**Technical Test**
There are no time restrictions on this test, but the sooner you return it, the sooner we can progress your application! You are welcome to do some research, but please don't get any outside help. There are sometimes no right or wrong answers. We want to understand how your mind works, and how you communicate your thought processes in written form.
Tasks / Questions

**Communications**
How would you explain each of these concepts or technologies to a non-technical person?

**Data warehouse**

- Whenever you access a website the data you see on the page is stored in a database somewhere and accessing that website is essentially doing a query to grab the data from a server and show it to you on the front end. Sometimes we want to store more long term data for historical or legal purposes like health records or sensitive government information. You would use a data warehouse for this because they are built in a way that optimizes how you can store and query for a lot of data. They can also be used to fuel large language models like chatgpt.

**Cloud computing**

- Imagine you have a personal computer at home that you use to play games, store documents, and do work. Once you don't have the physical device in front of you, you can't access that functionality anymore. Cloud computing on the other hand lets you access that computer and its resources from anywhere in the world as long as you have an internet connection.

**AWS: S3**

- Google drive but for aws allows you to store images, pdfs, text files, and all types of data on your cloud services

**AWS: Kinesis**

- A cost effective tool used to grab data from large streams for processing for later you would use something like this for a site like amazon that collects data about you every time you click a part of the page or scroll that data gets stored and processed in real time to give you better suggestions for products to buy

**AWS: EC2**

- A tool that allows you to put virtual servers or instances on the cloud a server allows you to host things like websites, automation tools, data pipelines and everything in between having this flexibility allows you to create a range of tools and programs from simple to complex with the flexibility of having it on the cloud so that anyone can connect to it

**AWS: DynamoDB**

- A database that allows you to store a bunch of information on the cloud in an organized way. Imagine you have a huge ecommerce website like amazon and you want to organize the information in a way that you'll be able to access them whenever you want them you would use DynamoDB for that.

**AWS: Redshift**

- If you have a lot of data that you need to organize like the ecommerce example with amazon redshift gives you tools that lets you analyze this data efficiently and give you insights on that data it also lets you organize them as columns instead of rows which makes it easier and much faster to search for data within this database

**GCP: Compute Engine**

- Imagine you want to try out a really data intensive game but you only have a chromebook instead of buying a brand new computer for $1000+ you rent one from a friend for 20$ and return it when you're done using it. This is similar to the way a GCP compute engine works. It allows you to rent powerful virtual machines in the cloud that can be customized to whatever you need it for if you need more ram, more memory, or even if you want more copies of that same machine you can do it with this service. It allows you to run your applications and process tasks effectively without the need for buying and owning the actual hardware or infrastructure.

**GCP: Storage**

- If you have a lot of files to store and access and you want to access them from any computer on the internet you would use google cloud storage. It works similar to google drive but allows much more flexibility and depth to the things you can do with the data you store. For example data warehousing, bucketing, and machine learning integrations are all things that gcp storage has that drive doesn't

**GCP: PubSub**

- Imagine you're at a sushi place but instead of standing in line you sit down at your table and there's a conveyor belt of sushi on plates that goes around the restaurant and you pick up the plate you want as it comes by your table. This is a simplified way of how GCP pubsub works you have multiple publishers that shoot data through a data stream and multiple subscribers that wait for that data to arrive and pull it from that data stream when it comes to them this way you can create an event driven architecture for sharing messages thats fast and reliable

**GCP: BigQuery**

- Imagine you have a database with a ton of data and you want to search through it, in normal databases these queries can take exponentially longer as the amount of data grows. Bigquery on the other hand optimizes this and works like a search engine specifically for handling huge amounts of data in a cost-effective way. It also integrates with other tools like tableau and data studio. And it lets you query through your data using a languages called SQL

**Google Analytics**

- Imagine you have a website and you have a ton of users coming in and out and you want to know how to improve your website, you want to know what part of the site users like and what part of the site users don't and you want to optimize that so you get more user traffic. Google analytics gives you tools to do that. It's a smart system that tracks

user behavior on your app and gives you reports so you better understand your customers needs and improve your website to meet them.

**Cookies**

- You notice that sometimes when you open your browser you're still logged in despite not explicitly logging in. That's because of cookies, they store information about you in a cache so that when you navigate to that site in the future all of your settings and preferences are stored. This is mainly done to reduce load on the servers so that every time that you login they don't have to do extensive queries on a database. These cookies can sometimes expire after a certain amount of time for security reasons as well.

**JSON**

- Stands for javascript object notation. It's a standardized way that the internet packages javascript to send it over different services the same way that when you send a package you need to have a standard format before the post office can accept it the internet uses this object standard for api services.

**JSON Schema**

- A way to validate and structure the content of a javascript object notation. The same way we have grammar rules for how to properly structure a sentence web technologies have a standardized way to format data. This makes it easy to make sure everyone is on the same page when sharing data across the internet.

**Respond to this unhappy customer:**

Hello, This is the fourth time I have chased your organization for a response on this support ticket. Can you help me or not? What do I need to do to get a resolution to my technical problem? Our company's reporting relies on this data and my boss is really unhappy with the delay!!

Dear Customer,

I apologize for any inconvenience we may have caused you regarding your support ticket. At Snowplow, we are actively working to resolve your ticket as quickly as possible, as it is our top priority. We strive to provide timely and efficient support, and we are committed to finding a resolution to your issue promptly.

To expedite the process, could you please provide me with your technical details once again and anyone you talked to about this issue? If you have any deadlines associated with the issue, kindly let me know, and I will ensure they are taken into consideration. Additionally, please provide any relevant logs, pictures, videos, or contextual information regarding your issue. Your cooperation in providing this information would be greatly appreciated as it will assist us in further investigating the issue.

Once again, we apologize for any inconvenience caused and appreciate your patience and understanding. If you require any further assistance or have any additional concerns, please do not hesitate to let me know.

**Help this neighbor:**

I have a huge room which I would like to fill with basketballs for my kid's party. How can I do That?

- I would ask the neighbor what type of basketball we are using for this and what is the size of the room were going to need to fill it with
- Once i have these measurements we can calculate exactly how many balls we need to fill the room completely
- Once we have that we need to determine if we want to order these balls pre-inflated or flat, flat balls are easier to store but take some time to inflate while inflated balls are ready out the box
- If the neighbor has a budget we can calculate the cost of buying inflated vs deflated and the time it would take to fill deflated balls up once we have a price point we can give them an estimate
- We also might want to consider if they're buying these balls or renting them
- Logistics of moving those balls to the party may be an issue as well though most wholesale places will deliver to your house if you buy enough in bulk https://www.bargainw.com/wholesale-product/922182/Basketball.html

**Logical thinking and problem solving**

Explain how you would solve these problems, showing your working:

Your mother planted some seeds in three flowerpots for you, and carefully labeled them. Unfortunately, the labels got muddled on the car journey to your house, and she put them back in a hurry. The only thing she is certain of is that the flowerpots are now all incorrectly labeled. She tells you one pot has violets, one has daisies, and one has a mix of violets and daisies. You wait for the seeds to grow so that you can put the labels on the correct pots. Eventually, one small flower appears, and you realize you can now label all the pots correctly without waiting for any more flowers to grow. How?

- By process of elimination the pot that grows the flower tells you the flower that that pot is in and that the pot that has that flower in it as the label will be the opposite of what its labeled as by process of elimination
- For example if flowers are labeled [key:value] where key = flower and value = label
- If were given the pots in a way that it looks like
- [daisy: violetLabel] [unknownA : daisyLabel] [unknownB : daisy&VioletLabel]
- We know that unknownB can't be a daisy because daisy already exists and it can't be a daisy and violet because the labels specifically say that they are all incorrectly labeled so unknownB must be violet

- This leaves unknownA to be  daisy & Violet by process of elimination

You have a rope that takes exactly one hour to burn from one end to the other, but it burns unevenly, meaning that it doesn't burn at a constant rate. How would you use this rope to measure exactly 45 minutes?

- If we assume that all threads from this rope will burn at the same rate no matter the thickness we can do this by taking strands from the rope that are the same length as the original rope
- We take two strands burn strand A from both sides and burn strand B from one side
- Once strand A has finished burning stop strand B and mark that on the original rope as the 30 minute point
- Repeat the same thing for two strands half the size of this to mark a 15 minute point
- From there you can make a 45 minute rope

The other way we can solve this is by taking two strands from the rope strandA and strandB
Burn strandA at both ends and burn strandB at one end
strandA will finish after 30 minutes
When that happens burn the other side of strandB so that it finishes in 15 minutes

**Technical knowledge**

Domain Name System (DNS)
Where is the domain name snowplowanalytics.com registered

- IONOS SE
- ionos.com

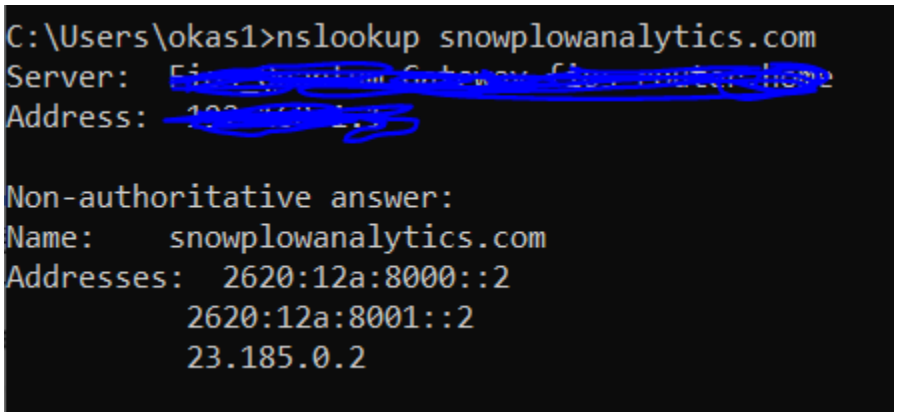
Where is the domain name snowplowanalytics.com hosted

- Hosted with AWS with the name servers
- NS-1292.AWSDNS-33.ORG,
- NS-2027.AWSDNS-61.CO.UK,
- NS-54.AWSDNS-06.COM,
- NS-653.AWSDNS-17.NET

## WHOIS search results

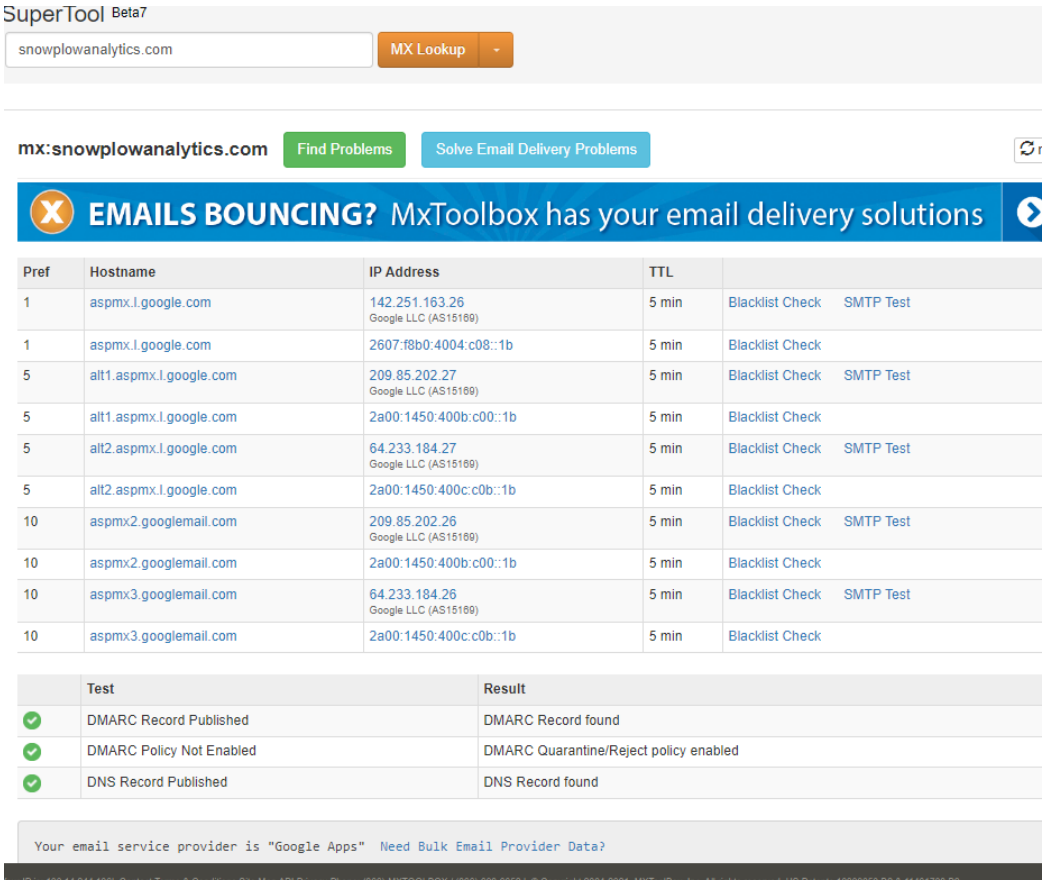
Domain Name: SNOWPLOWANALYTICS.COM
Registry Domain ID: 1707175393_DOMAIN_COM-VRSN
Registrar WHOIS Server: whois.ionos.com
Registrar URL: http://www.ionos.com
Updated Date: 2023-03-15T07:11:14Z
Creation Date: 2012-03-14T11:07:15Z
Registry Expiry Date: 2024-03-14T11:07:15Z
Registrar: IONOS SE
Registrar IANA ID: 83
Registrar Abuse Contact Email: abuse@ionos.com
Registrar Abuse Contact Phone: +1.6105601459
Domain Status: clientTransferProhibited
https://icann.org/epp#clientTransferProhibited
Name Server: NS-1292.AWSDNS-33.ORG
Name Server: NS-2027.AWSDNS-61.CO.UK
Name Server: NS-54.AWSDNS-06.COM
Name Server: NS-653.AWSDNS-17.NET
DNSSEC: unsigned

What is the IP address of the web server? Explain the result of your investigation.




IPv4 - 23.185.0.2
IPv6 - 2620:12a8000::2 and 2620:12a:8001::2

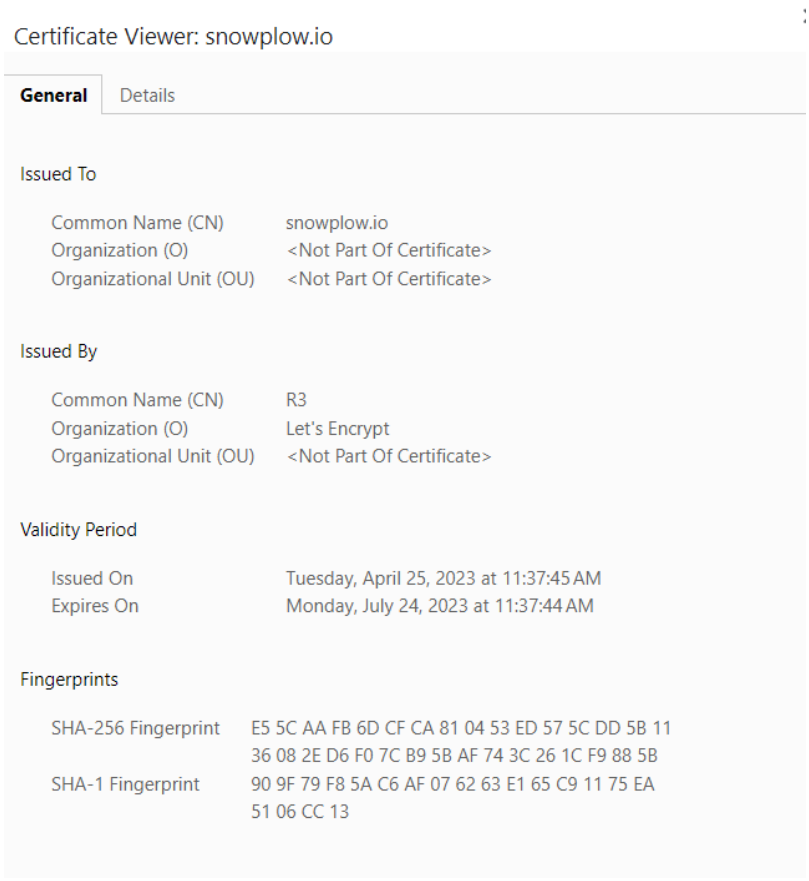
What company manages emails for the domain snowplowanalytics.com
According to mx lookup its google apps




SSL Certificates

Who issued the SSL certificate installed for https://snowplow.io/ ?

Issued by lets encrypt




When does the certificate expire?
Expires on july 24 2023

Is the certificate valid if installed for the website https://discourse.snowplow.io/ ? Why?

Yes because the dates are in the same/similar range and they're part of the same org and it looks like a wildcard certificate

Issued To

| | |
|---|---|
| Common Name (CN) | discourse.snowplow.io |
| Organization (O) | <Not Part Of Certificate> |
| Organizational Unit (OU) | <Not Part Of Certificate> |

Issued By

| | |
|---|---|
| Common Name (CN) | R3 |
| Organization (O) | Let's Encrypt |
| Organizational Unit (OU) | <Not Part Of Certificate> |

Validity Period

| | |
|---|---|
| Issued On | Thursday, April 27, 2023 at 8:00:13 PM |
| Expires On | Wednesday, July 26, 2023 at 8:00:12 PM |

Fingerprints

| | |
|---|---|
| SHA-256 Fingerprint | 9A C1 98 AE C6 C5 97 F8 C4 B2 1D C4 FC AC DC E8 36 92 FC 1F E9 29 BE 48 19 0A F8 8B 25 09 63 D4 |
| SHA-1 Fingerprint | E3 6D 19 13 24 5D BE C8 12 02 72 B8 C4 9D 43 84 53 E3 FB D4 |

Are the certificates installed with https://snowplow.io/ **and https://discourse.snowplow.io/ **the same? Why?

- No because the common name differs and they have different providers

**Structured Query Language (SQL)**

Count all the records that have the value "page_view" in the field event of the 'events' table in the 'atomic' schema.

There are 2 tables which share the same field 'id'. Select the first 100 records from both tables, 'events' and 'contexts', where the values of the field 'id' match.

Copy records for which 'timestamp' field equals "2017-10-05 00:11:54" from 'events' table into table 'events_new' in 'atomic' schema.

JSON

Fix the below JSON. List each correction.

Scripting Languages

```
{
    "id": "12345",
    "type": "event",
    "name": “click“,
    "12345"; {
}
}
`primary: true,
"image":`
```

```
    `{
        "url": "images/0001.jpg",
        "width": 200,
        "height": 200,`

    `},
"thumbnail":`

    `{
        "url": "images/thumbnails/0001.jpg",
        "width": 32,
        "height": 32`

    `},
"root": 'acme.com'
}`

Here is the corrected json
{
    "id": "12345",
    "type": "event",
    "name": "click",
    "12345": {
        "primary": true,
        "image": {
            "url": "images/0001.jpg",
            "width": 200,
            "height": 200
        },
        "thumbnail": {
            "url": "images/thumbnails/0001.jpg",
            "width": 32,
            "height": 32
        },
        "root": "acme.com"
    }
}
```

```
the quotation marks after "name" are incorrect it should be name:
"click"
12345 has the wrong type of colon it should be "12345": {}
12345 can't have empty brackets it should either be removed used to
store the part that has primary
then primary should have double quotes instead of single quotes
image shouldn't have single quotations
height shouldn't have the comma if its the last item in that list
we need to get rid of all of the extraneous single quotes (`)
the height in thumbnail has an extra single quote
acme.com should have double quotes
There seems to be an error around type where there are two types of
commas that are visually similar are treated very different by
javascript one of them is a (U+201A) comma which is not valid in json
the one we want to use is a (U+002C) comma not sure if this is
intentional but an interesting bug
```

JSON: http://jsonplaceholder.typicode.com/users
Using one of the scripting languages below print out the list of the cities.
bash
Python
Ruby

```
<html>
<head>
  <title>Snowplow</title>
</head>
<body>
    <button>Click me</button>
</body>
</html>
```

Insert the AlertifyJS library; it is used to generate dialogs and can be found at
** https://cdn.jsdelivr.net/npm/alertifyjs@1.13.1/build/alertify.min.js**
Insert CSS file hosted at
** https://cdn.jsdelivr.net/npm/alertifyjs@1.13.1/build/css/alertify.min.css**
The AlertifyJS library defines a simple method, alertify.alert(message) where the message is

a string representing the message displayed by the alert box. Add Javascript code to pop up an AlertifyJS’s alert box with the message “Well done!” on the button click.