{"version":3,"file":"enums.min.js","sources":["../../src/enums.js"],"sourcesContent":["// @flow\nexport const top: 'top' = 'top';\nexport const bottom: 'bottom' = 'bottom';\nexport const right: 'right' = 'right';\nexport const left: 'left' = 'left';\nexport const auto: 'auto' = 'auto';\nexport type BasePlacement =\n | typeof top\n | typeof bottom\n | typeof right\n | typeof left;\nexport const basePlacements: Array<BasePlacement> = [top, bottom, right, left];\n\nexport const start: 'start' = 'start';\nexport const end: 'end' = 'end';\nexport type Variation = typeof start | typeof end;\n\nexport const clippingParents: 'clippingParents' = 'clippingParents';\nexport const viewport: 'viewport' = 'viewport';\nexport type Boundary = Element | Array<Element> | typeof clippingParents;\nexport type RootBoundary = typeof viewport | 'document';\n\nexport const popper: 'popper' = 'popper';\nexport const reference: 'reference' = 'reference';\nexport type Context = typeof popper | typeof reference;\n\nexport type VariationPlacement =\n | 'top-start'\n | 'top-end'\n | 'bottom-start'\n | 'bottom-end'\n | 'right-start'\n | 'right-end'\n | 'left-start'\n | 'left-end';\nexport type AutoPlacement = 'auto' | 'auto-start' | 'auto-end';\nexport type ComputedPlacement = VariationPlacement | BasePlacement;\nexport type Placement = AutoPlacement | BasePlacement | VariationPlacement;\n\nexport const variationPlacements: Array<VariationPlacement> = basePlacements.reduce(\n (acc: Array<VariationPlacement>, placement: BasePlacement) =>\n acc.concat([(`${placement}-${start}`: any), (`${placement}-${end}`: any)]),\n []\n);\nexport const placements: Array<Placement> = [...basePlacements, auto].reduce(\n (\n acc: Array<Placement>,\n placement: BasePlacement | typeof auto\n ): Array<Placement> =>\n acc.concat([\n placement,\n (`${placement}-${start}`: any),\n (`${placement}-${end}`: any),\n ]),\n []\n);\n\n// modifiers that need to read the DOM\nexport const beforeRead: 'beforeRead' = 'beforeRead';\nexport const read: 'read' = 'read';\nexport const afterRead: 'afterRead' = 'afterRead';\n// pure-logic modifiers\nexport const beforeMain: 'beforeMain' = 'beforeMain';\nexport const main: 'main' = 'main';\nexport const afterMain: 'afterMain' = 'afterMain';\n// modifier with the purpose to write to the DOM (or write into a framework state)\nexport const beforeWrite: 'beforeWrite' = 'beforeWrite';\nexport const write: 'write' = 'write';\nexport const afterWrite: 'afterWrite' = 'afterWrite';\nexport const modifierPhases: Array<ModifierPhases> = [\n beforeRead,\n read,\n afterRead,\n beforeMain,\n main,\n afterMain,\n beforeWrite,\n write,\n afterWrite,\n];\n\nexport type ModifierPhases =\n | typeof beforeRead\n | typeof read\n | typeof afterRead\n | typeof beforeMain\n | typeof main\n | typeof afterMain\n | typeof beforeWrite\n | typeof write\n | typeof afterWrite;\n"],"names":["bottom","right","left","auto","basePlacements","start","end","variationPlacements","reduce","acc","placement","concat","placements","beforeRead","read","afterRead","beforeMain","main","afterMain","beforeWrite","write","afterWrite","modifierPhases"],"mappings":";;;;kPAEaA,EAAmB,SACnBC,EAAiB,QACjBC,EAAe,OACfC,EAAe,OAMfC,EAAuC,CAV1B,MAUgCJ,EAAQC,EAAOC,GAE5DG,EAAiB,QACjBC,EAAa,MAyBbC,EAAiDH,EAAeI,QAC3E,SAACC,EAAgCC,UAC/BD,EAAIE,OAAO,CAAKD,MAAaL,EAAmBK,MAAaJ,MAC/D,IAEWM,EAA+B,UAAIR,GAAgBD,IAAMK,QACpE,SACEC,EACAC,UAEAD,EAAIE,OAAO,CACTD,EACIA,MAAaL,EACbK,MAAaJ,MAErB,IAIWO,EAA2B,aAC3BC,EAAe,OACfC,EAAyB,YAEzBC,EAA2B,aAC3BC,EAAe,OACfC,EAAyB,YAEzBC,EAA6B,cAC7BC,EAAiB,QACjBC,EAA2B,aAC3BC,EAAwC,CACnDT,EACAC,EACAC,EACAC,EACAC,EACAC,EACAC,EACAC,EACAC,qJA7DgD,uFAKlB,8BACM,sCAtBZ,yCAiBU"}