import { popperGenerator, detectOverflow } from "./createPopper";

export \* from "./types";

declare const defaultModifiers: (import("./modifiers/popperOffsets").PopperOffsetsModifier | import("./modifiers/flip").FlipModifier | import("./modifiers/hide").HideModifier | import("./modifiers/offset").OffsetModifier | import("./modifiers/eventListeners").EventListenersModifier | import("./modifiers/computeStyles").ComputeStylesModifier | import("./modifiers/arrow").ArrowModifier | import("./modifiers/preventOverflow").PreventOverflowModifier | import("./modifiers/applyStyles").ApplyStylesModifier)[];

declare const createPopper: <TModifier extends Partial<import("./types").Modifier<any, any>>>(reference: Element | import("./types").VirtualElement, popper: HTMLElement, options?: Partial<import("./types").OptionsGeneric<TModifier>>) => import("./types").Instance;

export { createPopper, popperGenerator, defaultModifiers, detectOverflow };

export { createPopper as createPopperLite } from "./popper-lite";

export \* from "./modifiers";