**General Assessment Question**

1. **What will be your top 3 priorities for this year?**

**Answer:** As I am an undergraduate student who is still pursuing degree,

* My 1st priority will be Completing my studies with a satisfying academics.
* My 2nd primary priority is to start an internship and learn what goes on real world work as our education system and work market has a huge gap, I am eager to know and get myself to a state where I earn an position on a company and become professionally independent and financially independent.
* My 3rd priority is to stay determined on my work ethics and my work, not to forget take a good care of my health as we are amidst the current pandemic and we know if health is there only then other things exists.

1. **What action steps will you take to achieve your top 3 priorities?**

**Answer:** Our life is based on accomplishing the fundamental priorities. As we know a human strives towards achieving what he/she wants on life. So, it’s same on my case my priorities are set. Now in a path towards achieving it,

* As, I am currently taking my classes, doing my assignments and giving exams, these are the things one would do to complete his/her studies. But the thing is how much honestly and determined he/she is acting to perform these things. So, I’ll be focused and along with my professional life that’s in initiative phase I’ll be honest with myself and self-determined towards my studies.
* I am starting to learn JavaScript, its libraries, frameworks and .NET, C# for grooming myself to be a suitable candidate for internship. I have also started to do a self-learning amateur project for improving myself towards the foundation of programming and developing applications be web or desktop.
* My 3rd priority which is to be mentally and physically able to solve any problems that confronts me. For my mental health I perform exercise in weekly basis and refresh myself time to time which makes me always energized, determined and ready to stay on the correct path not to forget following conventions and precautions.

1. **What challenges do you anticipate to successfully achieve your top 3 priorities?**

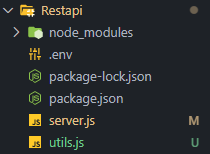
**Answer:** Telling about challenges yes, I have expected some which will hopefully I’ll be able to tackle.

* Time management: This is I think will be the hardest one. To manage time for classes, my extra course and all.
* Creativity: Not boasting myself and telling honestly everyone chokes somehow at solving problems and creating solutions which needs creativity so I think somehow this would also be one challenge for me.
* Distractions: To stay focused for hours and hours would be good but somehow, we have so many distractions presently.

Hope, I can overcome all these challenges gradually and achieve my goals for this year.

**Technical Skill Assessment Question**

**2) Develop an API for login with username and password with necessary [10] security handing like token verification(static) etc.**

****Source code:**

*require('dotenv').config();*

*const express = require('express');*

*const cors = require('cors');*

*const bodyParser = require('body-parser');*

*const jwt = require('jsonwebtoken');*

*const utils = require('./utils');*

*const app = express();*

*const port = process.env.PORT || 4000;*

*// static user details*

*const userData = {*

*password: "aayush123",*

*name: "Aayush Dura",*

*username: "duraaayush",*

*isAdmin: true*

*};*

*// enable CORS*

*app.use(cors());*

*// parse application/json*

*app.use(bodyParser.json());*

*// parse application/x-www-form-urlencoded*

*app.use(bodyParser.urlencoded({ extended: true }));*

*//middleware that checks if JWT token exists and verifies it if it does exist.*

*//In all future routes, this helps to know if the request is authenticated or not.*

*app.use(function (req, res, next) {*

*// check header or url parameters or post parameters for token*

*var token = req.headers['authorization'];*

*if (!token) return next(); //if no token, continue*

*token = token.replace('Bearer ', '');*

*jwt.verify(token, process.env.JWT\_SECRET, function (err, user) {*

*if (err) {*

*return res.status(401).json({*

*error: true,*

*message: "Invalid user."*

*});*

*} else {*

*req.user = user; //set the user to req so other routes can use it*

*next();*

*}*

*});*

*});*

*// request handlers*

*app.get('/', (req, res) => {*

*if (!req.user) return res.status(401).json({ success: false, message: 'Invalid user to access it.' });*

*res.send('Hey this is my first API' + req.user.name);*

*});*

*// validate the user credentials*

*app.post('/login', function (req, res) {*

*const user = req.body.username;*

*const pwd = req.body.password;*

*// return 400 status if username/password is not exist*

*if (!user || !pwd) {*

*return res.status(400).json({*

*error: true,*

*message: "Username or Password required."*

*});*

*}*

*// return 401 status if the credential is not match.*

*if (user !== userData.username || pwd !== userData.password) {*

*return res.status(401).json({*

*error: true,*

*message: "Username or Password is Wrong."*

*});*

*}*

*// generate token*

*const token = utils.generateToken(userData);*

*// get basic user details*

*const userObj = utils.getCleanUser(userData);*

*// return the token along with user details*

*return res.json({ user: userObj, token });*

*});*

*// verify the token and return it if it's valid*

*app.get('/verifyToken', function (req, res) {*

*// check header or url parameters or post parameters for token*

*var token = req.body.token || req.query.token;*

*if (!token) {*

*return res.status(400).json({*

*error: true,*

*message: "Token is required."*

*});*

*}*

*// check token that was passed by decoding token using secret*

*jwt.verify(token, process.env.JWT\_SECRET, function (err, user) {*

*if (err) return res.status(401).json({*

*error: true,*

*message: "Invalid token."*

*});*

*// get basic user details*

*var userObj = utils.getCleanUser(userData);*

*return res.json({ user: userObj, token });*

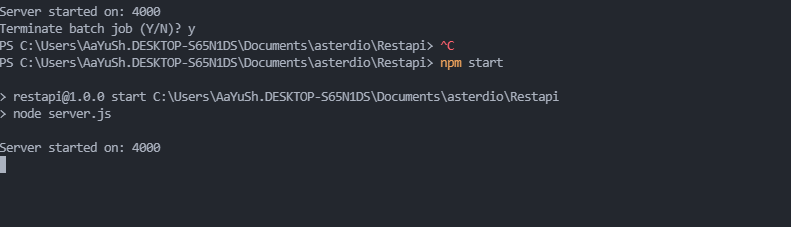
*});*

*});*

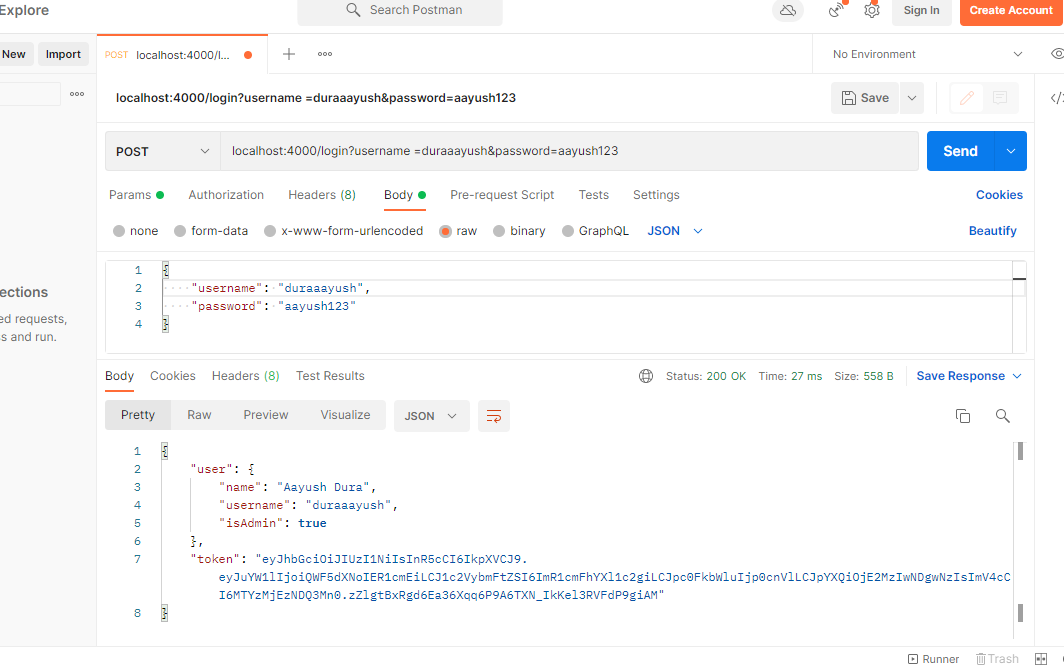
*app.listen(port, () => {*

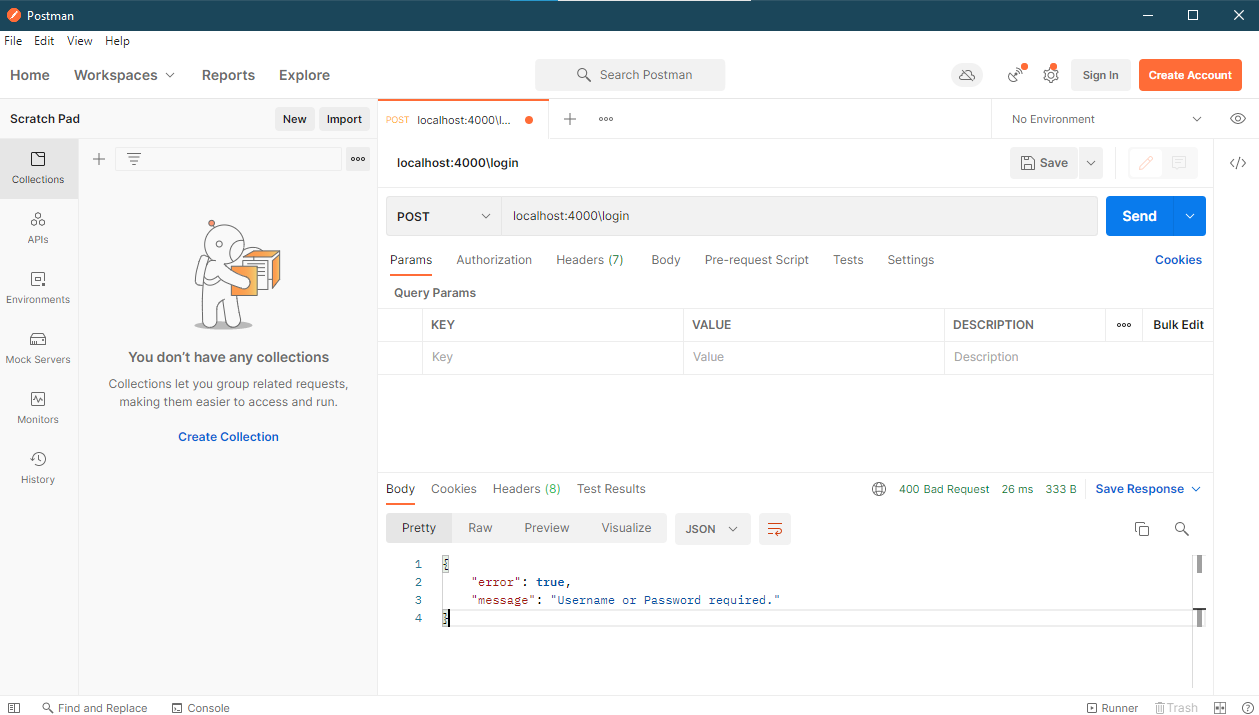
*console.log('Server started on: ' + port);*

*});*

  
**Terminal in VS code:**

**API testing using Postman.**





**4) String Reduction [20]**

**Have the function StringReduction(str) take the str parameter being**

**passed and return the smallest number you can get through the following reduction**

**method. The method is: Only the letters a, b, and c will be given in str and you must**

**take two different adjacent characters and replace it with the third. For example "ac"**

**can be replaced with "b" but "aa" cannot be replaced with anything. This method is**

**done repeatedly until the string cannot be further reduced, and the length of the**

**resulting string is to be outputted.**

**For example: if str is "cab", then "ca" can be reduced to "b" and you get "bb"**

**(you can also reduce it to "cc"). The reduction is done so the output should be 2. If**

**str is "bcab", "bc" reduces to "a", so you have "aab", then "ab" reduces to "c", and**

**the final string "ac" is reduced to "b" so the output should be 1.**

**Examples**

**Input: "abcabc"**

**Output: 2**

**Input: "cccc"**

**Output: 4**

**Source code:**

*function StringReduction(str) {*

*var i = 0;*

*var letters = str.split("");*

*do {*

*if ((letters[i] == "a" && letters[i + 1] == "b") || (letters[i] == "b" && letters[i + 1] == "a")) {*

*letters.splice(i + 1, 1);*

*letters[i] = "c";*

*i = 0;*

*} else if ((letters[i] == "b" && letters[i + 1] == "c") || (letters[i] == "c" && letters[i + 1] == "b")) {*

*letters.splice(i + 1, 1);*

*letters[i] = "a";*

*i = 0;*

*} else if ((letters[i] == "c" && letters[i + 1] == "a") || (letters[i] == "a" && letters[i + 1] == "c")) {*

*letters.splice(i + 1, 1);*

*letters[i] = "b";*

*i = 0;*

*} else {*

*i++;*

*}*

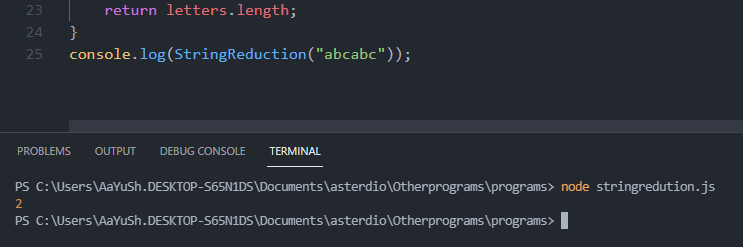
*} while (i < letters.length);*

*// Return the length of the transformed string*

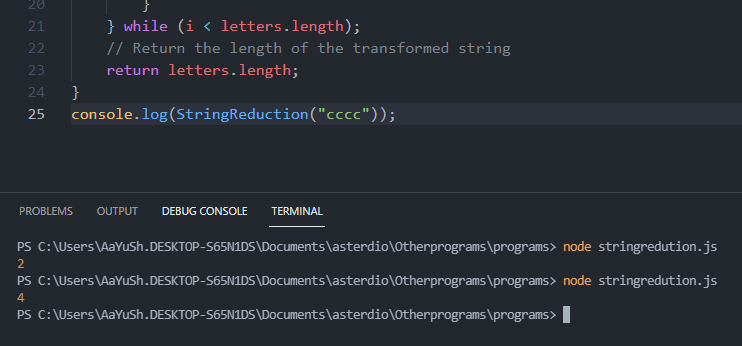
*return letters.length;*

*}*

*console.log(StringReduction("abcabc"));*

**Output:** for string str=”abcabc”

**Output:** for string str= ”cccc”

****

**8. How are duplicate nodes removed in an unsorted linked list?**

**Source code:**

*class node {*

*constructor(val) {*

*this.val = val;*

*this.next = null;*

*}*

*}*

*/\**

*Function to remove duplicates*

*from a unsorted linked list*

*\*/*

*function removeDuplicate( head) {*

*// Hash to store seen values*

*var hs = new Set();*

*/\* Pick elements one by one \*/*

*var current = head;*

*var prev = null;*

*while (current != null) {*

*var curval = current.val;*

*// If current value is seen before*

*if (hs.has(curval)) {*

*prev.next = current.next;*

*} else {*

*hs.add(curval);*

*prev = current;*

*}*

*current = current.next;*

*}*

*}*

*/\* Function to print nodes in a*

*given linked list \*/*

*function printList( head) {*

*while (head != null) {*

*console.log(head.val + " ");*

*head = head.next;*

*}*

*}*

*start = new node(10);*

*start.next = new node(12);*

*start.next.next = new node(11);*

*start.next.next.next = new node(11);*

*start.next.next.next.next = new node(12);*

*start.next.next.next.next.next = new node(11);*

*start.next.next.next.next.next.next = new node(10);*

*console.log(*

*"Linked list before removing duplicates :"*

*);*

*printList(start);*

*removeDuplicate(start);*

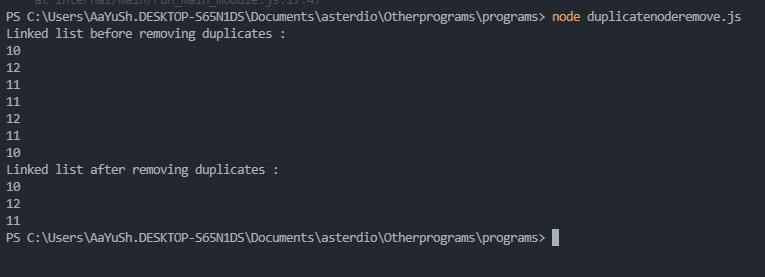
*console.log(*

*"Linked list after removing duplicates :"*

*);*

*printList(start);*

**Output:**

****

**7. How do you find the missing number in a given integer array of 1 to 100?**

**Source code:**

*//defining array containing 1 to 100*

*//as we need to find missing number from an array of 1 to 100*

*var n = 100;*

*var naturalsum = (n \* (n + 1)) / 2;*

*//defining array containing a missing element '20'*

*var arr = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10,*

*11, 12, 13, 14, 15, 16, 17, 18, 19,*

*21, 22, 23, 24, 25, 26, 27, 28, 29, 30,*

*31, 32, 33, 34, 35, 36, 37, 38, 39, 40,*

*41, 42, 43, 44, 45, 46, 47, 48, 49, 50,*

*51, 52, 53, 54, 55, 56, 57, 58, 59, 60,*

*61, 62, 63, 64, 65, 66, 67, 68, 69, 70,*

*71, 72, 73, 74, 75, 76, 77, 78, 79, 80,*

*81, 82, 83, 84, 85, 86, 87, 88, 89, 90,*

*91, 92, 93, 94, 95, 96, 97, 98, 99, 100];*

*console.log(naturalsum);*

*var missingsum = 0;*

*for (let i = 0; i < arr.length; i++) {*

*missingsum += arr[i];*

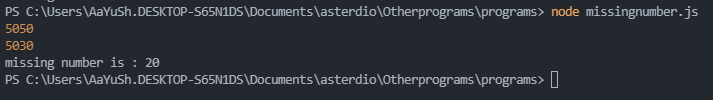
*}*

*console.log(missingsum);*

*var missingnumber = naturalsum - missingsum;*

*console.log('missing number is : ' + missingnumber);*

**Output:**

****

**9. How do you check if two rectangles overlap with each other?**

**Source Code:**

*class rectangle*

*{constructor(x, y)*

*{*

*this.x=x;*

*this.y=y;*

*}*

*}*

*function overlapcheck(l1,l2,r1,r2)*

*{*

*if(l1.x>=r2.x||l2.x>=r1.x){*

*return false;*

*}*

*else if(r1.y>=l2.y||r2.y>=l1.y){*

*return false;*

*}*

*return true;*

*}*

*var l1=new rectangle(20,50);*

*var r1=new rectangle(30,30);*

*var l2=new rectangle(10,20);*

*var r2=new rectangle(40,10);*

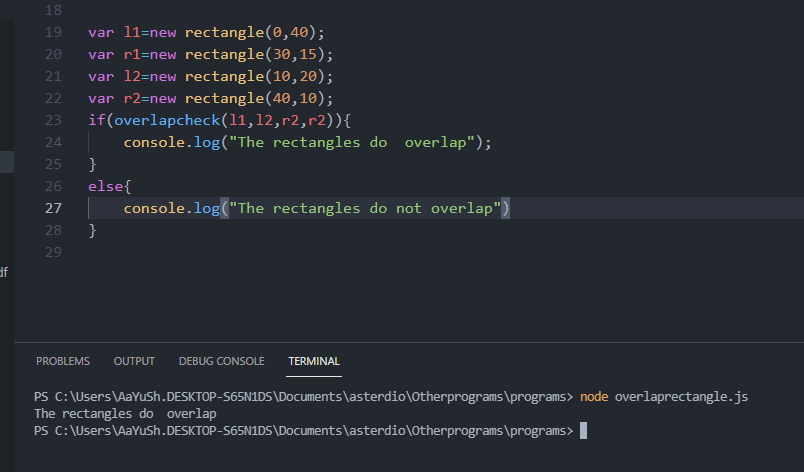
*if(overlapcheck(l1,l2,r2,r2)){*

*console.log("The rectangles do  overlap");*

*}*

*else{*

*console.log("The rectangles do not overlap")*

*****}*

**10. How do you design a vending machine? (Pseudocode only)?**

**Pseudocode for a vending machine:**

***function Welcomescreen()*** *//initial display and action of machine*

***{***

***Display("Hi there\n \*\*\*\*\*\*\*Quench your thirst\*\*\*\*\*\*\*");***

***vendingmachine();***

***}***

***function vendingmachine() {***

***var amount;*** *//price of respective drinks*

***Displaychoices();*** *//action to display all the available drinks and their types and size.*

***GetUserschoices()***

***{***

***choice = prompt('What are you on a mood for drinking?');***

***}***

***Displaycost(choice)***

***{***

***Case(choice)***

***{***

***Display("Please pay " + amount + "For" + choice + "drink");***

***}***

***}***

***Getpayment()***

***{***

***pay = prompt('Please enter coins or change if possible:');***

***if (pay > amount) {***

***Return(pay - cost);***

***}***

***else {***

***providedrink();***

***}***

***}***

***providedrink(choice, amount)***

***{***

***DisplayReciept(amount, choice);***

***ejectdrink(choice);*** *//action to provide drink to customer*

***}***

***}***