Neo4j.rb Screencast Series

Episode 4 Association chaining

```
9
```

```
require './config/environment'
   class User
     include Neo4j: ActiveNode
     has_many :out, :created_assets, type: :CREATED, model_class: :Asset
   end
9 class Asset
     include Neo4j: ActiveNode
10
11
     has_many :out, :categories, type: :HAS_CATEGORY
12
13 end
14
15 user = User.find_by(name: 'Sally')
16
```

NORMAL > master example.rb[+] ruby utf-8[unix] < 6%: 1: 1
12 fewer lines

```
2
```

```
require './config/environment'
 3 class User
      include Neo4j::ActiveNode
     has_many :out, :created_assets, type: :CREATED, model_class: :Asset
   end
 9 class Asset
     include Neo4j::ActiveNode
10
11
12
     has_many :out, :categories, type: :HAS_CATEGORY
13 end
14
15 user = User.find_by(name: 'Sally')
16
17 user.created_assets.to_a # => [#<Asset uuid: "c56caa73-8b00-4840-95e7-0b54e69b2376">, #<Asset uui
   d: "bcf88f93-b970-4f9e-a452-ae6fd03eab6d">, #<Asset uuid: "d137f683-bed5-4c0e-a315-6d5cb8799a55">
     #<Asset uuid: "0df38f05-298f-4c95-84c8-4ebdd1a0c170">1
18
19 # >> MATCH (n: 'User')
20 # >> WHERE (n.name = {n_name})
21 # >> RETURN n
22 # >> LIMIT {limit_1}
23 # >>
24 # >> MATCH user4865
25 # >> WHERE (ID(user4865) = {ID_user4865})
26 # >> MATCH user4865-[rel1:`CREATED`]->(result_created_assets:'Asset`)
27 # >> RETURN result_created_assets
28 # >>
                   example.rb[+]
                                                                       utf-8[unix] < 60%: 17: 24
NORMAL > master
                                                                ruby
17 lines filtered
```

```
4
```

```
3 class User
     include Neo4j ActiveNode
     has many :out, :created assets, type: :CREATED, model class: :Asset
   end
 9 class Asset
10
     include Neo4j::ActiveNode
11
12
    has many :out, :categories, type: :HAS_CATEGORY
13 end
14
15 user = User.find_by(name: 'Sally')
16
   user.created_assets.categories.to_a # => [#<Category uuid: "8753279d-754b-40cb-a05e-288f46c7ab87"
     created at: Tue, 28 Jul 2015 19:41:49 +0000, name: "Graph Databases", updated at: Tue, 28 Jul 2
   015 19:41:49 +0000>, #<Category uuid: "ae959f80-dacd-4272-939e-02d5b553f84f", created at: Tue, 28
    Jul 2015 19:55:41 +0000, name: "Poetry", updated_at: Tue, 28 Jul 2015 19:55:41 +0000>, #<Categor
   y uuid: "601c51ff-cf2e-439d-9109-481b24b4ae3f", created_at: Tue, 28 Jul 2015 20:22:33 +0000, name
   : "Databases", updated_at: Tue, 28 Jul 2015 20:22:33 +0000>, #<Category uuid: "8753279d-754b-40cb
   -a05e-288f46c7ab87", created_at: Tue, 28 Jul 2015 19:41:49 +0000, name: "Graph Databases", update
   d_at: Tue, 28 Jul 2015 19:41:49 +0000>, #<Category uuid: "1679a6a5-2e6b-435c-a3ef-eb0470d91dc0",
   created_at: Tue, 28 Jul 2015 19:55:00 +0000, name: "Tutorials", updated_at: Tue, 28 Jul 2015 19:5
   5:00 +0000>, #<Category uuid: "601c51ff-cf2e-439d-9109-481b24b4ae3f", created_at: Tue, 28 Jul 201
   5 20:22:33 +0000, name: "Databases", updated_at: Tue, 28 Jul 2015 20:22:33 +0000>, #<Category uui
   d: "8753279d-754b-40cb-a05e-288f46c7ab87", created_at: Tue, 28 Jul 2015 19:41:49 +0000, name: "Gr
   aph Databases", updated_at: Tue, 28 Jul 2015 19:41:49 +0000>]
18
                                                                                                 Þ
19 # >> MATCH (n:`User`)
20 # >> WHERE (n.name = {n_name})
21 # >> RETURN n
                                                     utf-8[unix] < 58%: 17: 31 <! trailing ?
NORMAL > master example.rb[+]
                                              ruby
```

```
4
```

```
has_many :out, :categories, type: :HAS_CATEGORY
13 end
14
15 user = User.find_by(name: 'Sally')
16
17 user.created_assets.categories.to_a # => [#<Category uuid: "8753279d-754b-40cb-a05e-288f46c7ab87"
    created_at: Tue, 28 Jul 2015 19:41:49 +0000, name: "Graph Databases", updated_at: Tue, 28 Jul 2
   015 19:41:49 +0000>, #<Category uuid: "ae959f80-dacd-4272-939e-02d5b553f84f", created_at: Tue, 28
    Jul 2015 19:55:41 +0000, name: "Poetry", updated_at: Tue, 28 Jul 2015 19:55:41 +0000>, #<Categor
   y uuid: "601c51ff-cf2e-439d-9109-481b24b4ae3f", created_at: Tue, 28 Jul 2015 20:22:33 +0000, name
   : "Databases", updated at: Tue, 28 Jul 2015 20:22:33 +0000>, #<Category uuid: "8753279d-754b-40cb
   -a05e-288f46c7ab87", created at: Tue, 28 Jul 2015 19:41:49 +0000, name: "Graph Databases", update
   d_at: Tue, 28 Jul 2015 19:41:49 +0000>, #<Category uuid: "1679a6a5-2e6b-435c-a3ef-eb0470d91dc0",
   created_at: Tue, 28 Jul 2015 19:55:00 +0000, name: "Tutorials", updated_at: Tue, 28 Jul 2015 19:5
   5:00 +0000>, #<Category uuid: "601c51ff-cf2e-439d-9109-481b24b4ae3f", created_at: Tue, 28 Jul 201
   5 20:22:33 +0000, name: "Databases", updated_at: Tue, 28 Jul 2015 20:22:33 +0000>, #<Category uui
   d: "8753279d-754b-40cb-a05e-288f46c7ab87", created_at: Tue, 28 Jul 2015 19:41:49 +0000, name: "Gr
   aph Databases", updated_at: Tue, 28 Jul 2015 19:41:49 +0000>]
18
19 # >> MATCH (n: 'User')
20 # >> WHERE (n.name = {n_name})
21 # >> RETURN n
22 # >> LIMIT {limit_1}
23 # >>
24 # >> MATCH user4865
25 # >> WHERE (ID(user4865) = {ID_user4865})
26 # >> MATCH user4865-[rel1: CREATED ]->(node3: Asset )
27 # >> MATCH node3-[rel2: HAS_CATEGORY] -> (result categories: Category)
28 # >> RETURN result categories
29 # >>
```

utf-8[unix] < 58%: 17: 31 < ! trailings

NORMAL > master example.rb[+]

```
2
```

```
has_many :out, :categories, type: :HAS_CATEGORY
13 end
14
15 user = User.find_by(name: 'Sally')
16
17 user.created_assets.categories.assets.to_a # => [#<Asset uuid: "67676eab-bbac-4398-80e6-f43c9e460
   78e">, #<Asset uuid: "c56caa73-8b00-4840-95e7-0b54e69b2376">, #<Asset uuid: "bcf88f93-b970-4f9e-a
   452-ae6fd03eab6d">, #<Asset uuid: "d137f683-bed5-4c0e-a315-6d5cb8799a55">, #<Asset uuid: "c56caa7
   3-8b00-4840-95e7-0b54e69b2376">, #<Asset uuid: "c56caa73-8b00-4840-95e7-0b54e69b2376">, #<Asset u
   uid: "bcf88f93-b970-4f9e-a452-ae6fd03eab6d">, #<Asset uuid: "67676eab-bbac-4398-80e6-f43c9e46078e
   ">, #<Asset uuid: "c56caa73-8b00-4840-95e7-0b54e69b2376">, #<Asset uuid: "bcf88f93-b970-4f9e-a452
   -ae6fd03eab6d">, #<Asset uuid: "d137f683-bed5-4c0e-a315-6d5cb8799a55">, #<Asset uuid: "bcf88f93-b
   970-4f9e-a452-ae6fd03eab6d">, #<Asset uuid: "c56caa73-8b00-4840-95e7-0b54e69b2376">, #<Asset uuid
   : "bcf88f93-b970-4f9e-a452-ae6fd03eab6d">, #<Asset uuid: "67676eab-bbac-4398-80e6-f43c9e46078e">,
    #<Asset uuid: "c56caa73-8b00-4840-95e7-0b54e69b2376">, #<Asset uuid: "bcf88f93-b970-4f9e-a452-ae
   6fd03eab6d">, #<Asset uuid: "d137f683-bed5-4c0e-a315-6d5cb8799a55">]
18
19 # >> MATCH (n:`User`)
20 # >> WHERE (n.name = {n_name})
21 # >> RETURN n
22 # >> LIMIT {limit 1}
23 # >>
24 # >> MATCH user4865
25 # >> WHERE (ID(user4865) = {ID_user4865})
26 # >> MATCH user4865-[rel1: CREATED ]->(node3: Asset)
27 # >> MATCH node3-[rel2: HAS_CATEGORY] -> (node4: Category)
28 # >> MATCH node4<-[rel3: HAS_CATEGORY ]-(result_assets: Asset)</pre>
29 # >> RETURN result assets
30 # >>
                   example.rb[+]
                                                     utf-8[unix] < 56%: 17: 37 < ! trailing ?
NORMAL > master
                                               ruby
29 lines filtered
```

```
9
```

```
has_many :out, :categories, type: :HAS_CATEGORY
13 end
14
15 user = User.find_by(name: 'Sally')
16
17 user.created_assets.categories.assets.to_cypher # => "MATCH user4865 WHERE (ID(user4865) = {ID_us
   er4865}) MATCH user4865-[rel1: CREATED ]->(node3: Asset ) MATCH node3-[rel2: HAS_CATEGORY ]->(node3-
   e4: 'Category') MATCH node4<-[rel3: 'HAS_CATEGORY']-(result_assets: 'Asset')"
18
19 # >> MATCH (n: 'User')
20 # >> WHERE (n.name = {n_name})
21 # >> RETURN n
22 # >> LIMIT {limit_1}
23 # >>
                                                      utf-8[unix] < 73%: 17: 47 < ! trailing 73
                   example.rb[+]
NORMAL > master
                                               ruby
30 lines filtered
```

```
2
```

```
14
15 user = User.find_by(name: 'Sally')
16
   user.created_assets.categories.assets.print_cypher # => nil
18
19 # >> MATCH (n:`User`)
   # >> WHERE (n.name = {n_name})
   # >>> RETURN n
   # >> LIMIT {limit_1}
24 # >> MATCH user4865
25 # >> WHERE (ID(user4865) = {ID_user4865})
   # >> MATCH user4865-[rel1: CREATED ]->(node3: Asset )
   # >> MATCH node3-[rel2:`HAS_CATEGORY`]->(node4:`Category`)
28 # >> MATCH node4<-[rel3: 'HA5_CATEGORY']-(result_assets: 'Asset')
                                                      utf-8[unix] < 60%: 17: 50 < ! trailing ?:
                   example.rb[+]
NORMAL > master
                                               ruby
23 lines filtered
```

has_many :out, :categories, type: :HAS_CATEGORY

13 end

```
2
```

```
has_many :out, :categories, type: :HAS_CATEGORY
13 end
14
15 user = User.find_by(name: 'Sally')
16
17 user.created_assets.where(public: true).categories.to_a # => [#<Category uuid: "8753279d-754b-400
   b-a05e-288f46c7ab87", created_at: Tue, 28 Jul 2015 19:41:49 +0000, name: "Graph Databases", updat
   ed_at: Tue, 28 Jul 2015 19:41:49 +0000>, #<Category uuid: "8753279d-754b-40cb-a05e-288f46c7ab87"
    created_at: Tue, 28 Jul 2015 19:41:49 +0000, name: "Graph Databases", updated_at: Tue, 28 Jul 20
   15 19:41:49 +0000>, #<Category uuid: "1679a6a5-2e6b-435c-a3ef-eb0470d91dc0", created_at: Tue, 28
   Jul 2015 19:55:00 +0000, name: "Tutorials", updated at: Tue, 28 Jul 2015 19:55:00 +0000>, #<Categ
   ory uuid: "601c51ff-cf2e-439d-9109-481b24b4ae3f", created_at: Tue, 28 Jul 2015 20:22:33 +0000, na
   me: "Databases", updated_at: Tue, 28 Jul 2015 20:22:33 +0000>)
18
19 # >> MATCH (n: 'User')
20 # >> WHERE (n.name = {n_name})
21 # >> RETURN n
22 # >> LIMIT {limit 1}
23 # >>
24 # >> MATCH user4865
25 # >> WHERE (ID(user4865) = {ID_user4865})
26 # >> MATCH user4865-[rel1:'CREATED']->(node3:'Asset')
27 # >> WHERE (node3.public = {node3_public})
28 # >> MATCH node3-[rel2: HAS_CATEGORY] -> (result_categories: Category)
29 # >> RETURN result categories
30 # >>
                                                     utf-8[unix] < 56%: 17: 40 < ! trailing ?
                  example.rb[+]
NORMAL > master
                                               ruby
```

28 lines filtered

```
9
```

```
has_many :out, :categories, type: :HAS_CATEGORY
13 end
14
15 user = User.find_by(name: 'Sally')
16
17 user.created_assets.categories(:category).pluck[['category.name', 'count(*)'] # => [["Poetry", 1],
    ["Tutorials", 1], ["Databases", 2], ["Graph Databases", 3]]
18
19 # >> MATCH (n: 'User')
20 # >> WHERE (n.name = {n_name})
21 # >> RETURN n
22 # >> LIMIT {limit_1}
23 # >>
24 # >> MATCH user4865
25 # >> WHERE (ID(user4865) = {ID_user4865})
26 # >> MATCH user4865-[rel1: CREATED ]->(node3: Asset )
   # >> MATCH node3-[rel2:`HAS_CATEGORY`]->(category:`Category`)
28 # >> RETURN
29 # >>
         category.name,
         count(*)
30 # >>
31 # >>
                                                     utf-8[unix] < 54%: 17: 76 <! trailing 7
                  example.rb[+]
NORMAL > master
                                              ruby
```

30 lines filtered

```
4
```

```
has_many :out, :categories, type: :HAS_CATEGORY
13 end
14
15 user = User.find_by(name: 'Sally')
16
17 user.created_assets.categories(:category, :has_category_rel).pluck[:category, :has_category_rel]
   # => [[#<Category uuid: "8753279d-754b-40cb-a05e-288f46c7ab87", created_at: Tue, 28 Jul 2015 19:4
   1:49 +0000, name: "Graph Databases", updated_at: Tue, 28 Jul 2015 19:41:49 +0000>, CypherRelation
   ship 25045], [#<Category uuid: "ae959f80-dacd-4272-939e-02d5b553f84f", created_at: Tue, 28 Jul 20
   15 19:55:41 +0000, name: "Poetry", updated_at: Tue, 28 Jul 2015 19:55:41 +0000>, CypherRelationsh
   ip 25051], [#<Category uuid: "601c51ff-cf2e-439d-9109-481b24b4ae3f", created_at: Tue, 28 Jul 2015
   20:22:33 +0000, name: "Databases", updated at: Tue, 28 Jul 2015 20:22:33 +0000>, CypherRelations
   hip 25053], [#<Category uuid: "8753279d-754b-40cb-a05e-288f46c7ab87", created_at: Tue, 28 Jul 201
   5 19:41:49 +0000, name: "Graph Databases", updated_at: Tue, 28 Jul 2015 19:41:49 +0000>, CypherRe
   lationship 25044], [#<Category uuid: "1679a6a5-2e6b-435c-a3ef-eb0470d91dc0", created_at: Tue, 28
   Jul 2015 19:55:00 +0000, name: "Tutorials", updated_at: Tue, 28 Jul 2015 19:55:00 +0000>, CypherR
   elationship 25050], [#<Category uuid: "601c51ff-cf2e-439d-9109-481b24b4ae3f", created_at: Tue, 28
   Jul 2015 20:22:33 +0000, name: "Databases", updated_at: Tue, 28 Jul 2015 20:22:33 +0000>, Cypher
   Relationship 25054], [#<Category uuid: "8753279d-754b-40cb-a05e-288f46c7ab87", created_at: Tue, 2
   8 Jul 2015 19:41:49 +0000, name: "Graph Databases", updated_at: Tue, 28 Jul 2015 19:41:49 +0000>,
    CypherRelationship 25043]]
18
19 # >> MATCH (n: 'User')
20 # >> WHERE (n.name = {n_name})
21 # >> RETURN n
22 # >> LIMIT {limit_1}
23 # >>
24 # >> MATCH user4865
25 # >> WHERE (ID(user4865) = {ID_user4865})
26 # >> MATCH user4865-[rel1: `CREATED`]->(node3: `Asset`)
27 # >> MATCH node3-[has_category_rel:`HAS_CATEGORY`]->(category:`Category`)
28 # >> RETURN
                                                     utf-8[unix] < 54%: 17: 96 < ! trailing
NORMAL > master example.rb[+]
```

```
2
```

<! trailing[7]

```
has_many :out, :categories, type: :HAS_CATEGORY
13 end
14
15 user = User.find_by(name: 'Sally')
16
   user.created_assets(:asset).categories(:category, :has_category_rel, optional: true).p
18
19 # >> MATCH (n: 'User')
   # >> WHERE (n.name = {n_name})
   # >> RETURN n
22 # >> LIMIT {limit_1}
24 # >> MATCH user4865
25 # >> WHERE (ID(user4865) = {ID_user4865})
26 # >> MATCH user4865-[rel1: CREATED ]->(node3: Asset )
   # >> MATCH node3-[has_category_rel:`HAS_CATEGORY`]->(category:`Category`)
   # >> RETURN
29 # >>
          category,
30 # >>
          has_category_rel
31 # >>
```

ruby

utf-8[unix] <

54%:

- INSERT --

example.rb[+]

```
2
```

```
has_many :out, :categories, type: :HAS_CATEGORY
13 end
14
15 user = User.find_by(name: 'Sally')
16
17 user.created_assets(:asset).categories(:category, :has_category_rel, optional: true).pluck[['asset]
   .title', 'collect(category.name)') # => [["A Database By Any Other Name", ["Graph Databases", "Po
   etry", "Databases"]], ["Database Principles", []], ["Graphs and You", ["Graph Databases"]], ["Bui
   lding With Graph Databases", ["Graph Databases", "Tutorials", "Databases"]]]
18
19 # >> MATCH (n:`User`)
20 # >> WHERE (n.name = {n_name})
21 # >> RETURN n
22 # >> LIMIT {limit_1}
23 # >>
24 # >> MATCH user4865
25 # >> WHERE (ID(user4865) = {ID_user4865})
26 # >> MATCH user4865-[rel1:`CREATED`]->(asset:`Asset`)
27 # >> OPTIONAL MATCH asset-[has_category_rel:`HAS_CATEGORY`]->(category:`Category`)
28 # >> RETURN
29 # >> asset.title,
         collect(category.name)
30 # >>
31 # >>
                                                     utf-8[unix] < 54%: 17:131 < ! trailing ?
NORMAL > master
                  example.rb[+]
                                               ruby
```

31 lines filtered

```
2
```

```
has_many :out, :categories, type: :HAS_CATEGORY
13 end
14
15 user = User.find_by(name: 'Sally')
16
17 user.created_assets(:asset).categories.where(name: /.*computer.*/i).pluck[['DISTINCT asset.title']
     # => ["A Database By Any Other Name", "Building With Graph Databases"]
18
19 # >> MATCH (n: 'User')
20 # >> WHERE (n.name = {n_name})
21 # >> RETURN n
22 # >> LIMIT {limit 1}
23 # >>
24 # >> MATCH user4865
25 # >> WHERE (ID(user4865) = {ID_user4865})
26 # >> MATCH user4865-[rel1: CREATED ]->(asset: Asset )
27 # >> MATCH asset-[rel2:`HAS_CATEGORY`]->(result_categories:`Category`)
28 # >> WHERE (result_categories.name =~ '(7i).*computer.*')
29 # >> RETURN DISTINCT asset.title
30 # >>
                                                      utf-8[unix] < 56%: 17: 97 <! trailing 7
                   example.rb[+]
NORMAL > master
                                               ruby
31 lines filtered
```

```
_
```

```
has_many :out, :created_assets, type: :CREATED, model_class: :Asset
     has_many :out, :viewed_assets, type: :VIEWED, model_class: :Asset
10 class Asset
11
     include Neo4j::ActiveNode
12
     has_many :out, :categories, type: :HAS_CATEGORY
13
14 end
15
16 user = User.find_by(name: 'Sally')
18 User.viewed_assets.to_a # => [#<Asset uuid: "d137f683-bed5-4c0e-a315-6d5cb8799a55">, #<Asset uuid
    "bcf88f93-b970-4f9e-a452-ae6fd03eab6d">, #<Asset uuid: "67676eab-bbac-4398-80e6-f43c9e46078e">,
    #<Asset uuid: "67676eab-bbac-4398-80e6-f43c9e46078e">, #<Asset uuid: "49f6c8b8-a4f7-48d1-9b2f-21
   08cad24ba6">, #<Asset uuid: "0df38f05-298f-4c95-84c8-4ebdd1a0c170">, #<Asset uuid: "d137f683-bed5
   -4c0e-a315-6d5cb8799a55">, #<Asset uuid: "bcf88f93-b970-4f9e-a452-ae6fd03eab6d">]
19
20 # >> MATCH (n: `User`)
21 # >> WHERE (n.name = {n_name})
22 # >> RETURN n
23 # >> LIMIT {limit_1}
24 # >>
25 # >> MATCH (node2: User )
26 # >> MATCH node2-[rel1:`VIEWED`]->(result_viewed_assets:`Asset`)
27 # >> RETURN result viewed assets
28 # >>
                                                      utf-8[unix] < 64%: 18: 23 < ! trailings?
NORMAL > master example.rb[+]
                                               ruby
1 change; before #113 7 seconds ago
```

```
has_many :out, :created_assets, type: :CREATED, model_class: :Asset
     has_many :out, :viewed_assets, type: :VIEWED, model_class: :Asset
 8 end
10 class Asset
     include Neo4j::ActiveNode
11
12
13
     has many :out, :categories, type: :HAS_CATEGORY
14 end
15
16 user = User.find_by(name: 'Sally')
18 User.as(:user).viewed_assets.categories(:category).pluck[['user.name', 'collect(category.name)']
    => [["Jim", ["Neo4j", "Graph Databases", "Love", "Graph Databases", "Computers", "Tutorials",
   raph Databases"]], ["Sally", ["Graph Databases", "Computers", "Tutorials", "Graph Databases", "Ne
   o4j", "Graph Databases"]]]
20 # >> MATCH (n: User )
21 # >> WHERE (n.name = {n_name})
22 # >> RETURN n
23 # >> LIMIT {limit_1}
24 # >>
25 # >> MATCH (user: 'User')
26 # >> MATCH user-[rel1: VIEWED] -> (node3: 'Asset')
27 # >> MATCH node3-[rel2: HAS_CATEGORY] -> (category: Category)
28 # >> RETURN
29 # >> user.name,
        collect(category.name)
30 # >>
31 # >>
                                                     utf-8[unix] < 58%: 18: 95 < ! trailing
```

ruby

NORMAL > master

example.rb[+]

```
4
```

```
has_many :out, :created_assets, type: :CREATED, model_class: :Asset
     has_many :out, :viewed_assets, type: :VIEWED, model_class: :Asset
 8 end
10 class Asset
     include Neo4j::ActiveNode
11
12
13
     has many :out, :categories, type: :HAS_CATEGORY
14 end
15
16 class Category
     include Neo4j::ActiveNode
17
18
19
     property :name
20
     scope :ordered, -> { order(:name) }
21 end
22
23 user = User.find_by(name: 'Sally')
24
25 user.created_assets.categories.ordered.pluck(:name) # => ["Computers", "Computers", "Graph Databa
   ses", "Graph Databases", "Graph Databases", "Poetry", "Tutorials"]
26
27 # >> MATCH (n: 'User')
28 # >> WHERE (n.name = {n_name})
29 # >> RETURN n
30 # >> LIMIT {limit_1}
31 # >>
   # >> MATCH user4865
33 # >> WHERE (ID(user4865) = {ID_user4865})
34 # >> MATCH user4865-[rel1: CREATED ]->(node3: Asset )
35 # >> MATCH node3-[rel2:`HAS CATEGORY`]->(result categories:`Category`)
                                                                           25: 51 < ! trailing 74
NORMAL > master example.rb[+]
                                              ruby
                                                     utf-8[unix] < 65%:
```

```
_
```

```
has_many :out, :created_assets, type: :CREATED, model_class: :Asset
     has many :out, :viewed assets, type: :VIEWED, model class: :Asset
 8 end
10 class Asset
11
     include Neo4j::ActiveNode
12
13
     has_many :out, :categories, type: :HAS_CATEGORY
14 end
15
16 class Category
17
     include Neo4j::ActiveNode
18
19
     property :name
20
     def self ordered
21
       all.order(:name)
22
     end
23 end
24
25 user = User.find_by(name: 'Sally')
26
   user.created_assets.categories.ordered.pluck(:name) # => ["Computers", "Computers", "Graph Databa
   ses", "Graph Databases", "Graph Databases", "Poetry", "Tutorials"]
28
29 # >> MATCH (n: 'User')
30 # >> WHERE (n.name = {n_name})
31 # >> RETURN n
32 # >> LIMIT {limit 1}
33
34 # >> MATCH user4865
35 # >> WHERE (ID(user4865) = {ID user4865})
36 # >> MATCH user4865-[rel1:`CREATED`]->(node3:`Asset`)
                                                     utf-8[unix] < 55%: 22: 5 < ! trailings
NORMAL > master example.rb[+]
                                              ruby
```

```
_
```

```
16 class Category
      include Neo4j :: ActiveNode
17
18
19
     property : name
20
     def self.most_recently_updated
        all.order(:updated_at).last
21
22
     end
23 end
24
25 user = User.find_by(name: 'Sally')
26
27 user.created_assets.categories.most_recently_updated # => #<Category uuid: "601c51ff-cf2e-439d-91
   09-481b24b4ae3f", name: "Computers">
28
29 # >> MATCH (n:`User`)
30 # >> WHERE (n.name = {n_name})
31 # >> RETURN n
32 # >> LIMIT {limit 1}
33 # >>
34 # >> MATCH user4865
35 # >> WHERE (ID(user4865) = {ID_user4865})
36 # >> MATCH user4865-[rel1:`CREATED`]->(node3:`Asset`)
37 # >> MATCH node3-[rel2:`HAS_CATEGORY`]->(n:`Category`)
38 # >> WITH n
39 # >> ORDER BY n.updated_at
40 # >> RETURN LAST(COLLECT(n)) as n
41 # >>
                                                      utf-8[unix] < 65%: 27: 52 <! trailing
                   example.rb[+]
NORMAL > master
                                               ruby
40 lines filtered
```

15