Java Full Stack Development

Phase 1

Agile

GIT

MYSQL

HTML

CSS

Javascript

Angular

Cucumber

SDLC

Software Development Life Cycle - it is a process used by software industries to design, develop, test and maintain applications

Agile model

* It is an iterative and incremental model

Agile principles

* Customers don’t have to wait for long time because they get the features early
* You will develop, test and release only a few features to the customers
* Changes can be done at the earlier stage based on the customer’s feedback.
* Standup meetings between the customers, business analysts, developers and testers

Benefits

* Releasing the features early - weekly
* Changes are allowed in the middle
* Good communication between the teams

Challenges of Agile

* Inadequate documentation

Scrum

* It is a framework that follows agile principles

Scrum includes group people

1. Product owner - gathers requirements and accept/reject the features
2. Scrum master - ensures everyone follows agile
3. Developers
4. QA

Scrum events

1. Daily standup meetings
2. Sprint planning
3. Sprint review
4. Backlog refinements

Jira:

It is a tool used by software and business teams for project management.

Scrum terms

* User stories
* User stories acceptance criteria

Example of user stories on product catalogue

1. As a customer, I want to filter products by category, price range and brand
2. As a customer, I want to sort products based on ratings

GIT

It is a distributed version controlling system that can remember every change made by the user, it also helps users to collaborate other user work

Pre-requisites

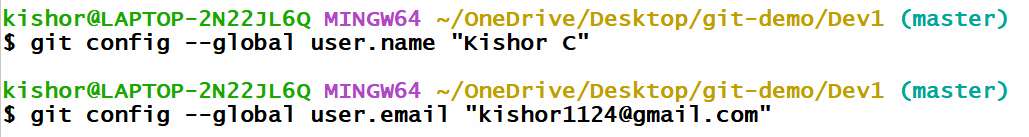
1. Git hub account
2. Git bash

Installing Git bash -> Just follow the steps

git init : It initializes the folder as a git repository so that it can track the changes.

You need to configure name and email to commit the changes and also you need to authorize to push the changes to the remote repository

To configure the user



To push the changes to the remote repository, git bash must know the repository should be uploaded to which remote repository

Command to push the changes to the remote

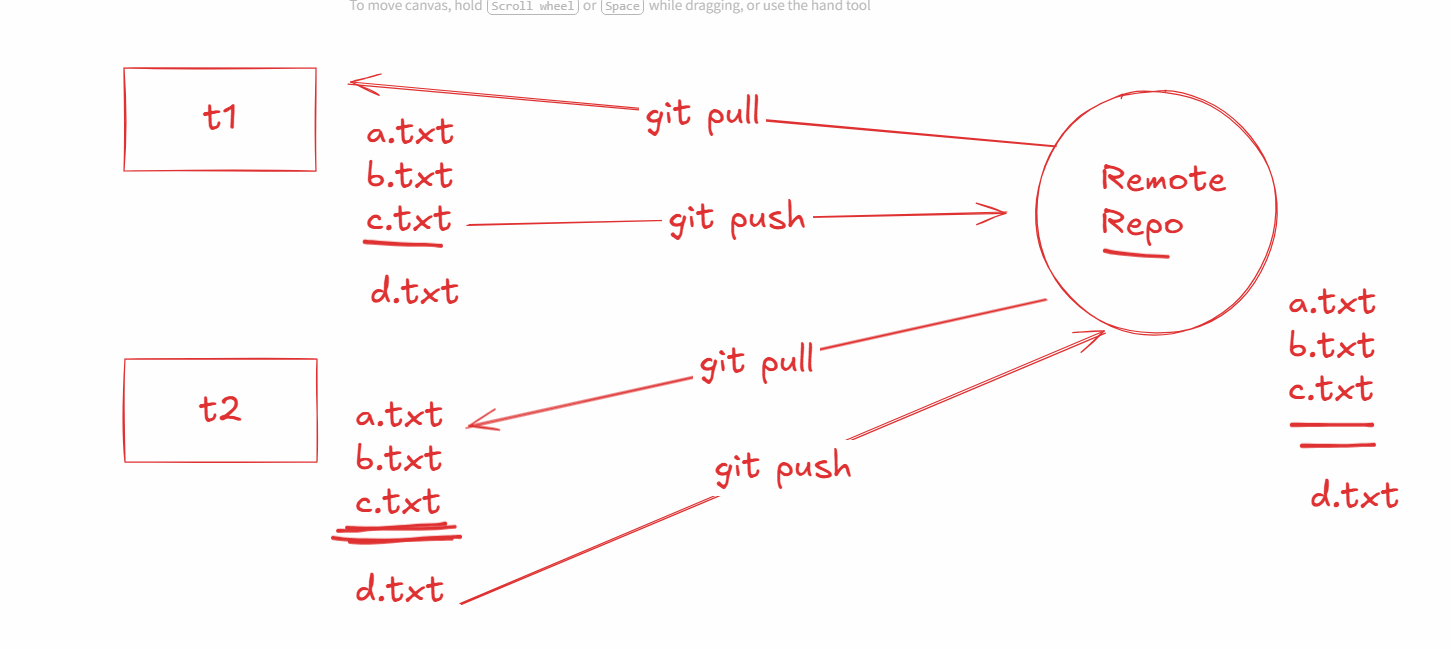


Summary

1. touch b.txt
2. git add b.txt
3. git commit -m ‘second commit’
4. git push -u origin master

Git clone: it is used to clone the remote repository to the local machine

command: git clone <<remote-repo-url>>



Activity:

Create d.txt in the 2nd terminal push that to the remote & in the 1st terminal pull the changes, and in both the terminal use ls to observe d.txt file

How does git knows what files to pull & merge to the local repository

Git creates a commit-id which is unique using that it can understand what version to pull and merge



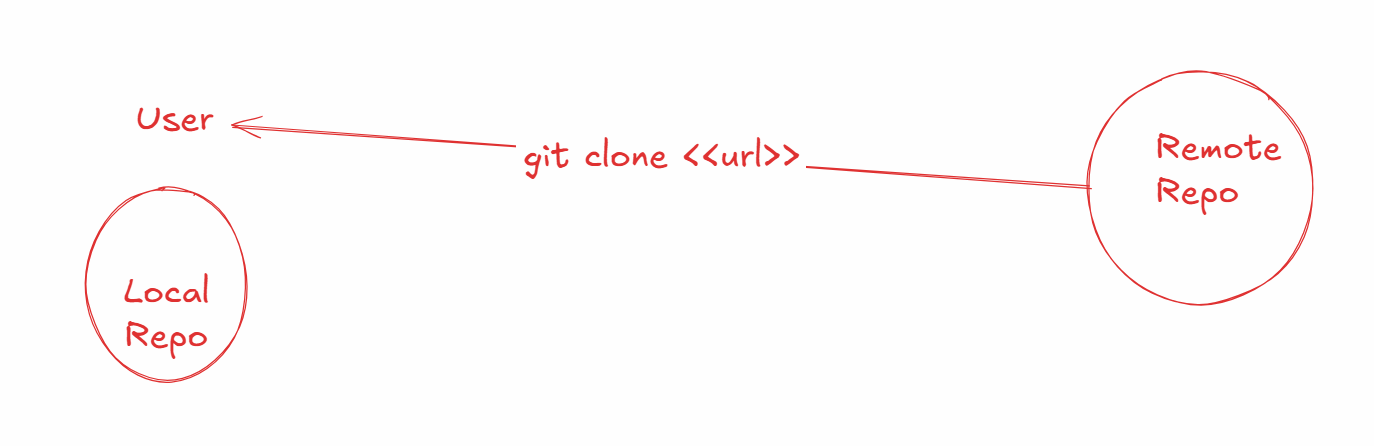
Summary:

1. git init
2. git config --global user.name “username”
3. git config --global user.email “emailid”
4. git add filename
5. git commit -m ‘message’
6. git push -u origin master
7. git pull
8. git log --online
9. git log
10. git clone <<remote-url>>

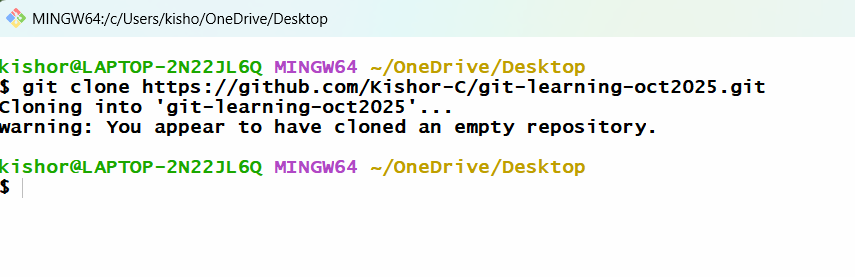
Day 2 Agenda

1. Creating remote repository
2. Cloning the remote repository
3. Understand branches in GIT
4. Merge conflicts
5. Stashing
6. Resetting
7. Fetch vs Merge

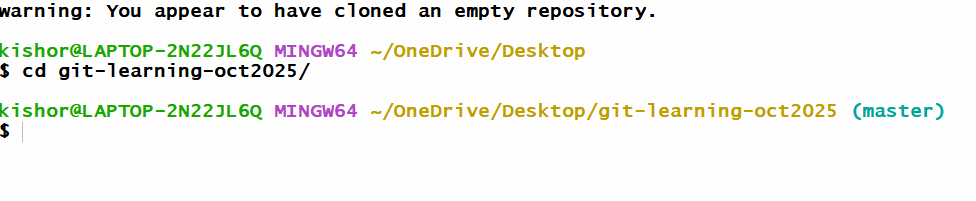
Step 1: Create a repository in the git hub account and clone it



Step 2:



Step 3: navigate to the repository



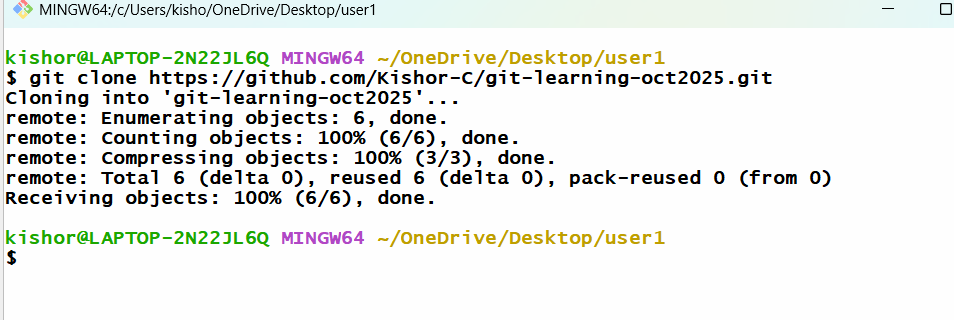
How to avoid conflicts

1. User needs to always create a separate branch called as feature branch instead of working in the main/master branch
2. In the feature branch users must do the changes and push to the remote repository
3. In the remote repository there will be a notification when changes are pushed
4. These changes must be merged with the master branch
5. If the changes doesn’t have conflict it can be merged successfully else it will be rejected
6. If the merge is rejected, then the user must pull the remote master changes to their local master which will merge the changes and then they must create a feature branch and make the changes and then push the changes to the remote

Create a 2 folders

1. user1
2. user2

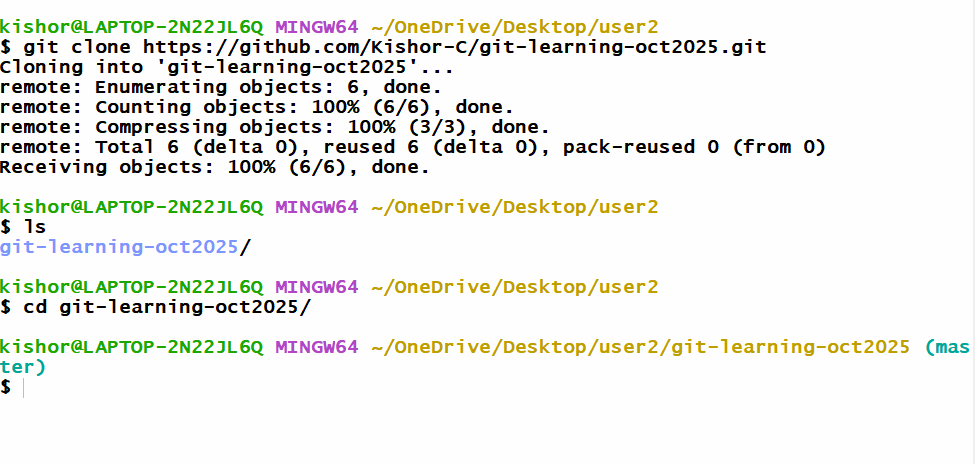
In User1 folder open git bash and clone the remote repository



Navigate to repository



In User2 folder clone the same repository and navigate to the repository



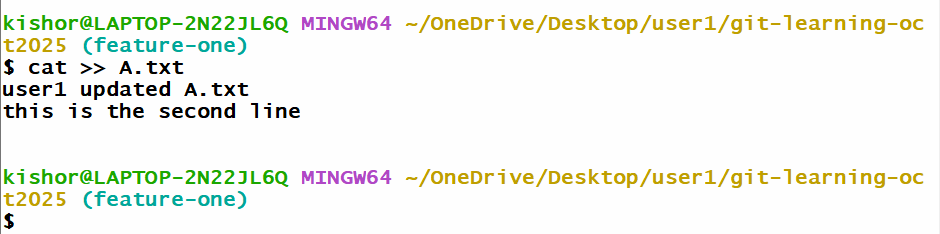
How to create a branch and switch to that branch

git branch branch-name: This creates the branch

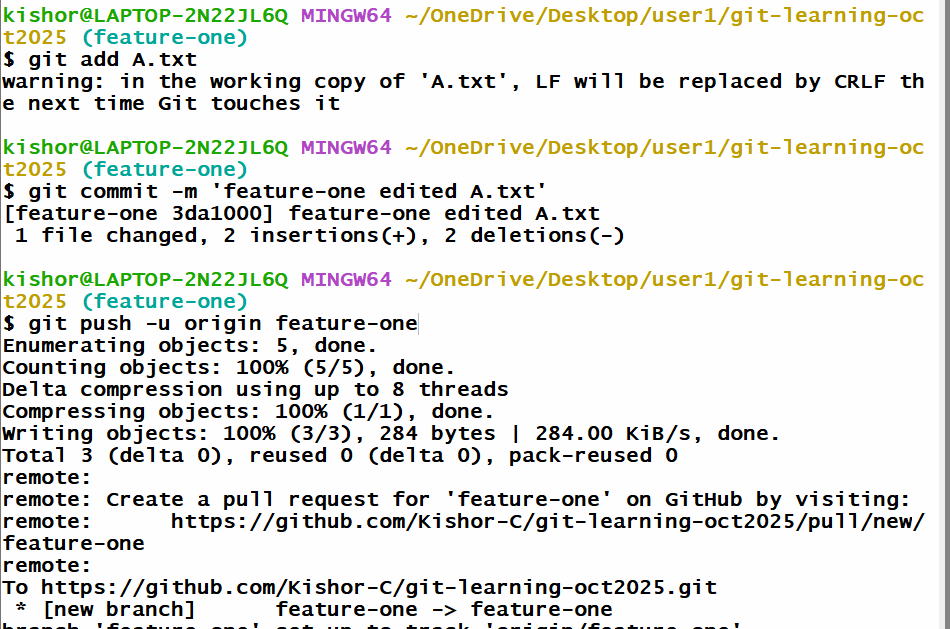
git checkout branch-name: This switches to the branch



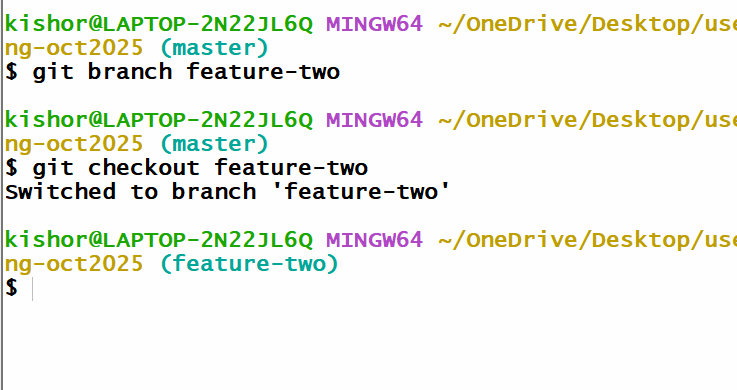
In User1 -> feature-one >> edit A.txt



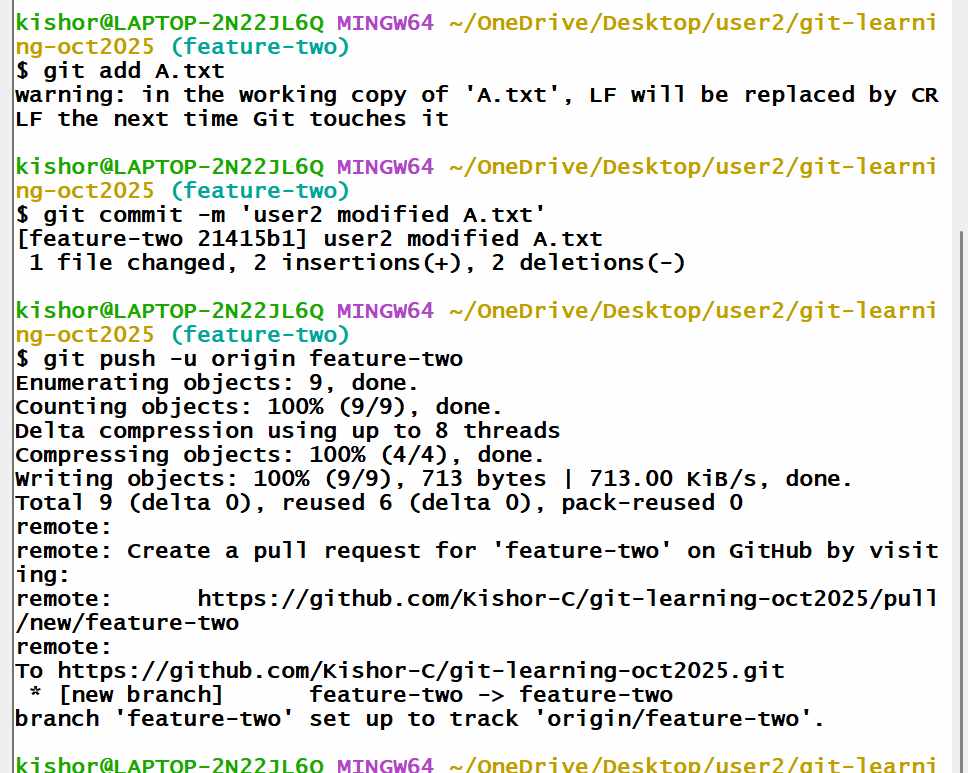
Push the feature-one to the remote



Create a branch in User2 and update A.txt and push to the remote



Push the feature-two to the remote

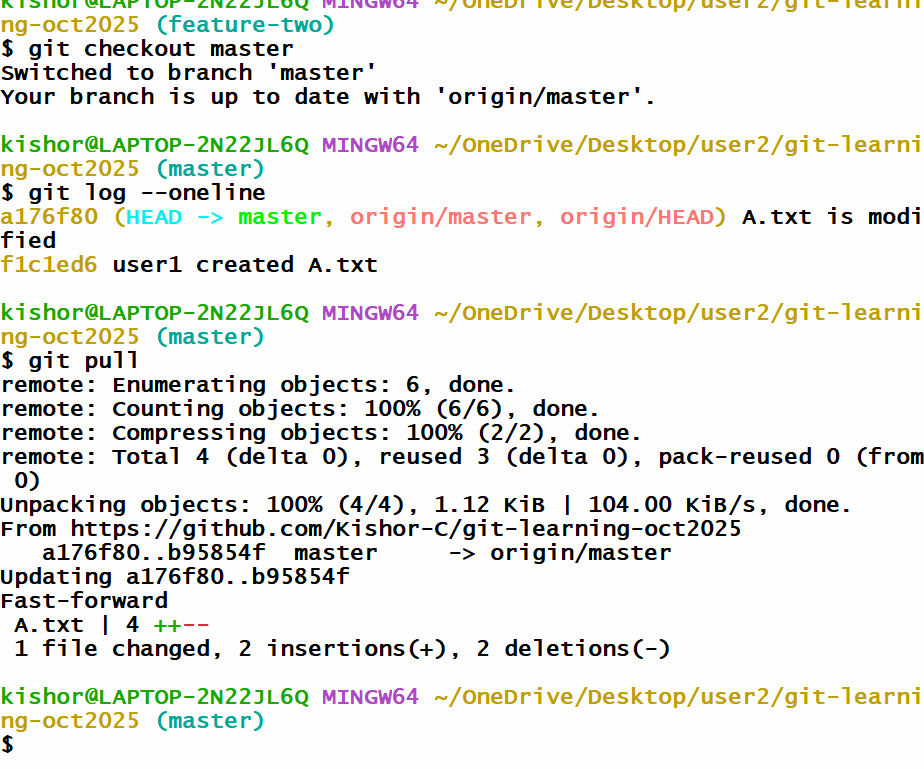


Note: Remote master can’t merge feature-two because of version mismatch

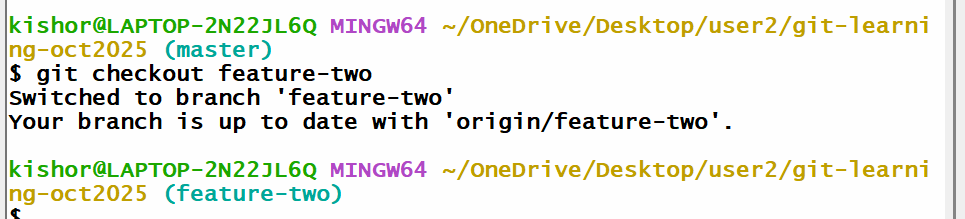
User2 must pull the remote master which will merge local master with remote master versions and then checkout to the feature-two and merge the local master with feature-two, here you will get the conflict which you will resolve with a new commit and then you will push

Steps to resolve

1. git checkout master
2. git pull -> this will merge local master with remote master
3. git checkout feature-two
4. git merge master -> gives conflict error -> resolve -> commit -> push the feature-two



Checkout to feature-two and merge - you will get conflict error - which you must resolve



Once you resolve you can push the feature-two

Note: If git-hub doesn’t show the notification you can click on Pull request tab

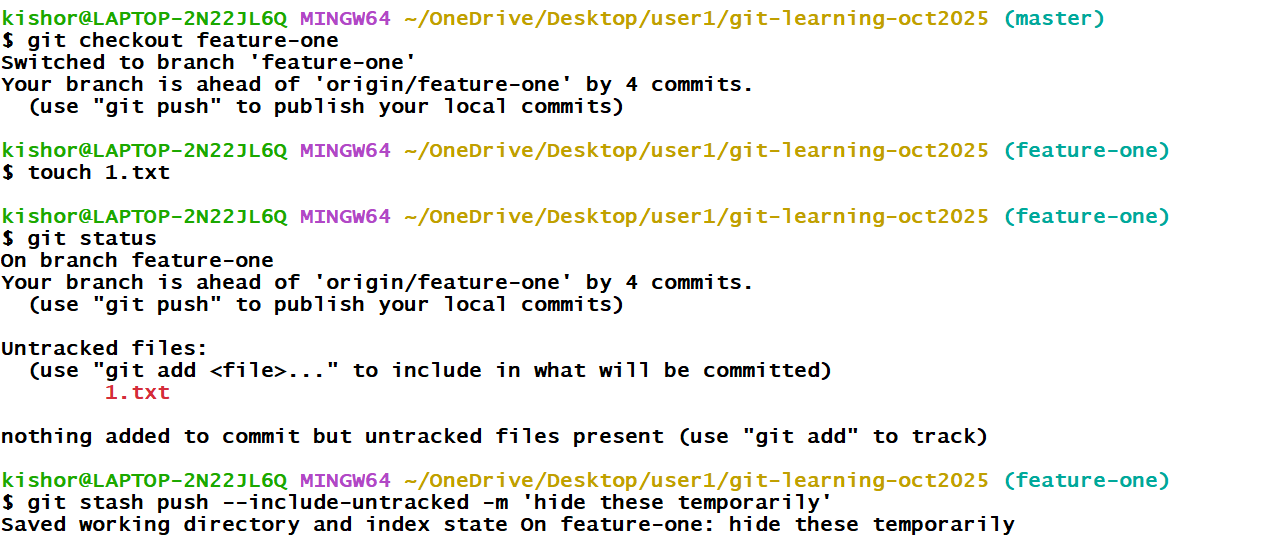
Day 3 agenda

1. Git stash
2. Git Reset
3. Fetch vs Merge
4. MySQL

Download MySQL 8 LTS version & Install it, ensure you install full products

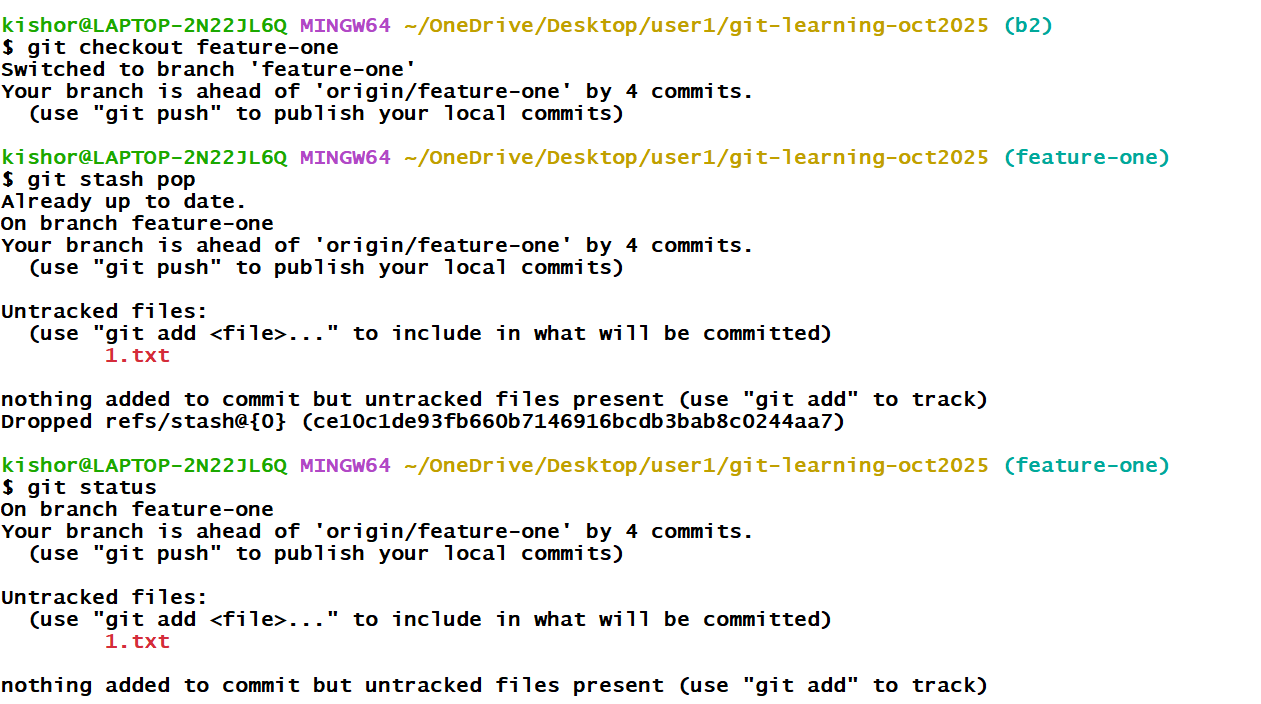
Git stash

Sometimes untracked files appear in other branches, to hide those untracked files you can use git stash push --include-untracked -m ‘some message’



Now you can create a new branch and checkout to the new branch and you don’t see the untracked files

Now you can checkout to the feature-one get those untracked files using git stash pop



Stash commands steps

1. git checkout feature-one
2. touch 1.txt
3. git stash push --include-untracked -m ‘some message
4. git checkout -B b2
5. git status
6. git checkout feature-one
7. git stash pop

Git Reset:

This command gives the version you want in your project, this done whenever you want to go back to old version of the project

syntax:

git reset --hard commit-id

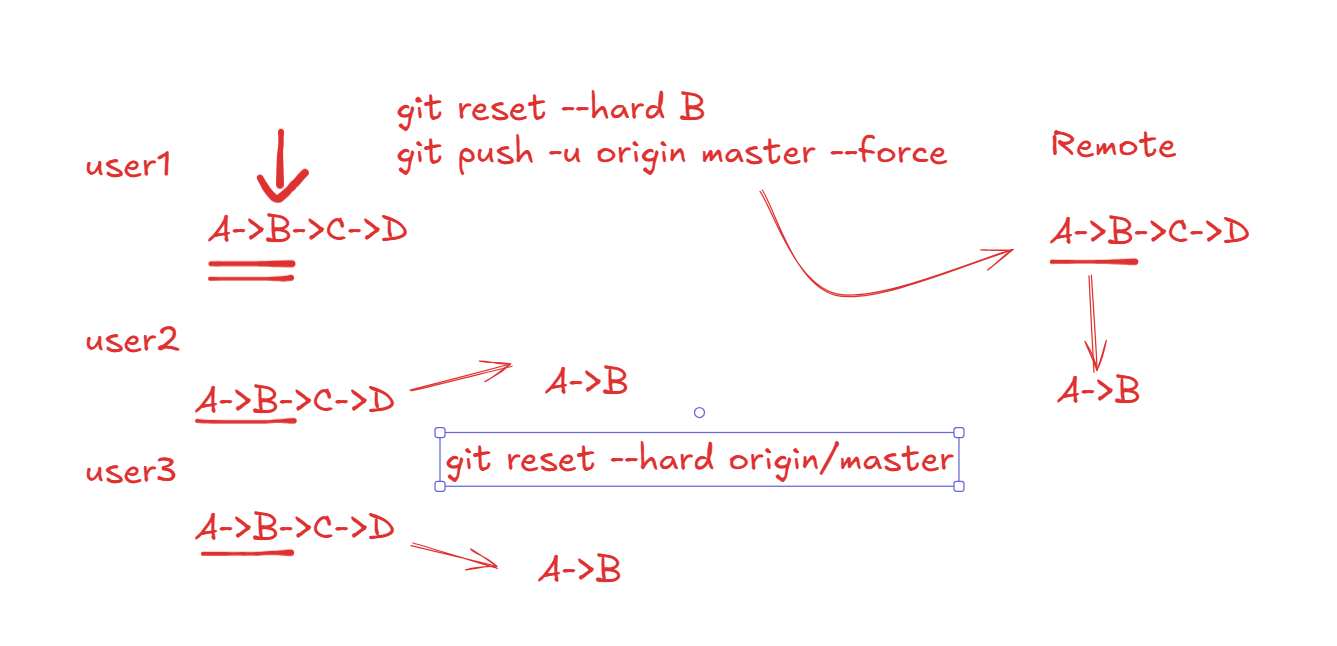
Once you reset the version, this is updated locally, but you may need to update the version to the remote, that is done using

git push -u origin master --force

Other users must also get these changes for that they need to use git pull & get reset below command

git pull

git reset --hard origin/master





git fetch vs git merge

fetch: get remote changes without merging the local branch, it stores the changes in the .git folder so later you can merge, this is a safe way of merging without immediately getting the conflicts, it is not risky

pull: get remote changes and merges, if there’s a conflict you need to resolve, here the risk of conflicts will be there

git pull = git fetch + git merge

MySQL database

Database: it is a collection of data or a record keeping system

DBMS: Database Management System, it is a software that helps to access the database to maintain the data like store, update, delete, read

RDBMS: Relational Database Management System, that maintains the data in a table format

We have many RDBMS software’s like

1. MySQL
2. Oracle
3. PostgreSQL

All these database understand a language called SQL (Structured Query Language)

SQL has sub types

1. DDL - Data Definition Language - create, alter, truncate, drop
2. DML - Data Manipulation Language - insert, update, delete
3. DQL - Data Query Language - select
4. DCL - Data Control Language - grant, revoke
5. TCL - Transaction Control Language - commit, rollback

MySQL gives you a terminal and also it gives you a workbench (auto-suggestions)

In MySQL you need to create a database

Syntax: create database database\_name;

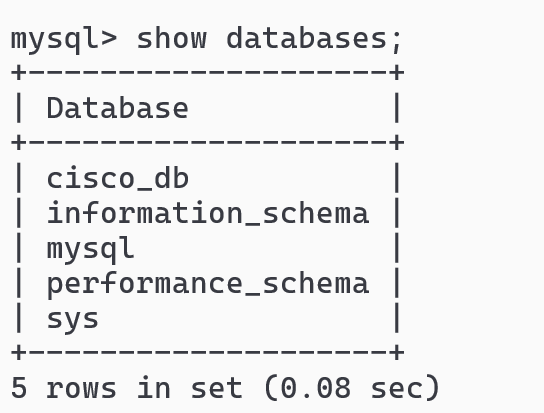
You can list all the database names using following command

show databases;

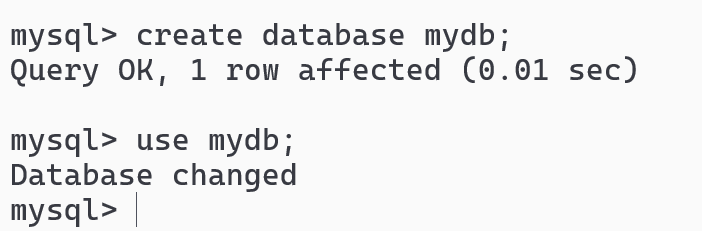
Switch to the database

Syntax: use database\_name;

Show databases



Create a database and switch to the database

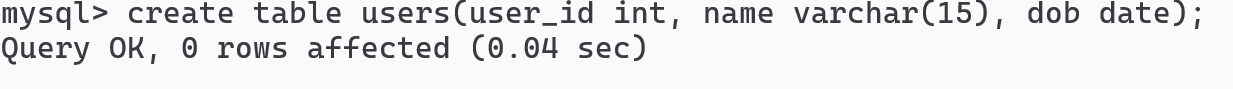


Syntax to create table

create table table\_name (column\_name type [constraint], column\_name type [constraint]);

table\_name can be like employee, customer, loan, profiles and etc

type: these are built-in types supported by the database which specifies what kind of value you can store like int, bigint, double, float, varchar, char, date, timestamp.

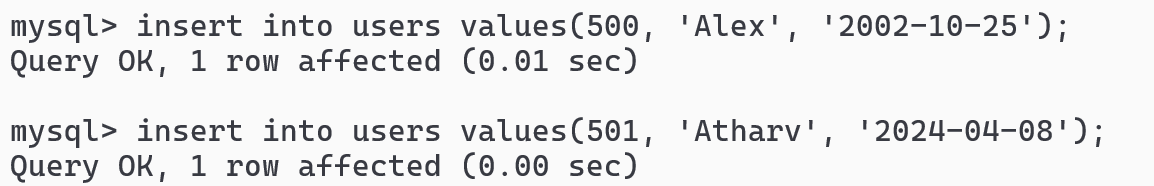


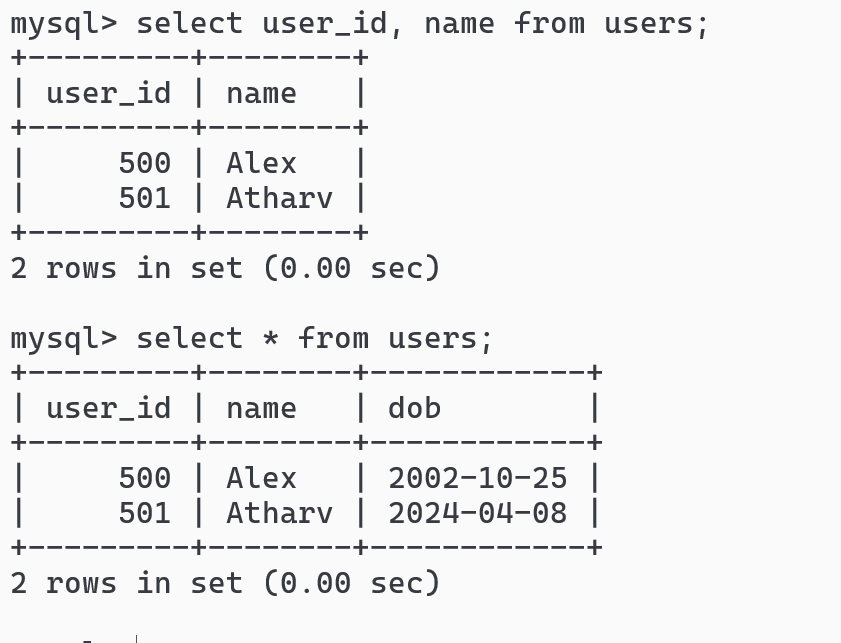
Insert & Select commands

Syntax: insert into table\_name values(v1, v2, v3,…)

Syntax: select column, column from table\_name or select \* from table\_name;

Note: Date must be stored in quotes in a yyyy-MM-dd format which is an ISO format that is globally supported





Suppose you want to add columns then you can use alter command

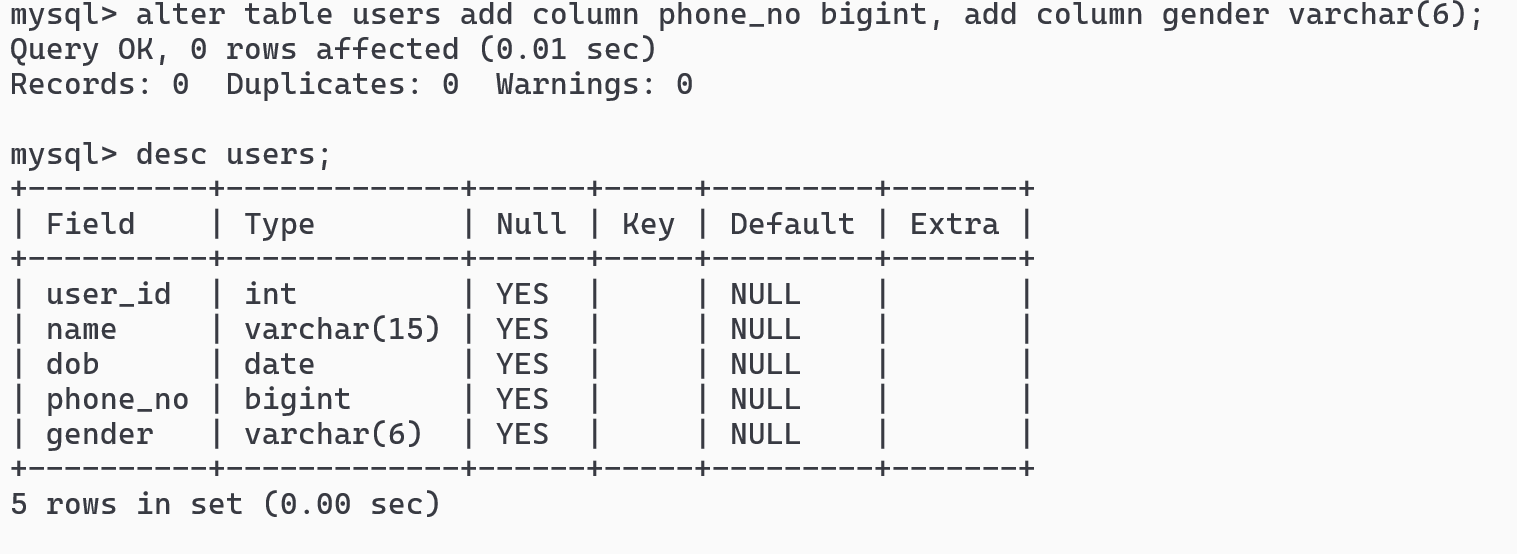
Syntax:

alter table table\_name add column column\_name type, column column\_name type,…

Example

alter table users add column phone\_no bigint, add column gender varchar(6);

Note: if the table is already present then the new columns get default values like NULL

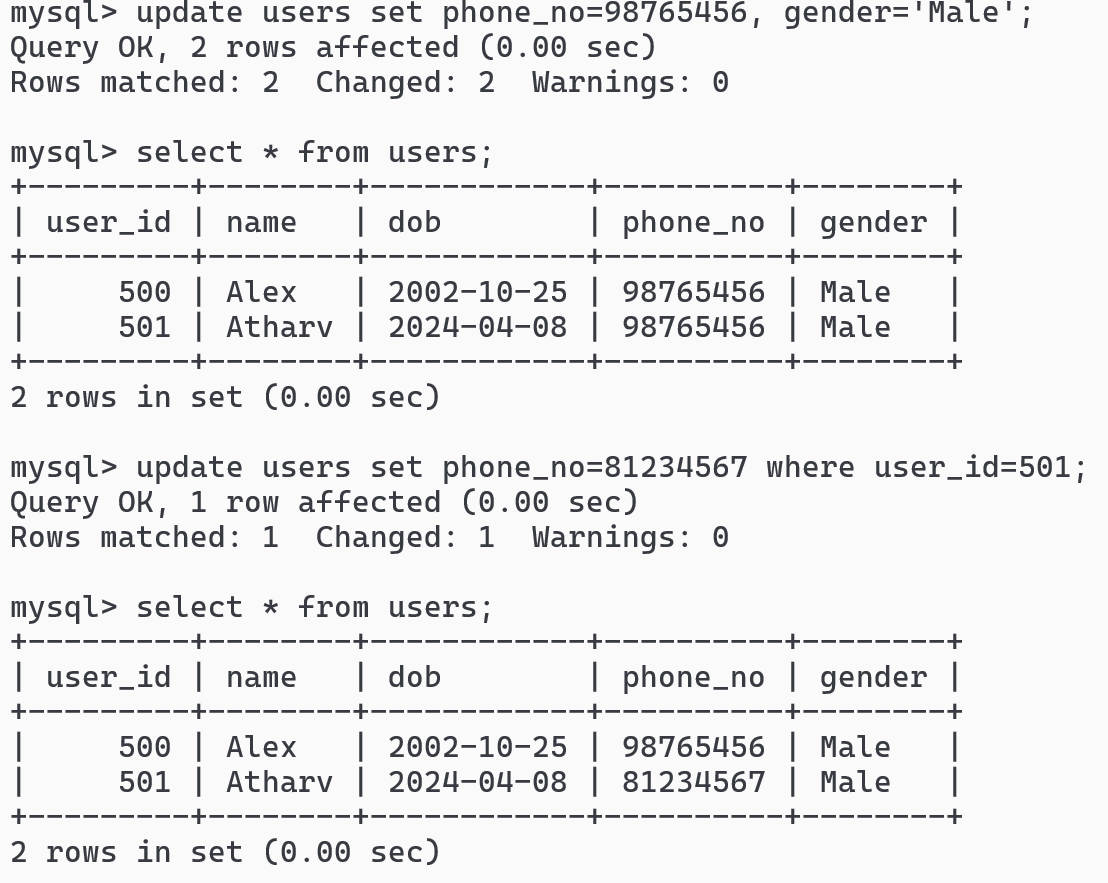


Update command:

You can update one or more columns

Syntax:

1. update table\_name set column\_name=value;
2. update table\_name set column\_name=value, column\_name=value;
3. update table\_name set column\_name=value where column\_name=value;
4. update table\_name set column\_name=value, column\_name=value where column\_name=value;



Alter command to remove the columns

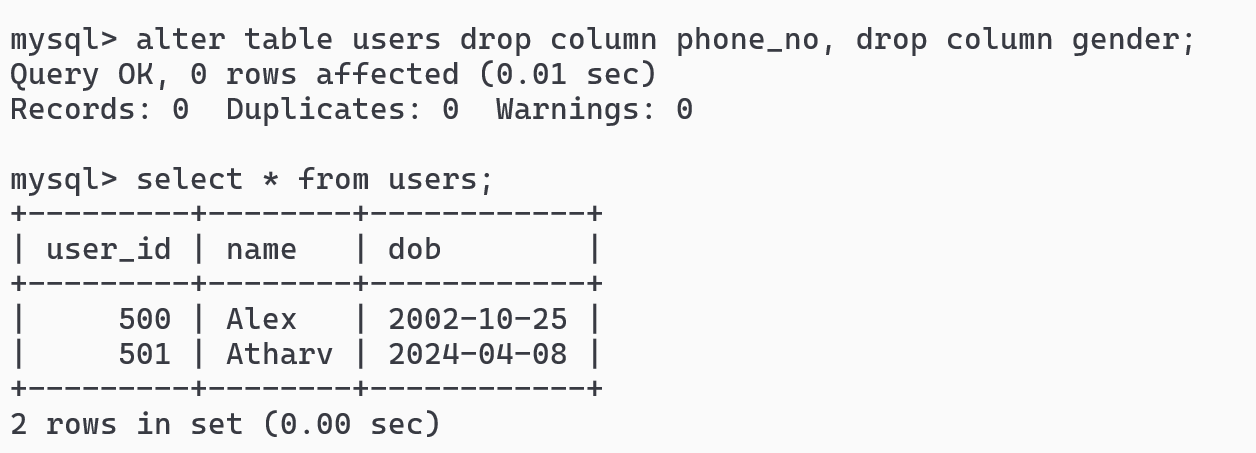
Syntax:

1. alter table table\_name drop column column\_name;
2. alter table table\_name drop column column\_name, drop column column\_name;

Example:

alter table users drop column phone\_no, drop column gender;

Note: If you drop the columns all of its data associated with the columns are also deleted

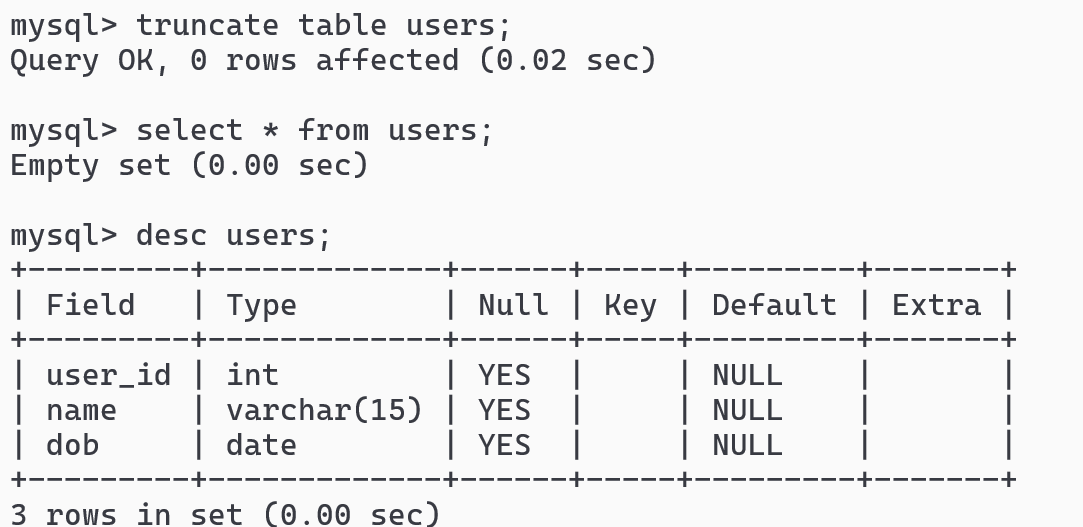


Truncate:

It is used to permanently delete all the records in a table, which you can’t undo, it retains the structure of the table

Syntax: truncate table table\_name;

Example: truncate table users;



Note: We also have delete command, but delete command can delete based on the conditions like one or more records and also delete command can be rolled back i.e., you can undo the delete

Note: truncate command deletes all the records and you can’t delete based on the condition, you can’t rollback the truncate

Drop:

It is used to permanently delete the table

Syntax: drop table table\_name;

Example: drop table users;



Constraints:

Constraints are the rules that can be applied for a table or a column, below are the constraints many database support



1. primary key: it can’t be duplicate, it is used to uniquely identify the records & it doesn’t support null
2. unique: it can’t be duplicate, however it supports null
3. not null: a column that mandatorily needs a value
4. check: a column that must have values that are defined in the check
5. foreign key: a table that is linked with another table

auto\_increment: It is a keyword that can be used for a primary key to automatically generate the value, by default it starts from 1, but you can start from any value.

Create command with constraints and auto\_increment

Syntax:

1. CREATE TABLE table\_name (column\_name type PRIMARY KEY AUTO\_INCREMENT, column\_name type NOT NULL, column\_name type CHECK( condition ), column\_name type UNIQUE)

The above table starts primary key from 1

1. CREATE TABLE table\_name (column\_name type PRIMARY KEY AUTO\_INCREMENT, column\_name type NOT NULL, column\_name type CHECK( condition ), column\_name type UNIQUE) AUTO\_INCREMENT=500

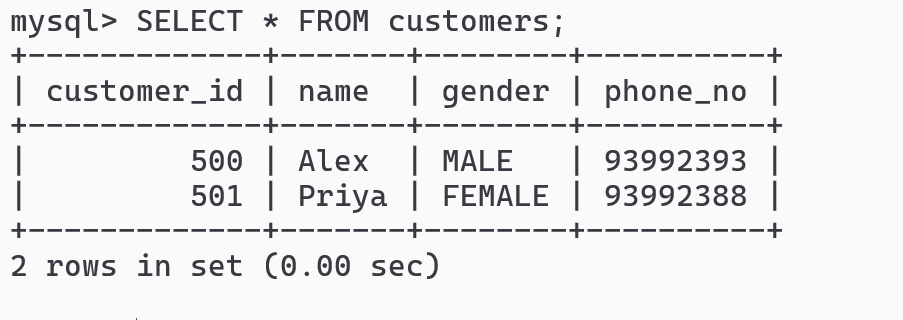
Example

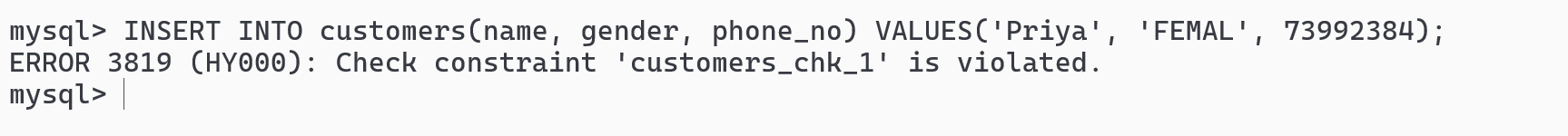
CREATE TABLE customers(customer\_id int PRIMARY KEY AUTO\_INCREMENT, name varchar(20) NOT NULL, gender varchar(6) CHECK (gender IN (‘MALE’,’FEMALE’)), phone\_no bigint UNIQUE)AUTO\_INCREMENT=500;

To store the records

INSERT INTO customers (name, gender, phone\_no) values (‘Alex’, ‘Male’, 939393)







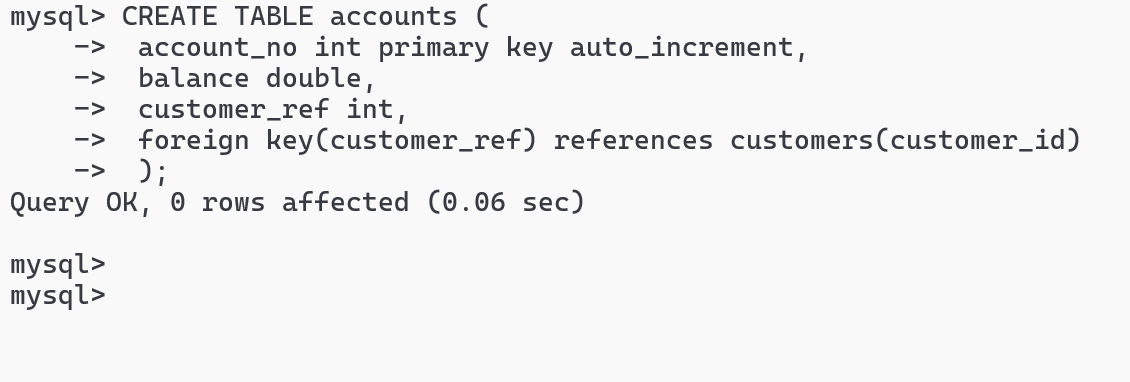
Foreign key constraints

It is used to link a child table from the parent table using the primary key of the parent table.

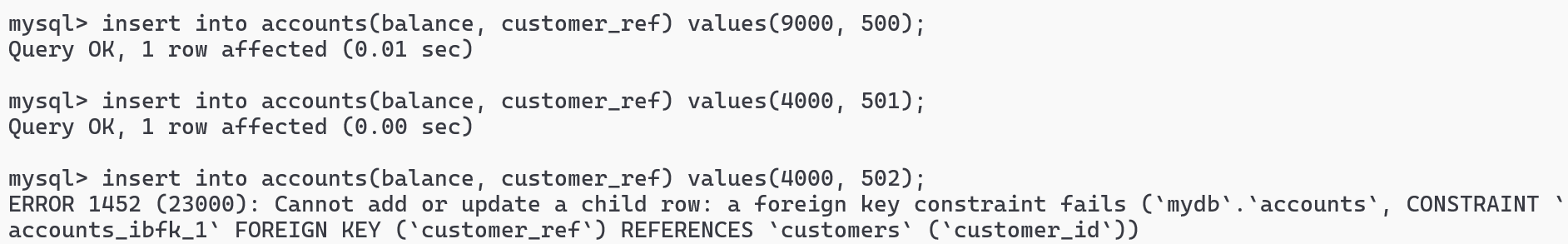
Syntax:

CREATE TABLE table\_name(column type, … column type, foreign key references parent\_table(primary\_key\_column));

Example



Note: Foreign key column must be the entries present in the parent table, it can be null and you can also have duplicate values to the foreign key



Backup and Restore

Backup is required to ensure that when anything goes wrong you could able to recover the data

The command used to take the backup is

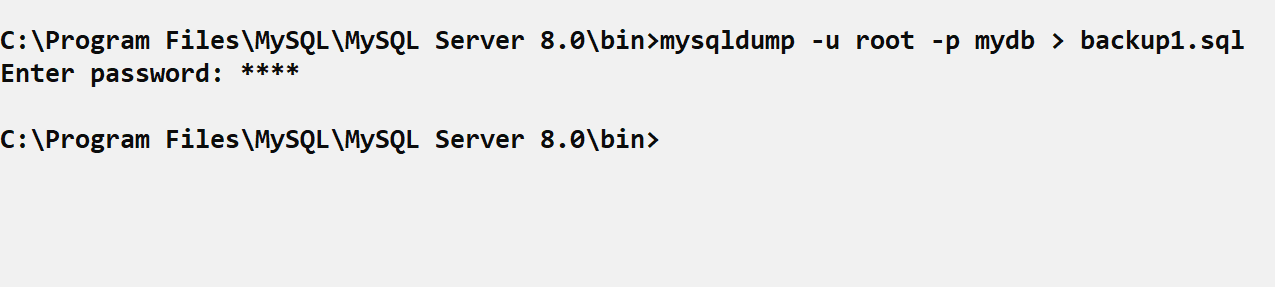
mysqldump -u root -p database\_name > file\_name.sql

To restore the command is

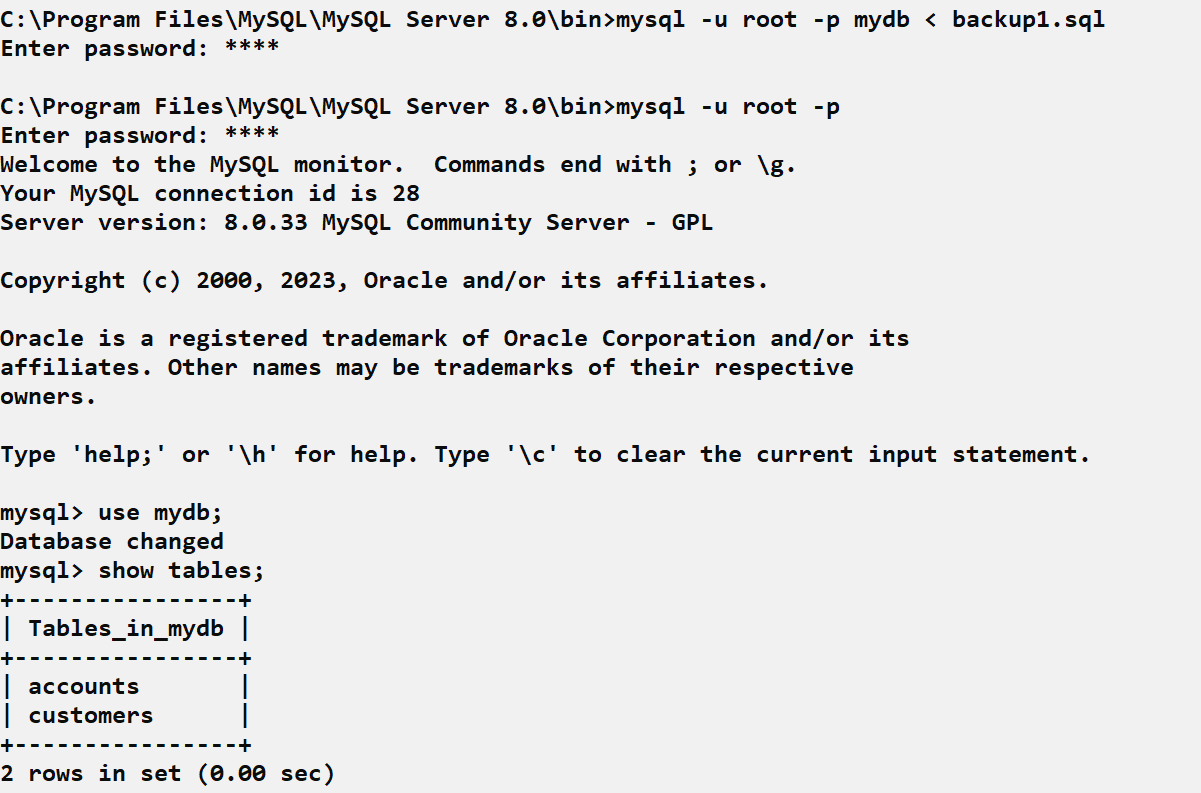
mysql -u root -p database\_name < file\_name.sql

Note: mysqldump is the command present in the C:\Program Files\MySQL\MySQL Server 8.0\bin in windows, for mac users it will be present in /usr/local/mysql/bin

Open the command prompt in the admin privilege, for mac use sudo



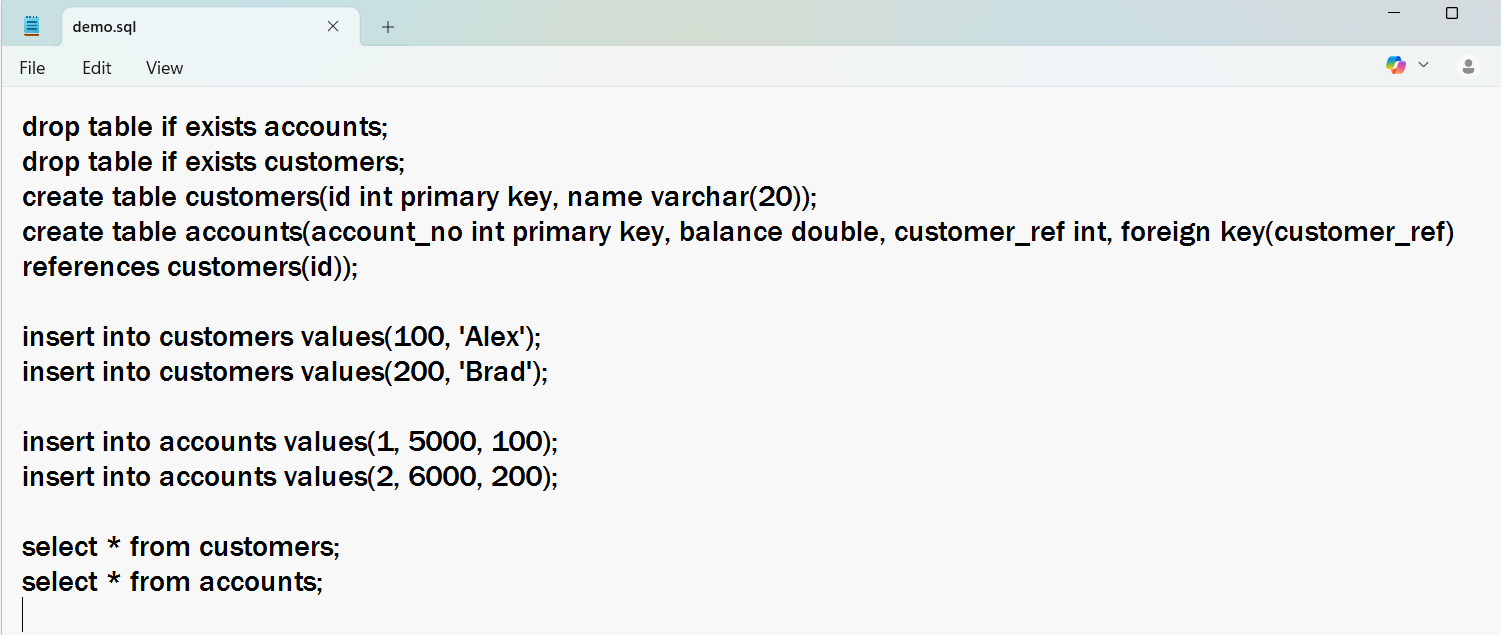
Now drop all the tables in mydb and then issue the following command from the command prompt.



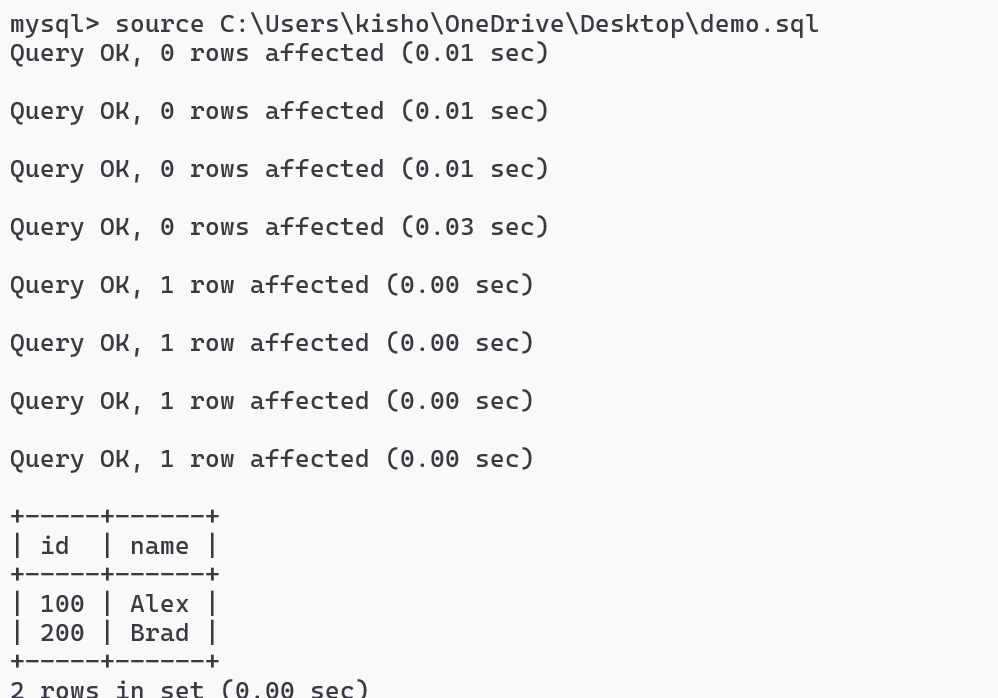
Running the SQL script files

Script file: It is an SQL file that will have SQL commands which you can run to execute all the commands present in it

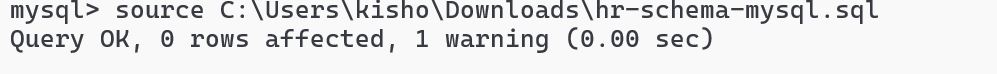
You can create demo.sql file in any location



To run the script file user source path



HR schema: it has some tables like employees, departments, jobs and etc with some records



Built-in functions in MySQL

There are two types of built-in functions

1. Single row functions: these functions return single result for each row

upper(), lower(), concat(), length(), date\_format()

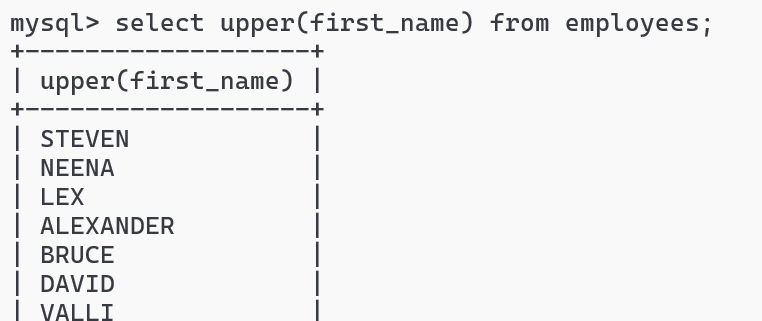
1. Aggregate functions: these functions combine all the rows and return a single result

count(), sum(), max(), min(), avg()

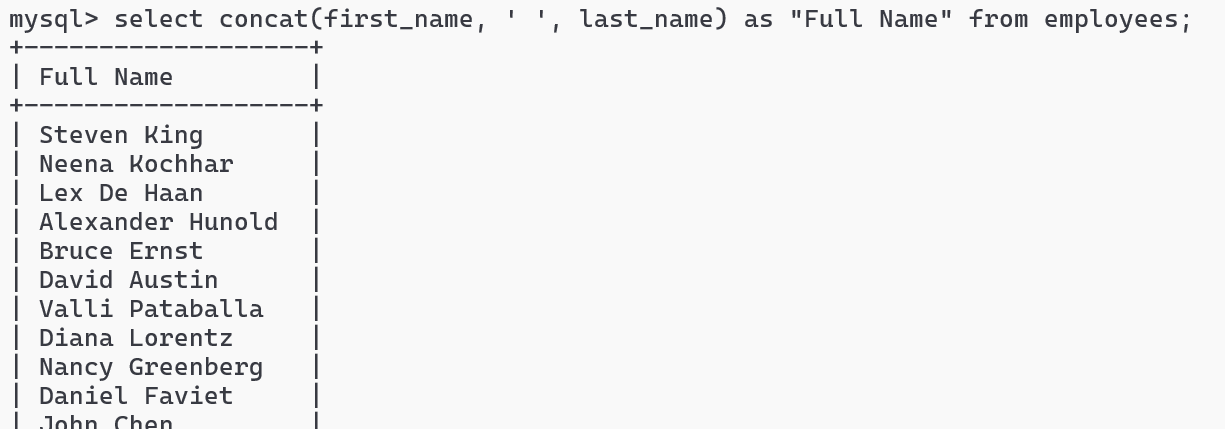
Using count



upper()



concat(arg, arg, arg): this concatenates multiple arguments and returns result for each row



Select comes with various clauses like - where, order by, between, or, in, and, having, group by and etc.

Order by salary:

select first\_name, last\_name, salary from employees order by salary;

Where

select employee\_id, first\_name, last\_name, salary from employees where employee\_id = 105;

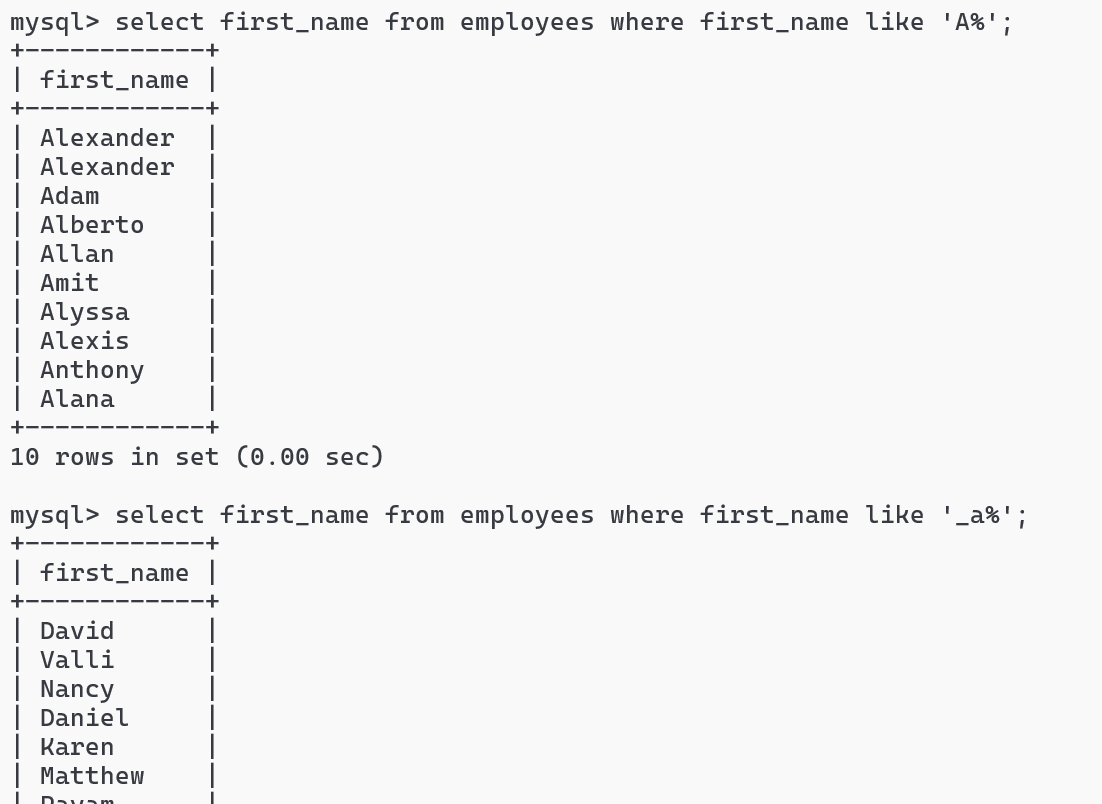
Between

select employee\_id, first\_name, last\_name, salary from employees where employee\_id between 100 and 110;

Like operator

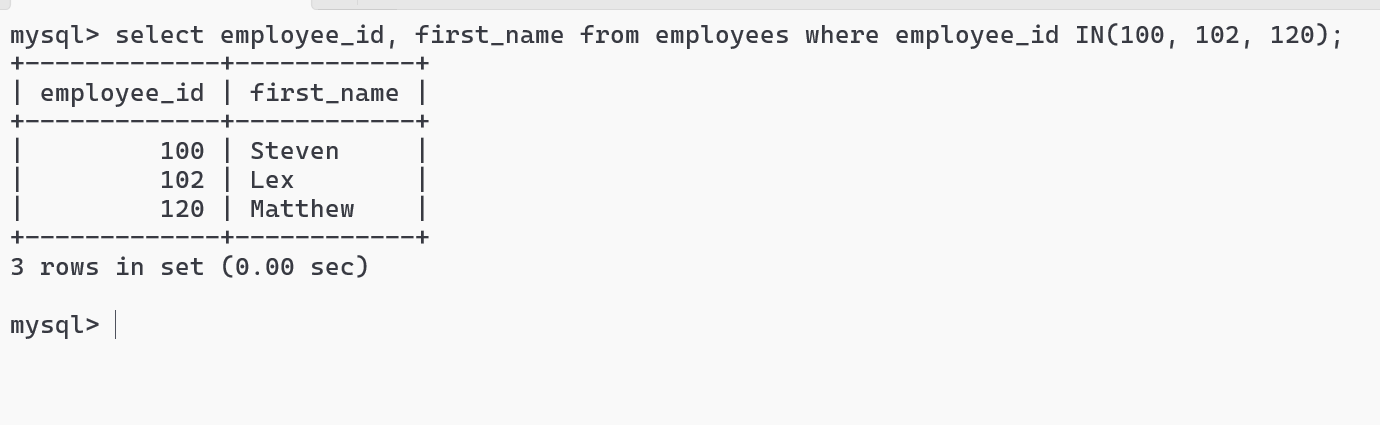
% any number of characters

\_ a single character that can be anything.



IN operator

select employee\_id, first\_name from employees where employee\_id IN (101, 102, 120);



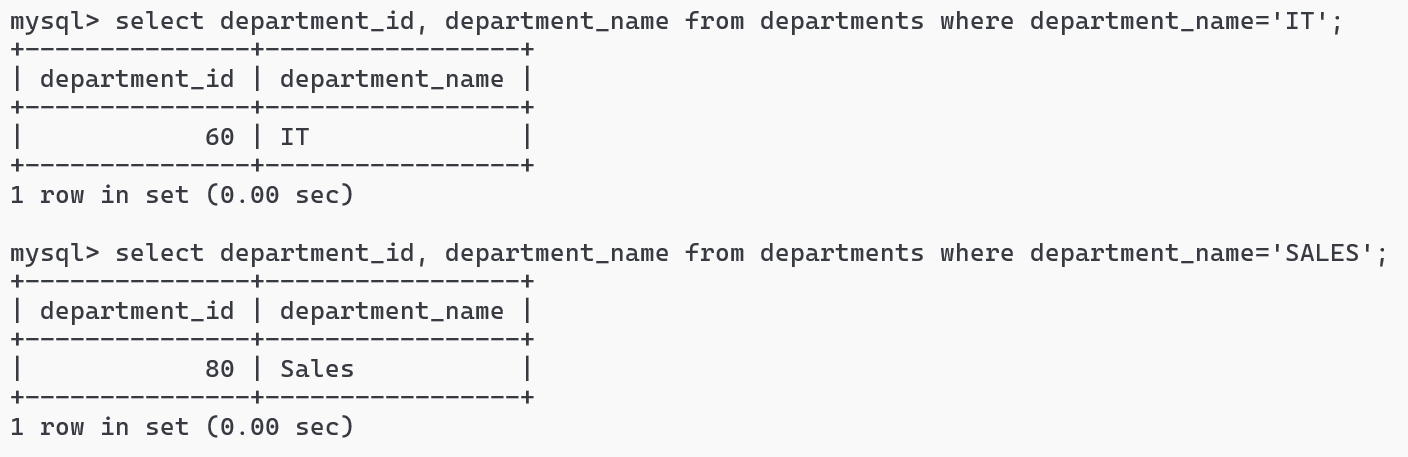
Sub queries or Nested queries

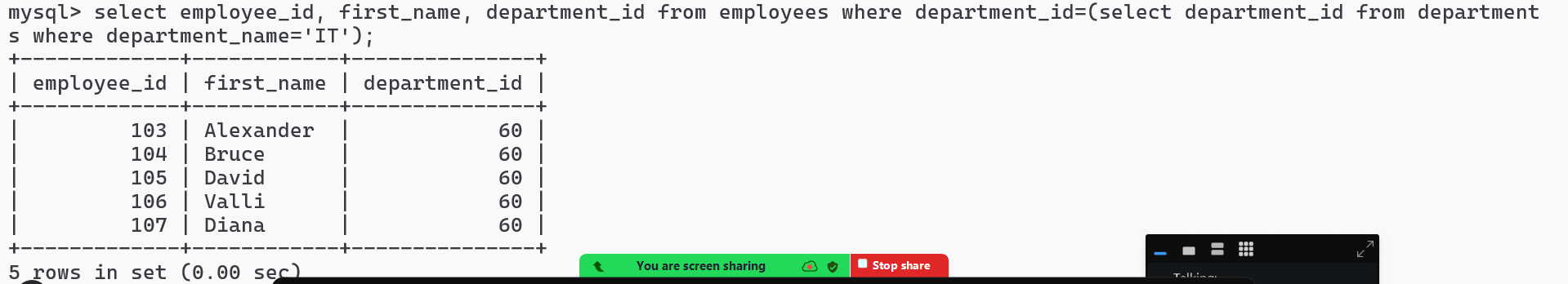
Sometimes a query needs a condition after executing another query

ex: You want all the employees belonging to IT or SALES, ACCOUNT department

Syntax:

outer query condition (inner query)





Activity:

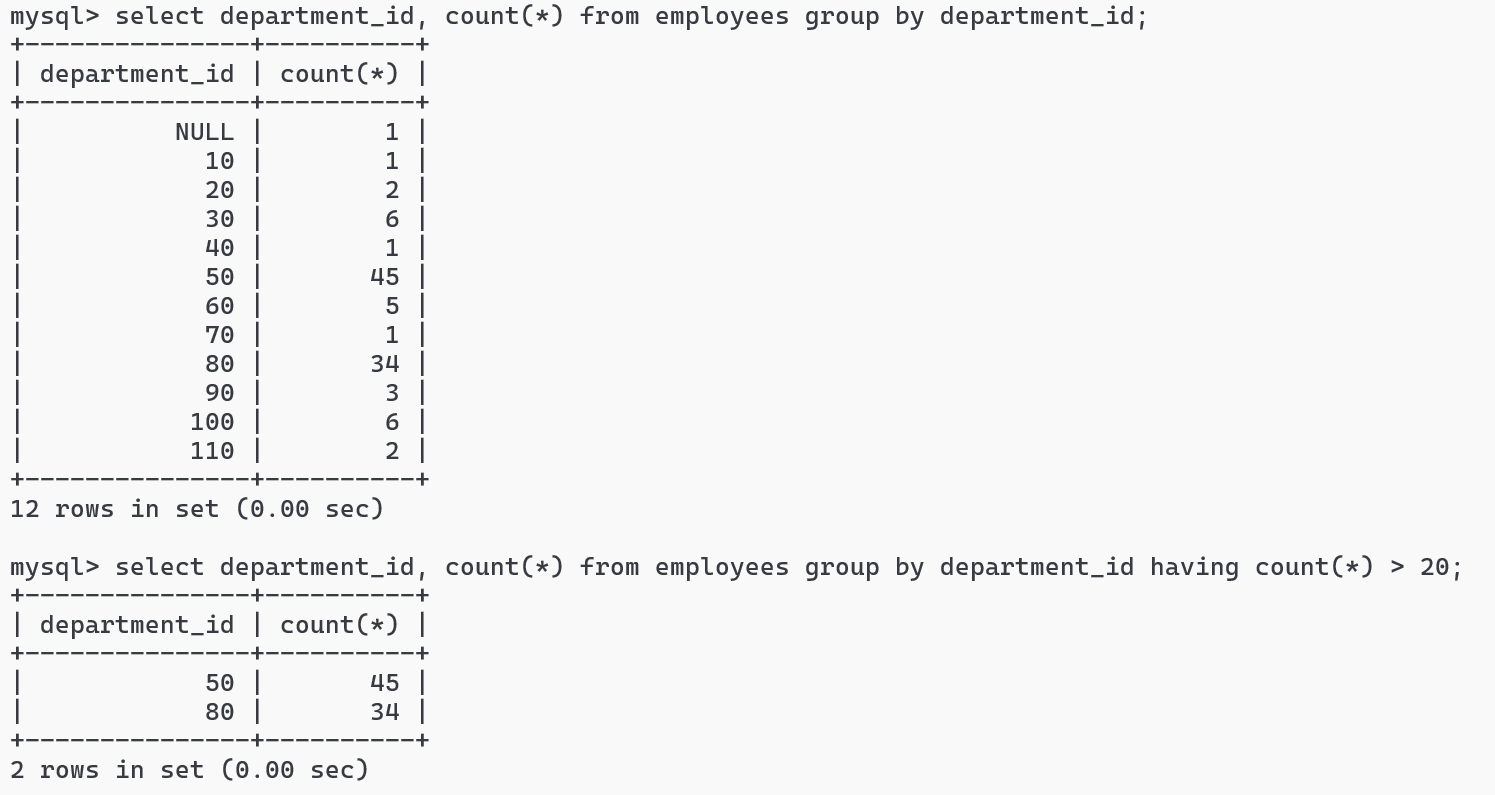
1. Show all the employees working is Canada
2. Show all the employees working in UK
3. Show all the employees belonging to FINANCE, IT and ACCOUNTING

GROUP BY & HAVING

It is used to group rows that have same values in one or more columns, it is often combined with aggregate functions.

HAVING is used when you want to apply condition with aggregation function

Total number of employees in each department and having count > 20.



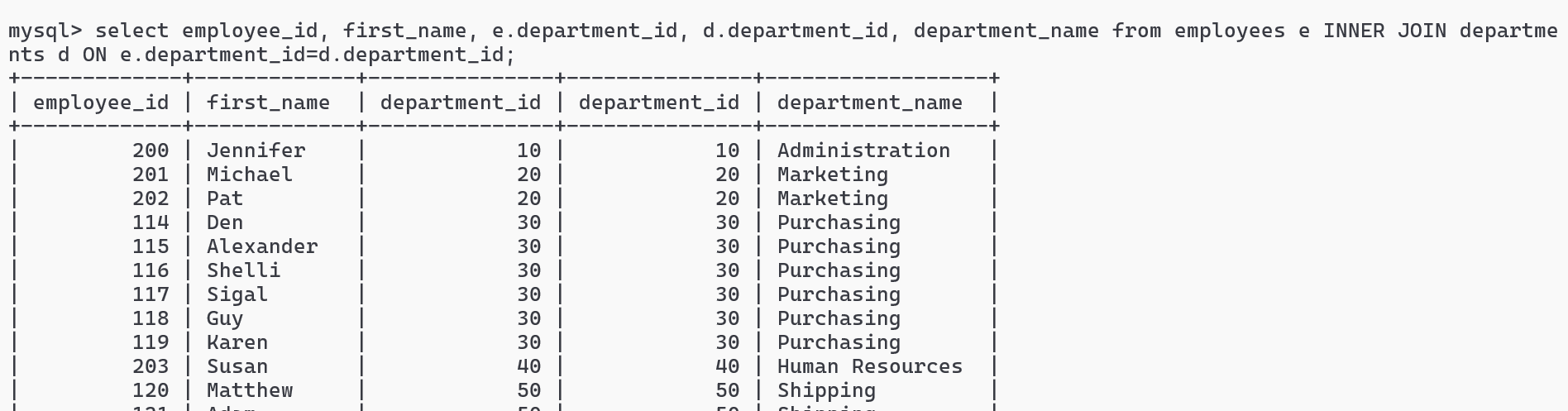
SQL Joins

It is used to join two or more tables and generate result based on some common value, following are the types of join

1. INNER JOIN
2. LEFT JOIN
3. RIGHT JOIN
4. FULL JOIN
5. CROSS JOIN

JOIN syntax

INNER JOIN: produces the result of left & right table of matching values



Activity:

Try LEFT, RIGHT and CROSS JOIN for employees & departments table and observe the output

Note: FULL JOIN is not supported in MySQL

CROSS JOIN: This gives you the cartesian product of two tables, where every row of a table is combined with every row of another table

Suppose you have 2 tables, A & B

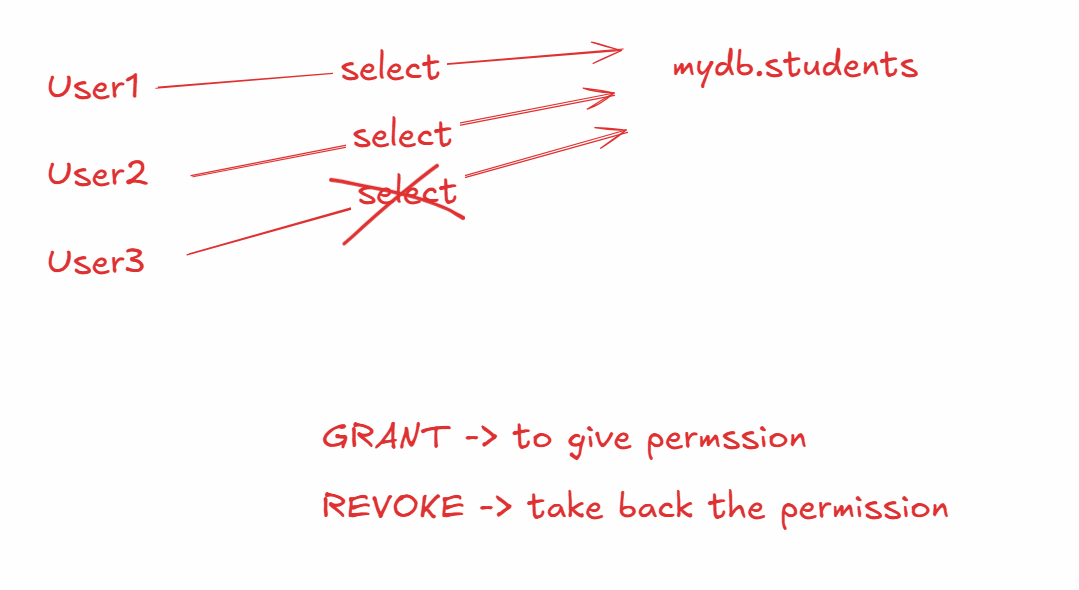
A has 2 rows

B has 3 rows

Then A cross join B results 6



DCL: Data Control Language - It is mainly used to GRANT or REOVKE permissions



Firstly you must create a user and provide a password for the user to login

Creating the user

CREATE USER ‘user1’@’localhost’ IDENTIFIED BY ‘Welcome123’;

GRANT permission

GRANT SELECT ON mydb.students to ‘user1’@’localhost’;

USER can login using mysql -u username

REVOKE permission

REVOKE SELECT on mydb.students from ‘user1’@’localhost’;

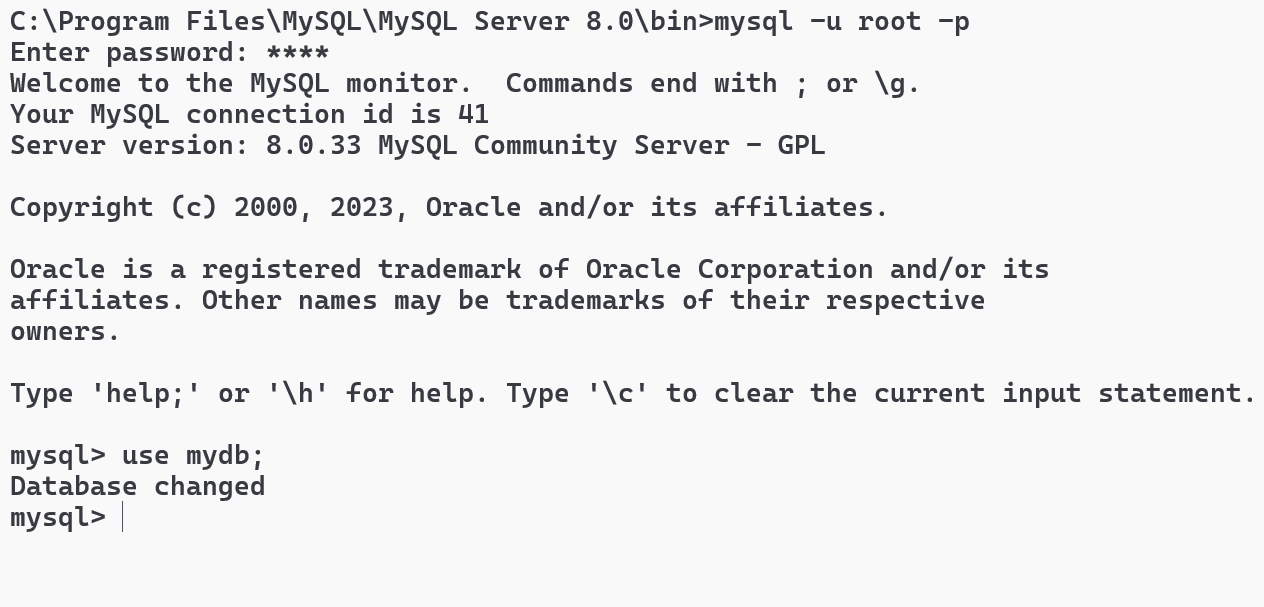
NOTE: You must use FLUSH PRIVILEGES to apply the permission after grant or revoke

NOTE: You can provide user to perform more than one operation using GRANT SELECT, INSERT on mydb.students to ‘user1’@’localhost’;

NOTE: You can use database\_name.\* if you want to select all the tables

ex: mydb.\* will select all the tables

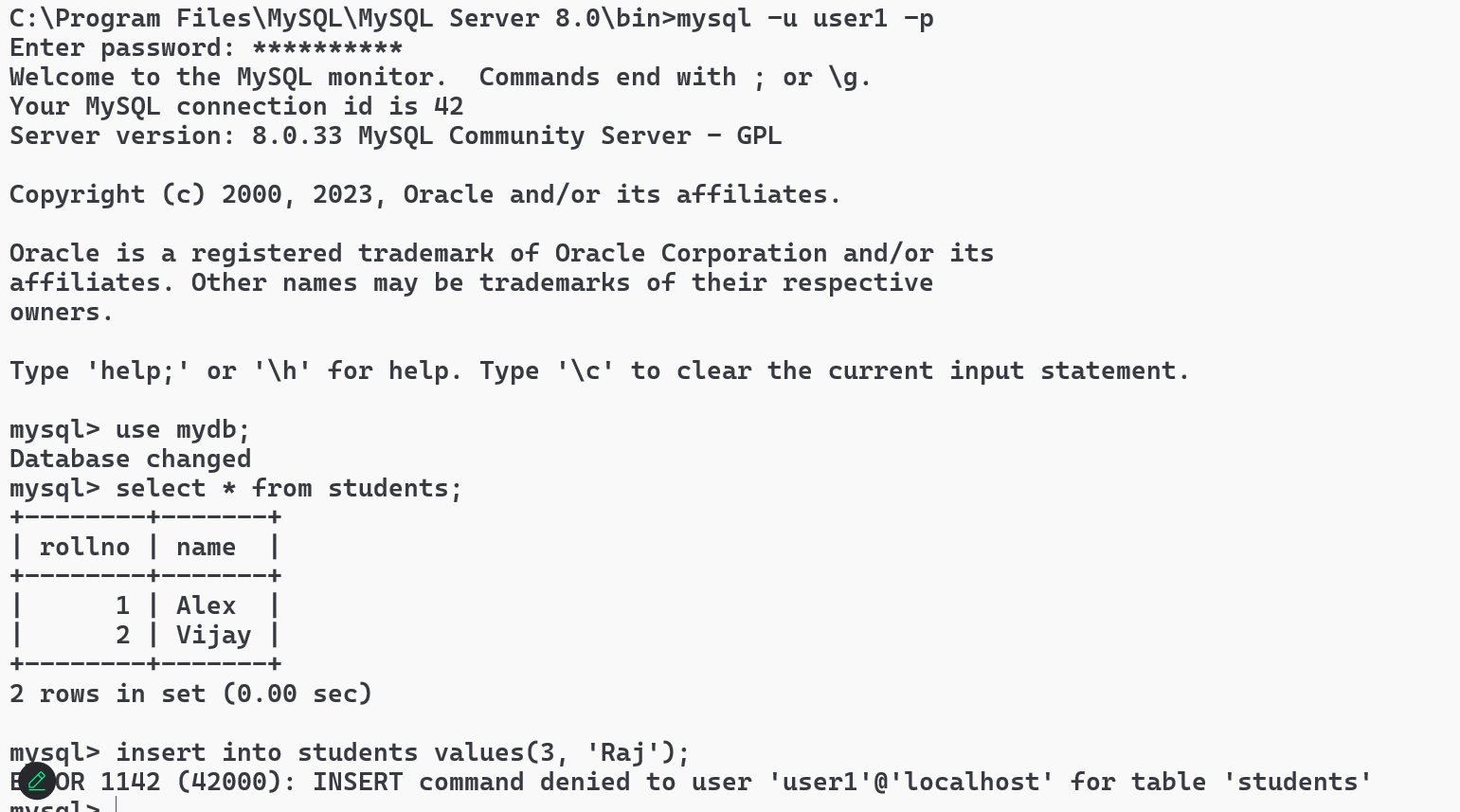
From the command prompt navigate to the mysql-server/bin folder



Grant SELECT permission & FLUSH privileges

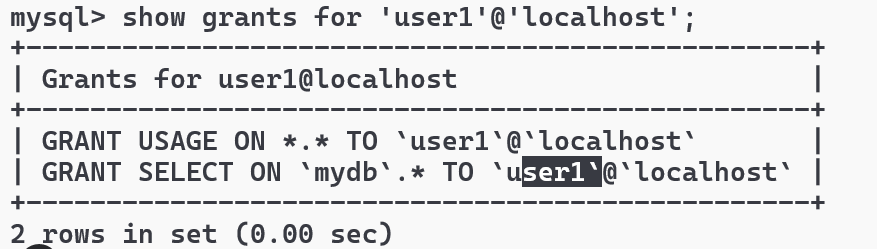


LOGIN as a user1 and try select & other commands on a table present in mydb

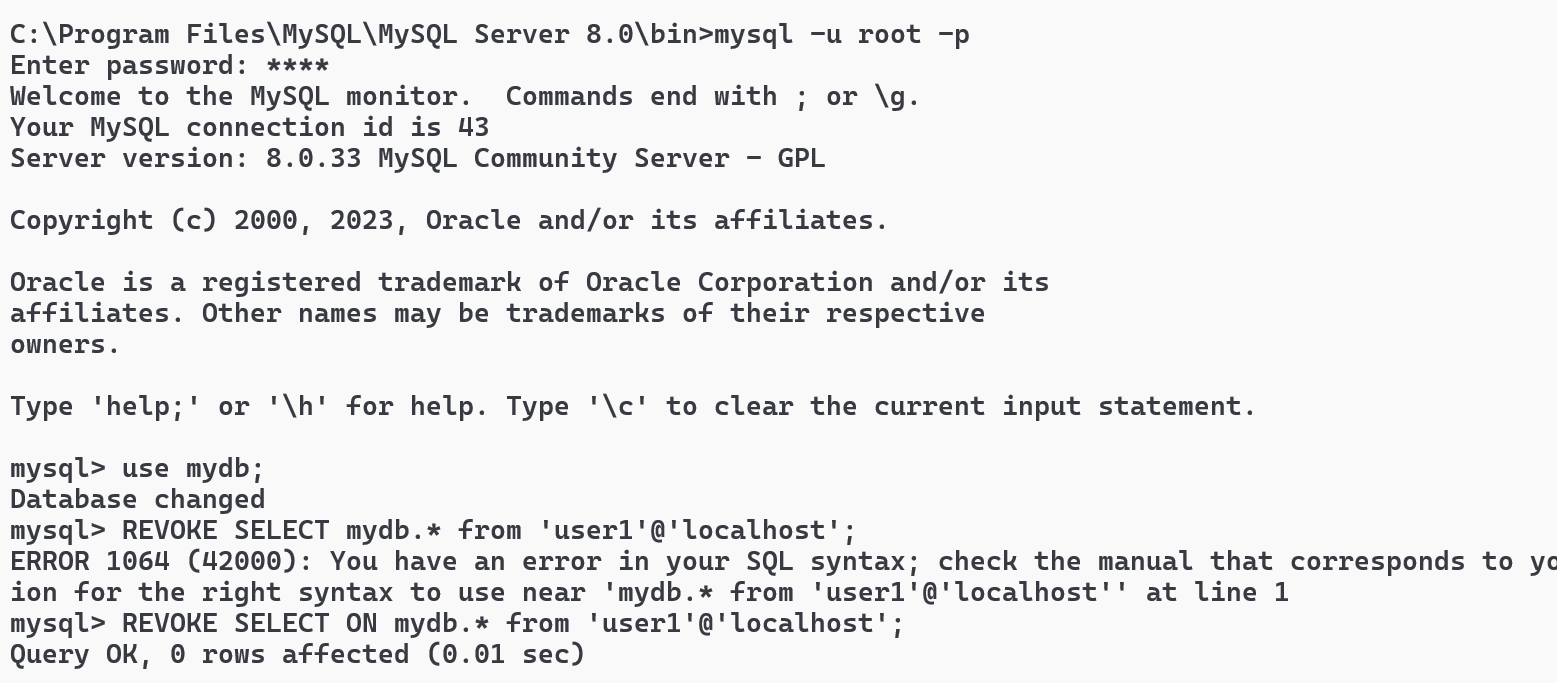


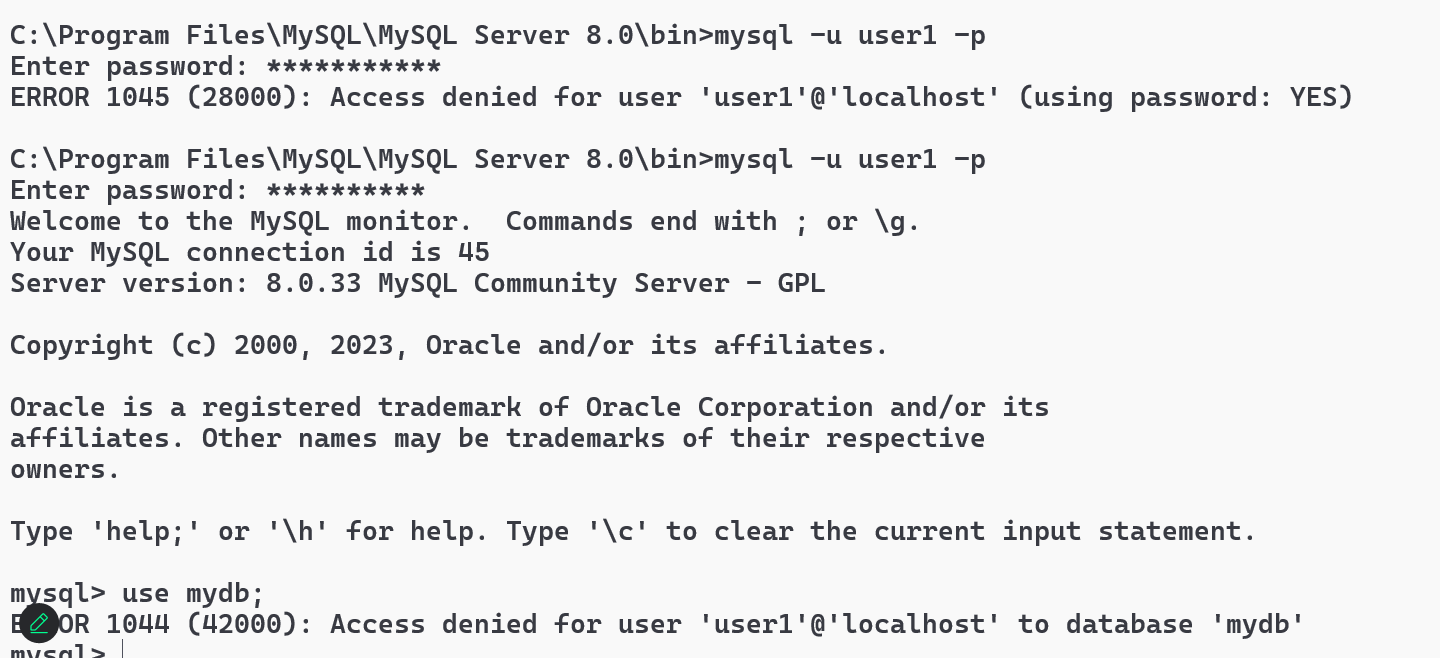
Note: You can see the permission denied error to user1

SHOW GRANTS to view what permissions the user has



REVOKE Command

Firstly you must login as root and then revoke, later login as the user1 

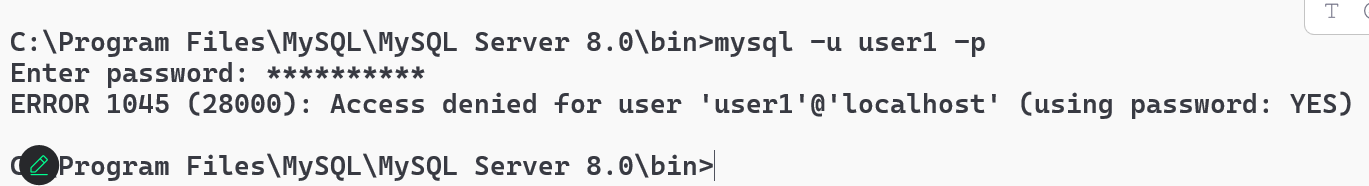


Note: Since user1 doesn’t have access to any tables in mydb, it is not allowing to use mydb

Finally you can drop the user so that he can’t login

Activity: Login as root & drop the user1 and then try to login as user1





Web technologies

1. HTML
2. CSS
3. Javascript

HTML: It is mainly to display the content

CSS: It is mainly to style the content

Javascript: It is used to add interactivity to the websites

HTML

Hyper Text Markup Language, it provides built-in tags to structure the web page

<html>  
<head>   
 <title>some title</title>  
 <link>..</link>  
</head>  
<body>  
 content of the web page  
</body>

</html>

Software requirement

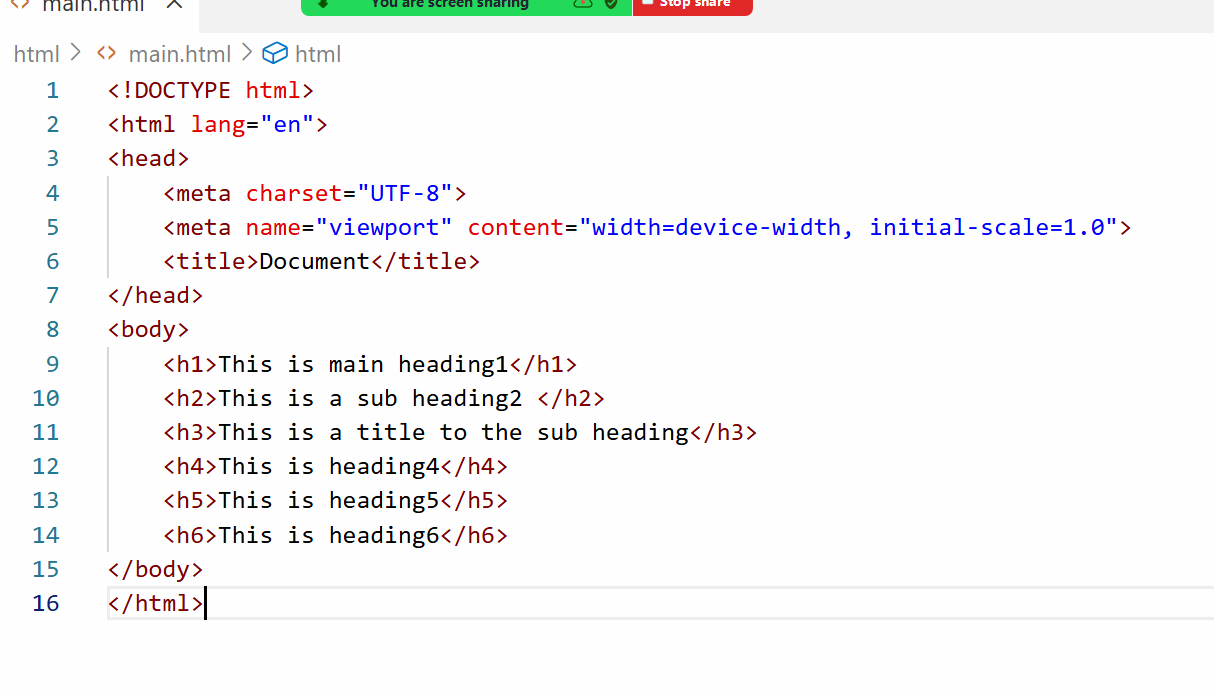
1. VS Code

HTML headings

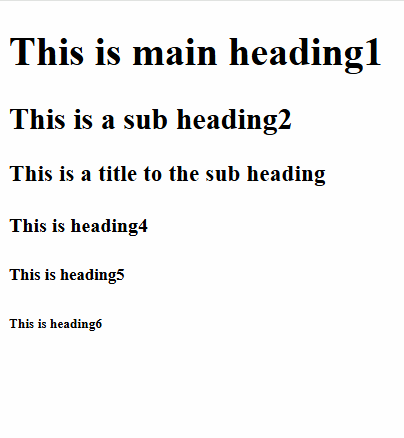
There are 6 heading tags

<h1>, <h2>, <h3>, <h4>, <h5>, <h6>

main.html



Output:



Lorem Ipsum: A short cut key to generate some random words

ex: lorem15 generates 15 random words

bold, italic, underline, strike tags

<b>, <i>, <u>, <stroke>

image tag

<img src = “URL” width = “200” height = “200” alt = “describe if image is not loaded”>

Subscript & Superscript tags

H<sub>2</sub>SO<sub>4</sub>

(a + b)<sup>2</sup>

HTML Entities: these display special characters

&pound; 10,000  
www.company.com &copy; 2025

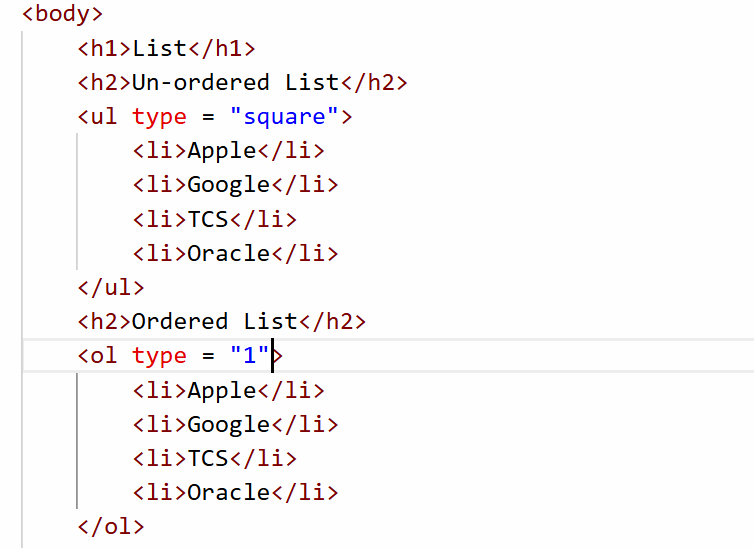
&#8377; 10,000

List tags

When you want to list some items you can use list tags, there are 2 types

1. ordered list <ol>
2. un-ordered list <ul>

Both the tags can have child tags like <li> which means list item or list index



Activity:

Try to create a nested list which will have following output

1. Operation Systems
   1. Windows
   2. Linux
   3. MacOS
2. Programming Languages
   1. Java
   2. Python
   3. C++

Media tags

<video> and <audio>

<audio controls>  
 <source src = “file.mp3” type = “audio/mp3”></source>  
</audio>

<video width = “300” height = “300” controls muted>  
 <source src = “file.mp4” type = “video/mp4”></source>  
</video>

List of attributes you can use in audio and video

muted: starts with sound off

autoplay: automatically starts the media

loop: repeats media automatically

Table tag

<table> to create a table, it has 2 tags for head of the table for heading & for body for the content of the table, you can use <thead> & <tbody> for heading & body, however to create row you must use <tr> and for columns you can use <th> or <td>

<table> supports attributes like border, cellpadding and cellspacing

cellpadding: it gives space between content and the border

cellspacing: it gives space between the cells

anchor tag

<a> an anchor tag is used to create an hyperlink that opens another page

<a href = “url”> opens another page

<a href = “url” target = “\_blank”> opens a page in a new tab

HTML Forms

HTML forms help user to enter data, using forms you can create text box, password field, radio buttons, check box, drop down, file upload field, buttons

<form> inside this form tag you can use many elements that accept the input.

form tag accepts 2 attributes

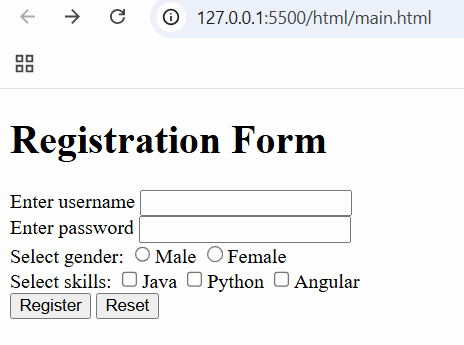
1. action: This takes the url of the server program usually Java, ASP.NET kind of programs
2. method: This is optional, there are two types of method you can use either GET or POST, by default method will be GET

GET vs POST

|  |  |
| --- | --- |
| GET | POST |
| The data will be sent via URL | The data will be send via Request body |
| The data will be visible to the end user in the browser address bar | The data will not be visible |
| This is insecure | This is secured |
| Limited characters | Unlimited characters |

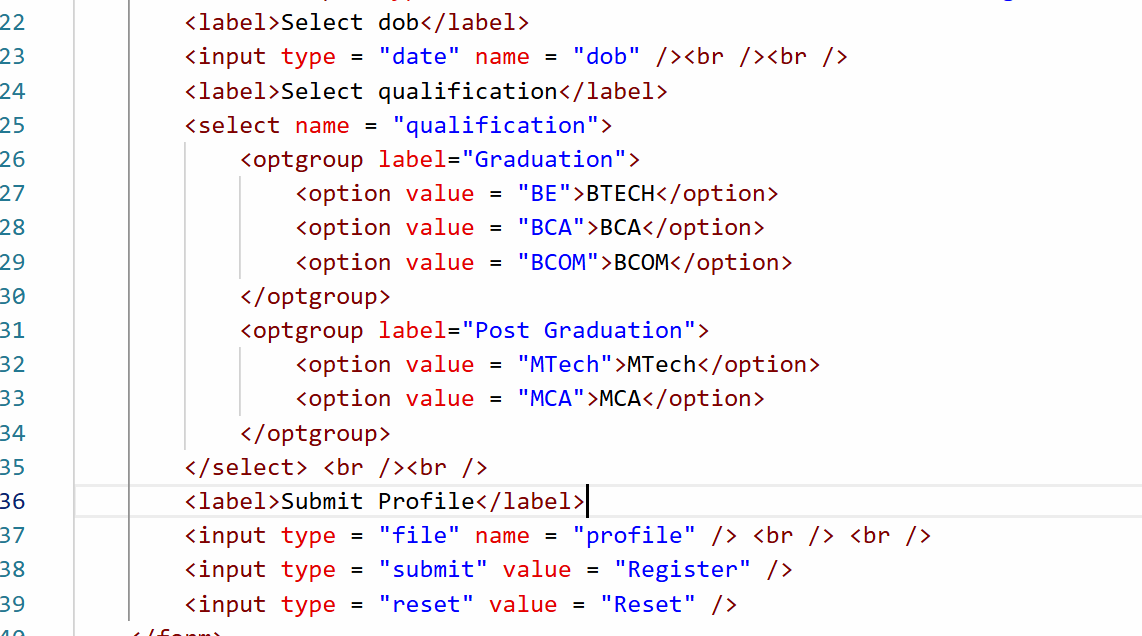


Output:

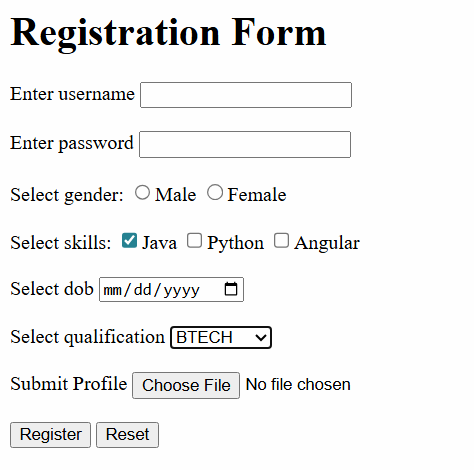


Other form controls are like date, file, email, range and so on

You can also create drop downs using <select> and <option> element



Output:



CSS

CSS stands for Cascading Style Sheet, it adds styles to the HTML elements, CSS has following selectors

1. element selector: selects the element by the name
2. class selector: selects the class associated with the element, multiple elements can have same class
3. id selector: selects the id associated with the element, id must be unique

Types of CSS

1. Inline CSS - You apply style to a particular element using style attribute  
   <h1 style = “color: blue”>Heading</h1>
2. Internal CSS - You can apply style to an entire HTML document  
   <style>  
    h1 { color : red; background-color : yellow }  
   </style>
3. External CSS - You can apply styles to multiple HTML documents  
   You will create an external css file i.e, global-styles.css

h1 { color : blue; background-color: yellow }  
h2 { color : green; background-color: yellow }

Note: HTML document must use <link> tag to refer the external css file

css/global-styles.css



main.html



Output:



What happens when style is applied in externa, internal and inline

Inline overrides internal style and internal style overrides external style.

Activity:

Create a table, an ordered list element and a form element, apply CSS styles for all of them.

class & id

These are the selectors which you can use to apply styles to the element

class: You can have multiple elements using the same class, so that they all will have common styles

id: you need to have unique id for any element

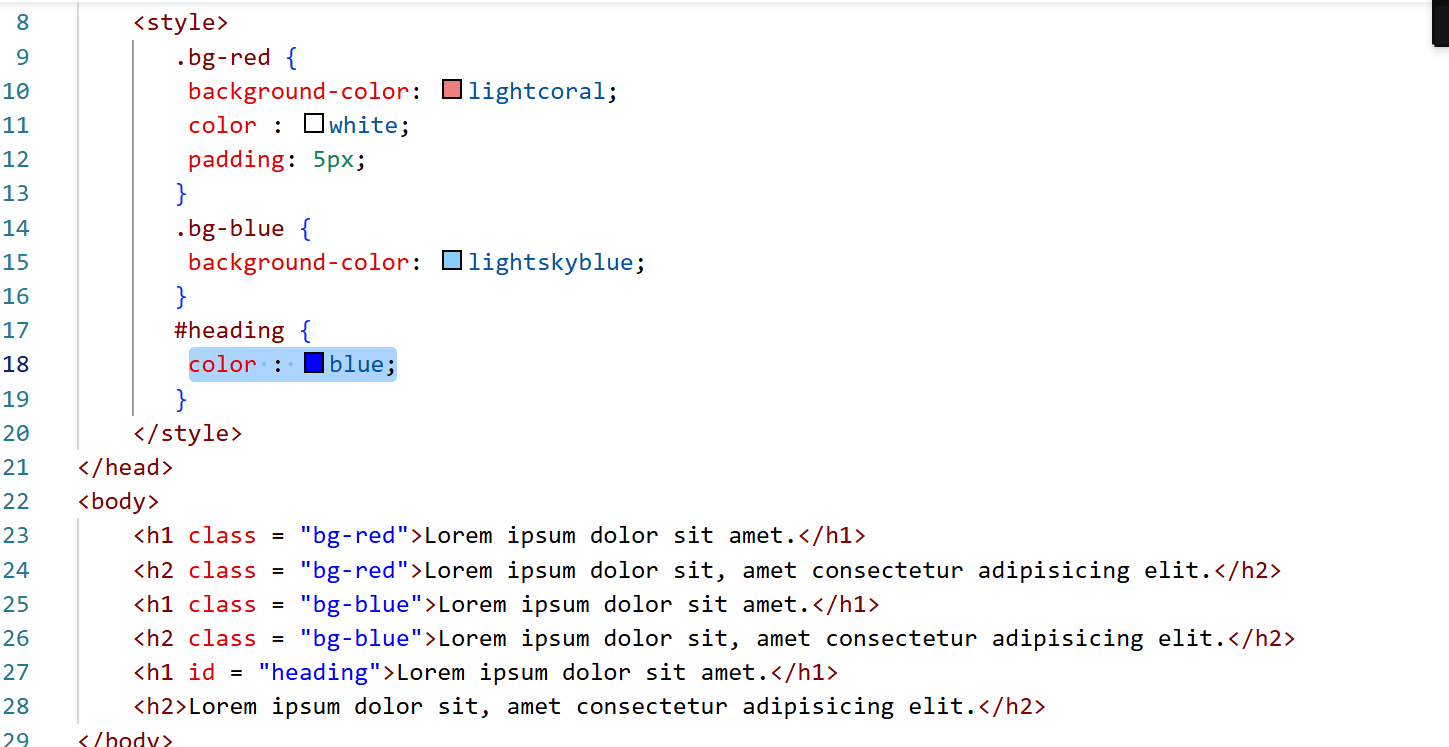
<p class = “text-danger”>Some content</p>  
<h1 class = “text-danger”>Some content</h1>

To apply CSS for class you must use dot followed by the class name

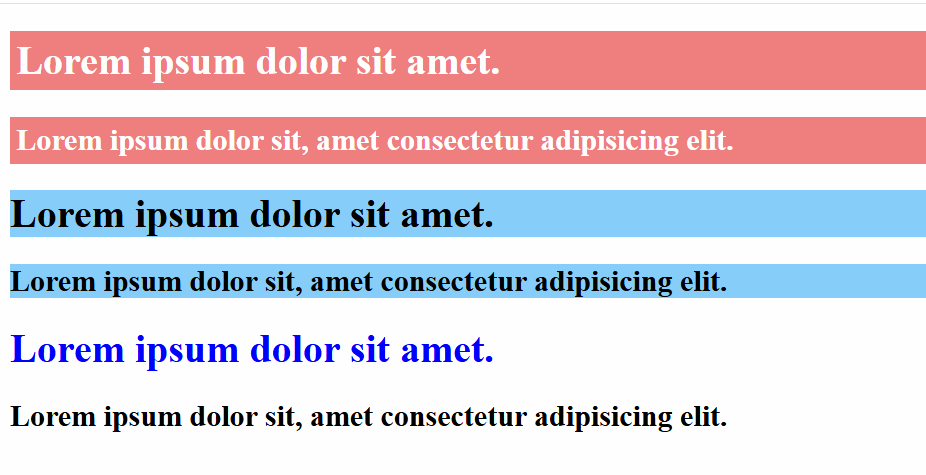
.text-danger { color : red }

<p id = “p1” >Some content</p>  
To apply CSS for id you must use # followed by the id name

#p1 { background-color : yellow }



Output:



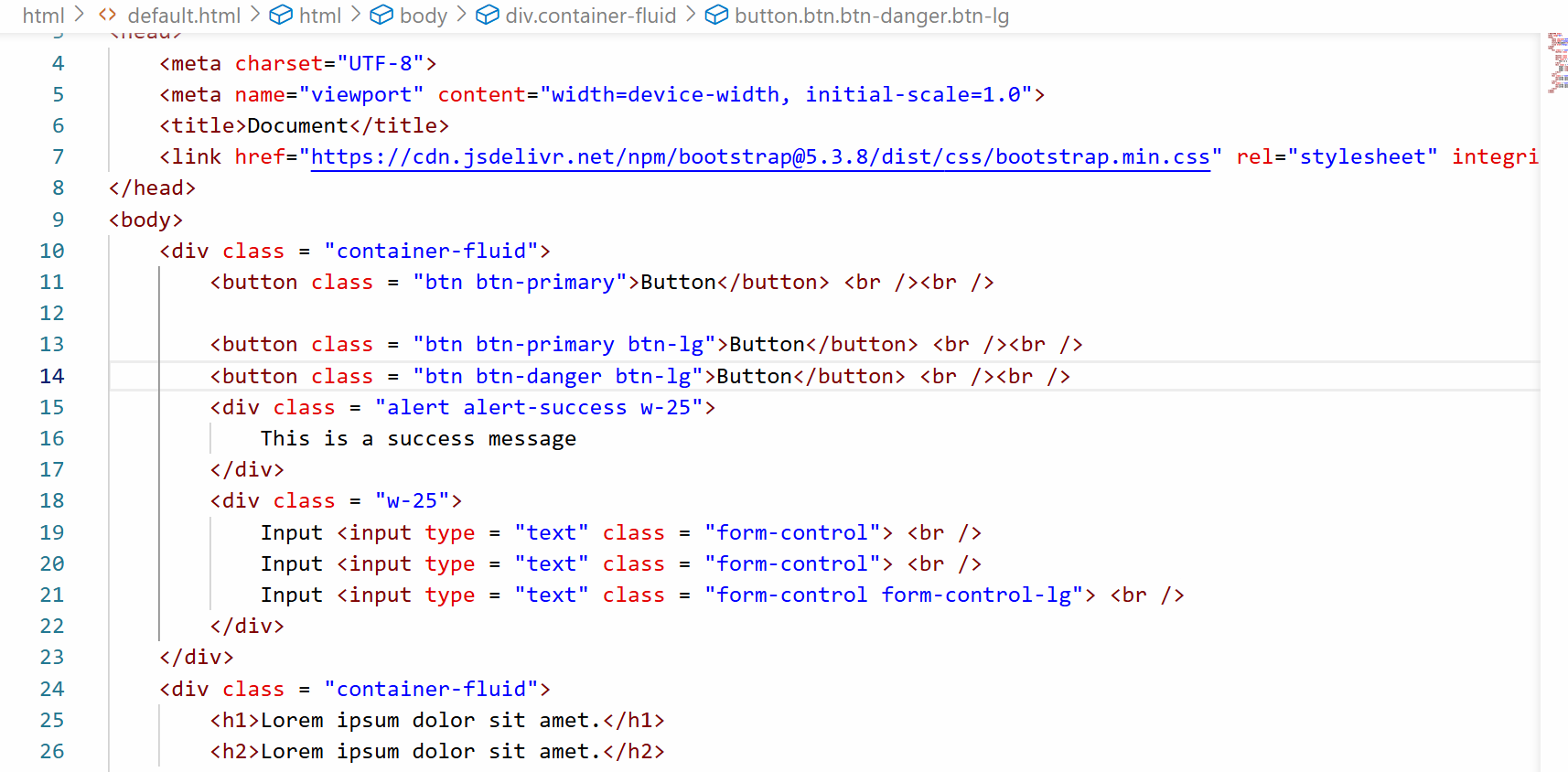
Bootstrap.js

It is a free library which you can use to apply CSS styles to your HTML elements, it gives lot of built-in classes that make your website to have a very good look and feel.

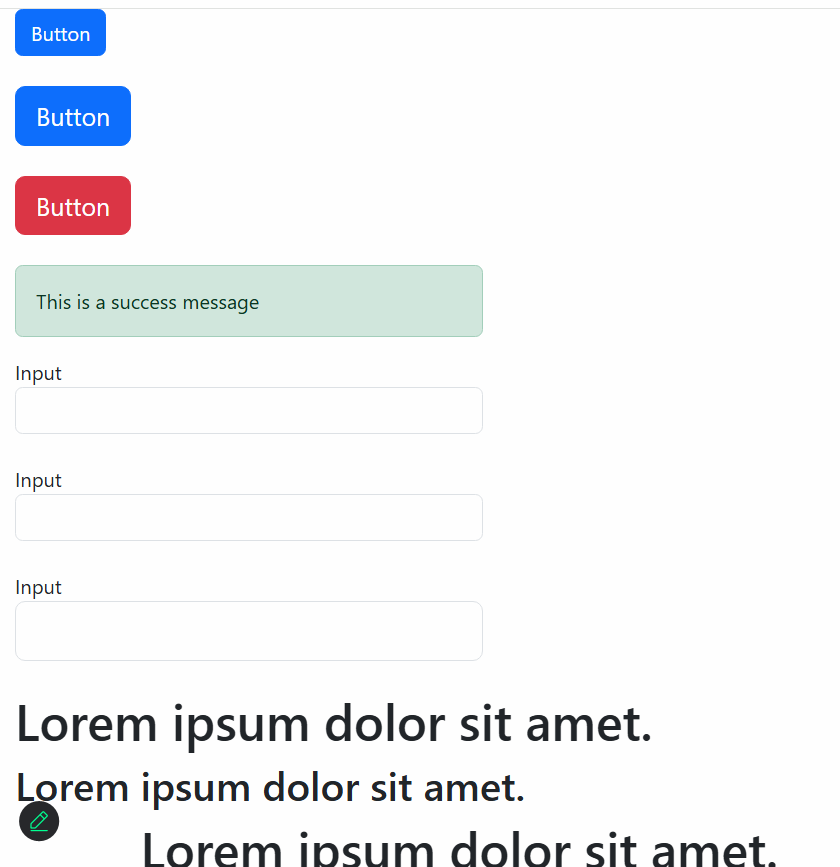
You need to just add the bootstrap.js library link in your HTML and go through the bootstrap.js website to see the built-class and how they appear on your page

Some common classes we can use from bootstrap

1. alert alert-primary
2. alert alert-danger
3. btn btn-primary
4. btn btn-secondary
5. btn btn-danger
6. btn btn-primary btn-lg
7. table
8. table table-striped
9. form-control
10. form-control form-control-lg



Output:



Javascript

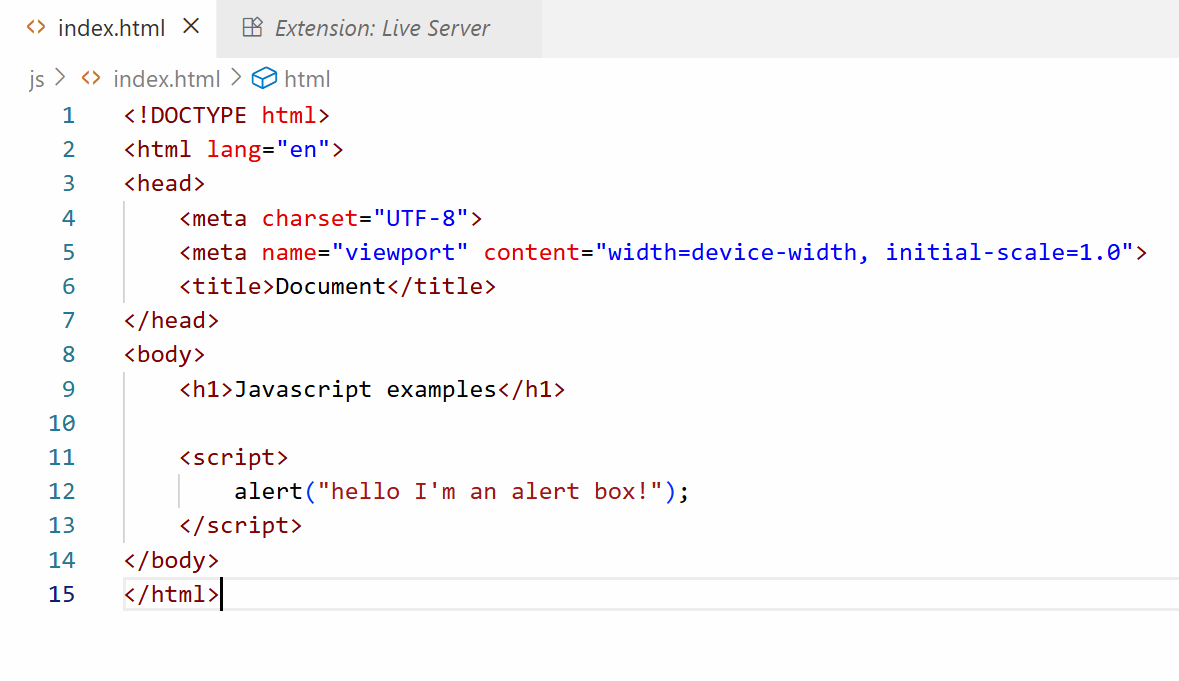
It is a programming language which make website interactive, it runs in the browser (client-side) and it is also evolved to run on servers through node.js

To add javascript in your web page, you must include it in the <script> tag, you can write javascript inside HTML document or outside HTML document and refer it using <script> tag.

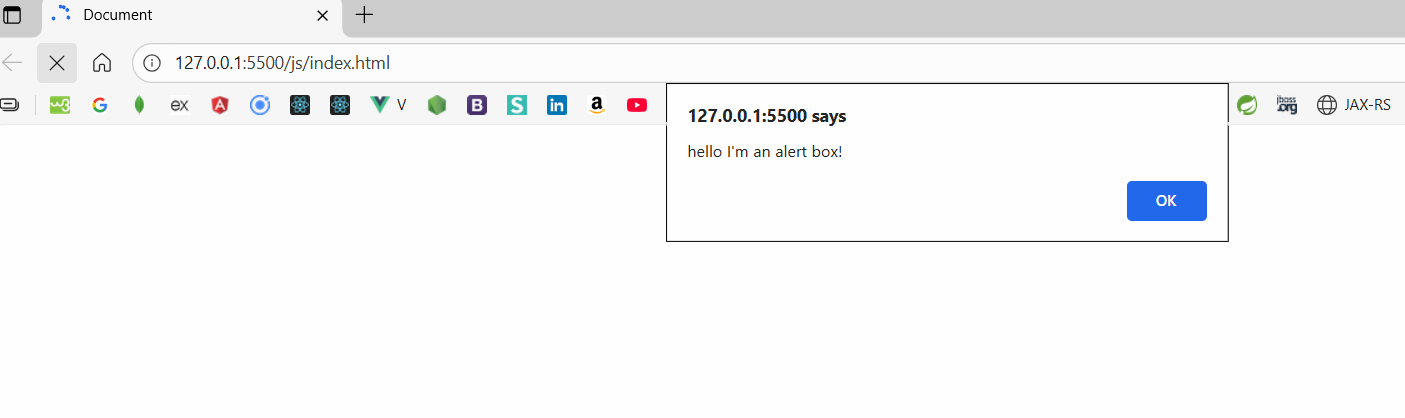
Software requirements

1. VS Code
2. Browser

index.html

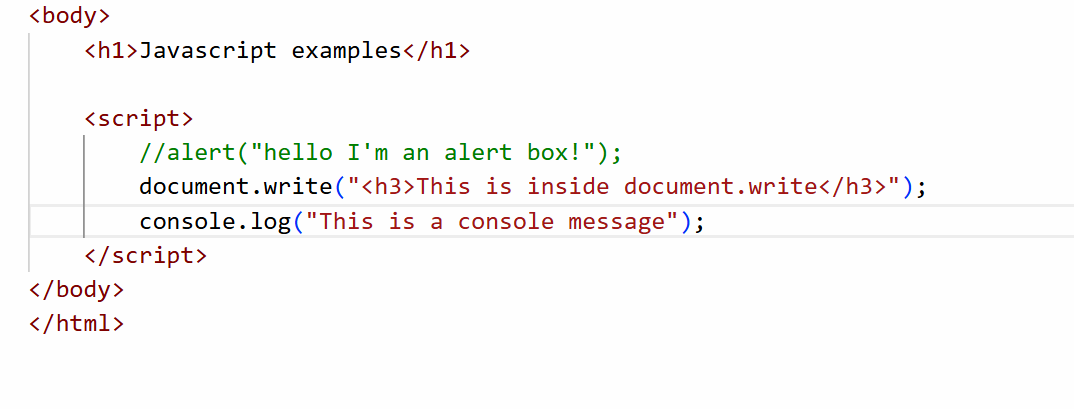


Output:

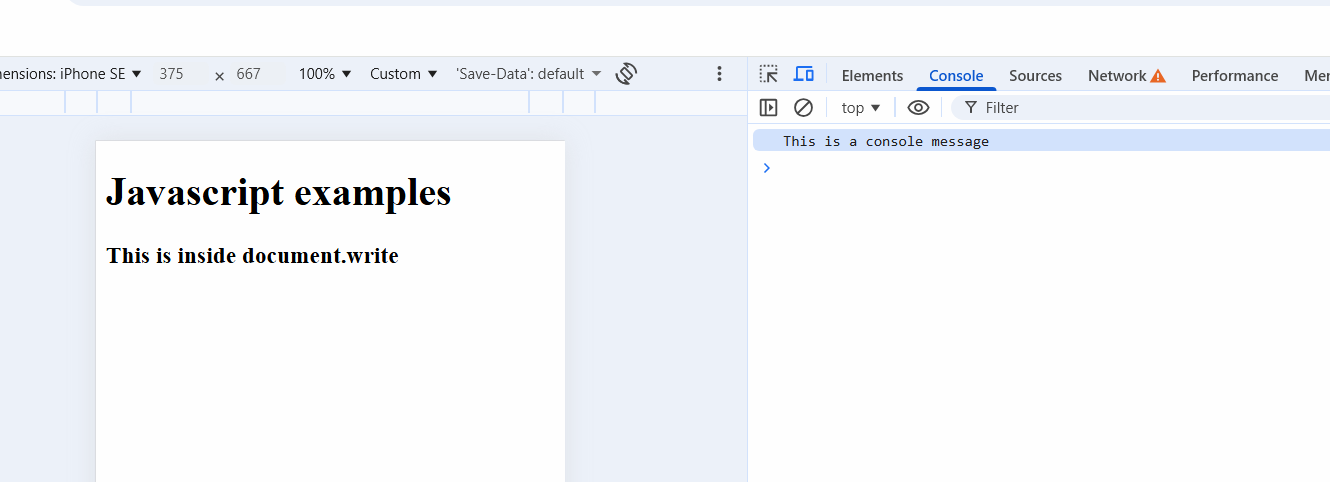


Various ways to display the messages

1. alert
2. document.write
3. console.log



Output:



Javascript Fundamentals

1. Variables & Functions
2. Operators
3. Conditional Statements
4. Loops
5. Arrays
6. Objects
7. Event Handling

Variables and Functions

Variables are placeholders or memory that can store data, it could be a simple data or a complex data, in Javascript you can declare variables using var, let, const keywords

Note: var is discouraged because it is an older approach and it’s not safe

Simple data

var username = “Kishor”;

var phone = 892892383;

var gender = ‘M’;

Functions: These are named blocks that will have set of instructions which you can reuse.

Function with a single argument

function greet(username) {   
 // this is a function that accepts an argument  
}

Function with multiple arguments

function add(x, y) {

// add x and y  
}

Function without any arguments

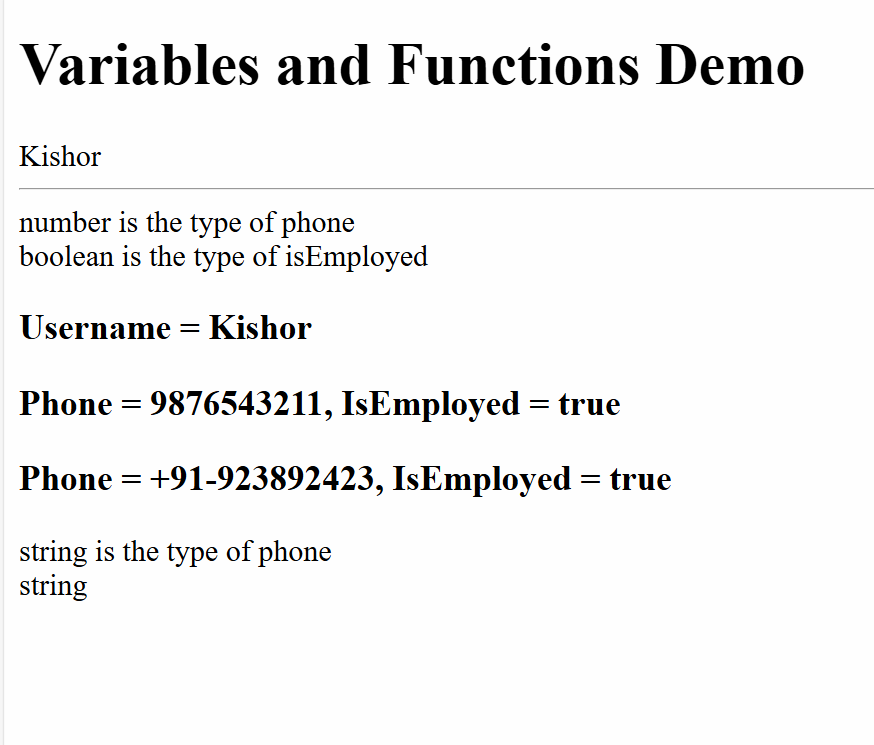
function test() {   
 // no argument  
}

greet(“Kishor”);  
add(20, 30);  
test();

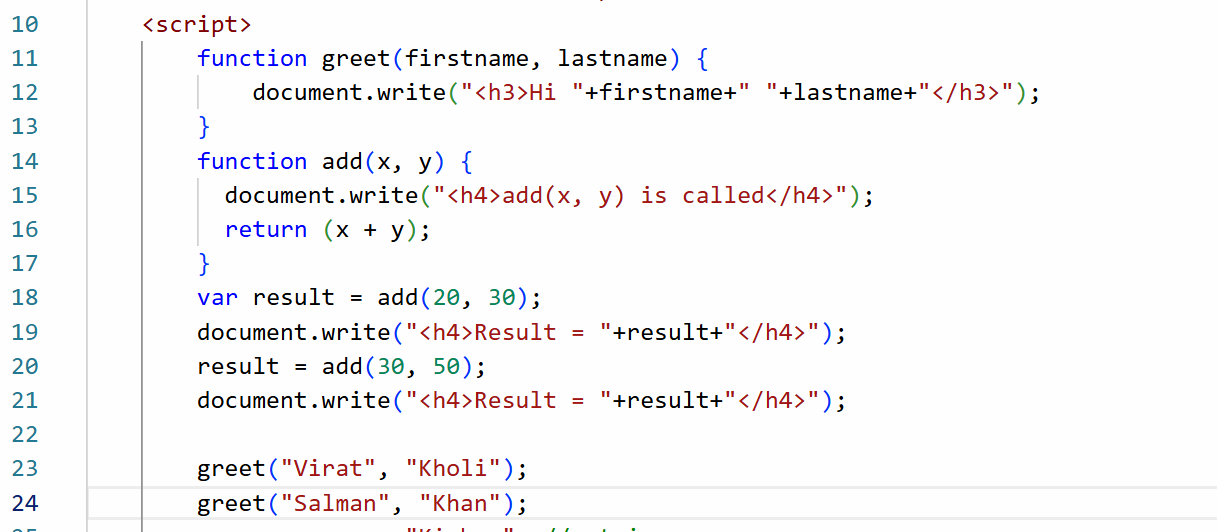
Variables Demo



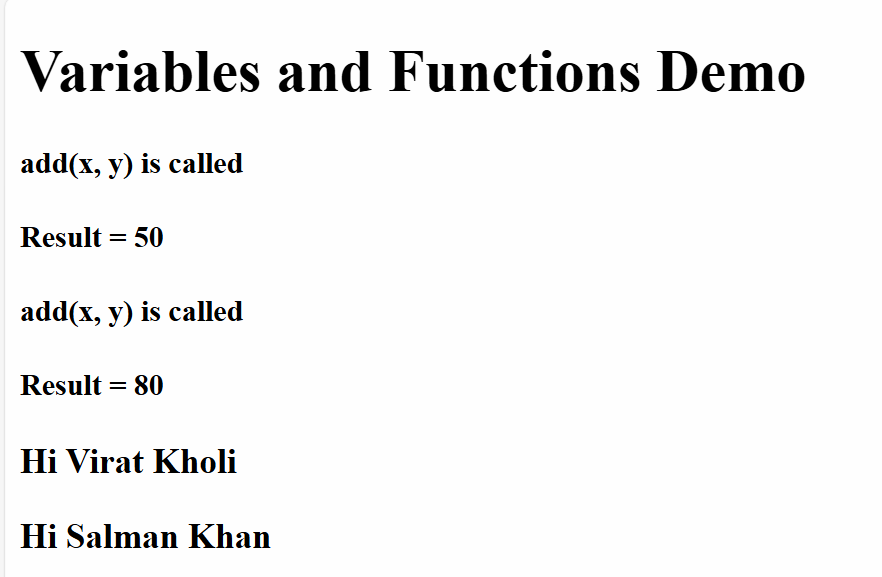
Output:



Function Demo



Output:

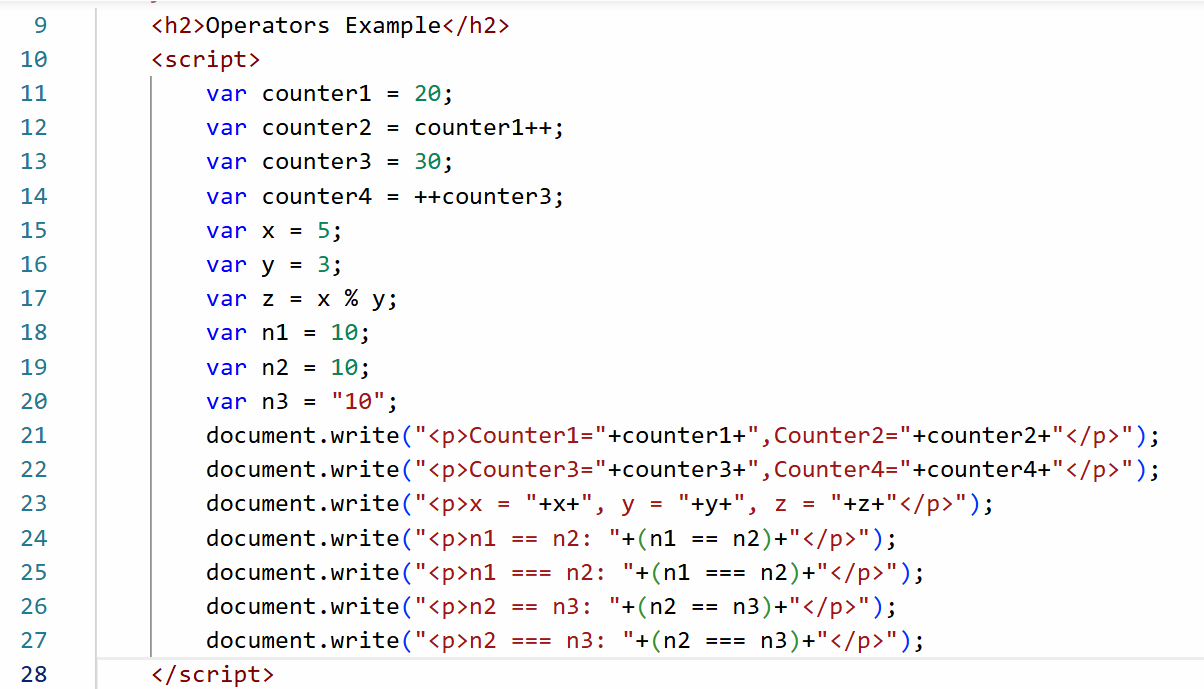


Operators

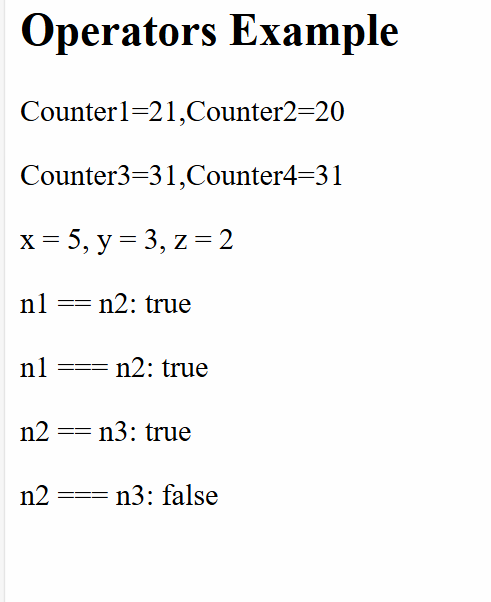
They perform operations on the variables or data, in Javascript there are many operations which are

+, -, /, \*, =, %, ==, ===, <, >, <=, >=, !=, ++, --, &&, ||

Operators Demo



Output:

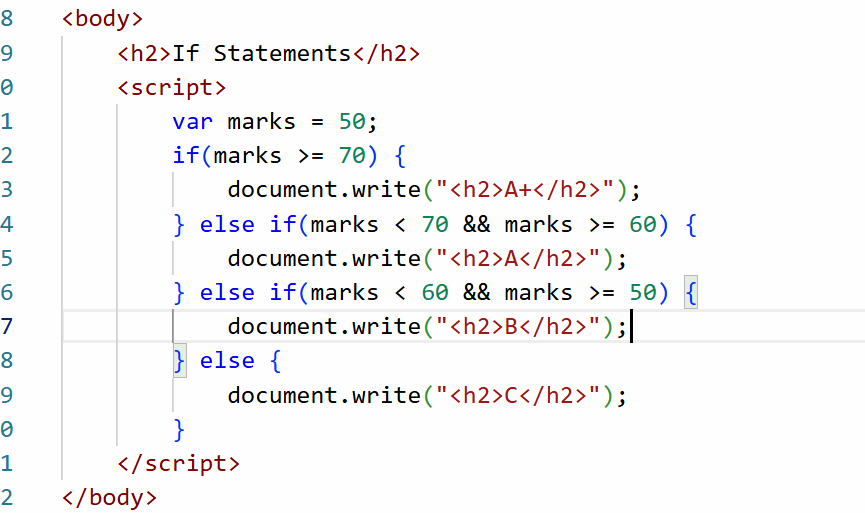


Conditional statements

Sometimes you want some code to be executed only when certain condition is met, there are following conditional statements in Javascript

1. if
2. if else
3. if else if else if .. else
4. switch

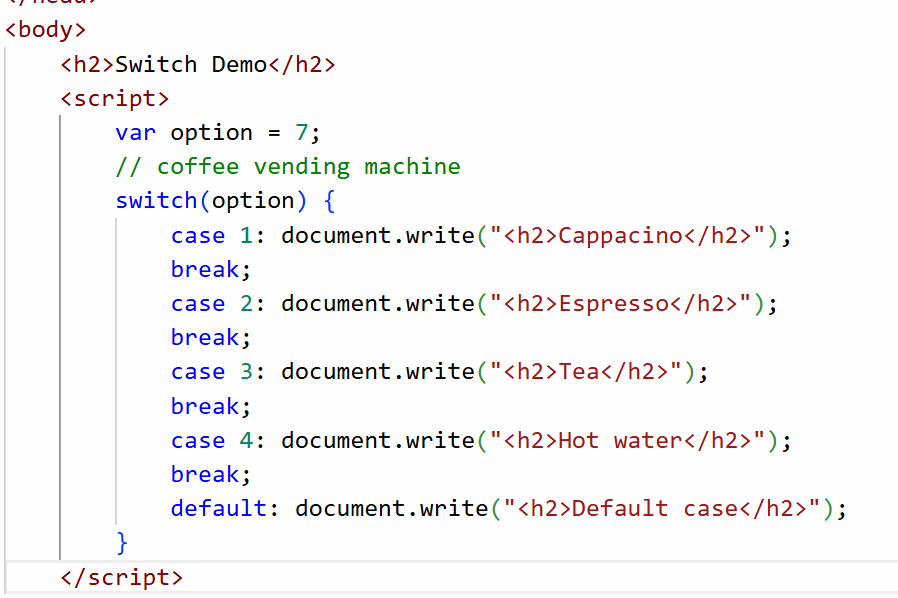
If Demo



Output:



Switch Demo



Output:



Complex types

These can store multiple values

1. arrays - can store multiple values
2. objects - they store properties and functions

Arrays:

let fruits = [ “apple”, “grapes”, “orange”, “mango”] ;  
let items = [20, 10, 50, 30];  
let items2 = [20, “apple”, true];

Objects

let employee = {id: 200, name : “Kishor”, gender: “Male”} ;

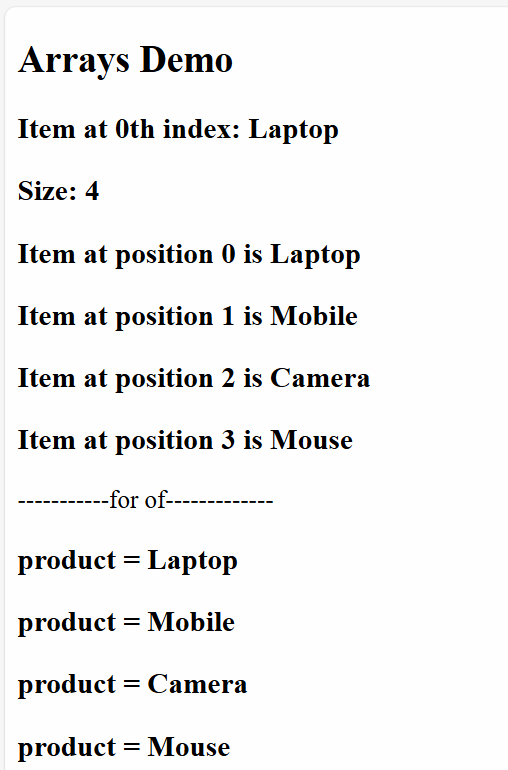
Object Array

let profiles = [ {id: 200, name: “Atharva”}, {id : 300, name: “Siddharth” }] ;

Array and for



Output:



Loops:

When you want to run the same statement repeatedly you use loops, there are 3 types of loops

1. for loop: When you know how many times to iterate

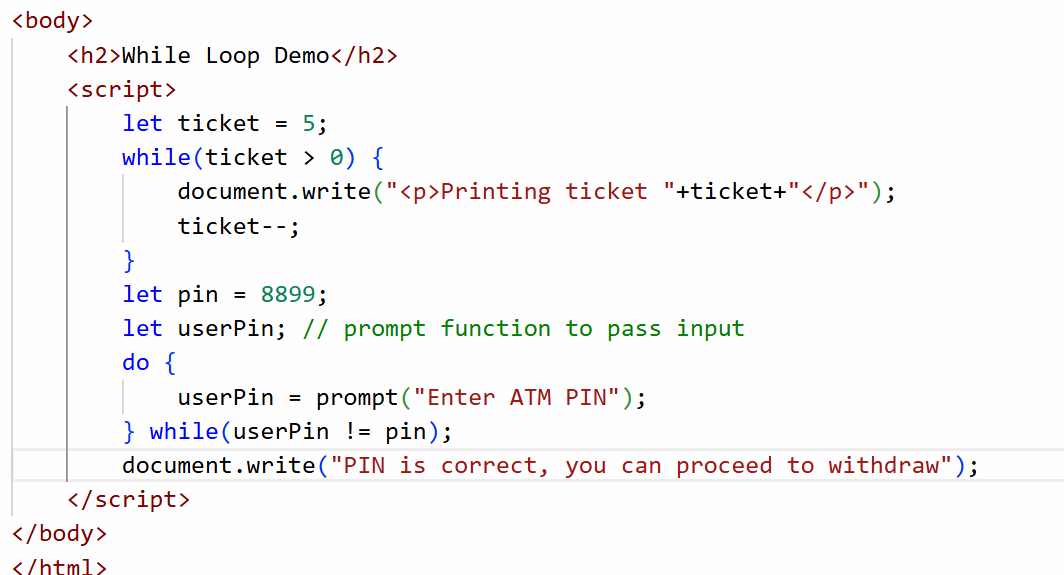
ex: iterating an array or printing a counter from 1 to 10

1. while loop: Run the statement as long as the condition is true, it checks the condition first and then runs the loop

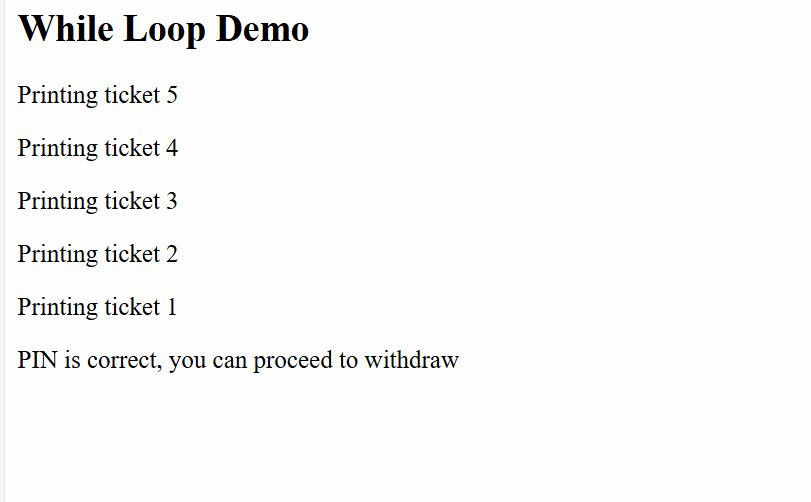
ex: print ticket until its available

1. do while loop: It runs atleast once and then checks the condition

ex: Asking for ATM pin and then proceeding to withdraw, if pin is wrong again you will ask user to enter pin



Output:

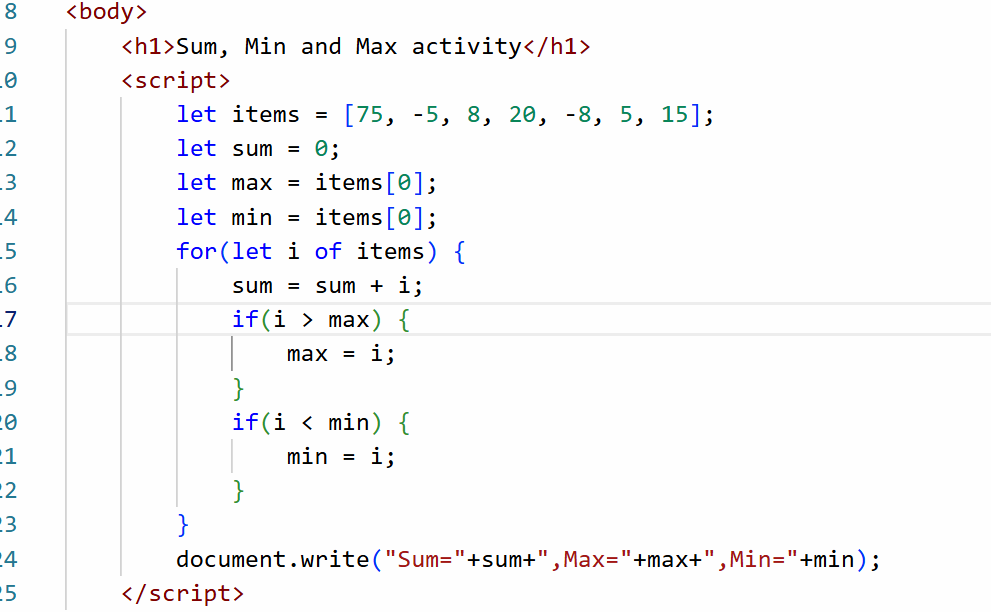


Activity:

In the following array 75, -5, 8, 20, -8, 5, 15 print the sum, max and min values

Expected output:

Max : 75  
Min: -8  
Sum: 110



Javascript objects

Objects are real world entities they will have properties and behaviour

properties - what object has like name, phone, gender, email, dob

behaviors - what object does like login, withdraw(), deposit(), these are functions of objects

In Javascript you can create objects in following ways

1. literal style

let emp = { id : 100, name: “Raj”, salary : 30000 }

1. class based

class Employee { … }

let emp = new Employee(100, “Raj”, 30000);

1. function based

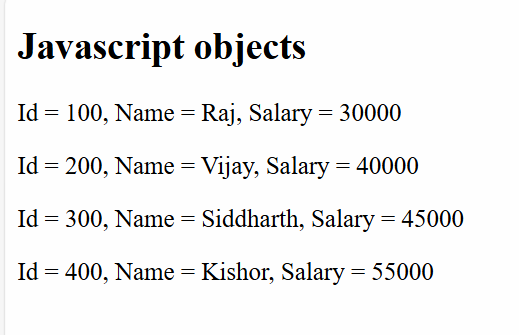
function Employee(id, name, salary) { … }

let emp = new Employee(100, “Raj”, 30000);

Javascript Object Demo



Output:



Literal style objects:

Best when you want a single object, because it is quicker to create, usually for configurations we use literal style objects

Here you can’t create similar objects without duplicating i.e., you need to copy-paste the same object and change their names

Function style objects

It uses a constructor function that specifies the structure of the object, you can create multiple objects with the same structure easily, you can use this approach when you want multiple objects with same structure.

How to add functions in an object

In literal style

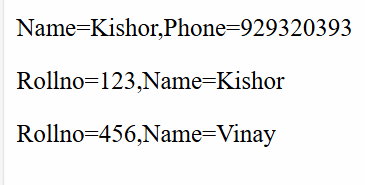
let emp = { id : 100, name : “Raj” , salary : 45000, display : function() { /\*code here\*/ } }

emp.display();

In Function style

function Employee(id, name, salary) {   
 this.id = id;  
 this.name = name;  
 this.salary = salary;  
 this.display = function() {   
 /\*code here \*/  
 }  
}

Output:



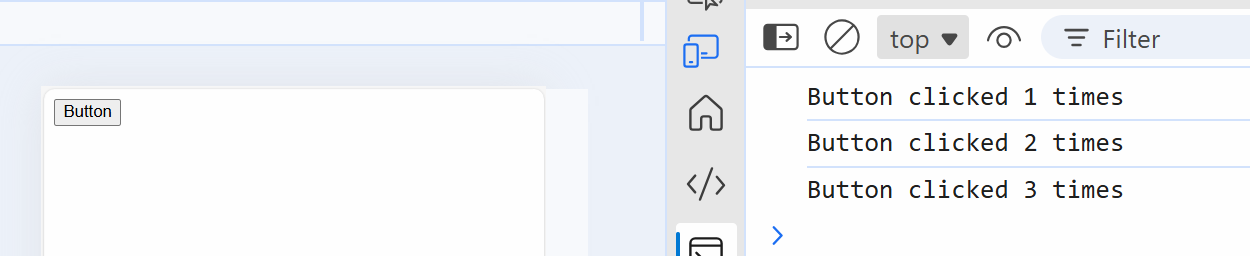
Event handling

Events are the things that occur when user interacts with HTML element like clicking, mouseover, keyup, keydown, form submission and etc.

Event handlers: These are javascript functions that are executed when a particular event occurs



Output:



document.write(): erases the existing content if a content is written later.

We must access an element based on some selectors like tag name, id or class, in the document object we have methods to access the elements

document.getElementById(“id”): This returns the element having an id

document.getElementsByClassName(“class”): This returns array of elements

document.getElementsByTagName(“tag”): This returns array of elements

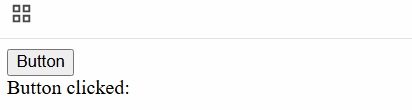
document.querySelector(“selector”): This returns the element based on the selector which could be id, class or tag names

document.getElementById(“id”): This returns an HTMLElement

HTMLElement: It is an object that has properties like innerHTML, textContent, style and so on



Output:

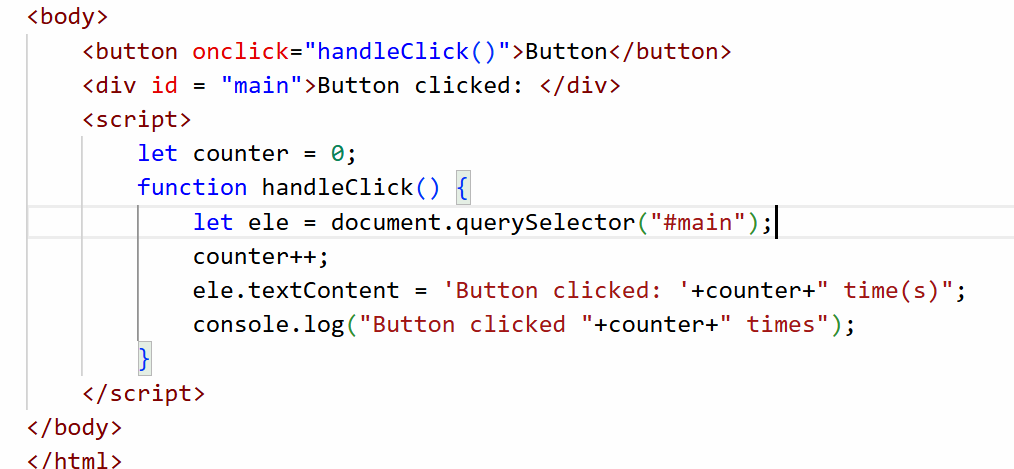


document.querySelector(“selector”): This is used to access an HTML element based on id, class or tag names, but it returns only one element if you use class or tag names

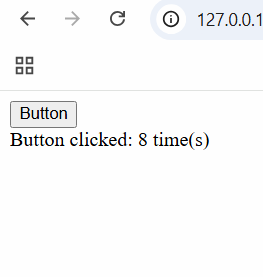
<p id = “x”>  
<p class = “y”>  
<div class = “y”>  
<p class = “y”>

document.querySelector(“#x”): #x means id = x, it selects <p id = “x”>  
document.querySelector(“p”): it selects there very first <p> i.e., <p id = “x”>  
document.querySelector(“.y”): It selects the class which has the class = y, here the very first element is selected i.e., <p class = “y”>

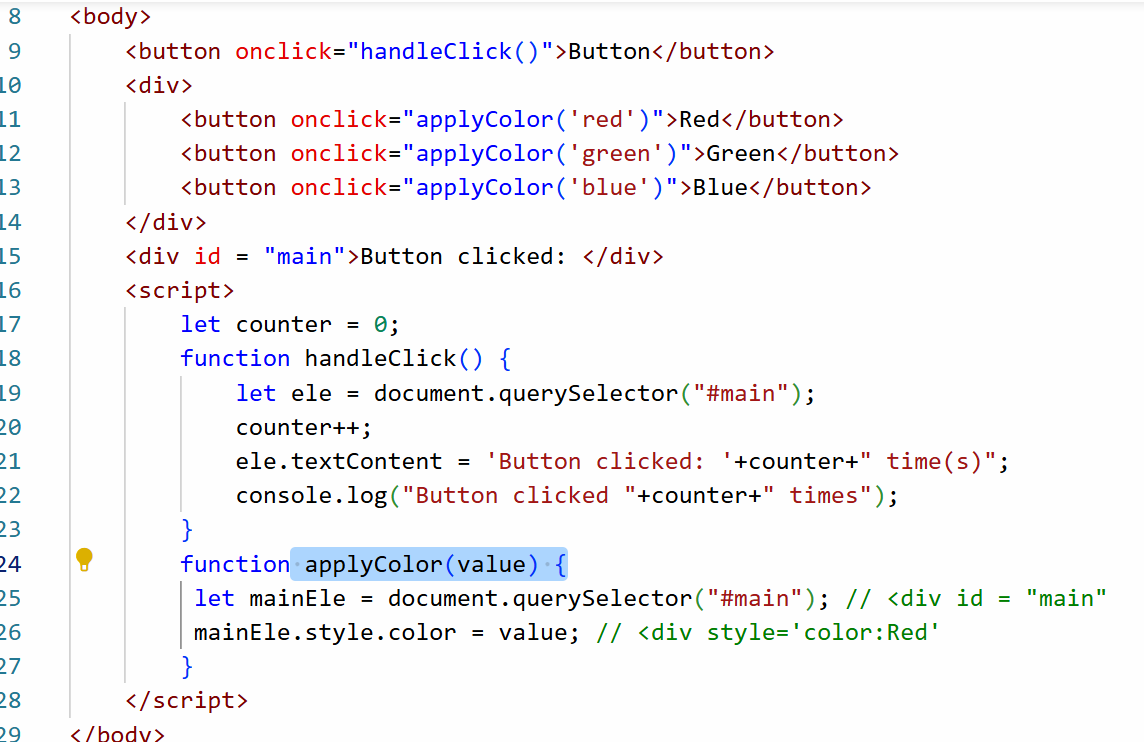
using document.querySelector



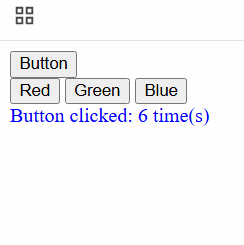
Output:



How to dynamically add styles to the element



Output:



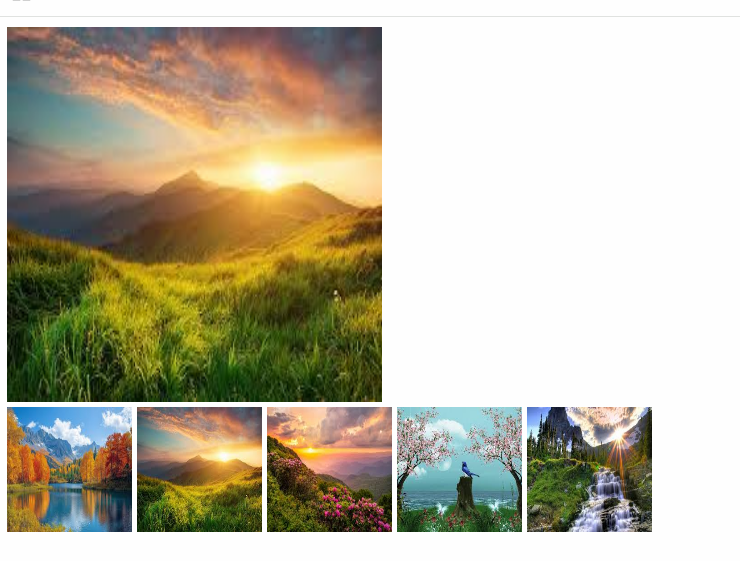
In Javascript you have an object called event that refers to the target element that generated the event, this helps you to dynamically access HTML element

<div onclick=”apply(event)”>  
 <h1>..</h1>  
 <h2>..</h2>  
</div>

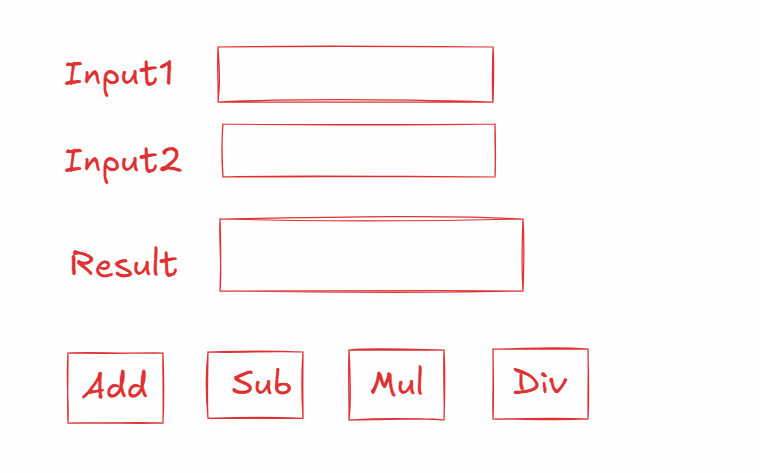
event.target that refers to the element that generated the event, this is another way of getting the element without using getElementById() or querySelector()



Output:

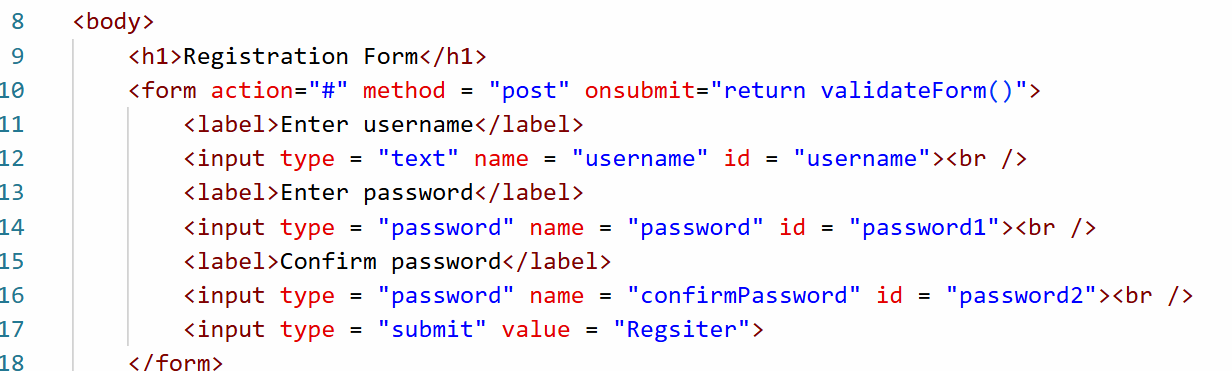


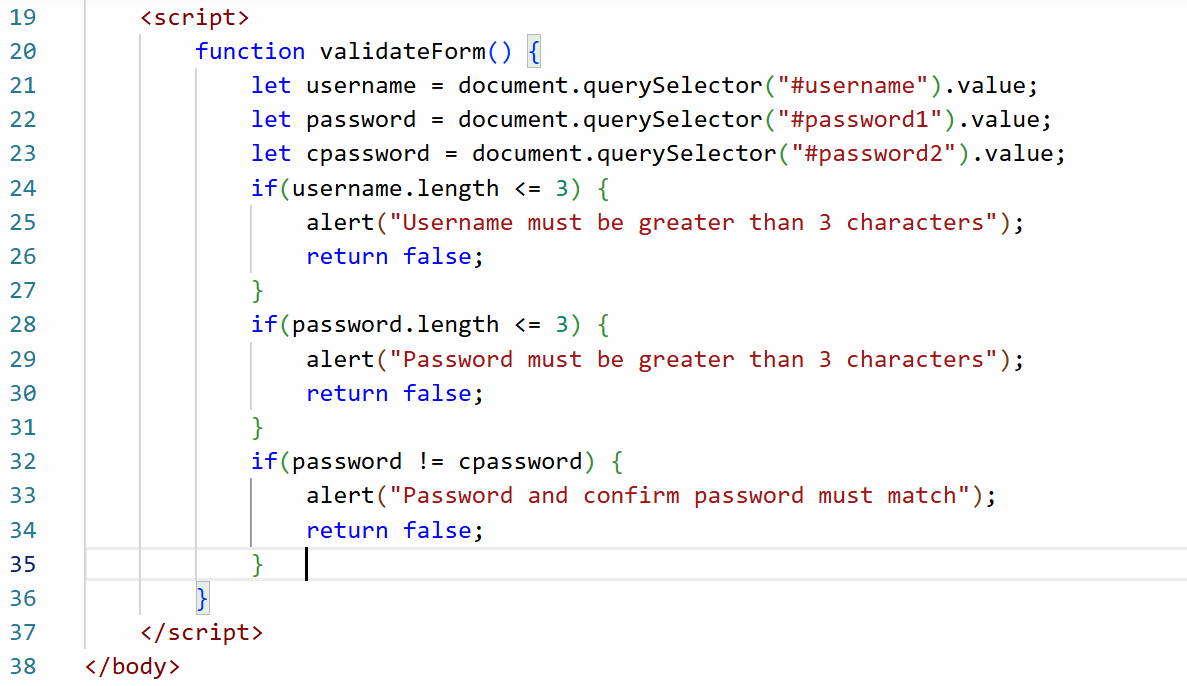
Activity: Enter inputs in Input1 and Input2 text box and Result must show the operation based on the buttons you click.



Validating form inputs

Javascript can perform client side validations as well like stopping user to send some invalid data to the server





Output:



22-11-2025

Summary of JS

* Fundamentals - operators, variables, functions, objects, arrays
* Event handling
* document.querySelector(“..”)
* document.getElementById(“..”)
* event.target
* Form validation
* var vs let vs const

Modern features of JS

1. let & const for variables
2. class, constructor - for objects
3. rest and spread operators
4. Template Strings
5. array and object Destructuring
6. arrow functions
7. Promises
8. async/await

class & constructor: It is used to create objects with a modern syntax instead of using function keyword

Old approach:

function Employee(name, age) {  
 this.name = name;  
 this.age = age;  
 this.display = function() { console.log(this.name, this.age) ; }  
}

New approach:

class Employee {   
 constructor(name, age) {   
 this.name = name;   
 this.age = age;   
 }  
 display() { … }  
}

let emp = new Employee(“Raj”, 35);  
emp.display();

Rest & Spread operators

Earlier you couldn’t write functions to take varying arguments

Old approach:  
function sum(x, y) { .. }  
You must pass 2 parameters to the sum() function, but what if at runtime you want to pass 3 or more parameters

Rest operator/parameter: It is a parameter to the function that can accept 0 or more parameters

Note: In a function you can have maximum one Rest parameter & it must appear as a last parameter

New approach: Rest parameters which can accept 0 or more parameters

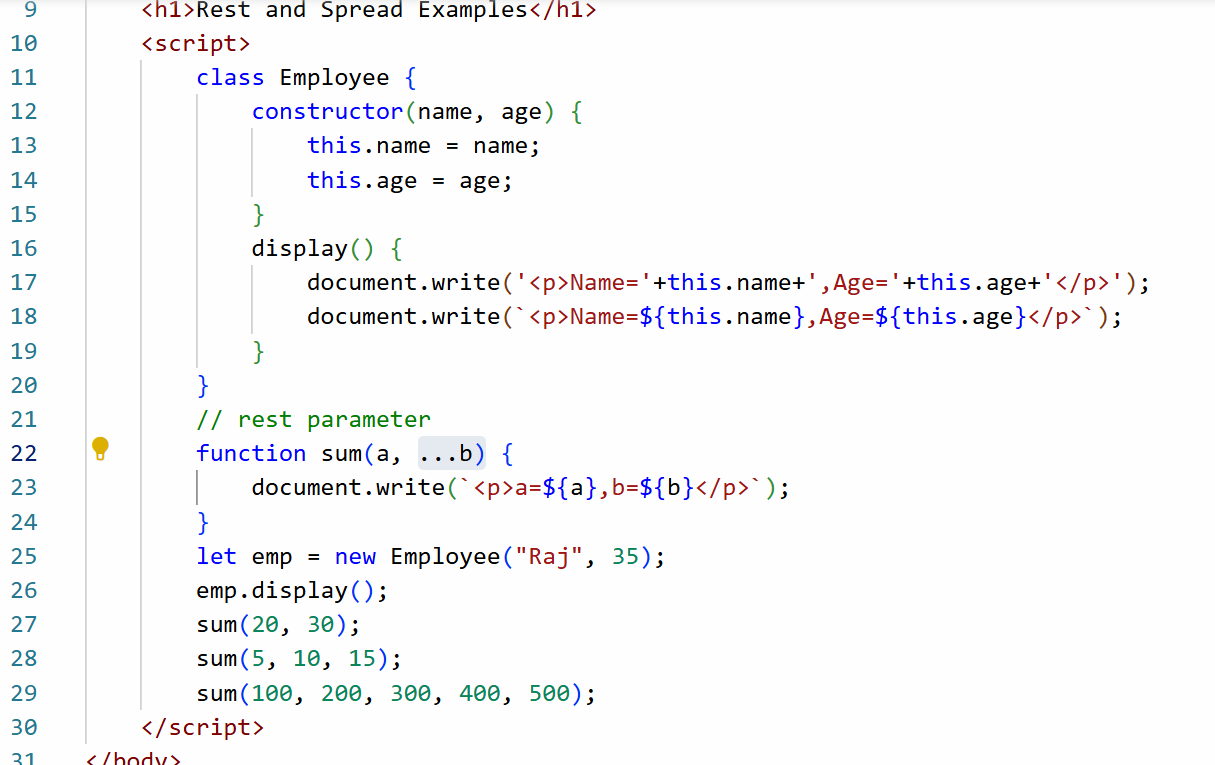
function sum(a, …b) { …………. }  
Now sum has …b which can accept 0 or more parameters

sum(20, 30); // a = 20, b = 30;  
sum(20), // a = 20, b =   
sum(20, 30, 40) // a = 20, b = [30, 40]  
sum(40, 50, 60, 70) // a = 40, b = [50, 60, 70]

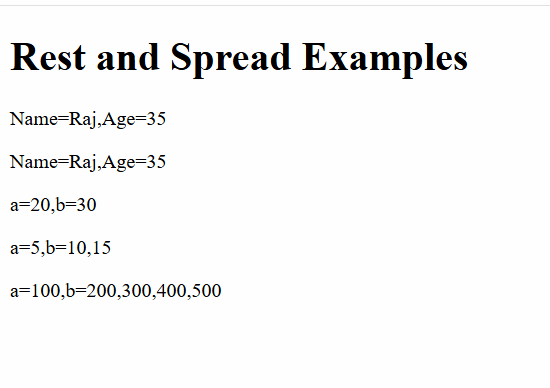
Template Strings: It uses back tick(`) character to easily concatenate string and javascript expression

Note: Single quote ‘ is different and back tick ` is different

`<p>Name = ${name}, Age = ${age}</p>`



Output:



Spread operator: It is used to spread multiple values of an array to multiple parameters of the function

Old approach

let items = [20, 30, 40]

function test(x, y, z) { … }

test(items[0], items[1], items[2]);

New approach

test(…items); // x = 20, y = 30, z = 40

the items value would spread across the parameters, if there are 3 elements in the array and 3 parameters in the function then it spreads properly

If there’s a mismatch in the array elements or parameters of the function

items1 = [3, 1, 6]

items2 = [10, 20, 30, 40, 50]

function test(a, b, c, d) { }

test(…items1); // a = 3, b = 1, c = 6, d = undefined

test(…items2); // a = 10, b = 20, c = 30, d = 40 and value 50 will be lost

So to avoid losing the values you need to combine rest & spread

function test(a, b, c, …d) { … }

test(…items2); // a = 10, b = 20, c = 30, d = 40, 50

rest vs spread

Rest parameter is used while creating the function parameter

Spread operator is used when you want to assign values to the variable i.e, while calling the function you can use spread operator

Destructuring

It is a feature that helps to easily assign values to the variables in a single line without declaring them in separate lines

Array Destructuring:

let items = [10, 20, 30];

old approach  
let x = items[0];  
let y = items[1];  
let z = items[2];

new approach

let [x, y, z] = items;

Object Destructuring

let emp = {id : 100, name : “Raj”, salary : 55000}

old approach

let id = emp.id;  
let name = emp.name;  
let salary = emp.salary;

new approach

let { id, name, salary } = emp;

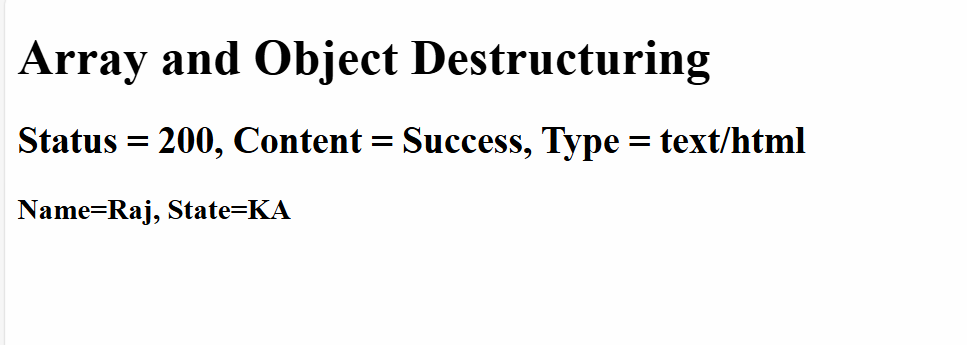
Note: variable names must match to the object properties, if you don’t give the same name then that variable will be undefined

let {x, y, z} = emp;

x = undefined, y = undefined, z = undefined

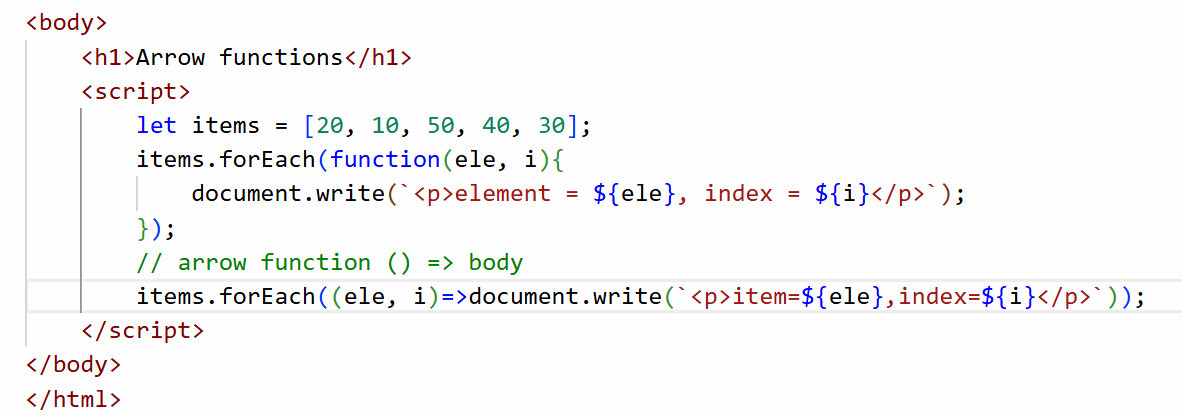


Output:



Arrow functions:

These are simplified form of an anonymous functions / callback functions in javascript



Promises in Javascript

These are objects which can perform asynchronous operations like handling the requests or any other background process which waiting for the response

Promise can return either with success or failed response, to handle these two status you have .then() and .catch() functions, these two functions take callbacks that are executed when the response is ready

