DEVOPS - DAY 1

Basic Definitions :

what is Operating System?

* The interface between user and hardware .

what is cloud computing?

* cloud computing is a on demand delivery of IT resourses over the internet. We can access ethical resources , technical resources such as storage, databases, compute, networking, security with **pay as you go.**

What is Devops

*devops is a set of practices , principles and cultural philosophies that aim to bridge the gap between software development and operations.

The goal of devops is to shorten the software development life cycle , improve collaboration between development and operations team and deliver high quality software faster and more efficiently.

Cloud providers in market

- AWS
- Microsoft Azure
- google cloud platform

SDLC - Software Development Life Cycle- its a structured step by step process that developing teams use to create high quality , cost effective and secure software.

what are the server components :

-OS(Linux, win, mac)

-RAM (6/8/12/16 etc GB)

-ROM (HDD, SSD)

-Networking

LINUX COMMANDS :

1.ls : it is a command to list files and directories(folders)

2.touch : is a command to create empty files.

ex: cat > kushi (to write something inside that file)

ex: clear

ex: cat kushi(file name)- to display all the content written inside that file.

3. rm filename (is a command to remove the files)

4. mkdir : is a command to create the empty folder.

ex : mkdir rishi

5.cd: change directory

ex : cd ..(exit from directory)

6. rmdir: is a command to delete the empty directory with files

rmdir directoryname

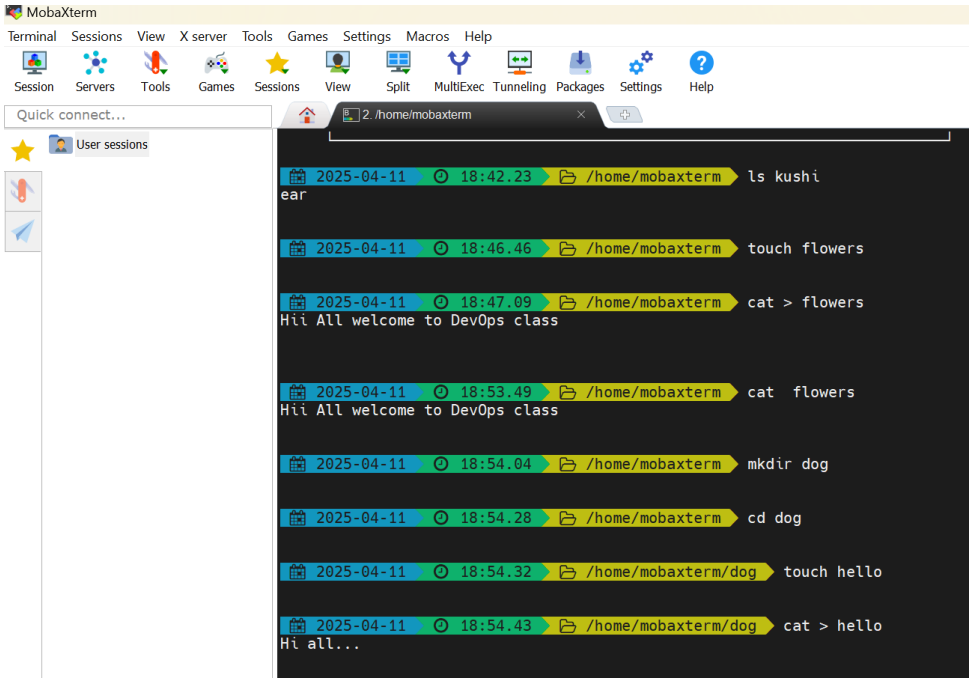
7.rm -r : is a command to delete directory with file

8.ls -l: list the files and directories with permissions.

9. pwd : print working directory/present

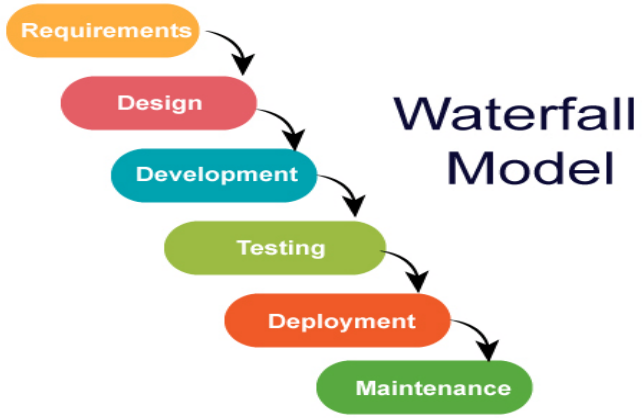
10. history - To get the history of the previous commands

11. cd ../.. to exit out of 2 directory.



```
2025-04-11 18:56.19 /home/mobaxterm cd dog

2025-04-11 18:57.48 /home/mobaxterm/dog touch eye

2025-04-11 18:57.56 /home/mobaxterm/dog touch apple

2025-04-11 18:58.07 /home/mobaxterm/dog touch orange

2025-04-11 18:58.15 /home/mobaxterm/dog ls -l
total 0
-rw-rw-r-- 1 Kushi None  0 2025-04-11 18:58 apple
-rw-rw-r-- 1 Kushi None  0 2025-04-11 18:57 eye
-rw-rw-r-- 1 Kushi None 10 2025-04-11 18:55 hello
-rw-rw-r-- 1 Kushi None  0 2025-04-11 18:58 orange
```

Water fall model :



1. Requirement Gathering and Analysis

- All possible requirements of the system are gathered and documented.
- Clear understanding is developed between the client and the development team.
- No development or designing is done in this phase.

Goal: Understand what the system should do.

2. System Design

- Based on requirements, the system architecture and design are created.

- High-level design (HLD) and low-level design (LLD) are prepared.
- Technologies, tools, programming languages, and hardware needs are selected.
- **Goal:** Plan how the system will be built.

3. Implementation (Coding)

- The actual code is written based on the design.
- The system is developed in small units/modules and integrated later.

Goal: Convert design into working code.

4. Integration and Testing

- All modules are integrated and tested as a whole system.
- Bugs are identified and fixed.
- Ensures the software meets the requirements.

Goal: Validate the system works as expected.

5. Deployment

- The software is deployed to the user environment or production system.
- It becomes available for use.

Goal: Deliver the product to the user.

6. Maintenance

- After deployment, any bugs, errors, or required updates are handled.
- New versions or patches may be released.

Goal: Keep the system running smoothly after release.

Limitations :

No Flexibility for Changes:

- Once a phase is completed, it's very difficult to go back and make changes.
- Not suitable if the requirements may evolve during the project.

Late Testing:

- Testing is done only after development is complete, so bugs are found late in the process.
- This increases the cost and effort to fix issues.

AGILE MODEL :

The Agile Model is an iterative and incremental approach to software development. It breaks the project into small parts called sprints (usually 1–4 weeks), where each sprint includes planning, designing, coding, and testing.

Teams can adapt to changes quickly, and customer feedback is gathered regularly.

Limitations:

Less Documentation:

- Focus on working software over detailed documents can lead to confusion later.

Requires Experienced Team:

- Agile needs skilled developers and good communication; not ideal for beginners.

Scope Creep Risk:

- Frequent changes and feedback can sometimes lead to uncontrolled changes in the project.

EC2 Configuration:

Setup steps:

1. Name of the server.

2. Application and OS images.(Amazon Machine Image)

-os selection

3. Instance type : choose number of CPU's and RAM

4. Keypair : used to connect securely to instance

5. Networking :

-firewalls

-VPC : virtual private cloud

-subnets

route tables

-NAT

-Internet gateway

-Elastic IPS

6. Configure storage

-EBS (Elastic Block store)

-EBS nothing but similar to hard disk

-persistent block level storage

- can create new volume and attach to running instance

7. Launch instance : web server/windows server