

```
const targetNodeTitle = "Crow Canyon NITRO Apps";

const targetNodeTitle2 = "Crow Canyon Branding (Modern sites)"; // <-- Replace with the name
of the link to remove

const deletedSites = [];
```

```
function getRequestDigest(webUrl) {
    return $.ajax({
        url: webUrl + "/_api/contextinfo",
        method: "POST",
        headers: { "Accept": "application/json; odata=verbose" }
    });
}
```

```
function getSubsites(webUrl) {
    return $.ajax({
        url: webUrl + "/_api/web/webs?$filter=WebTemplate ne 'APP'",
        method: "GET",
        headers: { "Accept": "application/json; odata=verbose" }
    });
}
```

```
function getQuickLaunchNodes(webUrl) {
    return $.ajax({
        url: webUrl + "/_api/web/Navigation/QuickLaunch",
        method: "GET",
        headers: { "Accept": "application/json; odata=verbose" }
    });
}
```

```
function deleteQuickLaunchNode(webUrl, nodeId, requestDigest) {
    return $.ajax({
```

```

url: webUrl + "/_api/web/Navigation/QuickLaunch(" + nodeId + ")/deleteObject",
method: "POST",
headers: {
    "Accept": "application/json; odata=verbose",
    "X-RequestDigest": requestDigest,
    "IF-MATCH": "*",
    "X-HTTP-Method": "DELETE"
}
});
}

```

```

function processSite(webUrl) {
    return getRequestDigest(webUrl).then(function (digestData) {
        const digestValue = digestData.d.GetContextWebInformation.FormDigestValue;

        return getQuickLaunchNodes(webUrl).then(function (data) {
            const nodes = data.d.results;
            const deletions = [];

            nodes.forEach(function (node) {
                if (node.Title === targetNodeTitle) {
                    console.log(` Deleting "${targetNodeTitle}" from ${webUrl}` );
                    deletions.push(deleteQuickLaunchNode(webUrl, node.Id, digestValue));
                } else if (node.Title === targetNodeTitle2) {
                    console.log(` Deleting "${targetNodeTitle2}" from ${webUrl}` );
                    deletions.push(deleteQuickLaunchNode(webUrl, node.Id, digestValue));
                }
            });

            return Promise.all(deletions);
        });
    });
}

```

```

    });
}

function processAllSites(rootWebUrl) {
    const queue = [rootWebUrl];

    function processNext() {
        if (queue.length === 0) return Promise.resolve();

        const currentWebUrl = queue.shift();
        return processSite(currentWebUrl).then(() => {
            return getSubsites(currentWebUrl).then(function (data) {
                const subsites = data.d.results;
                subsites.forEach(function (site) {
                    queue.push(site.Url);
                });
                return processNext();
            });
        });
    }

    return processNext();
}

var AllSiteCollections = [];
if (AllSiteCollections.length == 0) {
    AllSiteCollections.push(_spPageContextInfo.siteAbsoluteUrl);
}
var Index = 0;

var _CallprocessAllSites = function () {

```

```
const rootSiteUrl = AllSiteCollections[Index];
processAllSites(rootSiteUrl).then(function () {
  if (AllSiteCollections.length == Index) {
    alert("Quick Launch node deletion complete across all subsites.");
  } else {
    Index++;
    _CallprocessAllSites();
  }
}).catch(function (err) {
  console.error("Error during processing:", err);
});
}
```

```
_CallprocessAllSites();
```