

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 students that you teach. You are cringingly self-conscious about your productivity. So, you need to keep a ledger for all the information about these students. 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 student

- a. There will be 3 classes. Class 8, 9, or 10. You will have to choose one.
- b. It should have this information
 - i. Name
 - ii. A list of booleans of what subjects you teach. Math, English, Or Bangla
 - iii. Average of marks s/he get in the exam. (Will be 0 by default at first)
 - iv. Total days taught (Will be 0 by default at first)
 - v. Total earnings (Will be 0 by default at first)

2. Edit a student

- a. Adding days to the number of days taught. For every subject per day, you get BDT 1.
- b. Adding marks in exam s/he gets as an integer from 0 to 100. These marks will change the average.

3. Delete a student

- a. When a student is deleted, it will not show in the list but the earnings added because of him/her will still be there in total earning.

4. See the list of students individually

- a. If you choose a specific class, the list will include the name of the students, total earnings, average marks. 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 days taught across all classes
 - b. Individual days taught in each class
 - c. Total earnings
 - d. Individual earnings of each class
 - e. Individual earnings of each subject
 - f. Average marks of all students

After creating the application, assume you have nothing at the moment. Then add 10 students. 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.