

# Models - Access a Model

Any callback in the application that is passed the `request` object can access the Models programmatically. If the `actions` property in the model definition is set, some of the methods cannot be invoked. The `actions` property restricts which CRUD operations can be invoked on the models.

1. Retrieve an instance to API Builder using the `request.server` property.
2. Retrieve the Model instance using API Builder's `getModel('name')` method by passing it the name of the model.
3. Invoke one of the following methods on the Model instance and pass it a callback function, which is passed an `error` and `results` object:
  - `create(object, callback)`: Creates a model
  - `query(options, callback)`: Retrieves models specified by the query
  - `findAll(callback)`: Retrieves all models
  - `findById(id, callback)`: Retrieves the model specified by the id parameter
  - `update(instance, callback)`: Updates the passed model
  - `deleteAll(callback)`: Deletes all models
  - `delete(instance, callback)`: Deletes the passed model

## Example:

The Route example below retrieves all car model.

### web/routes/testroute.js

```
var Arrow = require('arrow');

var TestRoute = Arrow.Router.extend({
  name: 'car',
  path: '/car',
  method: 'GET',
  description: 'get some cars',
  action: function (req, res, next) {
    var model = req.server.getModel('car');
    model.findAll(function(err, results){
      if (err) {
        next(err);
      } else {
        req.log.info('got cars ' + JSON.stringify(results));
        res.render('car', {cars:results});
      }
    });
  }
});

module.exports = TestRoute;
```