Inspector on: lively.morphic.Box [Menu] [—] [X]

Filter: [show standard properties ▾] **(1)**

▼ this DatabaseConnector
  ▶ addToTable addToTable
  ▶ cachedBounds Rectangle
  ▶ createMetadataTables createMetadataTables
  ▶ createResource createResource
  ▶ createTableForForm createTableForForm
  ▶ dbQueryPostRoute evaljs
  ▶ derivationIds [...]
  ▶ distanceToDragEvent Point
  ▶ drop drop
  ▶ dropObject dropObject
  ▶ droppingEnabled true
  ▶ ensureFormIsPresent ensureFormIsPresent
  ▶ eventHandler EventHandler
 ▶ eventsAreDisabled false
  ▶ eventsAreIgnored false

this

☐ Bind 'this' to selection

StyleEditorPane [Menu] [—] [X]

DatabaseConnector

Appearance   CSS   Layout   **(2)**

Fill ☑
  Color [blue]
  Opacity [————]

Border ☑
  Color [   ]
  Width [○——————]
  Radius [○——————]
  Style [solid ▾]

Misc
  Clip mode [visible ▾]

ObjectEditor [Menu] [—] [X]

Tag: [all]   [<lively.morphic.Box#A13A5... - DatabaseConn 🔍] [save] [run]

Scripts [+] [-]
  -- ALL --
  addToTable
  createMetadataTables
  createResource
  createTableForForm
  drop
  dropObject
  ensureFormIsPresent
  exec
  execFunction
  fetchAll
  fetchFormForTable
  getDbHost
  getLastInsertId
  insert

Connections [+] [-]
  -- ALL --

```
this.addScript(function addToTable(dataForm) {

    if (!this.tableExists(dataForm.getTableName())) {
        this.createTableForForm(dataForm);
    }
    this.saveObject(dataForm.getTableName(), dataForm.asDataC
());
}).tag([]);

this.addScript(function createMetadataTables() {
    this.exec("create table metadata (tableName text, columnN
, type text, selector text, partName text)");
    this.exec("create table metadata_forms (tableName text, u
s integer, form blob)");
}).tag([]);

this.addScript(function createResource()      (3)
    var res = new WebResource('http://' +        ost() +
this.dbQueryPostRoute);
    return res;
}).tag([]);

this.addScript(function createTableForForm(dataForm) {
```