{"version":3,"file":"enums.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":["top","bottom","right","left","auto","basePlacements","start","end","clippingParents","viewport","popper","reference","variationPlacements","reduce","acc","placement","concat","placements","beforeRead","read","afterRead","beforeMain","main","afterMain","beforeWrite","write","afterWrite","modifierPhases"],"mappings":";;;;;;;;IACaA,GAAU,GAAG;IACbC,MAAgB,GAAG;IACnBC,KAAc,GAAG;IACjBC,IAAY,GAAG;IACfC,IAAY,GAAG;IAMfC,cAAoC,GAAG,CAACL,GAAD,EAAMC,MAAN,EAAcC,KAAd,EAAqBC,IAArB;IAEvCG,KAAc,GAAG;IACjBC,GAAU,GAAG;IAGbC,eAAkC,GAAG;IACrCC,QAAoB,GAAG;IAIvBC,MAAgB,GAAG;IACnBC,SAAsB,GAAG;IAgBzBC,mBAA8C,gBAAGP,cAAc,CAACQ,MAAf,CAC5D,UAACC,GAAD,EAAiCC,SAAjC;AAAA,SACED,GAAG,CAACE,MAAJ,CAAW,CAAKD,SAAL,SAAkBT,KAAlB,EAAqCS,SAArC,SAAkDR,GAAlD,CAAX,CADF;AAAA,CAD4D,EAG5D,EAH4D;IAKjDU,UAA4B,gBAAG,UAAIZ,cAAJ,GAAoBD,IAApB,GAA0BS,MAA1B,CAC1C,UACEC,GADF,EAEEC,SAFF;AAAA,SAIED,GAAG,CAACE,MAAJ,CAAW,CACTD,SADS,EAELA,SAFK,SAEQT,KAFR,EAGLS,SAHK,SAGQR,GAHR,CAAX,CAJF;AAAA,CAD0C,EAU1C,EAV0C;;IAc/BW,UAAwB,GAAG;IAC3BC,IAAY,GAAG;IACfC,SAAsB,GAAG;;IAEzBC,UAAwB,GAAG;IAC3BC,IAAY,GAAG;IACfC,SAAsB,GAAG;;IAEzBC,WAA0B,GAAG;IAC7BC,KAAc,GAAG;IACjBC,UAAwB,GAAG;IAC3BC,cAAqC,GAAG,CACnDT,UADmD,EAEnDC,IAFmD,EAGnDC,SAHmD,EAInDC,UAJmD,EAKnDC,IALmD,EAMnDC,SANmD,EAOnDC,WAPmD,EAQnDC,KARmD,EASnDC,UATmD;;;;;;;;;;;;;;;;;;;;;;;;;"}