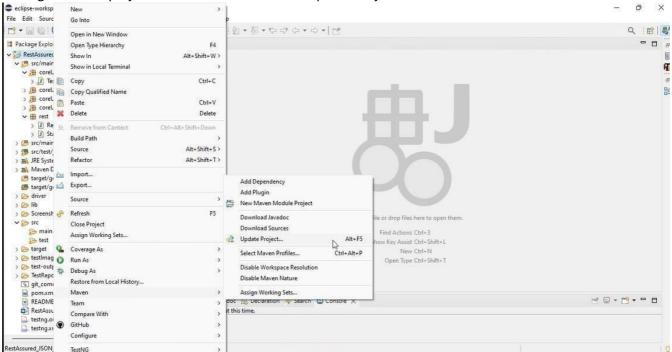
RESTASSURED API AUTOMATION PROJECT

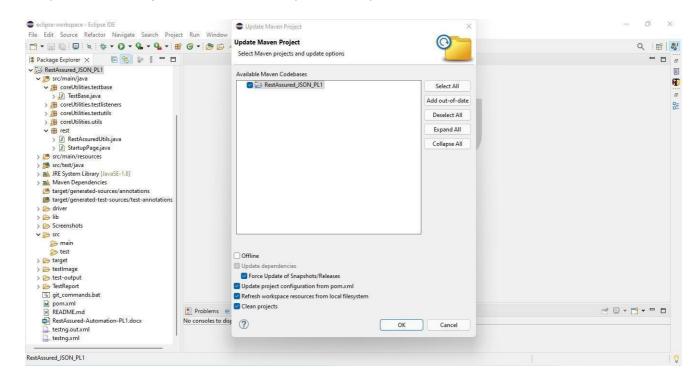
Pre-requisite:

As soon as you import a project in eclipse, update the project using the maven update option as below. This is to resolve issue if any maven dependency not downloaded properly:

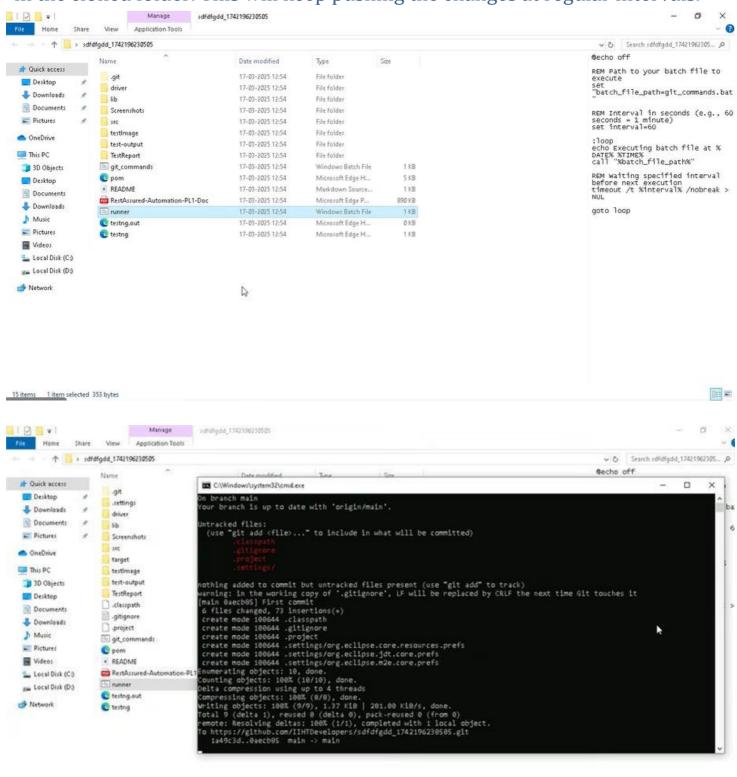
1. Right click on project : Go to "Maven" : Select "Update Project"



2. In Update Maven Project Box Select "Force Update of Snapshots/Releases" and click OK



** Before starting any implementation. Please execute this "runner" file present in the cloned folder. This will keep pushing the changes at regular intervals.



30 James 1 Jam referted 352 livites

Red to

Template Code Structure:

- a. Below are the packages and files you will be required to work upon.
- b. Other Files and packages you can ignore.
- c. In other Files and packages do not do any changes. It would affect your evaluation.
- d. You are not required to work in "Test" Folder. Files there are noneditable. Editing those files and trying to save them will throw error and would affect your evaluation.

Package	Class/File	Description
/src/main/java/rest	ApiUtil.java	All core activities to be performed here.
		2. The comments associated
		with each templated method here describe the expectation.
		3. Declare any variable/object you need to share data/status between different methods.
		4. Do not modify the signature of methods declared here.
		5. You can create additional supportive common methods in CommonEvents class.
/src/main/java/rest	AuthUtil.java	Class already defined to read and return bearer token from config.properties file.
/src/main/java/rest	CustomResponse	We have already created custom response object class. You need to
		use this and use it as return type from ApiUtil.java class methods.
/src/main/resources/	Config.xlsx	Data present to be used in
		Implementing functions.
/src/main/java/coreUtilities/utils	CommonEvents.java	 Contains all common activities. Certain templated
		common methods

	declared here. 3. You implement them as per your need. 4. You can add any additional method for common
Testng.xml	activity here Execution needs to kick started
	from TestNG xml

PROBLEM STATEMENT

Need to automate the following activities using RestAssured.

Key Activities to implement:

Below activities need to be implemented in the ApiUtil.java file present in src/main/java/rest package.

SI	Summary	Action	Expected Result
No.			
1	Retrieve all departments in Method: getAllDepartments(String endpoint, Object body)	1.Create an URL by combining the BASE_URL (already declared) and path parameter (provided as an argument in method). The final URL becomes: https://healthapp.yaksha.com/a pi/AssetReports/GetAllDepartm ents 2. Include a bearer token for authentication in authorization header. 3.Trigger a GET call to the specified endpoint. 4.Create an object of type CustomResponse and initialize it with values extracted from the Response object: Response statusCode (extracted from Response object) DepartmentId list (extracted from Response object) DepartmentName list (extracted from Response object) Status (extracted from Response object) Status (extracted from Response object) Status (extracted from Response object)	- Returns an object of type CustomResponse containing statusCode, status, DepartmentId list, DepartmentName list, and the complete Response object. - StatusCode should be 200. - Status should be OK. - DepartmentId and DepartmentName lists should not be empty and should not be null.
2	Retrieve all the list of items in Method: getAllItems(String endpoint, Object body)	 1.Create an URL by combining the BASE_URL (already declared) and path parameter (provided as an argument in method). The final URL becomes: https://healthapp.yaksha.com/a pi/AssetReports/GetAllItems 2. Include a bearer token for authentication in authorization header. 3.Trigger a GET call to the specified endpoint. 4.Create an object of type CustomResponse and initialize it with values extracted from the Response object: Response statusCode (extracted from Response object) ItemId list (extracted from 	- Returns an object of type CustomResponse containing statusCode, status, ItemId list, ItemName list, and the complete Response object. - StatusCode should be 200. - Status should be OK. - ItemId list and ItemName list should not be empty.

		Response object) ItemName list (extracted from Response object) Status (extracted from Response object) Response object) Return the CustomResponse object from the method.	
3	Retrieve incentive summary report in Method: getIncentiveSummaryReport(String URL, Object body)	1.The final URL is provided as a parameter (URL) and that is: https://healthapp.yaksha.com/B illingReports/INCTV_DocterSum mary?FromDate="+ fromDate + "&ToDate=" + toDate + "&IsRefferalOnly=" + isRefferal 2. Include a bearer token for authentication in authorization header. 3.Trigger a GET call to the specified endpoint. 4.Create an object of type CustomResponse and initialize it with values extracted from the Response object: Response statusCode (extracted from Response object) PrescriberId list (extracted from Response object as a list) PrescriberName list (extracted from Response object as a list) DocTotalAmount list (extracted from Response object as a list) TDSAmount list (extracted from Response object as a list) NetPayableAmount list (extracted from Response object as a list) Status (extracted from Response object as a list) Status (extracted from Response object as a list)	- Returns an object of type CustomResponse containing statusCode, status, PrescriberId list, PrescriberName list, DocTotalAmount list, TDSAmount list and the complete Response object. - StatusCode should be 200. - Status should be OK. - PrescriberId list, PrescriberName list, DocTotalAmount list and NetPayableAmount list should not be empty and should not be null.

4	Retrieve incentive referral summary report in Method:	1.The final URL is provided as a	- Returns an object of type
	, ,	parameter (URL) and that is:	CustomResponse
	getIncReffSummReport(String URL, Object body)	https://healthapp.yaksha.com/B	containing statusCode,
	OKL, Object body)	illingReports/INCTV_DocterSum	status, PrescriberId list,
		mary?FromDate="+ IncFromDate	PrescriberName list,
		+ "&ToDate=" + IncToDate +	DocTotalAmount list,
		"&IsRefferalOnly=" + isRefferal	TDSAmount list,
			NetPayableAmount list and
		2. Include a bearer token for	the complete Response
		authentication in authorization	object.
		header.	
		3.Trigger a GET call to the specified endpoint.	- StatusCode should be 200.
		4.Create an object of type	- Status should be OK.
		CustomResponse and initialize it	
		with values extracted from the	- PrescriberId list,
		Response object:	PrescriberName list,
		Response	DocTotalAmount list,
		 statusCode (extracted from 	TDSAmount list and
		Response object)	NetPayableAmount list
		 PrescriberName list (extracted 	should not be empty
		from Response object as a list)	and should not be null.
		 PrescriberId list (extracted from 	and should not be nam.
		Response object as a list)	
		DocTotalAmount list (extracted)	
		from Response object as a list)	
		TDSAmount list (extracted from	
		Response object as a list)	
		NetPayableAmount list	
		(extracted from Response	
		object as a list)	
		Status (extracted from	
		Response object)	
		5. Return the CustomResponse object	
		from the method.	
5	Retrieve hospital income	1.The final URL is provided as a	- Returns an object of type
	incentive report in Method:	parameter (endpoint) and that is:	CustomResponse
		https://healthapp.yaksha.com/R	containing statusCode,
	getHospIncIncReport(String	eporting/HospitalIncomeIncentiv	status,
	endpoint, Object body)	eReport?FromDate="+	ServiceDepartmentId list,
		IncFromDate + "&ToDate=" +	ServiceDepartmentName
		IncToDate +	list, HospitalNetIncome list,
		"&ServiceDepartments=" +	NetSales list,
		ServiceDepartments	OtherIncentive list,
			ReferralCommission list and
		2. Include a bearer token for	GrossIncome list and the
		authentication in authorization header.	complete Response object.
		3.Trigger a GET call to the specified endpoint.	- StatusCode should be 200.
		4.Create an object of type	- Status should be OK.
		CustomResponse and initialize it	
		with values extracted from the Response object:	- ServiceDepartmentId

Response object:

		Response	list,
		 statusCode (extracted from Response object) 	ServiceDepartmentNa me list,
		 ServiceDepartmentId list 	HospitalNetIncome list,
		(extracted from Response	NetSales list,
		object as a list)	OtherIncentive list,
		 ServiceDepartmentName list 	ReferralCommission
		(extracted from Response	list and GrossIncome
		object as a list)	list should not be
		NetSales list (extracted from	empty and should not
		Response object as a list)	be null.
		ReferralCommission list (outrasted from Baseause)	
		(extracted from Response object as a list)	
		GrossIncome list (extracted)	
		from Response object as a list)	
		OtherIncentive list (extracted)	
		from Response object as a list)	
		 HospitalNetIncome list 	
		(extracted from Response	
		object as a list)	
		Status (extracted from	
		Response object)	
		5. Return the CustomResponse object	
6	Retrieve employee bill	from the method. 1.Create an URL by combining the	- Returns an object of type
0	items in Method:	BASE_URL (already declared) and	CustomResponse
	items in Method.	path parameter (provided as an	containing statusCode,
	getEmpBillItem(String	argument in method). The final	status,
	endpoint, Object body)	URL becomes:	EmployeeIncentiveInfold
		https://healthapp.yaksha.com/a	object, Employeeld object,
		pi/Incentive/EmployeeBillItems?	IsActive object, FullName
		employeeld=" + employeeld	object, TDSPercent object,
			EmpTDSPercent object,
		2. Include a bearer token for	EmployeeBillItemsMap list,
		authentication in authorization	and the complete Response
		header. 3.Trigger a GET call to the specified	object.
		endpoint.	Chatrac Carda ah a vildila 200
		4.Create an object of type	- StatusCode should be 200.
		CustomResponse and initialize it	- Status should be OK.
		with values extracted from the	Status should be oit.
		Response object:	- EmployeeIncentiveInfold
		Response	object, Employeeld object,
		statusCode (extracted from	IsActive object , FullName
		Response object)	object , TDSPercent object ,
		 EmployeeIncentiveInfold object (extracted from Response 	EmpTDSPercent object ,
		object)	EmployeeBillItemsMap list
		Employeeld object (extracted)	should not be null and
		from Response object)	should not be empty.
		FullName object (extracted	
		from Response object)	
		 TDSPercent object (extracted 	

		from Response object) • EmpTDSPercent object (extracted from Response object) • IsActive object (extracted from Response object) • EmployeeBillItemsMap list (extracted from Response object as a list) • Status (extracted from Response object) 5. Return the CustomResponse object from the method.	
7	Retrieve inventory fiscal years in Method: getInvntryFiscalYrs(String endpoint, Object body)	1.Create an URL by combining the BASE_URL (already declared) and path parameter (provided as an argument in method). The final URL becomes: https://healthapp.yaksha.com/a pi/Inventory/InventoryFiscalYear s 2. Include a bearer token for authentication in authorization header. 3.Trigger a GET call to the specified endpoint. 4.Create an object of type CustomResponse and initialize it with values extracted from the Response object: Response statusCode (extracted from Response object) FiscalYearId list (extracted from Response object as a list) FiscalYearName list (extracted from Response object as a list) StartDate list (extracted from Response object as a list) EndDate list (extracted from Response object as a list) IsActive list (extracted from Response object as a list) Status (extracted from Response object as a list) Status (extracted from Response object as a list)	- Returns an object of type CustomResponse containing statusCode, status, FiscalYearId list, FiscalYearName list, StartDate list, EndDate list, IsActive list and the complete Response object. - StatusCode should be 200. - Status should be OK. - FiscalYearId list, FiscalYearName list, StartDate list, EndDate list, IsActive list should not be null.

8	Activate inventory in Method:	1.Create an URL by combining the BASE_URL (already declared) and	- Returns an object of type CustomResponse
	getActInventory(String endpoint, Object body)	path parameter (provided as an argument in method). The final URL becomes: https://healthapp.yaksha.com/api/ActivateInventory/	containing statusCode, status, StoreId list, Name list, StoreDescription list and the complete Response object.
		 Include a bearer token for authentication in authorization header. Trigger a GET call to the specified endpoint. Create an object of type CustomResponse and initialize it with values extracted from the Response object: Response statusCode (extracted from Response object) Storeld list (extracted from Response object as a list) Name list (extracted from Response object as a list) StoreDescription list (extracted from Response object as a list) Status (extracted from Response object as object as a list) Status (extracted from Response object as object) Return the CustomResponse object from the method. 	 StatusCode should be 200. Status should be OK. StoreId list, Name list and StoreDescription list should not be empty and should not be null.
9	Retrieve inventory subcategories in Method: getInvSubCat(String endpoint, Object body)	1.Create an URL by combining the BASE_URL (already declared) and path parameter (provided as an argument in method). The final URL becomes: https://healthapp.yaksha.com/a pi/Inventory/SubCategories	- Returns an object of type CustomResponse containing statusCode, status, SubCategoryId list and SubCategoryName list, and the complete Response object.
		 Include a bearer token for authentication in authorization header. Trigger a GET call to the specified endpoint. Create an object of type CustomResponse and initialize it with values extracted from the Response object: Response statusCode (extracted from Response object) SubCategoryId list (extracted from Response object as a list) SubCategoryName list (extracted from Response 	 Status Code should be 200. Status should be OK. SubCategoryId list and SubCategoryName list should not be empty and should not be null.

		object as a list) • Status (extracted from Response object) 5. Return the CustomResponse object from the method.	
10	Retrieve available item quantity by store ID in Method: getAvlQtyByStoreId(String endpoint, Object body)	1.Create an URL by combining the BASE_URL (already declared) and path parameter (provided as an argument in method). The final URL becomes: https://healthapp.yaksha.com/api/Inventory/AvailableQuantityByItemIdAndStoreId?itemId="+reqItemId+"&storeId="+reqStoreId" 2. Include a bearer token for authentication in authorization header. 3.Trigger a GET call to the specified endpoint. 4.Create an object of type CustomResponse and initialize it with values extracted from the Response object: Response statusCode (extracted from Response object) ItemId object (extracted from Response object) AvailableQuantity object (extracted from Response object) StoreId object (extracted from Response object) Status (extracted from Response object)	- Returns an object of type CustomResponse containing statusCode, status, ItemId, AvailableQuantity, StoreId, and the complete Response object. - StatusCode should be 200. - Status should be OK. - ItemId, AvailableQuantity and StoreId fields should not be null.

Retrieve requisition items by ID in Method:

getReqItemsById(String endpoint, Object body)

- 1.Create an URL by combining the BASE_URL (already declared) and path parameter (provided as an argument in method). The final URL becomes:
 - https://healthapp.yaksha.com/a pi/Inventory/RequisitionItemsFo rView?requisitionId=" + requisitionId
- 2. Include a bearer token for authentication in authorization header.
- 3. Trigger a GET call to the specified endpoint.
- 4.Create an object of type
 CustomResponse and initialize it
 with values extracted from the
 Response object:
 - Response
 - statusCode (extracted from Response object)
 - CreatedByName object (extracted from Response object)
 - RequisitionNo object (extracted from Response object)
 - RequisitionStatus object (extracted from Response object)
 - RequisitionItems list (extracted from Response object as list)
 - Status (extracted from Response object)
- 5. Return the CustomResponse object from the method.

- Returns an object of type CustomResponse containing statusCode, status, CreatedByName field, RequisitionStatus field, RequisitionItems list, and the complete Response object.
- StatusCode should be 200.
- Status should be OK.
- CreatedByName field, RequisitionNo field, RequisitionStatus field and RequisitionItems list should not be null.

	I = 1 1	I	
12	Track requisition by ID in Method: trackReqItemById(String endpoint, Object body)	1.Create an URL by combining the BASE_URL (already declared) and path parameter (provided as an argument in method). The final URL becomes: https://healthapp.yaksha.com/a pi/Inventory/TrackRequisition?r equisitionId="+ trackByRequisitionId 2. Include a bearer token for authentication in authorization header. 3.Trigger a GET call to the specified endpoint. 4.Create an object of type CustomResponse and initialize it with values extracted from the Response object: Response statusCode (extracted from Response object) RequisitionId object (extracted from Response object) Response object) Status object (extracted from Response object) Dispatchers list (extracted from Response object as list) Status (extracted from Response object) Status (extracted from Response object)	- Returns an object of type CustomResponse containing statusCode, status, RequisitionId field, CreatedBy field, Status field , Dispatchers list and the complete Response object. - StatusCode should be 200. - Status should be OK. - Dispatchers list should not be empty and should not be null.
		from the method.	
13	Retrieve inventory items by store ID in Method: getInvItem(String endpoint, Object body)	 1.Create an URL by combining the BASE_URL (already declared) and path parameter (provided as an argument in method). The final URL becomes: https://healthapp.yaksha.com/api/WardSupply/GetInventoryItemsByStoreId/7 2. Include a bearer token for authentication in authorization header. 3.Trigger a GET call to the specified endpoint. 4.Create an object of type CustomResponse and initialize it with values extracted from the Response object: 	- Returns an object of type CustomResponse containing statusCode, status, ItemId list, ItemName list AvailableQuantity list and the complete Response object. - StatusCode should be 200. - Status should be OK. - ItemId, ItemName and AvailableQuantity list should not be null and should not be empty.
		ResponsestatusCode (extracted from	- AvailableQuantity should

		Response object) • ItemId list (extracted from Response object as list) • ItemName list (extracted from Response object as list) • AvailableQuantity list (extracted from Response object as list) • Status (extracted from Response object) 5. Return the CustomResponse object from the method.	be greater than 0.
14	Retrieve most sold medicines within a specified date range in Method: getMostSoldMed(String endpoint, Object body)	1.The final URL is provided as a parameter (endpoint) and that is: https://healthapp.yaksha.com/P harmacyDashboard/GetPharmac yDashboardMostSoldMedicine?F romDate="+ FromDate + "&ToDate="+ IncToDate" 2. Include a bearer token for authentication in authorization header. 3.Trigger a GET call to the specified endpoint. 4.Create an object of type CustomResponse and initialize it with values extracted from the Response object: Response statusCode (extracted from Response object) ItemName list (extracted from Response object as list) SoldQuantity list (extracted from Response object as list) Status (extracted from Response object) Status (extracted from Response object) Status (extracted from Response object) 5. Return the CustomResponse object from the method.	 Returns an object of type CustomResponse containing statusCode, status, ItemName list, SoldQuantity list, and the complete Response object. StatusCode should be 200. Status should be OK. ItemName list and SoldQuantity list should not be empty and should not be null. SoldQuantity should be greater than 0.

NOTE: "Please do not delete any file in the src folder. But you are free to add any other file".

Expectations:

- 1) Learners should write automation scripts using Java and REST Assured to automate the API testing for all the provided methods (e.g., GET, POST, PUT, DELETE). In other words, the automation script should perform all mentioned API interactions, including validation of responses.
- 2) Learners should not use any pre-built libraries or tools to validate API responses (e.g., JSON schema validation tools). They should manually validate the response content (e.g., status codes, response body, etc.) by writing their own logic for assertion.

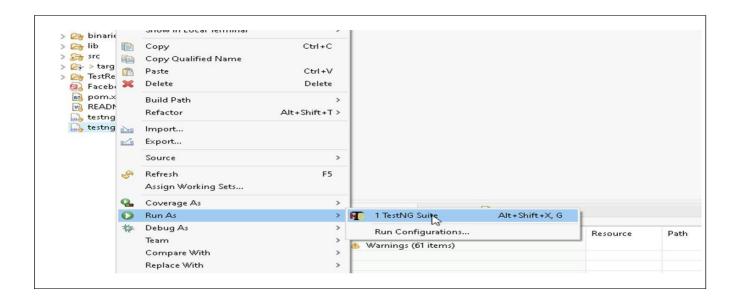
IMPLEMENTATION/FUNCTIONAL REQUIREMENT

1.1 CODE QUALITY/OPTIMIZATIONS

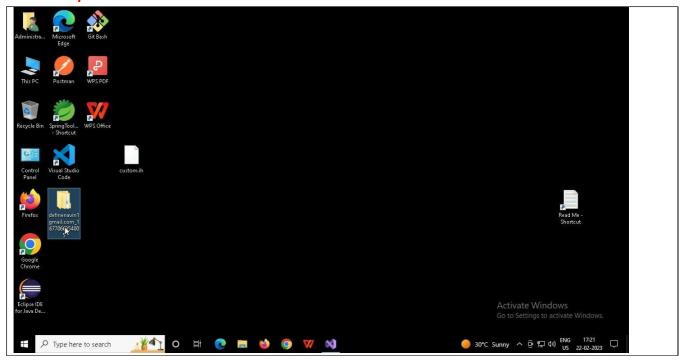
- 1. Associates should have written clean code that is readable.
- 2. Associates need to follow SOLID programming principles.

EXECUTION STEPS TO FOLLOW

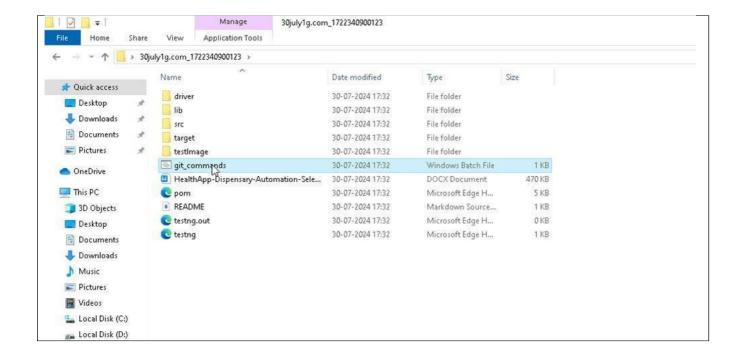
- 1. You are mandatorily required to run test cases for applications before final submission. Without which project evaluation will not happen.
- 2. You can launch test cases any time as follows: Right click on testng.xml and run TestNGSuite



3. Before final submission, you are also required to push your code to GIT. Following are the steps to follow:



In your project folder, you will find a batch file named git_commands



Double-click the batch file to run it. It will run the commands to push your code to GIT.

```
| | | = |
                                                                                                      Manage
                                                                                                                                             30july1g.com_1722340900123
                      Home
                                              Share
                                                                    View
                                                                                            Application Tools
                       C:\Windows\system32\cmd.exe
                                                                                                                                                                                                                                                                                                                                                                                                * Ou Your branch is up to date with 'origin/main'.
                    Changes not staged for commit:
(use "git add <file>..." to update what will be committed)
(use "git restore <file>..." to discard changes in working directory)
        no changes added to commit (use "git add" and/or "git commit -a")

or [main ce9ec82] First commit

1 file changed, 1 insertion(+), 1 deletion(-)

The Enumerating objects: 13, done.

Counting objects: 180% (13/13), done.

30 elta compression using up to 4 threads

Compressing objects: 180% (7/7), done.

Writing objects: 180% (7/7), 618 bytes | 389.80 KiB/s, done.

Total 7 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)

remote: Resolving deltas: 180% (2/2), completed with 2 local objects.

To https://github.com/IIHTDevelopers/30july1g.com_1722348988123.git

123cadc..ce9ec82 main -> main

press any key to continue . . .
        1
        7
        in Ne
```

All the Best