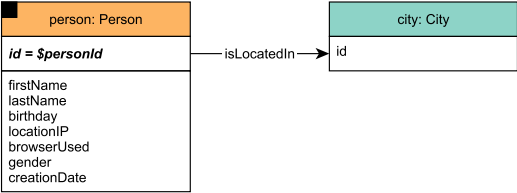


IS 1  
IS 2  
IS 3  
IS 4  
IS 5  
IS 6  
IS 7

Interactive / short / 1

query	Interactive / short / 1				
title	Profile of a person				
pattern					
description	Given a start Person with ID \$personId, retrieve their first name, last name, birthday, IP address, browser, and city of residence.				
params	1	\$personId	ID		
result	1	person.firstName	String	R	
	2	person.lastName	String	R	
	3	person.birthday	Date	R	
	4	person.locationIP	String	R	
	5	person.browserUsed	String	R	
	6	city.id	ID	R	
	7	person.gender	String	R	
	8	person.creationDate	DateTime	R	

## Interactive / short / 2

IS 1

IS 2

IS 3

IS 4

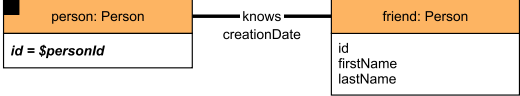
IS 5

IS 6

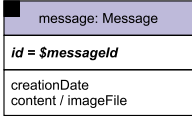
IS 7

query	Interactive / short / 2																																							
title	Recent messages of a person																																							
pattern	<div> <div> <div>person: Person</div> <div>id = \$personId</div> </div> <div> <div>message: Message</div> <div>id content / imageFile creationDate</div> </div> <div> <div>originalPoster: Person</div> <div>id firstName lastName</div> </div> <div> <div>post: Post</div> <div>id</div> </div> <div> <div>hasCreator</div> <div>replyOf*0..</div> </div> </div>																																							
description	Given a start Person with ID \$personId, retrieve the last 10 Messages created by that user. For each Message, return that Message, the original Post in its conversation (post), and the author of that Post (originalPoster). If any of the Messages is a Post, then the original Post (post) will be the same Message, i.e. that Message will appear twice in that result.																																							
params	<div> <div>1</div> <div>\$personId</div> <div>ID</div> <div></div> </div>																																							
result	<table> <tr> <td>1</td> <td>message.id</td> <td>ID</td> <td>R</td> <td></td> </tr> <tr> <td>2</td> <td>message.content or message.imageFile (for photos)</td> <td>Text</td> <td>R</td> <td></td> </tr> <tr> <td>3</td> <td>message.creationDate</td> <td>DateTime</td> <td>R</td> <td></td> </tr> <tr> <td>4</td> <td>post.id</td> <td>ID</td> <td>R</td> <td></td> </tr> <tr> <td>5</td> <td>originalPoster.id</td> <td>ID</td> <td>R</td> <td></td> </tr> <tr> <td>6</td> <td>originalPoster.firstName</td> <td>String</td> <td>R</td> <td></td> </tr> <tr> <td>7</td> <td>originalPoster.lastName</td> <td>String</td> <td>R</td> <td></td> </tr> </table>					1	message.id	ID	R		2	message.content or message.imageFile (for photos)	Text	R		3	message.creationDate	DateTime	R		4	post.id	ID	R		5	originalPoster.id	ID	R		6	originalPoster.firstName	String	R		7	originalPoster.lastName	String	R	
1	message.id	ID	R																																					
2	message.content or message.imageFile (for photos)	Text	R																																					
3	message.creationDate	DateTime	R																																					
4	post.id	ID	R																																					
5	originalPoster.id	ID	R																																					
6	originalPoster.firstName	String	R																																					
7	originalPoster.lastName	String	R																																					
sort	<table> <tr> <td>1</td> <td>message.creationDate</td> <td>↓</td> <td></td> </tr> <tr> <td>2</td> <td>message.id</td> <td>↓</td> <td></td> </tr> </table>					1	message.creationDate	↓		2	message.id	↓																												
1	message.creationDate	↓																																						
2	message.id	↓																																						
limit	10																																							

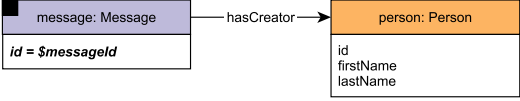
### Interactive / short / 3

IS 1	query	Interactive / short / 3			
IS 2	title	Friends of a person			
IS 3	pattern				
IS 4					
IS 5					
IS 6					
IS 7	description	Given a start Person with ID \$personId, retrieve all of their friends, and the date at which they became friends.			
	params	1	\$personId	ID	
	result	1	friend.id	ID	R
		2	friend.firstName	String	R
		3	friend.lastName	String	R
		4	knows.creationDate	DateTime	R
	sort	1	knows.creationDate	↓	
		2	friend.id	↑	

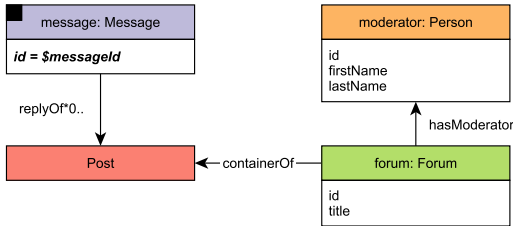
### Interactive / short / 4

IS 1	query	Interactive / short / 4			
IS 2	title	Content of a message			
IS 3	pattern				
IS 4					
IS 5					
IS 6					
IS 7	description	Given a Message with ID \$messageId, retrieve its content and creation date.			
	params	1	\$messageId	ID	
	result	1	message.creationDate	DateTime	R
		2	message.content or message.imageFile (for photos)	Text	R

## Interactive / short / 5

IS 1	query	Interactive / short / 5			
IS 2	title	Creator of a message			
IS 3	pattern	 <pre> graph LR     Message["message: Message&lt;br/&gt;id = \$messageId"] -- hasCreator --&gt; Person["person: Person&lt;br/&gt;id&lt;br/&gt;firstName&lt;br/&gt;lastName"]           </pre>			
IS 4					
IS 5					
IS 6					
IS 7	description	Given a Message with ID \$messageId, retrieve its author.			
	params	1	\$messageId	ID	
	result	1	person.id	ID	R
		2	person.firstName	String	R
		3	person.lastName	String	R

## Interactive / short / 6

IS 1	query	Interactive / short / 6			
IS 2	title	Forum of a message			
IS 3	pattern	 <pre> graph TD     Message["message: Message&lt;br/&gt;id = \$messageId"] -- replyOf --&gt; Post["Post"]     Forum["forum: Forum&lt;br/&gt;id&lt;br/&gt;title"] -- containerOf --&gt; Post     Forum -- hasModerator --&gt; Person["moderator: Person&lt;br/&gt;id&lt;br/&gt;firstName&lt;br/&gt;lastName"]           </pre>			
IS 4					
IS 5					
IS 6					
	description	Given a Message with ID \$messageId, retrieve the Forum that contains it and the Person that moderates that Forum. Since Comments are not directly contained in Forums, for Comments, return the Forum containing the original Post in the thread which the Comment is replying to.			
	params	1	\$messageId	ID	
	result	1	forum.id	ID	R
		2	forum.title	Long String	R
		3	moderator.id	ID	R
		4	moderator.firstName	String	R
		5	moderator.lastName	String	R

## Interactive / short / 7

IS 1  
IS 2  
IS 3  
IS 4  
IS 5  
IS 6  
IS 7

query	Interactive / short / 7																																							
title	Replies of a message																																							
pattern																																								
description	<p>Given a Message with ID \$messageId, retrieve the (1-hop) Comments that reply to it.</p> <p>In addition, return a boolean flag knows indicating if the author of the reply (replyAuthor) knows the author of the original message (messageAuthor). If author is same as original author, return False for knows flag.</p>																																							
params	<table><tr><td>1</td><td>\$messageId</td><td>ID</td><td></td></tr></table>					1	\$messageId	ID																																
1	\$messageId	ID																																						
result	<table><tr><td>1</td><td>comment.id</td><td>ID</td><td>R</td><td></td></tr><tr><td>2</td><td>comment.content</td><td>Text</td><td>R</td><td></td></tr><tr><td>3</td><td>comment.creationDate</td><td>DateTime</td><td>R</td><td></td></tr><tr><td>4</td><td>replyAuthor.id</td><td>ID</td><td>R</td><td></td></tr><tr><td>5</td><td>replyAuthor.firstName</td><td>String</td><td>R</td><td></td></tr><tr><td>6</td><td>replyAuthor.lastName</td><td>String</td><td>R</td><td></td></tr><tr><td>7</td><td>knows</td><td>Boolean</td><td>C</td><td>True if the knows edge exists between the replyAuthor and the messageAuthor nodes, False otherwise (including the case when the two nodes are the same)</td></tr></table>					1	comment.id	ID	R		2	comment.content	Text	R		3	comment.creationDate	DateTime	R		4	replyAuthor.id	ID	R		5	replyAuthor.firstName	String	R		6	replyAuthor.lastName	String	R		7	knows	Boolean	C	True if the knows edge exists between the replyAuthor and the messageAuthor nodes, False otherwise (including the case when the two nodes are the same)
1	comment.id	ID	R																																					
2	comment.content	Text	R																																					
3	comment.creationDate	DateTime	R																																					
4	replyAuthor.id	ID	R																																					
5	replyAuthor.firstName	String	R																																					
6	replyAuthor.lastName	String	R																																					
7	knows	Boolean	C	True if the knows edge exists between the replyAuthor and the messageAuthor nodes, False otherwise (including the case when the two nodes are the same)																																				
sort	<table><tr><td>1</td><td>comment.creationDate</td><td>↓</td><td></td></tr><tr><td>2</td><td>replyAuthor.id</td><td>↑</td><td></td></tr></table>					1	comment.creationDate	↓		2	replyAuthor.id	↑																												
1	comment.creationDate	↓																																						
2	replyAuthor.id	↑																																						