

## Interactive / short / 2

IS 1	query	Interactive / short / 2			
IS 2	title	Recent messages of a person			
IS 3	pattern	 <pre> graph LR     person[person: Person] -- hasCreator --&gt; message[message: Message]     message -- replyOf*0.. --&gt; post[post: Post]     post -- hasCreator --&gt; originalPoster[originalPoster: Person]     </pre> <p>The diagram illustrates the following entities and relationships:</p> <ul style="list-style-type: none"> <li><b>person: Person</b> (orange box) with attribute <code>id = \$personId</code>.</li> <li><b>message: Message</b> (purple box) with attributes <code>id</code>, <code>content / imageFile</code>, and <code>creationDate</code>.</li> <li><b>post: Post</b> (red box) with attribute <code>id</code>.</li> <li><b>originalPoster: Person</b> (orange box) with attributes <code>id</code>, <code>firstName</code>, and <code>lastName</code>.</li> <li>Relationships: <ul style="list-style-type: none"> <li><code>person: Person</code> hasCreator <code>message: Message</code>.</li> <li><code>message: Message</code> replyOf*0.. <code>post: Post</code>.</li> <li><code>post: Post</code> hasCreator <code>originalPoster: Person</code>.</li> </ul> </li> </ul>			
	desc.	<p>Given a start Person, 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.</p>			
	params	1	personId	ID	
	result	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	1	message.creationDate	↓	
		2	message.id	↓	