FSD Training

Linux

GIT

Java

UI – HTML, CSS, Javascript

Angular & React Introduction

Hibernate

Spring

Spring Boot

Cloud computing technologies

GIT:

It is a version controlling system, which helps people to collaborate their work so that at the end every user will have other users work in their machine as well as in remote machine.

Git provides an Online Remote Repository which users need to clone

List of GIT commands:

git clone: Clones the remote repository in your local machine

git push: Pushes the local repository changes to the remote repository

git pull: Pulls the remote repository changes to the local repository

git commit: Commits the changes done by the user & creates one unique id

git add: Tracks the changes in the working directory for commit

git status: Shows list of tracked & untracked changes in the working directory

Things you need are:

Git account: Create it from your personal ID

Git Bash: Install GIT in your machine so that you get GIT bash, which is a terminal

Git Branch:

Branch is like a pointer which will have the work with series of some commit ids, by default GIT maintains a default branch with a name called master/main

Steps we did

1. Created a Remote Repository
2. Cloned the Remote repository in a folder
3. Navigated to the repository in the local machine
4. Created a file and added some content
5. Used git add command
6. Used git commit command
7. Used git push command
8. Observed that Remote repository got the updated from the local repository
9. Created another folder Developer1 and cloned the remote repository
10. Created another folder Developer2 and cloned the remote repository
11. In Developer1 folder navigated to the local repository & created some files, then entered git add, git commit, git push
12. In Developer2 folder navigated to the local repository & created some files, then entered git add, git commit, git push
13. Remote rejected the push, so in the local repository we entered git pull, here if its fast forward merge then it doesn’t create a commit else it asks us to create a commit for merge

What happens if both the developers try to edit the same file & uploads to the Remote Git

Then Git doesn’t automatically merge instead users need to manually resolve the conflicts and merge the work

Why we should not work with the master branch?

In Real time there could be chance that users may push errors in master and master branch will be automatically merged without any review by any users, hence we must control it by creating a feature branch which is a copy of master branch, so that users can work with the feature branch & push the feature branch so that somebody will review that branch before merge

Feature branch

>> git branch branch-name

The above command creates feature branch

Checkout to branch

When you want to work in a different branch you need to checkout, the work you do in one branch is not updated to another branch automatically

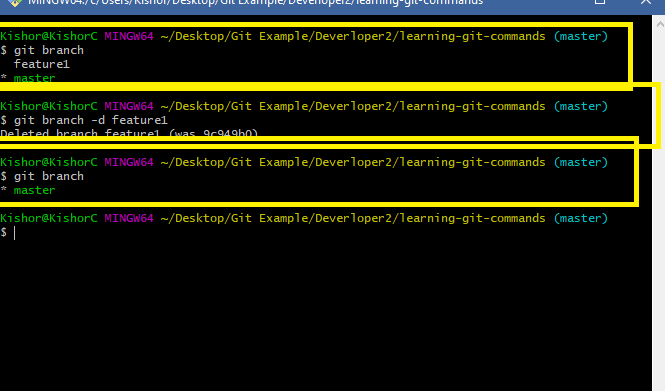
>> git checkout branch-name

Note:

1. Never work in the master branch
2. Always make commits or new changes in the Feature branch
3. Push the feature branch to the remote
4. In Remote the feature branch can be merged or it will be closed without merging

How to delete a branch

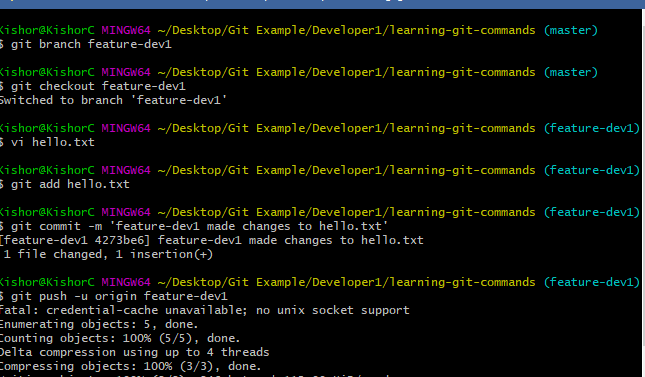
>> git branch -d branch-name



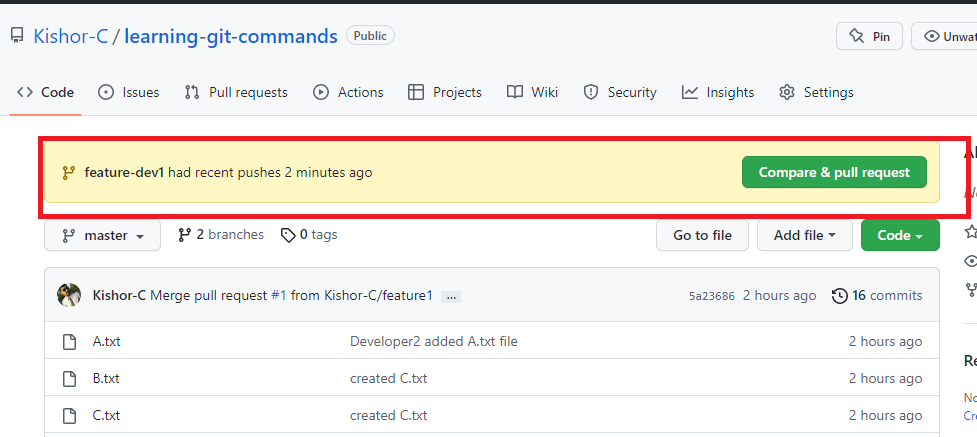
What to do when the remote can’t merge the feature branch

In the remote whenever there’s a Merge conflict you can close the pull request without merging & delete the branch, so that the user who pushed the branch should able to resolve the conflict and again push the feature branch

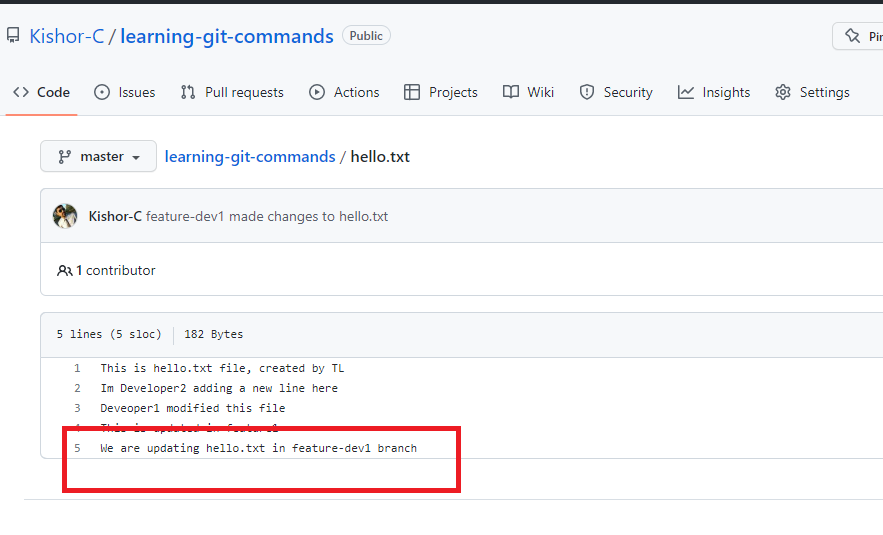
Let us create a new branch in the developer1 terminal & edit hello.txt with some content & push that new feature branch



We must see the same changes in the Remote, but we get a pull request in the Remote

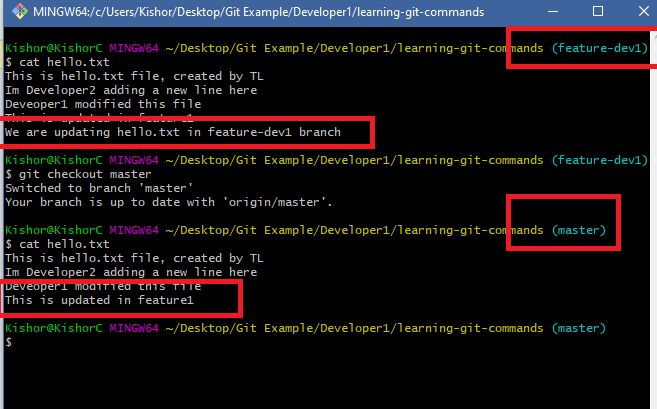


You need to click on Compare & pull request & check for merge option, if possible you can merge else you can close pull request



The highlighted part is the change done by developer1 in the feature-dev1 branch.

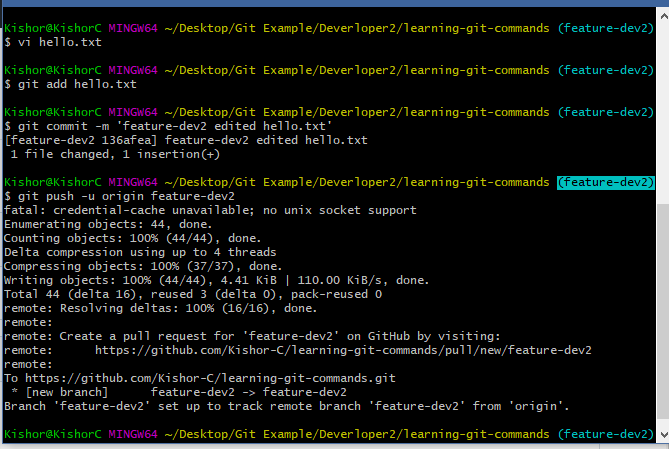
Note: Even in Developer1 terminal we have a master branch which is also not having this change



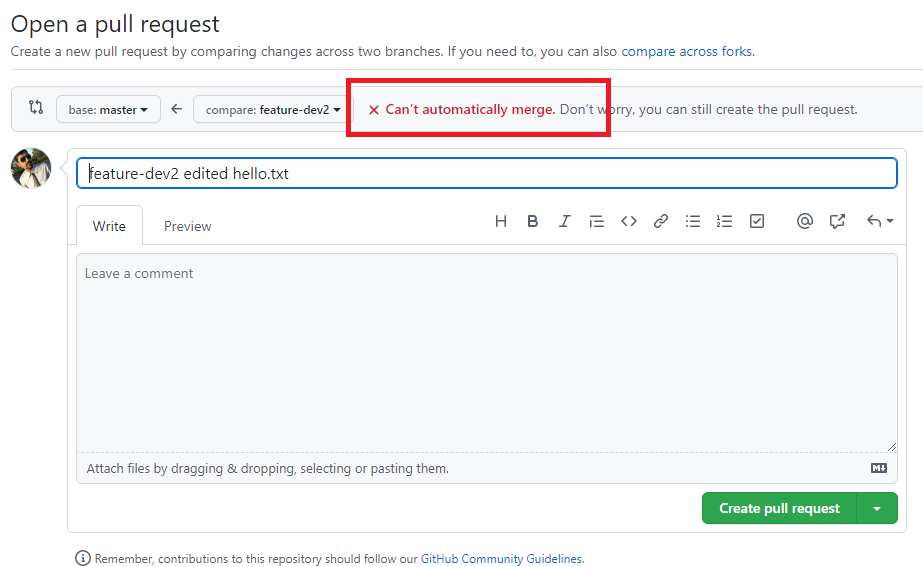
There are two ways in Developer1 Local repository to get master branch up to date changes of feature branch

1. use git merge in local repository in the master branch(git merge feature-dev1): This is not recommended as it gets only local repository changes, but it doesn’t get any remote repository changes
2. use git pull in local repository so that it pulls all the updates from remote repository: This is appropriate because it always gets changes done by other users from the remote

What happens if Developer2 terminal make changes to the ‘hello.txt’ without updating the local master



Here the Developer2 pushed feature-dev2 hence there was no rejection at the remote, however when the feature branch is tried to merge with remote master you may get conflict because the feature branch is not having other user work.



Here we can’t merge hence we can create pull request & close it so that the Developer2 will manually resolve, he must follow below steps

1. Pull the remote master to local master
2. Checkout to feature branch
3. Merge local master with feature branch if conflict resolve

Steps:

>> git checkout master

>> git pull

>> git checkout feature-dev2

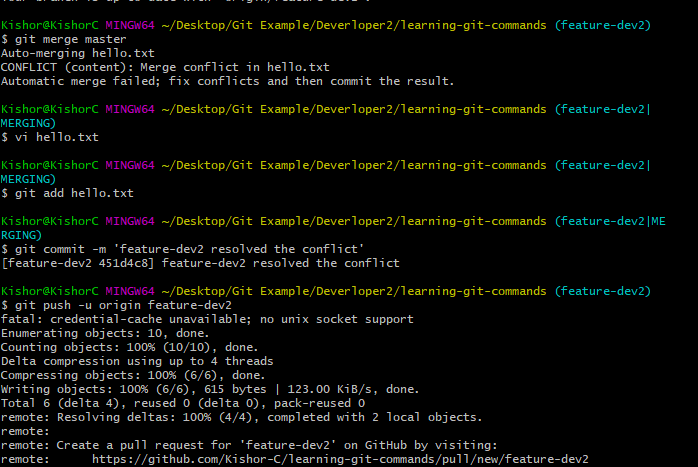
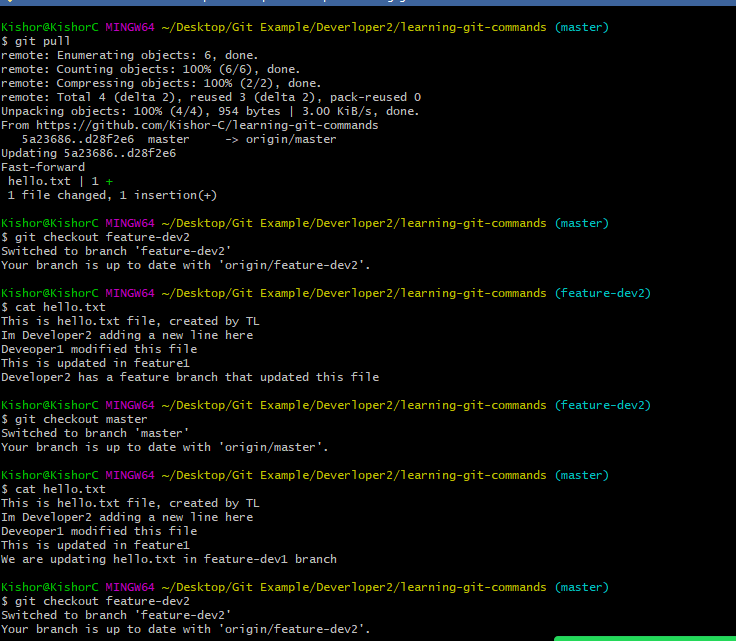
>> git merge master

# resolve the conflict when you get auto-merge failed

>> git add hello.txt

>> git commit -m ‘some message’

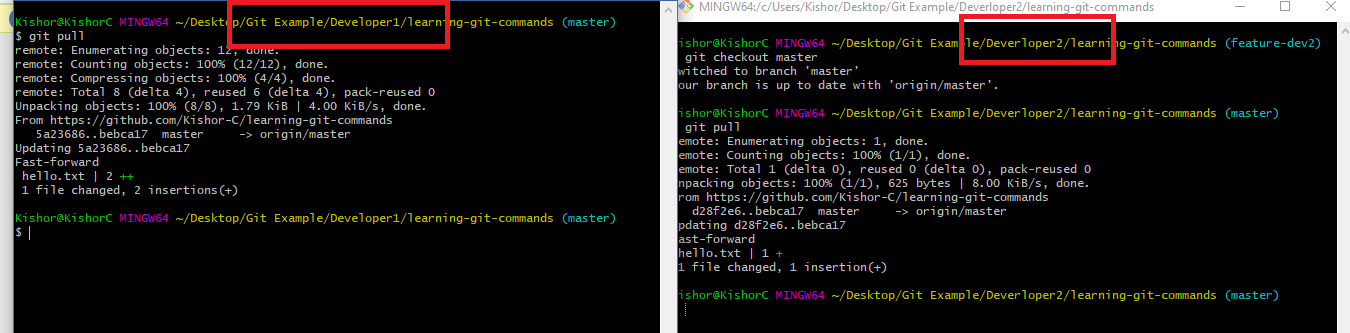
>> git push -u origin feature-dev2



Now in Remote there wouldn’t be any conflict to merge this feature branch

Final step:

Pull remote in local master in both the developers terminal to keep up to date.



Summary of Git steps to be followed when working in the project

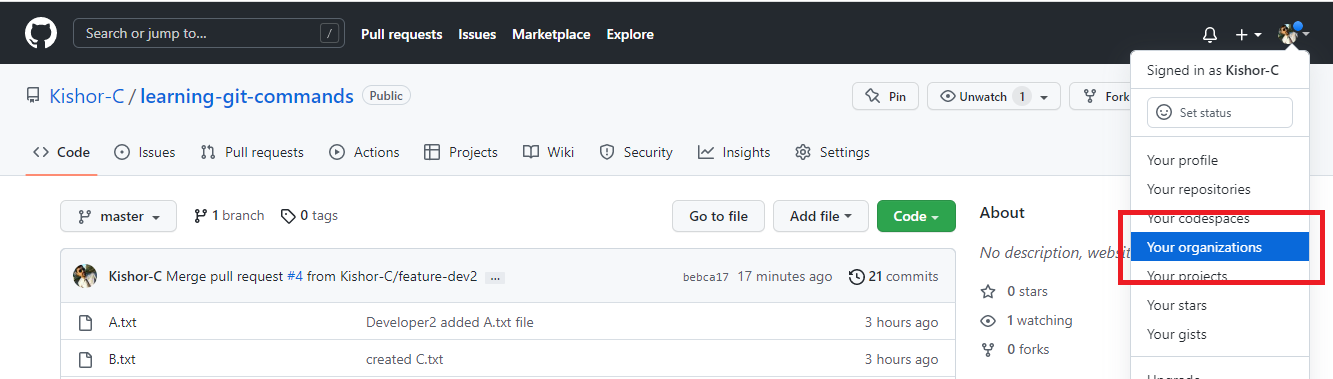
1. Git clone the remote repository if you don’t have local repository
2. Always make sure that you don’t work in master/main branch
3. Create feature branch to make any changes
4. Push the feature branch to the remote
5. If Pull request failed to merge then update the local master with remote master first using ‘git pull’
6. Once git pull updates the local master, checkout to feature branch and merge local master branch with feature branch using ‘git merge master’
7. You may get conflict while merging hence you can resolve it by editing the file having conflict.
8. Once conflict resolved commit & push the feature branch to the Remote repository for merge, but before that you can use git pull to ensure you have up to date changes

Git Organization:

It is a feature in the GIT which makes people to work in a team to collaborate their work, you need to add members here so that only members in the organization can work with the repository

Here someone must add the members by mentioning the mail-ids of their Git account so that each member will accept the invitation over the mail

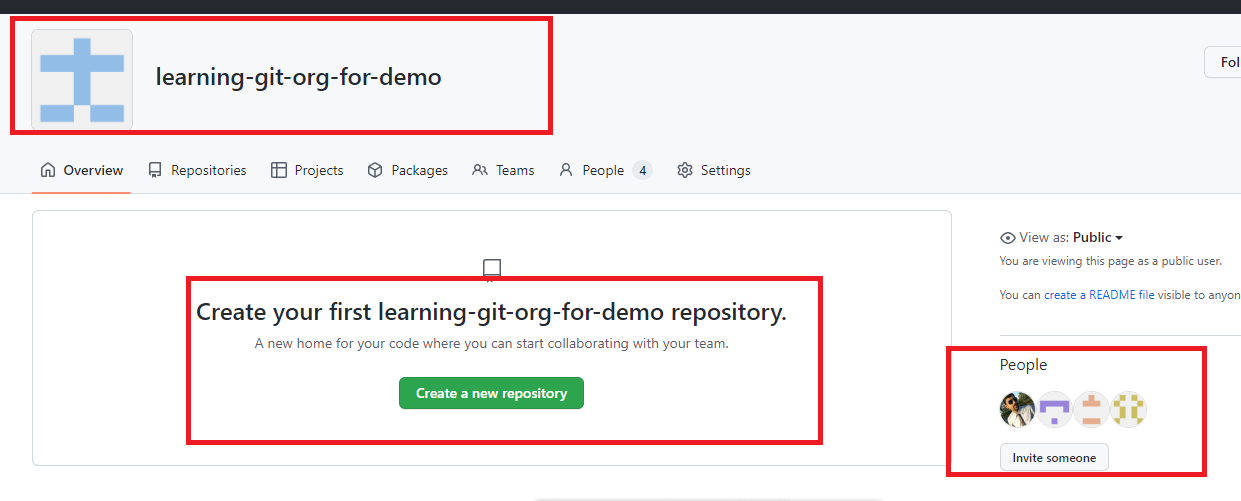
How to create Organization



Steps

1. You can create a new organization & add members.
2. You can pick Free plan
3. Enter organization name & your git mail id to create organization
4. Add members by entering mail ids
5. Members need to accept invitation
6. You can make all the members owner to enable them do push/pull task

You must able to see members in the organization



Activity

1. Try out all the GIT commands taught in the Session
2. Create an organization (any one person in the team), add the members, one person in the team must create the Repository & all the members must clone it
3. Each member can create a text file with their name in the feature branch & push that feature branch to the organization repository
4. Someone in the team must take care of merging the pull request

Java

Software required

* Eclipse IDE / STS

Web Development

There are 3 main technologies we use to develop web pages

1. HTML
2. CSS
3. Javascript

HTML stands for Hyper Text Markup Language, it is mainly used to create contents

CSS stands for Cascading Style Sheet, it is mainly used to add styles to the HTML

Javascript makes web page more dynamic by manipulating HTML & CSS at runtime

HTML uses elements or tags to create contents, it as many predefined elements that browser can understand some of them are

<p>, <span>, <div>, <table>, <img>, <form>, <h1>, <h2>, <h3>, <h4>, <h5>, <h6>, <ol>, <ul>, <li>

Software required for HTML, CSS, & Javascript

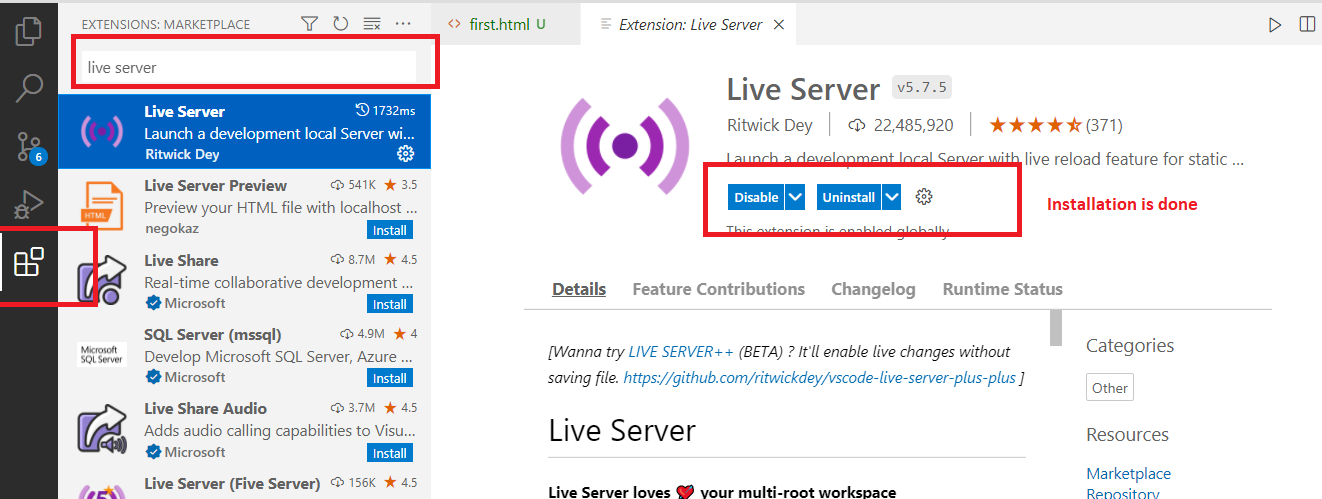
1. Editors: Notepad, VSCode, Brackets, Online editors(Code pen, vscode.dev)

You can get online VSCode editor from <https://vscode.dev/>

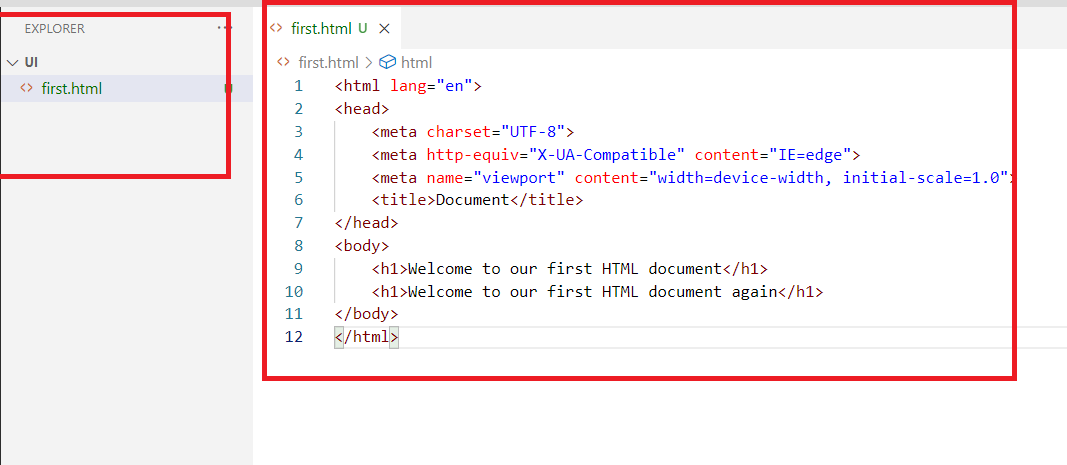
1. Browser to preview the output

VSCode: It is available both online as well offline, but in offline you can add plugins to the VSCode to get a better development experience

ex: Live Server which auto-reloads the browser when you modify the file

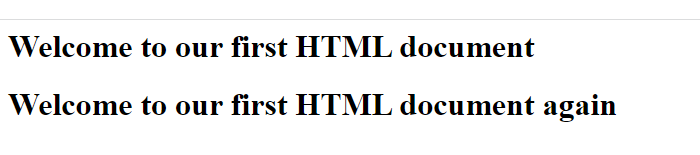


first.html



After you have installed Live Server extension, you can right click on the file and open with Live Server to see the output

Output:

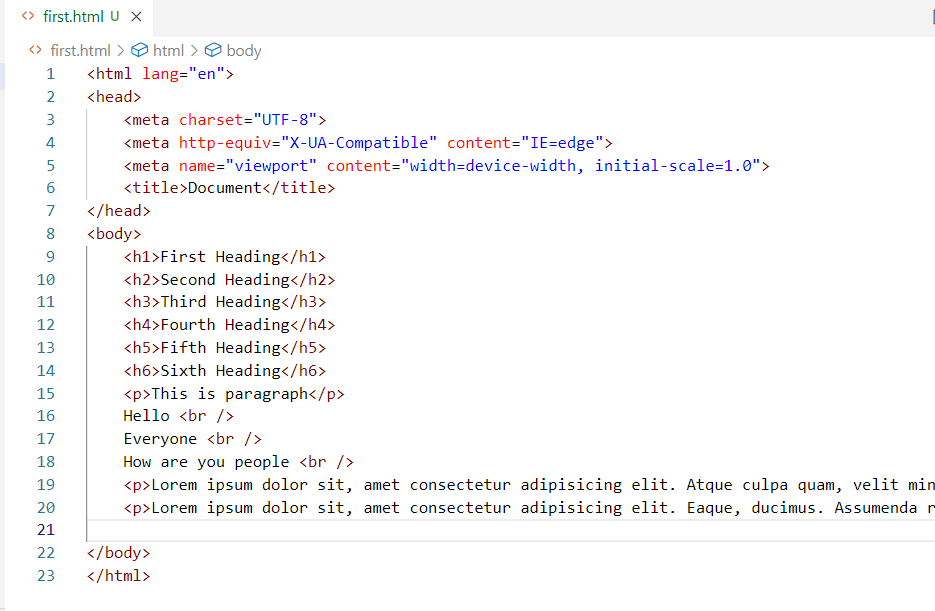


HTML has 6 heading tags

Their font size varies according to the heading tag number

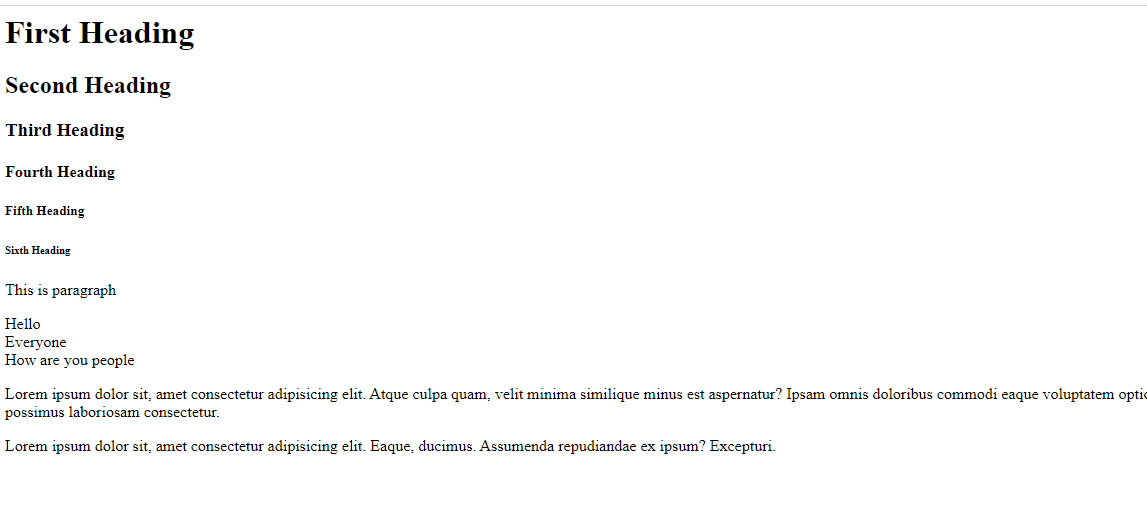
h1 will have a bigger font, h2 will be slightly lower than h1, same way you have h3, h4, h5, h6

first.html



You can type lorem10 to generate 10 words of Lorem ipsum

Output:



HTML Images & their attributes

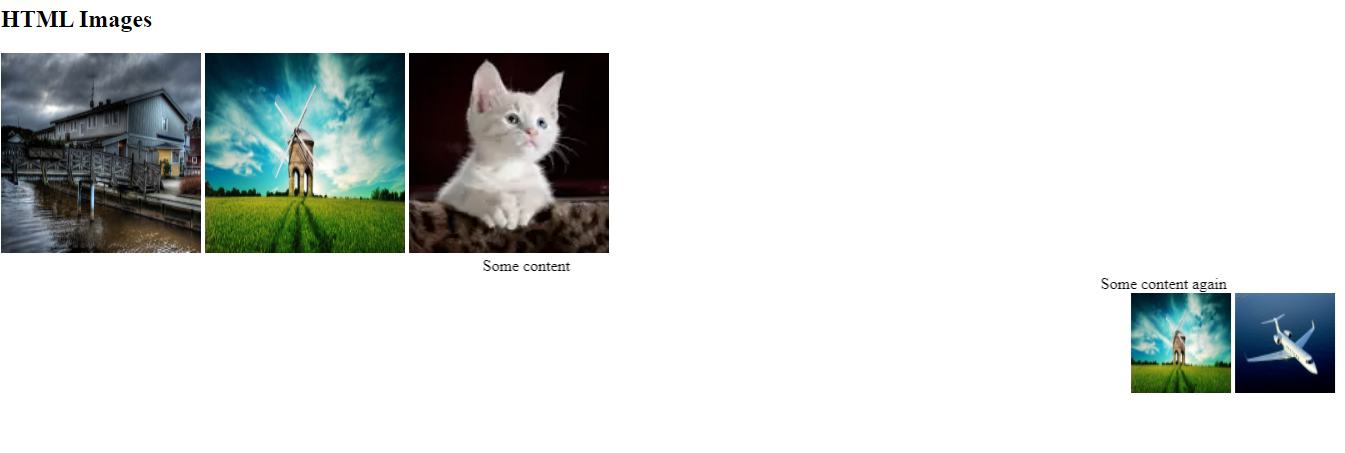
<img> tag is used to add images, we can add the image present in our local machine or we also add online image url and load in the web page.

It has attributes like src, width, height, alt



div: It is a container tag used to include other tags inside it, this is one of the very useful tag in HTML

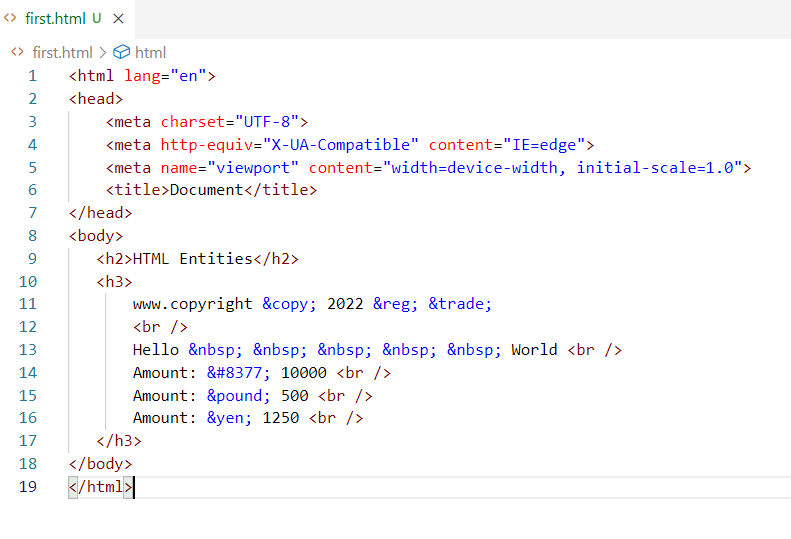
Output:



Entities

These are some special characters that will have a name or a number so that they can displayed, they may be not be present on the keyboard, they are used with &entityName; or #&entityNumber;

Ex: © ® ™ All these are some special characters that can be displayed using the entity names like &copy; creates ©, &reg; creates ®, &trade; creates ™, similarly we have entity numbers for currency symbols like pound, rupee, yen



Output:



Comments in HTML will be



This same comments works even in XML File also, they are multi line comments

HTML Lists

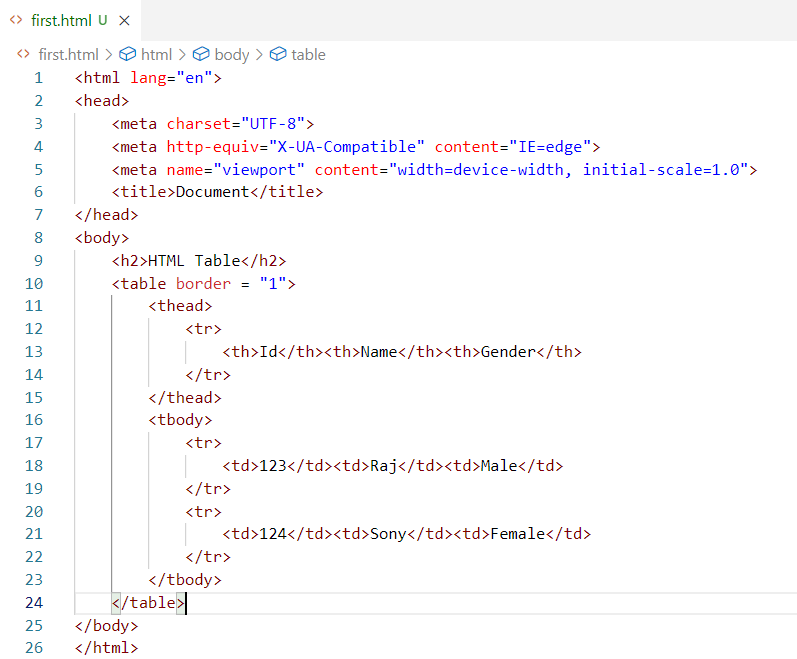
In HTML we have tags to show the contents in an ordered & unordered way, for that we have <ol> & <ul> tags both takes <li> as a child tag to show the list of items



In <ol> you can use type as 1, A, i, and in <ul> you can use type as disc, square, circle.

HTML Tables

This helps to create tables that will have rows & columns, <table> is used to create table & <tr> is used to create rows, in the row we can have either <th> or <td> to create the data for the columns, <table border = “1”> will give some border lines in the table, maximum number you can give is 8



Git Hub Activity Link

<https://github.com/Kishor-C/publicis-sapient-student-activities.git>

Steps to perform

1. Fork the remote repository so that it is cloned in your account remotely
2. Clone the remote repository of your account to your local machine (Ensure you have not cloned the below URL

<https://github.com/Kishor-C/publicis-sapient-student-activities.git>

1. Create day wise folder on each day and create separate files to perform the hands-on exercises
2. Push the changes to your remote repository

HTML Forms

Forms help user to enter inputs, select controls like radio, checkbox, dropdown and so on

<form> tag is used to create the form, it provides 2 attributes action & method

<form action=”url” method=”httpMethod”>

Form action is the backend resource url the form will submit the input,

Form method is http method like GET or POST which is used to wrap the data in the HTTP request either in the body or url of the request.

GET: It is the default method the form uses, here the data is submitted via URL, it means the data will be visible to the end user

POST: It sends the data via the request body & it wouldn’t be visible to the end user

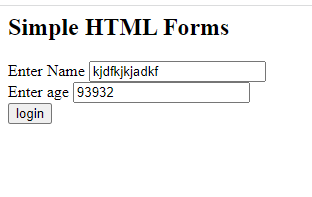


GET vs POST

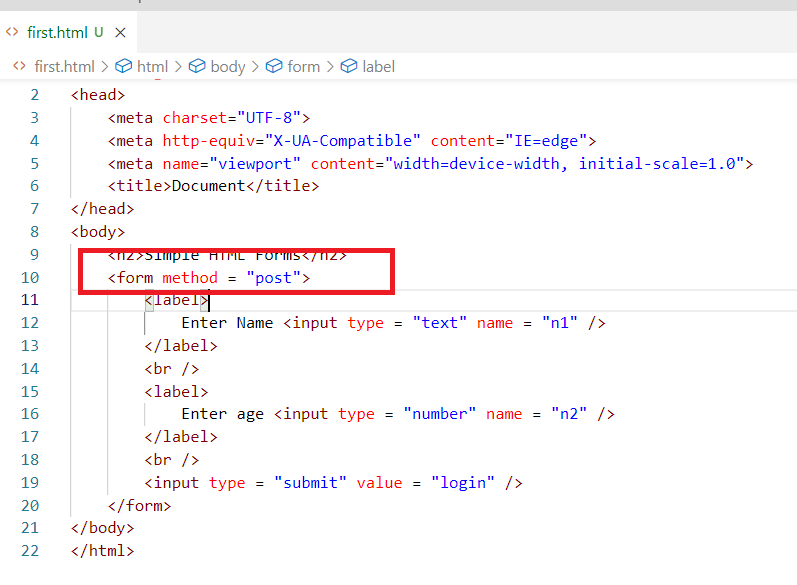
|  |  |
| --- | --- |
| GET | POST |
| GET sends data via URL | POST sends data via Request body |
| Insecure | Secured |
| Faster | Slower compare to GET |
| Supports only 256 characters | Supports unlimited characters |
| Default form submission is GET | It is not chosen by default, we need to use <form method = “POST”> to make form submission post |



Output:



You can use method = “Post”



Here you don’t see the data in the URL.

There are other controls you can create in forms like radio, checkbox, dropdown, date and so on.

For radio button you can use <input type = ‘radio’>,

For checkbox you can use <input type = ‘checkbox’>

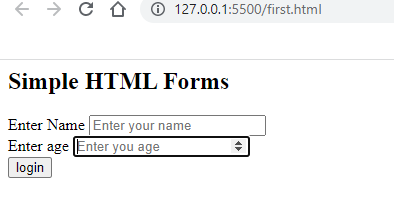
Form Control attributes

autocomplete: You can use this when the input box shouldn’t show the history or previous input, you can make it false.

placeholder: You can use this when you want to display the hint in the input box what to enter.



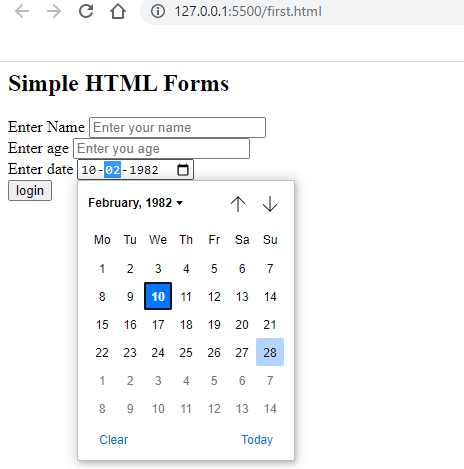
Output:



date: It is a type in the <input> which shows the calendar



Output:



input name: These are placeholder which make server side programs to read the values

Example in Java we have Servlets that can handle these input names in a request parameter as below



CSS: Cascading Style Sheet

It is used to style the HTML elements, it uses a syntax which will have css property & value to it

You can embed CSS into HTML in 3 ways

1. Inline CSS
2. Internal CSS
3. External CSS

Inline CSS: Adding styles to the HTML elements directly so that it applies only to that particular element, it can be added using ‘style’ attribute

<p style = ‘color:red’>Some content in p element </p>  
<p>some content again in p element </p>

Internal CSS: Adding styles to the entire HTML documents, the styles can be used by multiple elements in the same HTML document

<head>  
 <style>  
 p { color : red } /\* all the <p> gets red color\*/  
 h1 { color : blue } /\*all the <h1> tag gets blue color\*/  
 </style>  
</head>

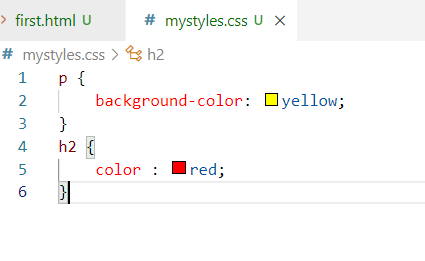
<body>  
 <p>…</p>,  
</body>

External CSS: It is a style that exists outside the html, it can be applied to multiple HTML documents, it should be created in a css file & you must link the CSS using <link> tag

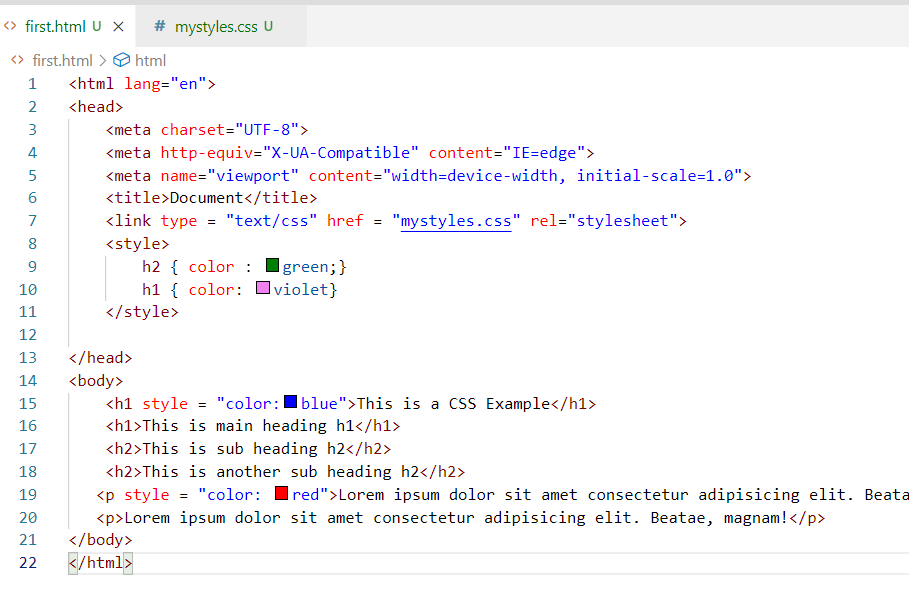
What happens if the same element is styles with External, Internal & Inline

Internal overrides External, and Inline overrides Internal

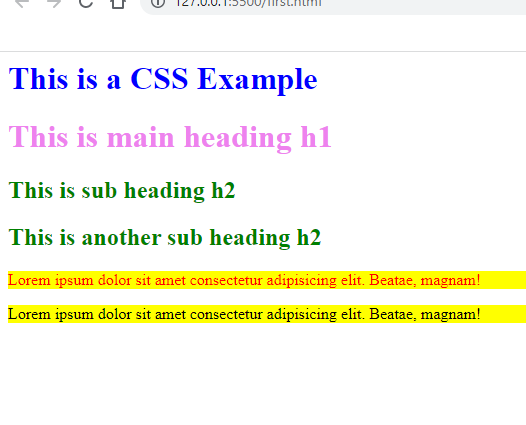
mystyles.css



first.html



Here external css is overridden by internal & internal is overridden by inline css



There are many CSS properties which makes HTML elements look better, but as a CSS developer we need to know the syntax of how to use CSS instead of remembering the CSS properties.

Different types of selectors

Multiple elements selector: This is used to add css to the multiple elements,

ex: h1, h2, h3 { property : value } /\* h1, h2, h3 gets same CSS\*/

ex: \* { property : value } /\* this to select all the elements \*/

Tag selector: This is to select a particular tag

ex: p { property : value }

Class selector: This is to select multiple elements with the same class

ex: div.c1 { property: value}

Here <div class = “c1”> get the style, we can have one or more div with the same class and also we can apply classes to multiple html elements

ex: .c2 { property : value }

Here <p class = “c2”> <div class = “c2”> can get the css of c2

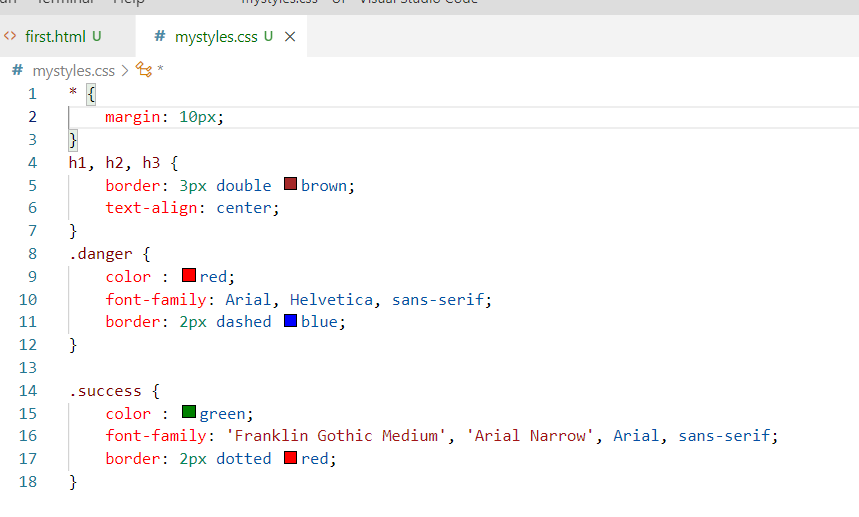
Id selector: This is used when you have an unique element with an unique id

#a { property : value}

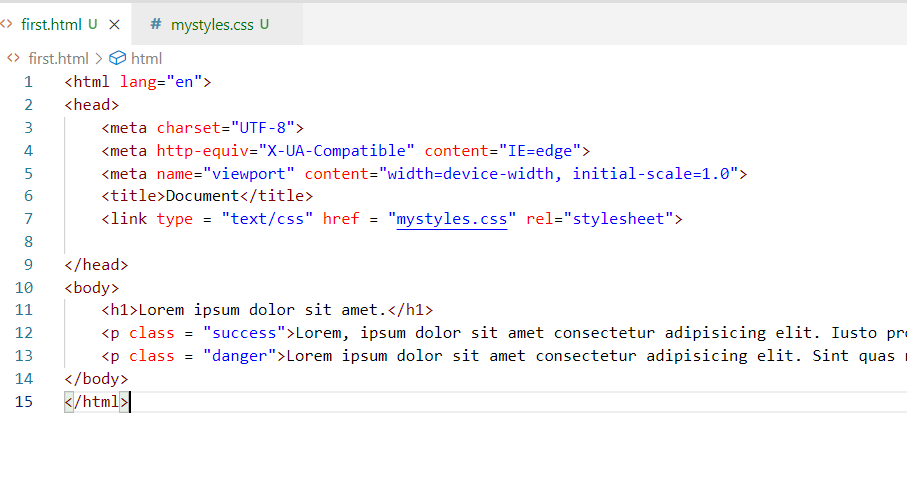
Here an element can have id a i.e., <p id = “a”>

Note: Id must be always unique, though it works in CSS, it doesn’t work in Javascript

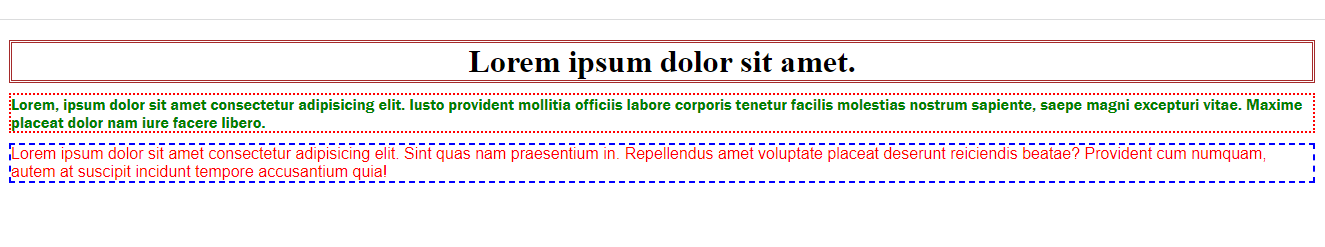
mystyles.css



first.html



Output:

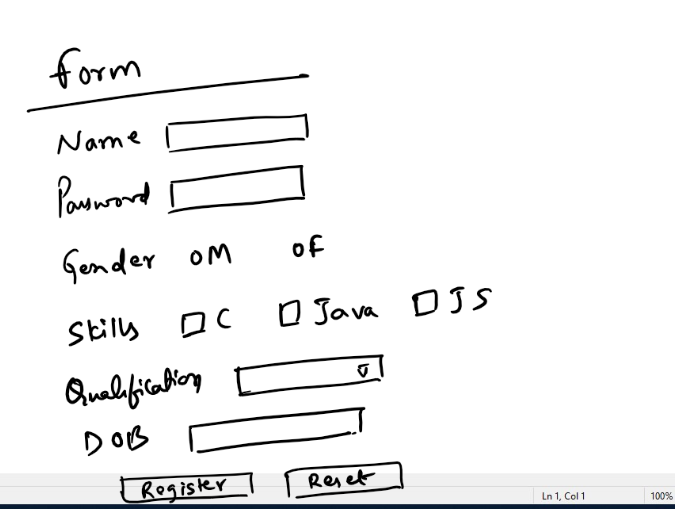


Activity:

1. Create separate files for each examples & update in the Git repository that was forked

ex: day6/ex1.html, day6/ex2.html and so on (or) you can give meaningful names to the html files

1. Create a form that will have all the controls like text, password, radio, checkbox, dropdown, use <table> without border so that all the labels & controls will have proper alignment



1. Using CSS try to create a web page with a layout having only borders instead of contents, use <div> as an element to create the layout, the layout must look as below

