dsa and database

Trick	What it means
pwd	Know where you are
ls	See files
mkdir / touch	Make directory / file
cp / mv / rm	Copy, move/rename, delete
sudo	Run commands as superuser
cat	View file contents

- `chmod` → change permissions
- `chown` → change owner
- `grep` → search in files
- `ps` → list running processes
- `kill` → stop processes

Topic	Quick Tip
SDLC	Plan → Analyze → Design → Develop → Test → Deploy
Agile	Deliver small working parts (incremental)
Scrum	Work in short sprints (~2 weeks)
Product Owner	Manages Product Backlog
Standup	15 minutes only

Directed Graph	Edges have direction (arrows)
Undirected Graph	Edges have no direction
DFS (Depth First Search)	Go deep first (stack)
BFS (Breadth First Search)	Go level by level (queue)
Cycle in Graph	Reach same node again (loop)

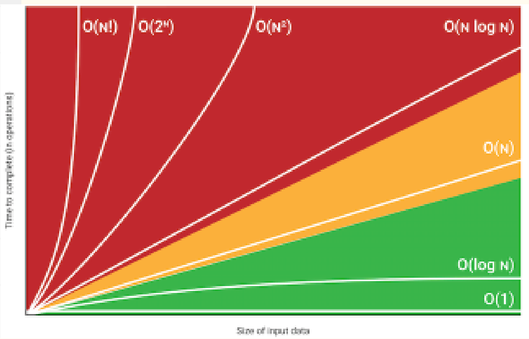
Traversal Type	Visit Order
Inorder	Left → Root → Right
Preorder	Root → Left → Right
Postorder	Left → Right → Root

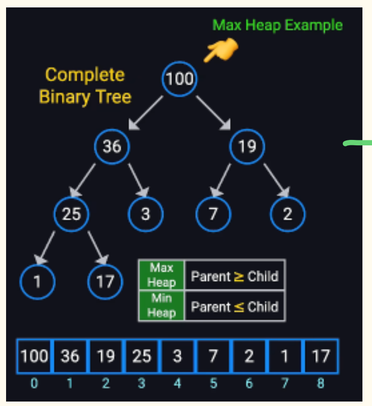
Topics for CS

Applied CS Concepts

1. Linux, Git or Version Control
2. SDLC and Agile
3. Big-O Notation
4. Tree and Graphs
5. Stack, Queue, Hash & Heaps
6. Sorting and Searching
7. Popular Algorithms
8. Database Concept
9. SQL
10. REST API

Algorithm	Best case	Average Case	Worst Case	Space Complexity
Bubble Sort	O(n)	O(n^2)	O(n^2)	O(1)
Insertion Sort	O(n)	O(n^2)	O(n^2)	O(1)
Selection Sort	O(n^2)	O(n^2)	O(n^2)	O(1)
Merge Sort	O(n log n)	O(n log n)	O(n log n)	O(n)
Quick Sort	O(n log n)	O(n log n)	O(n^2)	O(log n)
Heap Sort	O(n log n)	O(n log n)	O(n log n)	O(1)
Binary Search	O(1)	O(log n)	O(log n)	O(1)





to sort:

- in ascending, uses max heap
- in descending, uses min heap
- ani tyo point gareko value last box ma rakhne which will be sorted last ma

- ✅ REST = Representational State Transfer
- ✅ API = Application Programming Interface

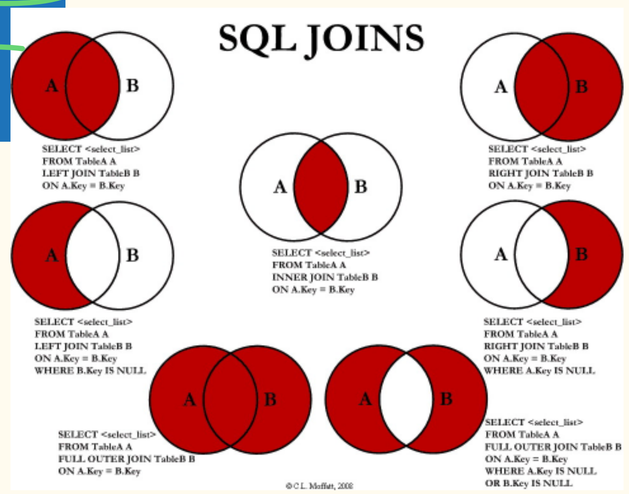
Simple words:

> "REST API allows two systems to talk using standard web methods (like GET, POST, etc.)."

- ✅ REST follows **rules** to make communication **simple, scalable, and stateless**.

HTTP Method	Purpose	Example
`GET`	Retrieve data	`GET /users/5` (Get user with ID 5)
`POST`	Create new data	`POST /users` (Create new user)
`PUT`	Update full data	`PUT /users/5` (Update all data of user 5)
`PATCH`	Update partial data	`PATCH /users/5` (Update some fields of user 5)
`DELETE`	Delete data	`DELETE /users/5` (Delete user 5)

Status Code	Meaning
`200 OK`	Successful GET, PUT, PATCH, DELETE
`201 Created`	Successful POST (resource created)
`204 No Content`	Successful DELETE (nothing to return)
`400 Bad Request`	Client error (wrong input)
`401 Unauthorized`	Login required
`403 Forbidden`	Access denied
`404 Not Found`	Resource does not exist
`500 Internal Server Error`	Server-side crash



SELECT	Fetch data from table
INSERT	Add new data into table
UPDATE	Modify existing data
DELETE	Remove data
JOIN	Combine rows from multiple tables
INNER JOIN	Returns matching rows between tables
LEFT JOIN	All from left + matching from right
RIGHT JOIN	All from right + matching from left
FULL JOIN	All rows from both tables
Index	Speed up searching (like a book index)
View	Virtual table based on query (no real storage)
Transaction	Group of operations treated as one (ACID properties)
ACID	Atomicity, Consistency, Isolation, Durability