|  |  |
| --- | --- |
| **Name:** |  |
| **Email:** |  |

**Question 1**

In your own words explain the 4 sub constraints in REST’s Uniform Interface. Give an everyday example to illustrate each of the constraint.

**Question 2**

What is the difference between the following HTTP methods?

1. POST, PUT and PATCH
2. GET and HEAD

**Question 3**

You have a monolithic web application for managing warehouses. The application exposes the following end points

* /warehouses – list of all warehouses
* /warehouse/<warehouse\_id> – returns the warehouse’s details
* /warehouse/<warehouse\_id>/inventories – inventory list for the warehouse
* /inventories – list of all the inventories
* /inventory/<inventory\_id> – inventory detail
* /inventory/<inventory\_id>/report – generate a report

Describe how you can scale this application

1. By duplication
2. By functional decomposition
3. By data partitioning

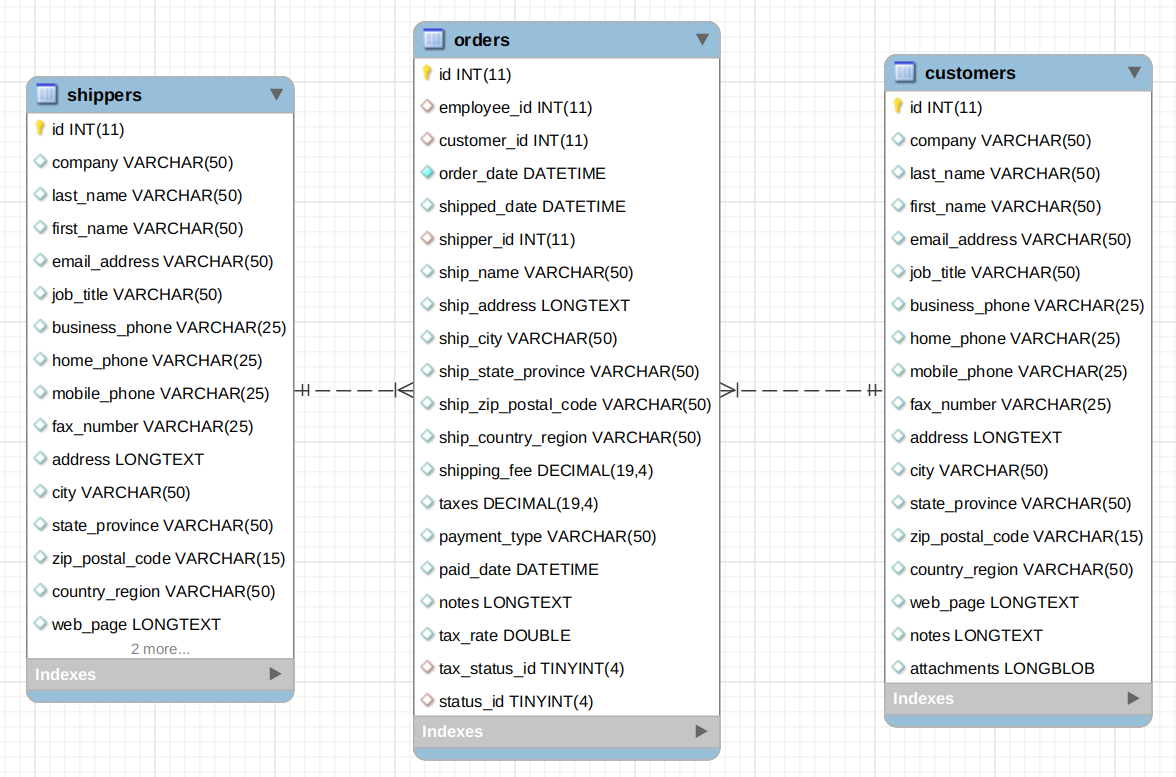
**Question 4**

Study the top headlines REST API from newsapi.org. Answer the following questions

1. List the different ways you can present the API key when performing an invocation
2. Construct a URL to get 30 technologies headlines from Japan
3. What is the status code if an incorrect API key is used?
4. How long will the result be cached?

**Question 5**

Study the following entity-relationship diagram



Both customer and shippers has a one to many relationships with orders.

Answer the following questions.

1. Design one or more API endpoints to return a list of customers and a single customer
2. What are some criteria and how might you might include in your endpoint (wrt Q5a)?
3. Show a sample output of a customer’s list as a result of performing a GET on the resource. (wrt Q5a)
4. How do you provide flow control or pagination support (wrt Q5a)?

**Question 6**

You have deployed a service to encode video viz. convert AVI to mp4, etc. Subscribers of your service uploads their video to the service; after conversion the converted video is returned to the subscriber (assume that the conversion time is short).

You charge the subscribers based on the 2 criteria.

1. Subscription rates based on the cumulative video sizes: 500GB, 1TB, 1.5TB, etc. A subscriber who subscribe to the 500GB package can upload a maximum amount of 500GB videos.
2. Charge the subscribe based on their ingress and egress traffic viz. the upload and downloads of the videos.

Design an API for this encoding service to give your subscribers control over their encoding process.

You can ignore authentication.

**Submission**

Copy this Word document to your repository and commit it.

git add .

git commit -m ‘worksheet01’

git push origin master