lode Three

This helper module simplyfies loading lode-generated LOD artifacts in a Three.js application.

Usage

Import the module and the lode-manifest and create the `lodeContext` by calling the `createContext` method.

```
1 import * as lodLoader from '@kreativwebdesign/lode-three'
2 import manifest from "./lode-build/lode-manifest.json";
3
4 const lodeContext = lodeLoader.createContext({
5   basePath: "./lode-build",
6   manifest,
7 });
```

and then load the artifacts:

```
1 lodeLoader.loadModel({
2     lodeContext,
3     artifactName: 'path-to/artifact',
4 })
```

API

createContext

This method accepts the basePath to the lode-build output and the lode-manifest.

```
1 {
2   basePath: "./lode-build", // Relative path to the lode-build
        output
3   manifest, // the manifest generated by the lode-build, by
        default at lode-build/lode-manifest.json
4 }
```

loadModel

This method accepts the basePath of the generated artifacts where all the artifacts are and the specific artifact name.

```
1 {
2     lodeContext, // Object returned from the createContext method
3     artifactName: 'path-to/artifact', // path to the artifact
          base
4 }
```

This would work for the following folder structure:

```
 1 |-index.js // where the code is
 2 |-lode-build
 3   |-lode-manifest.json
 4   |-path-to
 5     |-artifact
 6       |-lod-0
 7       | |-artifact.gltf
 8       | |-artifact.bin
 9       |-lod-1
10       | |-artifact.gltf
11       | |-artifact.bin
```

Outlook

- Add culled lod at the end of each lod if the last one is not set to infinity.
- Add helper for loading manifest async
- Add helper to progressively load LOD-artifacts

Dev

Run `yarn setup` to get up and running

Publish

- Prepare versions and ensure clean setup state
- `npm publish --access public`