Here are two problems that you may try to solve using any tech stack of your choosing. The works should be uploaded to individual git repos and sent as a reply in the email. Problem 1 is mandatory. The latter one is to show off your web skills. You may opt-out without any hesitation. For any kind of queries, please contact us at nahid.rahman@dsinnovators.com. The email subject should include your name, the position you are applying for, and the date you have been given this assignment on.

Problem 1

You want to keep track of the pieces of art (series, movies, books) that you consume. You are cringingly self-conscious about your productivity. So, you need to keep a ledger for all the time you spend doing this. When you run the program, it will be able to accept command line inputs. And, will show the result in the command line. When engaging with the command line, it will give you options to

1. Add a consumable
    a. There will be 3 types. Books, series, or movie. You will have to choose one.
    b. It should have this information
        i. Name
        ii. Consumption starting date in the format YYYY-MM-DD (could be blank)
        iii. Consumption ending date in the same format (could be blank)
        iv. Total consumption time in hours (will be 0 by default at first)
        v. Your personal rating X.YZ out of 10 (could be blank)
        vi. Total days of consumption (will be 0 by default at first)

2. Edit a consumable
    a. When a product is being edited. You will have the option to add a time in hrs which will be added to the total consumption time.
    b. While adding consumption time you will have an option to add a day to days of consumption.
    c. You can change the rating.
    d. You can also add the consumption ending date. But, once added, the consumable can not be edited anymore.

3. Delete a consumable
    a. When a consumable is deleted, it will not show in the list but the times and days added because of it will still be there in total consumption time and days.

4. See the list of consumables and individually
    a. If you choose a specific type, the list will include the name of the consumable, total days of consumption, total hrs of consumption, rating. Showing these in a tabular form is not mandatory. But, anyone would love that, right?
    b. You can pick one and see the full details.

5. See overall info
    a. The total consumption time in hours across all types
    b. Individual consumption time in hours of each type
    c. The total days of consumption across all types
    d. Individual days of consumption of each type
    e. Average rating across all types
    f. Average individual rating of each type
    g. Total number of consumable across all types
    h. Individual number of consumable of each type

After creating the application, assume you have nothing at the moment. Then add 10 consumables with at least 2 from each type. Edit some of them so that the overall info can be produced visibly.

Problem 2 (optional)

Your friend has an idea of a company, he wants your help building its website. Using the website a customer will see specific products which are actually available on Amazon, can add them to their cart, pay the price of those products in BDT and get them for a little extra as a service charge. You need not build the whole website. What you will build is

1. A usable link in localhost that'll take the browser to a webpage.
2. That webpage will consist of 10 products with name, price in BDT, and image.
3. If an image is clicked the browser will take you to that product on the Amazon website. The products can be anything of your liking.

As we might not be able to run every project sent to us, screenshots of the webpage should be included in a folder of the repo.

Bonus: To make you excited, there are also some bonus factors. These are not mandatory. But if you have it, it will act as some bonus points for you.

These are

1. Responsive UI
2. Framework
3. Dependency Management
4. Database
5. Object Relation Mapping
6. Pull the images directly from amazon
7. Sortability
8. Searchability
9. A suitable look for the use case

You can use any Language you want to build the application. It's totally up to you.
Languages Preferred: JAVA, JavaScript(NodeJS Environment), Python.
Using your work, we aim to judge your ability and concept knowledge in OOP. We want to see how you understand the use case and structure your code efficiently.
Enjoy making systems. Wish you good luck.