to{opacity:0}}.outer\_\_\_2lDXy{background-color:rgba(0,0,0,.85);top:0;left:0;right:0;bottom:0;z-index:1000;width:100%;height:100%;-ms-content-zooming:none;-ms-user-select:none;-ms-touch-select:none;-ms-touch-action:none;touch-action:none}.outerClosing\_\_\_1EQGK{opacity:0}.image\_\_\_2FLq2,.inner\_\_\_1rfRQ{position:absolute;top:0;left:0;right:0;bottom:0}.image\_\_\_2FLq2{margin:auto;max-width:100%;max-height:100%;-ms-content-zooming:none;-ms-user-select:none;-ms-touch-select:none;-ms-touch-action:none;touch-action:none}.imageNext\_\_\_1uRqJ,.imagePrev\_\_\_F6xVQ{@extends .image}.imageDiscourager\_\_\_3-CUB{background-repeat:no-repeat;background-position:50%;background-size:contain}.navButtons\_\_\_3kNVF{border:none;position:absolute;top:0;bottom:0;width:20px;height:34px;padding:40px 30px;margin:auto;cursor:pointer;opacity:.7}.navButtons\_\_\_3kNVF:hover{opacity:1}.navButtons\_\_\_3kNVF:active{opacity:.7}.navButtonPrev\_\_\_2vBS8{left:0;background:rgba(0,0,0,.2) url("") no-repeat 50%}.navButtonNext\_\_\_30R2i{right:0;background:rgba(0,0,0,.2) url("") no-repeat 50%}.downloadBlocker\_\_\_3rU9-{position:absolute;top:0;left:0;right:0;bottom:0;background-image:url("");background-size:cover}.caption\_\_\_3vDh\_,.toolbar\_\_\_1xYly{background-color:rgba(0,0,0,.5);position:absolute;left:0;right:0;display:-webkit-box;display:-ms-flexbox;display:flex;-webkit-box-pack:justify;-ms-flex-pack:justify;justify-content:space-between}.caption\_\_\_3vDh\_{bottom:0;max-height:150px;overflow:auto}.captionContent\_\_\_30kw2{padding:10px 20px;color:#fff}.toolbar\_\_\_1xYly{top:0;height:50px}.toolbarSide\_\_\_3FYWk{height:50px;margin:0}.toolbarSideNoFlex\_\_\_KxqgW{height:auto;line-height:50px;max-width:48%;position:absolute;top:0;bottom:0}.toolbarLeftSide\_\_\_8beAg{padding-left:20px;padding-right:0;-webkit-box-flex:0;-ms-flex:0 1 auto;flex:0 1 auto;overflow:hidden;text-overflow:ellipsis}.toolbarLeftSideNoFlex\_\_\_3O3cZ{left:0;overflow:visible}.toolbarRightSide\_\_\_1Sdfc{padding-left:0;padding-right:20px;-webkit-box-flex:0;-ms-flex:0 0 auto;flex:0 0 auto}.toolbarRightSideNoFlex\_\_\_oa0FT{right:0}.toolbarItem\_\_\_3WbMb{display:inline-block;line-height:50px;padding:0;color:#fff;font-size:120%;max-width:100%;overflow:hidden;text-overflow:ellipsis;white-space:nowrap}.toolbarItemChild\_\_\_2U\_MP{vertical-align:middle}.builtinButton\_\_\_1zqo6{width:40px;height:35px;cursor:pointer;border:none;opacity:.7}.builtinButton\_\_\_1zqo6:hover{opacity:1}.builtinButton\_\_\_1zqo6:active{outline:none}.builtinButtonDisabled\_\_\_3uvqe{cursor:default;opacity:.5}.builtinButtonDisabled\_\_\_3uvqe:hover{opacity:.5}.closeButton\_\_\_3BdAF{background:url("") no-repeat 50%}.zoomInButton\_\_\_3xtuX{background:url("") no-repeat 50%}.zoomOutButton\_\_\_38PZx{background:url("") no-repeat 50%}.outerAnimating\_\_\_2-fZi{-webkit-animation-name:closeWindow\_\_\_2Hlon;animation-name:closeWindow\_\_\_2Hlon}@-webkit-keyframes pointFade\_\_\_2RA5J{0%,19.999%,to{opacity:0}20%{opacity:1}}@keyframes pointFade\_\_\_2RA5J{0%,19.999%,to{opacity:0}20%{opacity:1}}.loadingCircle\_\_\_3JNJg{width:60px;height:60px;position:relative}.loadingCirclePoint\_\_\_3md-S{width:100%;height:100%;position:absolute;left:0;top:0}.loadingCirclePoint\_\_\_3md-S:before{content:"";display:block;margin:0 auto;width:15%;height:15%;background-color:#fff;border-radius:30%;-webkit-animation:pointFade\_\_\_2RA5J 1.2s infinite ease-in-out both;animation:pointFade\_\_\_2RA5J 1.2s infinite ease-in-out both}.loadingCirclePoint\_\_\_3md-S:first-of-type{-webkit-transform:rotate(0deg);-ms-transform:rotate(0deg);transform:rotate(0deg)}.loadingCirclePoint\_\_\_3md-S:first-of-type:before,.loadingCirclePoint\_\_\_3md-S:nth-of-type(7):before{-webkit-animation-delay:-1.2s;animation-delay:-1.2s}.loadingCirclePoint\_\_\_3md-S:nth-of-type(2){-webkit-transform:rotate(30deg);-ms-transform:rotate(30deg);transform:rotate(30deg)}.loadingCirclePoint\_\_\_3md-S:nth-of-type(8){-webkit-transform:rotate(210deg);-ms-transform:rotate(210deg);transform:rotate(210deg)}.loadingCirclePoint\_\_\_3md-S:nth-of-type(2):before,.loadingCirclePoint\_\_\_3md-S:nth-of-type(8):before{-webkit-animation-delay:-1s;animation-delay:-1s}.loadingCirclePoint\_\_\_3md-S:nth-of-type(3){-webkit-transform:rotate(60deg);-ms-transform:rotate(60deg);transform:rotate(60deg)}.loadingCirclePoint\_\_\_3md-S:nth-of-type(9){-webkit-transform:rotate(240deg);-ms-transform:rotate(240deg);transform:rotate(240deg)}.loadingCirclePoint\_\_\_3md-S:nth-of-type(3):before,.loadingCirclePoint\_\_\_3md-S:nth-of-type(9):before{-webkit-animation-delay:-.8s;animation-delay:-.8s}.loadingCirclePoint\_\_\_3md-S:nth-of-type(4){-webkit-transform:rotate(90deg);-ms-transform:rotate(90deg);transform:rotate(90deg)}.loadingCirclePoint\_\_\_3md-S:nth-of-type(10){-webkit-transform:rotate(270deg);-ms-transform:rotate(270deg);transform:rotate(270deg)}.loadingCirclePoint\_\_\_3md-S:nth-of-type(4):before,.loadingCirclePoint\_\_\_3md-S:nth-of-type(10):before{-webkit-animation-delay:-.6s;animation-delay:-.6s}.loadingCirclePoint\_\_\_3md-S:nth-of-type(5){-webkit-transform:rotate(120deg);-ms-transform:rotate(120deg);transform:rotate(120deg)}.loadingCirclePoint\_\_\_3md-S:nth-of-type(11){-webkit-transform:rotate(300deg);-ms-transform:rotate(300deg);transform:rotate(300deg)}.loadingCirclePoint\_\_\_3md-S:nth-of-type(5):before,.loadingCirclePoint\_\_\_3md-S:nth-of-type(11):before{-webkit-animation-delay:-.4s;animation-delay:-.4s}.loadingCirclePoint\_\_\_3md-S:nth-of-type(6){-webkit-transform:rotate(150deg);-ms-transform:rotate(150deg);transform:rotate(150deg)}.loadingCirclePoint\_\_\_3md-S:nth-of-type(12){-webkit-transform:rotate(330deg);-ms-transform:rotate(330deg);transform:rotate(330deg)}.loadingCirclePoint\_\_\_3md-S:nth-of-type(6):before,.loadingCirclePoint\_\_\_3md-S:nth-of-type(12):before{-webkit-animation-delay:-.2s;animation-delay:-.2s}.loadingCirclePoint\_\_\_3md-S:nth-of-type(7){-webkit-transform:rotate(180deg);-ms-transform:rotate(180deg);transform:rotate(180deg)}.loadingCirclePoint\_\_\_3md-S:nth-of-type(13){-webkit-transform:rotate(1turn);-ms-transform:rotate(1turn);transform:rotate(1turn)}.loadingCirclePoint\_\_\_3md-S:nth-of-type(7):before,.loadingCirclePoint\_\_\_3md-S:nth-of-type(13):before{-webkit-animation-delay:0ms;animation-delay:0ms}.loadingContainer\_\_\_2vaJ-{position:absolute;top:0;right:0;bottom:0;left:0}.loadingContainer\_\_icon\_\_\_1wQQz{color:#fff;position:absolute;top:50%;left:50%;-webkit-transform:translateX(-50%) translateY(-50%);-ms-transform:translateX(-50%) translateY(-50%);transform:translateX(-50%) translateY(-50%)}', "" ]),

// exports

exports.locals = {

outer: "outer\_\_\_2lDXy",

outerClosing: "outerClosing\_\_\_1EQGK",

inner: "inner\_\_\_1rfRQ",

image: "image\_\_\_2FLq2",

imagePrev: "imagePrev\_\_\_F6xVQ",

imageNext: "imageNext\_\_\_1uRqJ",

imageDiscourager: "imageDiscourager\_\_\_3-CUB",

navButtons: "navButtons\_\_\_3kNVF",

navButtonPrev: "navButtonPrev\_\_\_2vBS8",

navButtonNext: "navButtonNext\_\_\_30R2i",

downloadBlocker: "downloadBlocker\_\_\_3rU9-",

caption: "caption\_\_\_3vDh\_",

toolbar: "toolbar\_\_\_1xYly",

captionContent: "captionContent\_\_\_30kw2",

toolbarSide: "toolbarSide\_\_\_3FYWk",

toolbarSideNoFlex: "toolbarSideNoFlex\_\_\_KxqgW",

toolbarLeftSide: "toolbarLeftSide\_\_\_8beAg",

toolbarLeftSideNoFlex: "toolbarLeftSideNoFlex\_\_\_3O3cZ",

toolbarRightSide: "toolbarRightSide\_\_\_1Sdfc",

toolbarRightSideNoFlex: "toolbarRightSideNoFlex\_\_\_oa0FT",

toolbarItem: "toolbarItem\_\_\_3WbMb",

toolbarItemChild: "toolbarItemChild\_\_\_2U\_MP",

builtinButton: "builtinButton\_\_\_1zqo6",

builtinButtonDisabled: "builtinButtonDisabled\_\_\_3uvqe",

closeButton: "closeButton\_\_\_3BdAF",

zoomInButton: "zoomInButton\_\_\_3xtuX",

zoomOutButton: "zoomOutButton\_\_\_38PZx",

outerAnimating: "outerAnimating\_\_\_2-fZi",

closeWindow: "closeWindow\_\_\_2Hlon",

loadingCircle: "loadingCircle\_\_\_3JNJg",

loadingCirclePoint: "loadingCirclePoint\_\_\_3md-S",

pointFade: "pointFade\_\_\_2RA5J",

loadingContainer: "loadingContainer\_\_\_2vaJ-",

loadingContainer\_\_icon: "loadingContainer\_\_icon\_\_\_1wQQz"

};

}, /\* 5 \*/

/\*\*\*/

function(module, exports) {

/\*

MIT License http://www.opensource.org/licenses/mit-license.php

Author Tobias Koppers @sokra

\*/

// css base code, injected by the css-loader

module.exports = function() {

var list = [];

// return the list of modules as css string

// import a list of modules into the list

return list.toString = function() {

for (var result = [], i = 0; i < this.length; i++) {

var item = this[i];

item[2] ? result.push("@media " + item[2] + "{" + item[1] + "}") : result.push(item[1]);

}

return result.join("");

}, list.i = function(modules, mediaQuery) {

"string" == typeof modules && (modules = [ [ null, modules, "" ] ]);

for (var alreadyImportedModules = {}, i = 0; i < this.length; i++) {

var id = this[i][0];

"number" == typeof id && (alreadyImportedModules[id] = !0);

}

for (i = 0; i < modules.length; i++) {

var item = modules[i];

// skip already imported module

// this implementation is not 100% perfect for weird media query combinations

// when a module is imported multiple times with different media queries.

// I hope this will never occur (Hey this way we have smaller bundles)

"number" == typeof item[0] && alreadyImportedModules[item[0]] || (mediaQuery && !item[2] ? item[2] = mediaQuery : mediaQuery && (item[2] = "(" + item[2] + ") and (" + mediaQuery + ")"),

list.push(item));

}

}, list;

};

}, /\* 6 \*/

/\*\*\*/

function(module, exports, \_\_webpack\_require\_\_) {

var content = \_\_webpack\_require\_\_(4), insertCss = \_\_webpack\_require\_\_(7);

"string" == typeof content && (content = [ [ module.id, content, "" ] ]), module.exports = content.locals || {},

module.exports.\_getContent = function() {

return content;

}, module.exports.\_getCss = function() {

return content.toString();

}, module.exports.\_insertCss = function(options) {

return insertCss(content, options);

};

}, /\* 7 \*/

/\*\*\*/

function(module, exports, \_\_webpack\_require\_\_) {

"use strict";

function \_interopRequireDefault(obj) {

return obj && obj.\_\_esModule ? obj : {

default: obj

};

}

// Base64 encoding and decoding - The "Unicode Problem"

// https://developer.mozilla.org/en-US/docs/Web/API/WindowBase64/Base64\_encoding\_and\_decoding#The\_Unicode\_Problem

function b64EncodeUnicode(str) {

return btoa(encodeURIComponent(str).replace(/%([0-9A-F]{2})/g, function(match, p1) {

return String.fromCharCode("0x" + p1);

}));

}

/\*\*

\* Remove style/link elements for specified node IDs

\* if they are no longer referenced by UI components.

\*/

function removeCss(ids) {

ids.forEach(function(id) {

if (--inserted[id] <= 0) {

var elem = document.getElementById(prefix + id);

elem && elem.parentNode.removeChild(elem);

}

});

}

/\*\*

\* Example:

\* // Insert CSS styles object generated by `css-loader` into DOM

\* var removeCss = insertCss([[1, 'body { color: red; }']]);

\*

\* // Remove it from the DOM

\* removeCss();

\*/

function insertCss(styles) {

for (var \_ref = arguments.length > 1 && void 0 !== arguments[1] ? arguments[1] : {}, \_ref$replace = \_ref.replace, replace = void 0 !== \_ref$replace && \_ref$replace, \_ref$prepend = \_ref.prepend, prepend = void 0 !== \_ref$prepend && \_ref$prepend, ids = [], i = 0; i < styles.length; i++) {

var \_styles$i = (0, \_slicedToArray3.default)(styles[i], 4), moduleId = \_styles$i[0], css = \_styles$i[1], media = \_styles$i[2], sourceMap = \_styles$i[3], id = moduleId + "-" + i;

if (ids.push(id), !inserted[id] || replace) {

inserted[id] = 1;

var elem = document.getElementById(prefix + id), create = !1;

elem || (create = !0, elem = document.createElement("style"), elem.setAttribute("type", "text/css"),

elem.id = prefix + id, media && elem.setAttribute("media", media));

var cssText = css;

sourceMap && "function" == typeof btoa && (// skip IE9 and below, see http://caniuse.com/atob-btoa

cssText += "\n/\*# sourceMappingURL=data:application/json;base64," + b64EncodeUnicode((0,

\_stringify2.default)(sourceMap)) + "\*/", cssText += "\n/\*# sourceURL=" + sourceMap.file + "?" + id + "\*/"),

"textContent" in elem ? elem.textContent = cssText : elem.styleSheet.cssText = cssText,

create && (prepend ? document.head.insertBefore(elem, document.head.childNodes[0]) : document.head.appendChild(elem));

} else inserted[id]++;

}

return removeCss.bind(null, ids);

}

var \_stringify = \_\_webpack\_require\_\_(8), \_stringify2 = \_interopRequireDefault(\_stringify), \_slicedToArray2 = \_\_webpack\_require\_\_(9), \_slicedToArray3 = \_interopRequireDefault(\_slicedToArray2), prefix = "s", inserted = {};

module.exports = insertCss;

}, /\* 8 \*/

/\*\*\*/

function(module, exports) {

module.exports = require("babel-runtime/core-js/json/stringify");

}, /\* 9 \*/

/\*\*\*/

function(module, exports) {

module.exports = require("babel-runtime/helpers/slicedToArray");

}, /\* 10 \*/

/\*\*\*/

function(module, exports) {

module.exports = require("prop-types");

}, /\* 11 \*/

/\*\*\*/

function(module, exports) {

module.exports = require("react");

}, /\* 12 \*/

/\*\*\*/

function(module, exports) {

module.exports = require("react-modal");

} ]);

});

//# sourceMappingURL=react-image-lightbox.js.map