Write_Up_for capston:

Prerequisite- First we need to set up running web page of flyaway website in Apache tomcat server . All have been set up by our instructor itself in live class. Then basically we have to do mainly three things inside this project. 1.End-to-end Automation of flyaway website with the help of java , selenium and unit testing with the help of testng .Also running the testng class with pom.xml then Jenkins server.

2.performace testing with jmeter . 3.API testing with the help of postman and Rest-Assured (inside Eclipse). First let's light on How we have do End to End Automation inside Eclipse and then Unit Testing with the help of Testng. After we have fully running tomcat server and inside it flyaway website, it has some property by which if we signup it directly inserted inside table name f_user of Flyaway Database(we not need to focus it is done , this is not the aim of project). We are staring from Flyaway class . Inside exp_details function we are first creating one instance of chrome

website then opening http://localhost:8080/FlyAway/ Apache tomcat server link for flyaway website , then we click on signup and before that whatever we enter to create account we are first storing inside variable then we are passing variable inside sendkeys , we are doing so that we can validate the details later with the help of these variable. So , from this function we return array of string consists of user details. Now inside Login_SearchFlight_Validate function , first we are login with the help of details we have created inside above function, after login we are clicking Home button ,there we can see table consisting of list of flight details , now we are storing data from table inside a string , then we are selecting source and destination from dropdown menu and also we are storing these value in a variable so that we can verify later, after we have selected source and destination then we click on search button , if those flight are in our database after searching we can see only those flights details in next page , as we can see inside screenshot we are able to see that flight only that we have selected , now again there is table , again we are storing

value of table inside array , now we are ready to validate whatever we are looking for we are getting the same flight details , for that we are using two if-else condition and also we are returning int from this function , first we can see int ret initialize as 0 and if the source is validated then we are increasing the value by again and same we are doing for destination, later inside assert function of testng class we are checking whether they are equal to 2 or not.

Afte this we have Book_Show_and_validate where ae are clicking on book flight then click to complete booking and finally see your booking . After that we can list of booking we have done so far inside table. Now we are again storing value of table inside array and again we are validating the booking is same as we expected. Here , again we are doing same as previous function we are returning int ,if source and destination matches we are incrementing the value of that value by 1. This is all we are doing in Flyaway class. Now let's see what is happening inside JDBC class . It consists of only one function i.e. jdbc , inside this function we are connecting to database with the help of

JDBC(java database connection) and running query

Select * from f_user where name ='Lokesh JD'

, and then fetching details of user inside variables , and we have already created array of string , inside this array we are first inserting all variables and at the end we are returning this array , so that later we can validate whatever we have enter details while signup , it is same in database. Now Let's see what is happening inside Validate class , It has 3 functions and all functions are returning int only .

1.Validate_DB : inside this function we first calling two array of string that we have created inside above class . one that array that consists of user details while signup and another that consists of user details from database, also I'm maintained the order of both array same i.e. first index will have name only for both array so that we can validate with the help of one loop only , now after calling we stored both array inside a variable and then we run a loop and as we have declared a int ret again , initialize its value to 0 and if the values

from arrays matches we are changing value of int ret to 1 else it will remain same . and at the end we are returning this int ret so that we can use later inside testng class.

2.Validate_Table:inside this we are validating tables , we are calling desired function from above class and as those function return int as we already seen above , here again we are returning same int, we are doing nothing special here. 3.BookShowValidate: again we are calling function , the will return int , and we return same int from this function also , same as above function. This is all we are doing inside this class. Now , testrunn Class is testng class , we are running the entire project from this class only. Earlier I tried to put some set of automation inside beforemethod but as every function require different webpage to start i.e. why I decided to code for special function for every functionality, Now , there is 3 test we are performing inside this class. Blueprint to execute every test case is same . as we can see we are importing all function of Validate class inside these testcases ,as these function will retirn int , in case of Test testcase

function ,we are validating database with the help of assertequal , if the function return 1 then our test cas