## Pretty-print [

{"version":3, "file": "externalsregistry.inline.1bb9c887.bundle.min.js","mappings":"0HAGAA,0AAOC,mBAAqBD,0AAOC,oBAAsB,CAAEC,IA AK,CAAC,GAEjE,MAAM,OAAEC,EAAM,MAAEC,EAAK,SAAEC,EAAQ,eAAEC,EAAC,UAAEC,GAAeP,OAAOQ,kBAAoB,CAC1FL ,OAAQ,CAAC,EACTC,MAAO,CAAC,EACRC,SAAU,CAAC,EACXC,eAAqB,CAAC,EACjBC,UAAW,CAAC,GAGbD,EAAeG,OAAS, IAAIC,QAASC,IACpCL,EAAeM,OAASD,IAGzBR,EAAOM,OAAS,IAAIC,QAASC,IAC5BR,EAAOS,OAASD,IAGjBJ,EAAUE,O AAS,IAAIC,QAASC,IAC/BJ,EAAUK,OAASD,IAIfX,OAAOa,WAEXb,OAAOc,kBAAoBd,OAAOa,SAAW,CAAEE,SAAS,IAEzD V,EAASI,OAAS,IAAIC,QAASC,IAC9BN,EAASO,OAAS,KAIjBI,OAAOC,OAAOjB,OAAOc,mBAAqB,CAAC,EAAGd,OAAOa,S AAU, CAAEE, SAAS, IAC1EJ, OAIGX, OAAOkB, QAEX1B, OAAOmB, eAAiBnB, OAAOkB, MAAQ, CAAEH, SAAS, IAEnDX, EAAMK, O AAS, IAAIC, QAASC, IAC3BP, EAAMQ, OAAS, KAIdI, OAAOC, OAAOjB, OAAOmB, gBAAkB, CAAC, EAAGnB, OAAOkB, MAAO, CAA EH, SAAS, IACPEJ, OAIFX, OAAOOB, uBAAyBV, QAAQW, IAAI, CAACjB, EAAMK, OAAQJ, EAASI, SACPET, OAAOSB, iC", "sou rces":["webpack:///./externals-registry.ts"], "sourcesContent":["// expose externals (lodash, react, react dom, ...) libs on the window for chunks to be able to import them before they are loaded\n\n// set \_\_imageClientApi\_\_ to be an object so it can be passed as reference to functions and set as ref when it is downloaded by customElement\nwindow.\_\_imageClientApi\_\_ window. imageClientApi\_ || { sdk: {} }\n\nconst { lodash, react, reactDOM, imageClientApi, clientSdk } = (window.externalsRegistry = {\n\tlodash: {} as typeof window.externalsRegistry['lodash'],\n\treact: {} as typeof window.externalsRegistry['react'],\n\treactDOM: {} as typeof window.externalsRegistry['reactDOM'],\n\timageClientApi: {} as typeof window.externalsRegistry['imageClientApi'],\n\tclientSdk: {} as typeof window.externalsRegistry['clientSdk'],\n})\n\nimageClientApi.loaded = new Promise((resolve) => {\n\timageClientApi.onload = resolve\n})\n\nlodash.loaded = new Promise((resolve) => {\n\tlodash.onload = resolve\n})\n\nclientSdk.loaded = new Promise((resolve) => {\n\tclientSdk.onload = resolve\n})\n\n// sometimes React is loaded before externals-window.ReactDOM = { loading: true }\n}\nreactDOM.loaded = new Promise((resolve) =>  $\{\n\treactDOM.onload = () => \{\n\t// if a chunk that is dependent on react-dom was evaluated$ prior to evaluating react-dom,\n\t\t// we need to override the stale reactDOMReference this chunk holds\n\t\t// with window.ReactDOM which is overridden by react-dom evaluation.\n\t\t0bject.assign(window.reactD0MReference || {}, window.ReactD0M, { loading: false  $)\n\t\ensuremath{\line}$  (!window.React)  $\n\t//\$ @ts $ignore\n\twindow.reactReference = window.React = { loading: true }\n\twindow.react.loaded = new }$ Promise((resolve) =>  $\{\n\treact.onload = () => \{\n\treact.onload = ($ was evaluated prior to evaluating react,\n\t\t// we need to override the stale reactReference this chunk holds\n\t\// with window.React which is overridden by react evaluation.\n\t\t0bject.assign(window.reactReference || {}, window.React, { loading: false })\n\t\tresolve()\n\t}\n})\n\nwindow.reactAndReactDOMLoaded = Promise.all([react.loaded, reactDOM.loaded])\nwindow.resolveExternalsRegistryPromise()\n"], "names":
["window", "\_\_imageClientApi\_\_", "sdk", "lodash", "react", "reactDOM", "imageClientApi", "clientSdk",
"externalsRegistry", "loaded", "Promise", "resolve", "onload", "ReactDOM", "reactDOMReference", "load
ing", "Object", "assign", "React", "reactReference", "reactAndReactDOMLoaded", "all", "resolveExterna lsRegistryPromise"], "sourceRoot":""}