

C3 AI Suite Tips and Tricks

- [Fetch Method](#)
 - ['this'](#)
- [Clearing Metadata Errors](#)
- [Cleaning Your Tag](#)
 - [Cleaning Type Data](#)
 - [Cleaning All Data](#)
- [Checking Your Authorization](#)
- [Checking C3 Server Version](#)
- [Checking the Provisioned Package](#)
- [Listing all Types in a package](#)

Fetch Method

'this'

When fetching a Type which has as property another Type, you can use the 'this' keyword in the 'include' field of the fetch spec to get the entire subordinate object. For example:

```
entity type TypeA schema name "TypeA" {  
  name: string  
  child: TypeB  
}
```

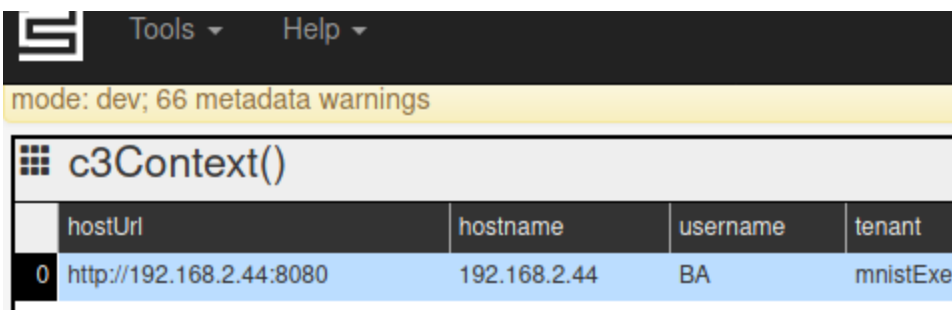
Fetch command:

```
TypeA.fetch({'include': 'child.this'})
```

```
c3.TypeA.fetch({'include': 'child.this'})
```

Clearing Metadata Errors

Whenever you provision a package to your tag you will get a multitude of metadata warnings, and in some cases errors. This results in the following yellow or red banner at the top of your static console:



To clear these errors, simply issue the following in static console:

```
DbMetadataIssue.removeAll()
```

Or this for a connected python session:

```
c3.DbMetadataIssue.removeAll()
```

Cleaning Your Tag

When developing a package, you frequently need to provision and re-provision progressively changing versions. This can result in older versions of your data lingering on your tag even though you're no longer using it. Additionally, when using your tag for a different package, you may leave behind old data which is not used.

There are two options for cleaning data.

Cleaning Type Data

Suppose you want to remove the data associated with the Type 'typeName'. All that's needed in this case is the 'removeAll' command in static console:

```
typeName.removeAll()
```

Or this for a connected python session:

```
c3.TypeName.removeAll()
```

Cleaning All Data

You can remove all data from your tag with the 'removeData'. Please be aware that not all users are authorized to do this.

From static console:

```
Tag.removeData()
```

From a connected python session:

```
c3.Tag.removeData()
```

Warning: Please be aware that this command will remove all data from your tag. Please be sure you are really finished using that data, or back it up if necessary.

Checking Your Authorization

You may not be authorized to execute some of the above commands. To check this, you can use the Authorizer Type. For example, to check whether Tag.removeData is allowed, execute the following in static console:

```
Authorizer.isAuthorized("Tag", "removeData")

// or

Authorizer.actionAuthzRoles({typeName: {typeName: 'Tag'}, action: 'removeData'})
```

And the same in Python would be:

```
c3.Authorizer.isAuthorized("Tag", "removeData")

# or

c3.Authorizer.actionAuthzRoles(spec={'typeName': {'typeName': 'Tag'}, 'action': 'removeData'})
```

Checking C3 Server Version

Sometimes you may need to know the version of the C3 Server you're using. You can do this from the static console with the following:

```
server_version = Cluster.hosts()[0].serverInfo.buildCITag
```

Checking the Provisioned Package

```
c3Grid(TagProvisionLog.fetch({  
  order: "descending(deployDate)",  
  limit: 1  
}))
```

Listing all Types in a package

```
c3Grid(TagMetadataStore.typesByPackage( "packagename" ) )
```