Prev Next

Exercise (Instructions): Loopback Relations		
Objectives and Outcomes		
In this exercise you will explore the use of model relations in Loopback and howev can link various models by defining relations among them. In additionn you will explore the use of a timestamp mixin. At the end of this		
exercise, you will be able to: • Define model relations among various Loopback models		
Make use of a mixin within the Loopback server		
Add a Comments Model Create a new model called Comments by typing the following at the prompt:		
stream a new moder camed comments by typing the rollowing at the prompt. 1 s1c loopback:model		
- Use the Notional's options: Section Section Section		
2 Date Source Prograd 2 Date Source Prograd 3 Date Source Program 5 Pauls Feders: common 7 Date Source Program 7 Date Source Program		
7 Name: Rating B Type: number 9 Required: Yes		
10 overalt: 5 11 Name: comment 12 Type: String 13 Required: Yes		
14 Default: (empty)		
Setting up Model Relations • To define relationships, type the following at the command prompt:		
1 sic loopback:relation		
First the relation between dishes and Comments, use the following options:		
1 Roda's dishar		
2 Relation type: has many 3 Relationshy with: Comments 4 Name: comments 5 Foreign key: rome 6 Through model: no		
Now define a relation between dishes and customers, use the following options:		
1 Model: dishes 2 Belation type: has many 3 Belationship with: Customer 4 Name: Customer 5 Forein Nav: nome		
2 Redain States 2 Selection types has many 3 Malestonskip with Customer 4 Name: Continers 5 Foreign key: nome 6 Through modify mig		
Between Comments and Dishes, use the following options:		
1 Model: Comments 2 Mealston type: belongs to 3 Mealstonship with: dishes 4 Name: dishes		
Asser: dishes Foreign Rey: none Between Comments and Customer, use the following options:		
1 Model: Comments		
2 Relation type: belongs to 3 Relationship with: Cutomor 4 Name: customer 5 Foreign key: customerid		
Between Customer and Comments, use the following options:		
1 Rodel: Customer 2 Salation type: has many 3 Salationship with Commant		
2 Relation type: has many 3 Relationship with: Comment 5 Foreign key: customerId 6 Require through model: ngl		
Define and Use a Mixin		
Install the loopback-ds-timestamp-mixin as follows:		
1 npm install loopback-ds-timestamp-mixinsave		
Open model-config.json in the server folder, edit the mixins as follows:		
1 "minista": [2 "lemphoxic/memon/minista"; 3 "lemphoxic/memon/minista"; 4 "./nommundsis/lemphoxic-de-timentamp-minist"; 5 "./common/minista"; 6 "./sitia"; 7 "./sitia";		
5 "./commo/nixina", 6 "./mixina", 7		
 To use the mixin, add the following code to both comments, json and dishes, json in the common folder, after the properties: 	•	
1 "mixins": { 2		
Configuring Access Control		
You will now set access control for both dishes and Comments by typing the following at the prompt:		
1 slc loopback:scl		
For the dishes model, use the following settings:		
1 Model: dishes 2 Scope: All methods and properties 3 access type: Write 4 role: other 5 role among defin		
4 role: other 5 role name: addin 6 Permission: Explicitly grant access		
For the Comments model, use the following options:		
1 Model: Comments 2 Scope: All methods and properties 3 access type: All 4 role: All users		
Total All users Persistation: Explicitly deny access Now to allow customers to read comments, use the following options:		
1 Hotel: Commants		
2 Scope: All setheds and properties 3 access type: Read 4 role: Any anthenticated user 5 Permission: Epolicity grant access		
To allow customers to post comments, use the following options:		
1 Nodel: Comments 2 Scope: A single method 3 method name: create 4 role: May suthenticated user		
5 Permission: Explicitly grant access		
To allow a customer that posted a comment to edit or delete the comment, use the following options: Model: Comments		
1 Model: Comments 2 Scope: All methods and properties 3 access type: Write the Comment of the Comme		
Permission: Explicitly grent access Start the server and explore the REST API using the API explorer.		
 In particular if you get comments with the following filter: ("include": "dishes", customer")), the system will include the dish information and customer information into the comments. 		
Conclusions		
In this exercise you explored the use of model relations in Loopback and used a timestamp mixin.	,	
Mark as completed	J	
C	O Po	

1 of 1 12/14/2016 5:52 PM