Week 1

OS: interface between computer hardware and user

Allows to run multiple programs

Allows communication between user and computer

Used in security for authentication/authorization and file management

Windows: not open source

Mac: partially open source

Linux: open source, security

ChromeOS: partially, education

Android: open source

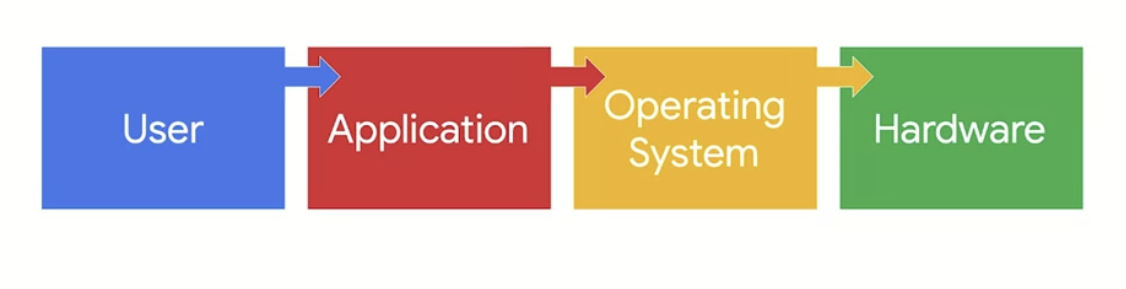
IOS: partially open source

Legacy: outdated but companies still use the software because it’s hard to update to be compatible with newer versions

OS: runs all the messy steps for a computer to do tasks without needing direct user control. E.g. person pushes the pedal, and the car moves. The OS takes care of all the things like starting the engine and keeping oil running.

Starting a computer = starting OS. Power button opens BIOs or UEFI microchip. This starts a program called the boot loader. Boot loader starts the OS.

App: program that performs a task



User uses an application, which sends a request to the operating system which accesses the components of hardware needed

Basic input and output system (BIOS): microchip with instructions to start the computer, old

Unified Extension Firmware Interface (UEFI): same as bios, new security features

Bootloader: boots the OS

### **User**

The first part of the process is the user. The user initiates the process by having something they want to accomplish on the computer. Right now, you’re a user! You’ve initiated the process of accessing this reading.

### **Application**

The application is the software program that users interact with to complete a task. For example, if you want to calculate something, you would use the calculator application. If you want to write a report, you would use a word processing application. This is the second part of the process.

### **Operating system**

The operating system receives the user’s request from the application. It’s the operating system’s job to interpret the request and direct its flow. In order to complete the task, the operating system sends it on to applicable components of the hardware.

### **Hardware**

The hardware is where all the processing is done to complete the tasks initiated by the user. For example, when a user wants to calculate a number, the CPU figures out the answer. As another example, when a user wants to save a file, another component of the hardware, the hard drive, handles this task.

After the work is done by the hardware, it sends the output back through the operating system to the application so that it can display the results to the user.

Some tasks require more resources than others

OS: Conductor, allocates resources for different programs

Virtual machine: virtual version of a computer; no hardware. All hardware is replaced by code (cpu, hard drive, etc.)

Virtualization: making software versions/representations of physical objects

Can run VMs on a single computer, but it takes resources (16gb ram = 3 VMs. 4 ram per computer.)

VMs can be used for security: sandboxes, isolated testing environments. Malware can escape the virtualization and attack the host system.

VM Efficiency: can streamline/multi task, increasing security efficiency

Hypervisors: manage VMs. Allocates resources to VMs.

Kernel based Virtual Machine (KVM): VM accessible from the linux OS with no further software downloads

User Interface: program that allows user to communicate with the OS

GUI: interface that uses icons on the screen to communicate with the computer. Simple. Apps, toolbar with commands

CLI (command line interface): Text based interface that communicates with the computer. No icons. More powerful, more control over the OS. Can do tasks more efficiently, e.g. move all files instead of doing it one by one with the GUI.

-Efficient

-History file

Linux: stores history of CLI commands.

WEEK 2:

Linux: checking logs, authorizing users.

Linux is based on the UNIX OS, but makes it accessible to everyone- open source



User: person interacting with the computer. Multiple users can use the resources through linux

App: program that performs a task.

Shell: Command Line Interpreter. How you communicate with the system and get results back.

Filesystem Hierarchy Standard: Component of linux OS that organizes data. Filing cabinet.

Kernel: manages processes/memory. Communicates with the hardware to do tasks

Hardware: physical components.

Package installer: app that helps users download, install, and manage packages/apps.

Package: piece of software that can be combined w/ other packages to form an app

FHS: manages directories (collections of files), and other storages to ensure they are found by the OS

CPU: performs computing tasks of the computer. Executes instructions

RAM: stores short term memory. E.g. this document is in RAM as I am using it, once I’m done it will be closed and stored on the hard drive. CPU uses data from here to run programs.

Hard drive: long term memory used to store files which can be accessed later.

Distributions: different versions of Linux used for different purposes. Includes a linux kernel, package installer, utilities

Kali Linux: offensive security. Made for penetration testing and forensics. Should be used on a VM as it’s risky.

Penetration attacks: simulating attacks to find vulnerabilities in assets

-metasploit: test for weaknesses on machines

-burp suite: test for vulnerabilities in web apps

-john the ripper: solve passwords

Digital forensics: collecting/analyzing data to figure out what happened

-tcpdump: CLI log network traffic

-wireshark: GUI to live capture web traffic

-autopsy: used to analyze hard drives and smartphones

Ubuntu: user friendly cyber security distribution, CLI and GUI. comes with a lot of apps, apps can also be downloaded. Used in cloud computing.

Parrot: also has GUI. User friendly. Pen testing, digital forensics (similar to KALI)

Red Hat Enterprise Linux: for enterprise use, comes at a price. Dedicated support team.

Cent OS: similar to Red Hat, but open source. Not as good for enterprise use.

Packages: pieces of software. Can be put together to create an app. Includes the instructions to install, along with dependencies (peripheral software needed to run the app)

Package manager: use to install, remove, and manage packages. Linux uses multiple

Package managers vary by distribution. Important to know the parent distribution, uses the same package managers.

**Debian**: KALI Linux, Ubuntu, Parrot; PM: dpkg file extension: .deb; PMT: Advanced Package Tool (APT) to manage, search, install packages from shell

*Package\_Version-Release\_Architecture.deb*

**Red Hat**: CentOS; PM: Red Hat Packet Manager (RPM) file extension: .rpm; PMT: Yellowdog Update Modifier (YUM) to manage, search, install packages from shell

*Package-Version-Release\_Architecture.rpm*

Package management tools: used to manage packages from the shell. Easier to perform basic tasks (e.g. installing a new package)

Shells: CLI allows communication of user with computer. Sends information to the kernel, and displays back the results to the user through the CLI

All shells use common linux commands, but may differ in other features.

Bash: default, user friendly shell. Used widely in cybersecurity. Basic commands, larger projects.

Standard input: info received as input on the command line

Echo: linux command that outputs a specific string

String data: consists of ordered sequence of characters

Standard output: info received from the OS to the shell

Standard error: error messages returned to the shell by the OS

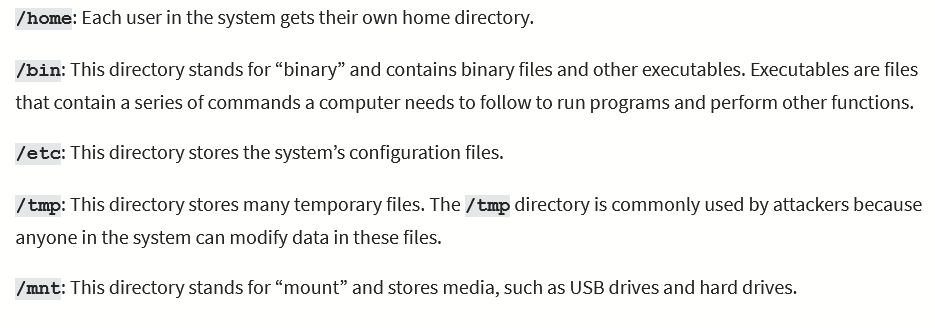
Command: instructing computer to do something

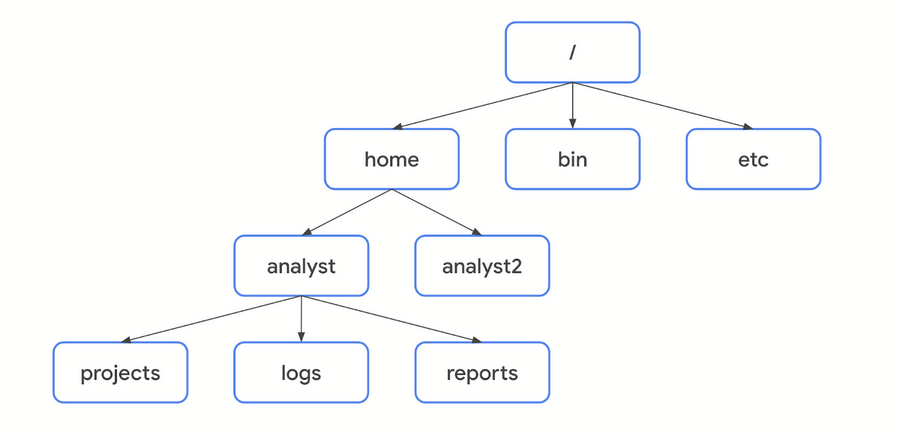
Argument: info provided as input with command to produce output

Filesystem Hierarchy Standard: component of Linux OS that organizes data

Root directory: highest level directory in linux

Below root: Standard directories



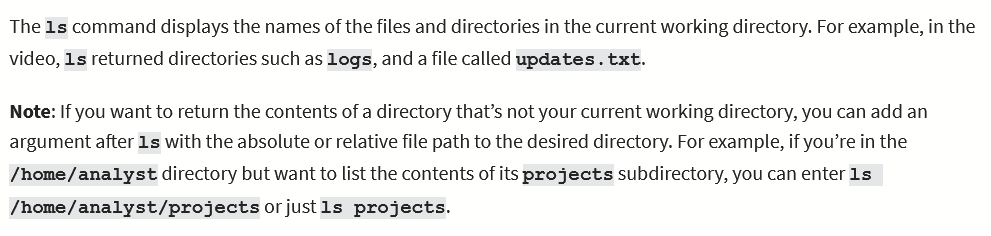


Subdirectories indicated by /. /home/analyst. First slash indicates root

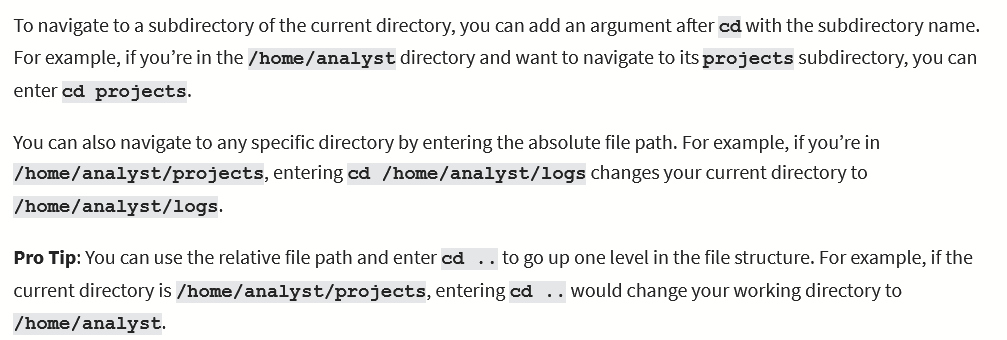
**File commands:**

**Pwd: prints working directory and absolute file path**

**Ls: displays the names and files of current working directory**

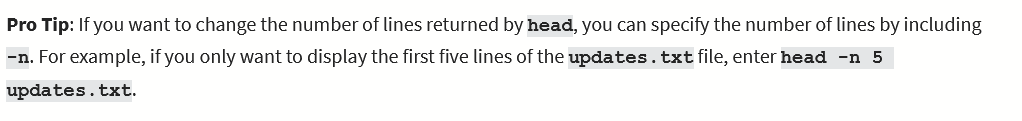
****

**Cd: navigates between directories.**

****

**Cat: displays contents of a file**

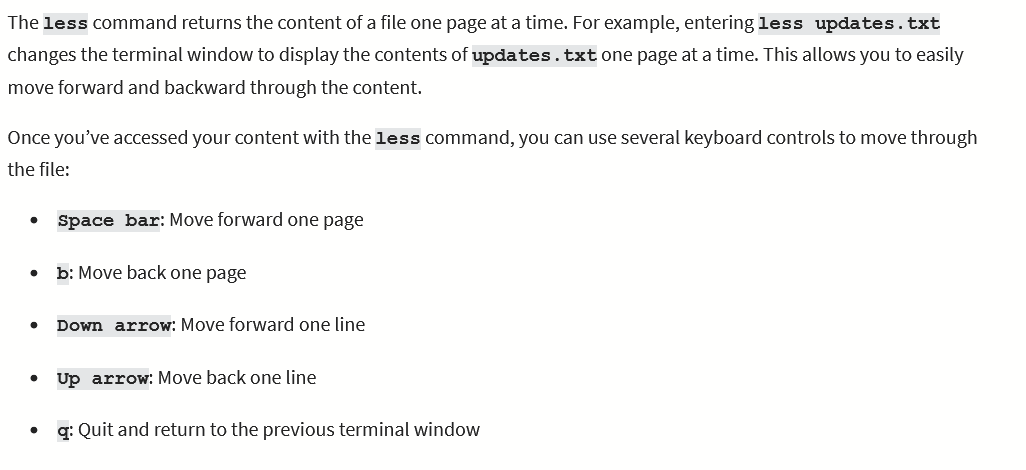
**Head: displays beginning of a file (10 lines)**

****

**Tail: displays end of a file (10 lines)**

****

**Less: displays contents of a file one page of a time. Use keyboard to navigate file**

****

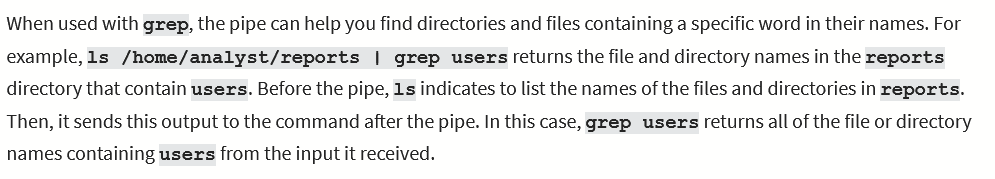
**Man hier: learn more about FHS and standard directories**

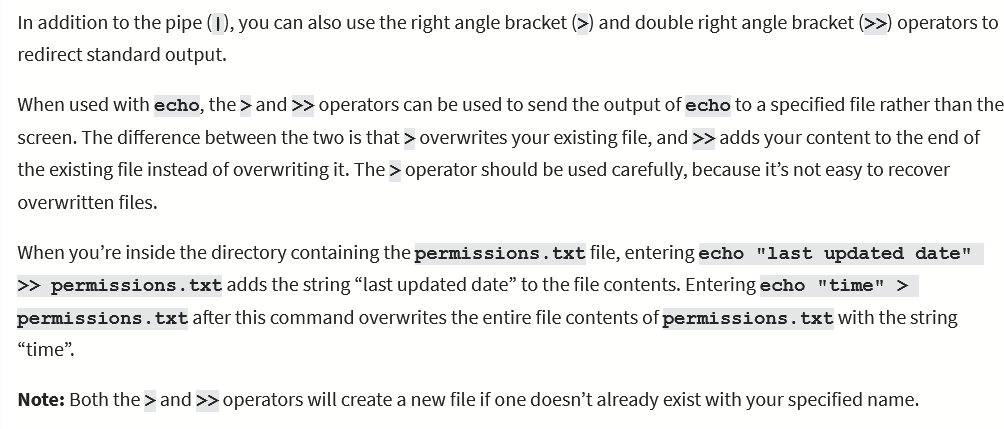
**Whoami: returns username of current user**

**Grep: searches file, and returns all lines that contain string argument**

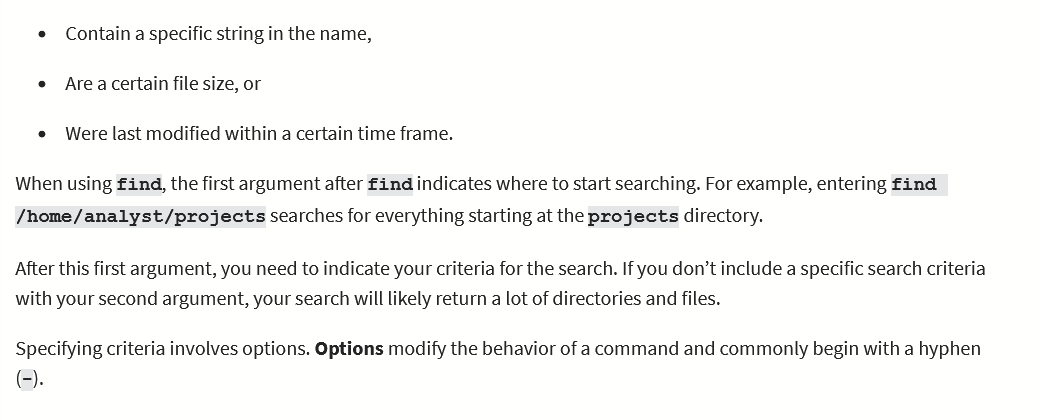
****

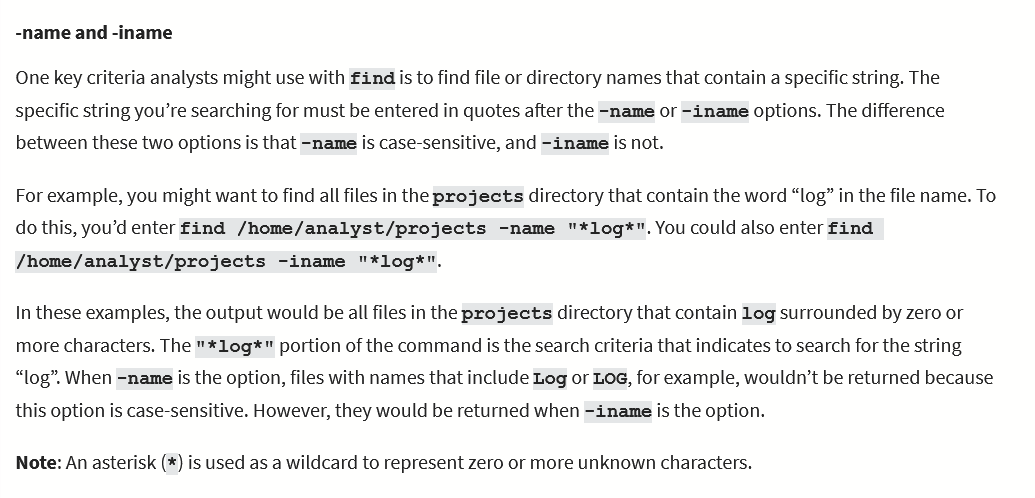
**I/>/>> (piping): sends the std output of one command as std input of another command**

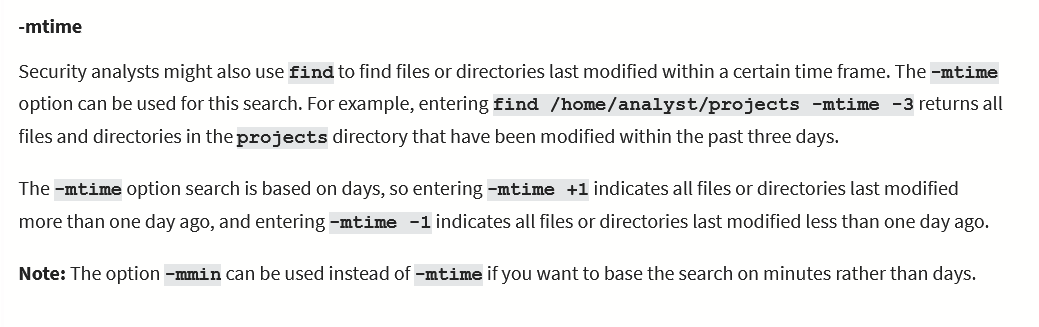
****

****

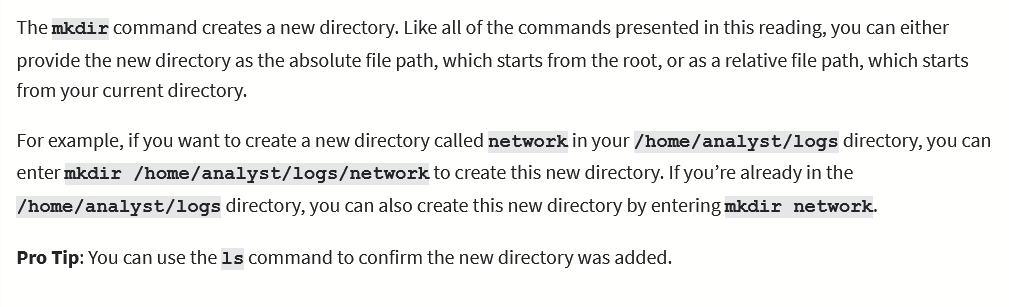
**Find: searches directories and files based on specific criteria**

****

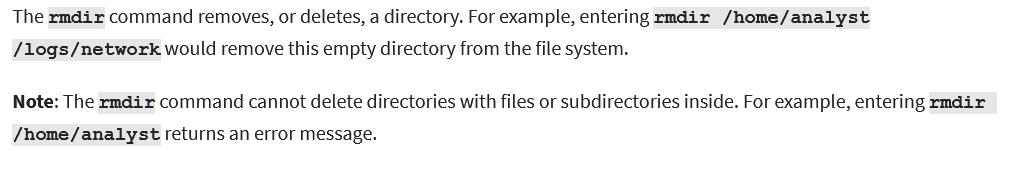
****

****

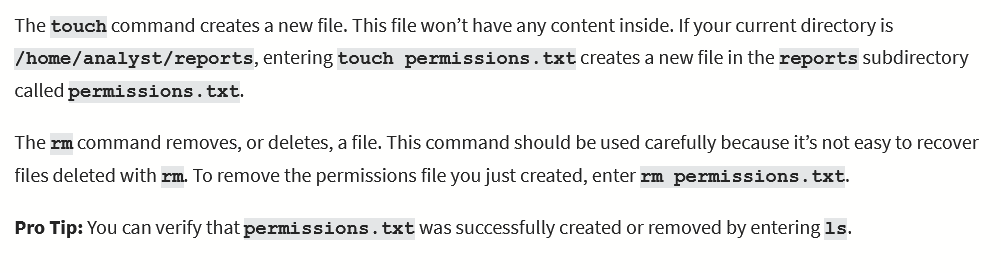
**Mkdir: makes a directory by entering absolute file path to directory**

****

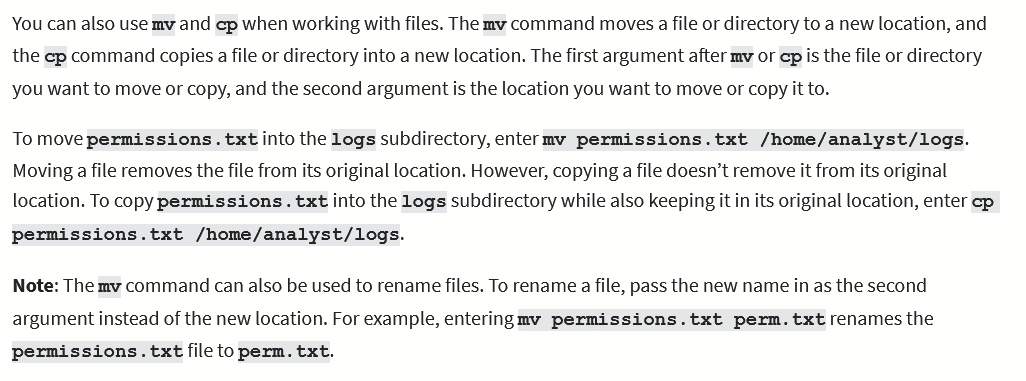
**Rmdir: removes/deletes directories (can’t have files/subdirectories in them)**

****

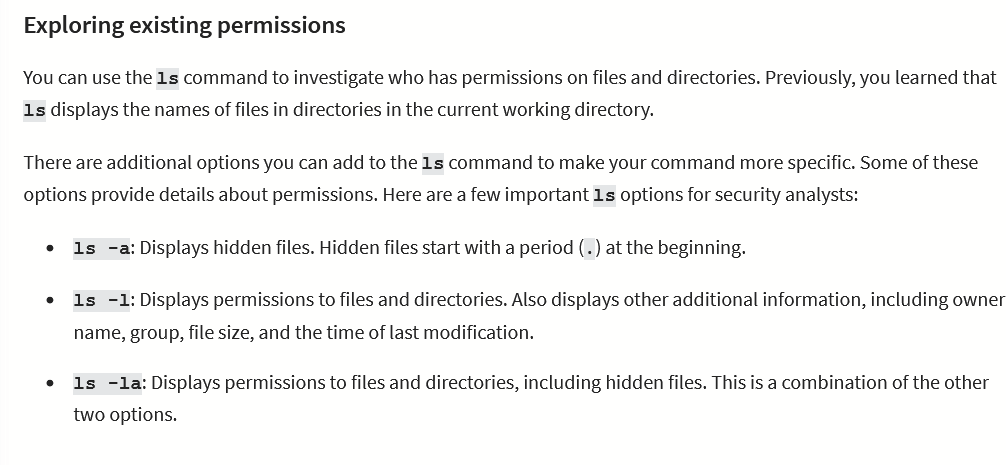
**Touch/rm: creates/removes file.**

****

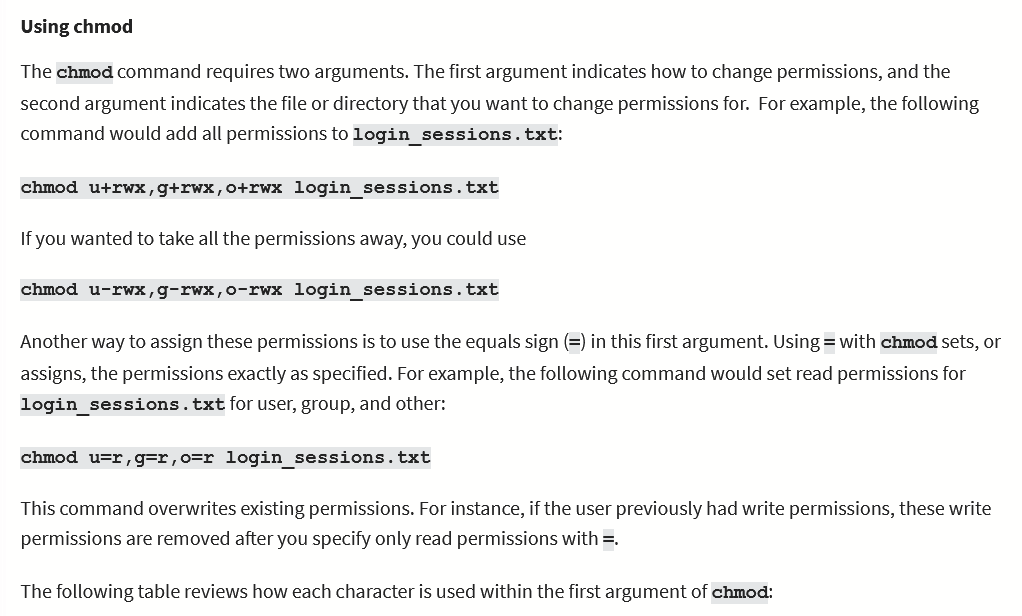
**mv/cp: moves/copies file/dir into another directory. First argument: file/dir, 2nd argument: destination**

****

**Permissions**

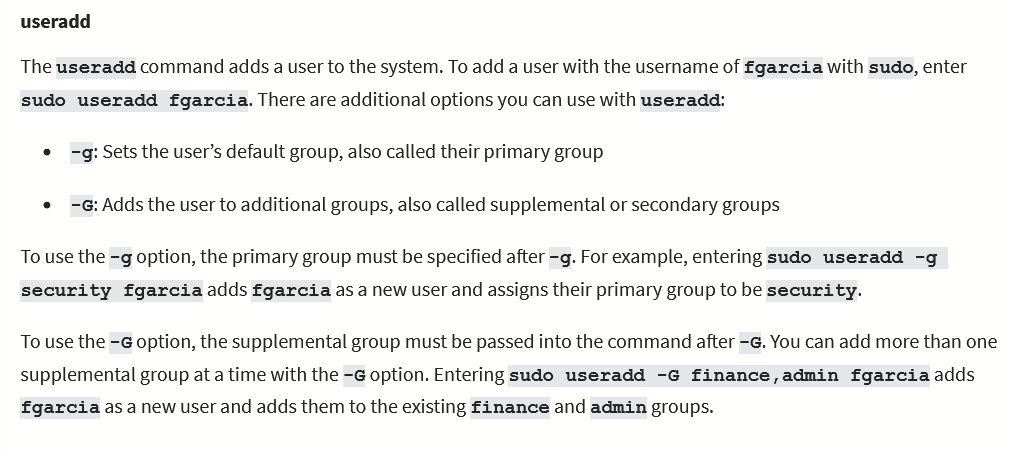
****

**Chmod:**

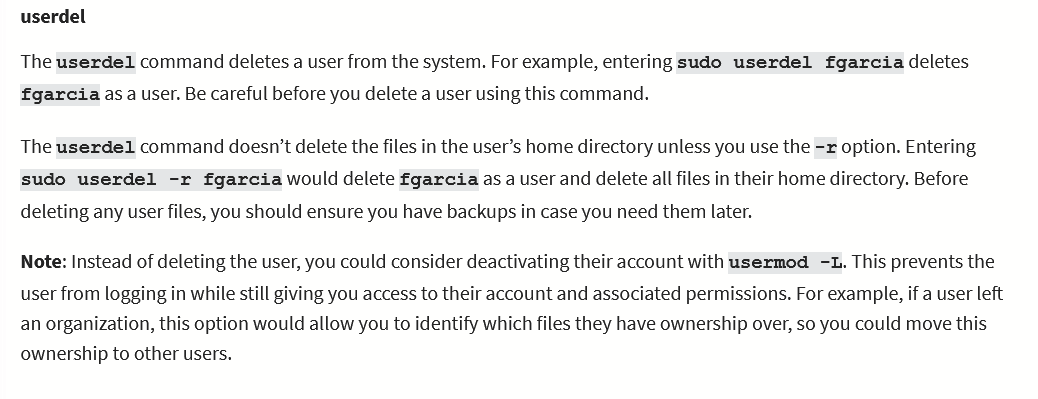
****

**Sudo: super users do. Temporary root user commands to modify users in a system**

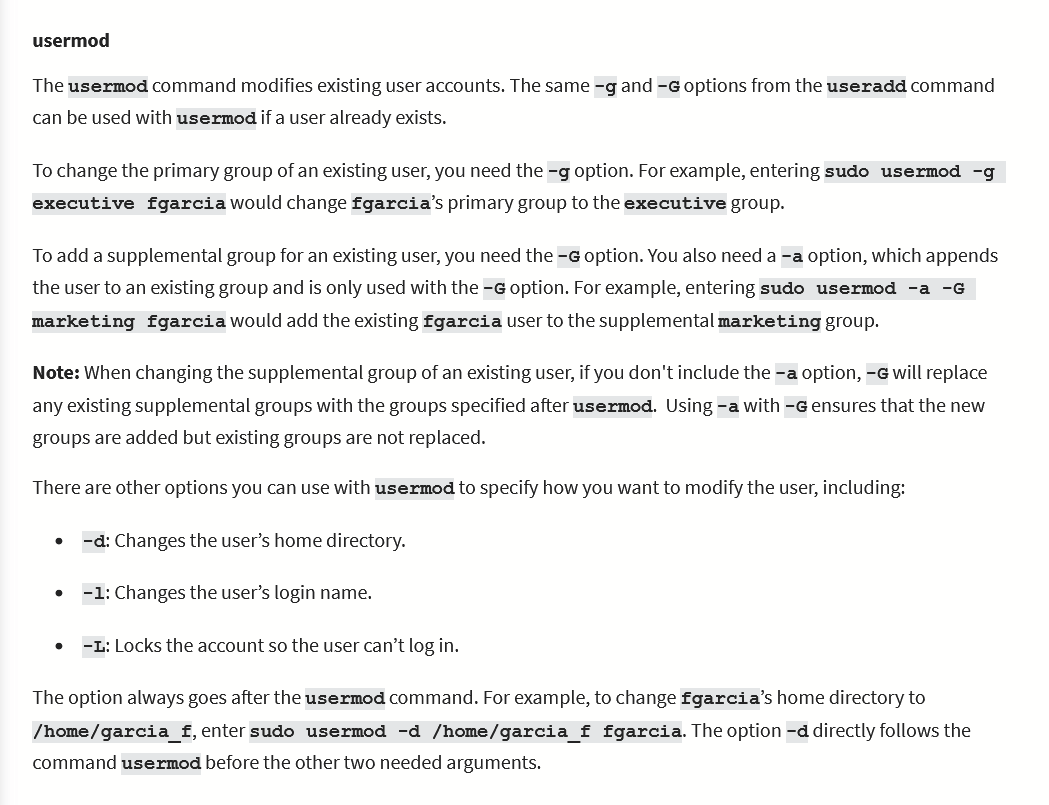
**Useradd**

****

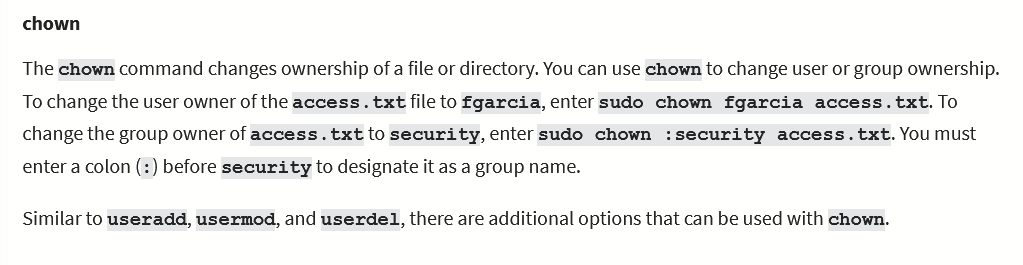
**Userdel**

****

**Usermod**

****

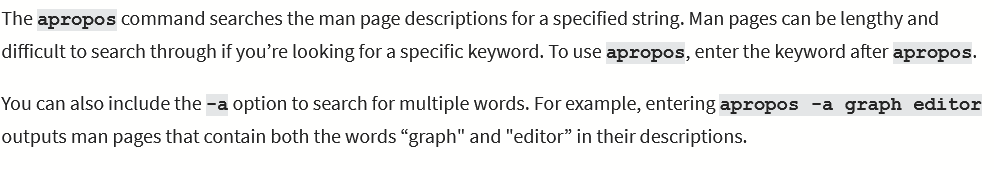
**Chown**

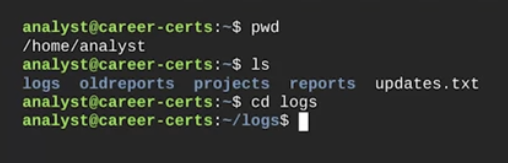
****

**Man: displays help, detailed**

**Whatis: displays briefly what a command does**

**Apropos: searches manuel description for specific string**

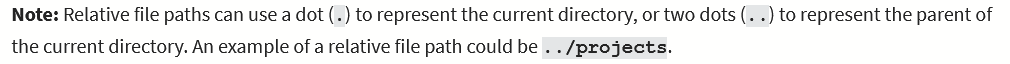
****

****

Blue = directories, white = files

Absolute path: path all the way from the beginning to the directory

Relative path: use dots to represent current/parent directories to lead to the directory



When path leads from user’s home directory, ~ can be used to replace home directory. E.g. /home/analyst/logs = ~/logs

Filtering: finding specific information in a wide set of data

Nano: command line editor widely used. Use $ nano and the name of the file in the working directory to open in nano 

If permissions.txt doesn’t exist in reports directory, the same command will create a new file

Three permissions on linux:

Read: able to read all files in a specific directory/read a single file

Write: able to write on a file/create files in a directory

Execute: access all files in a directory

Three types of owners on linux:

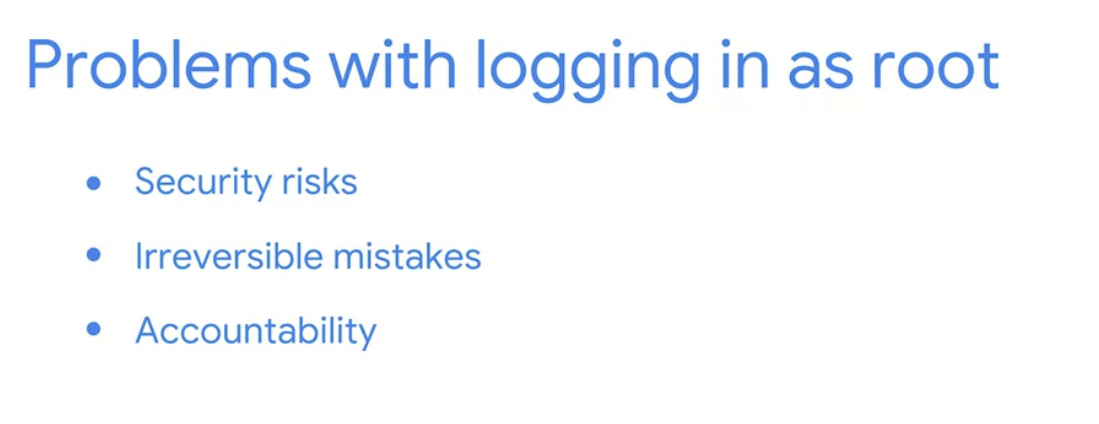
User: owner/creator of file

Group: group that the user is a part of

Other: all other users on the system



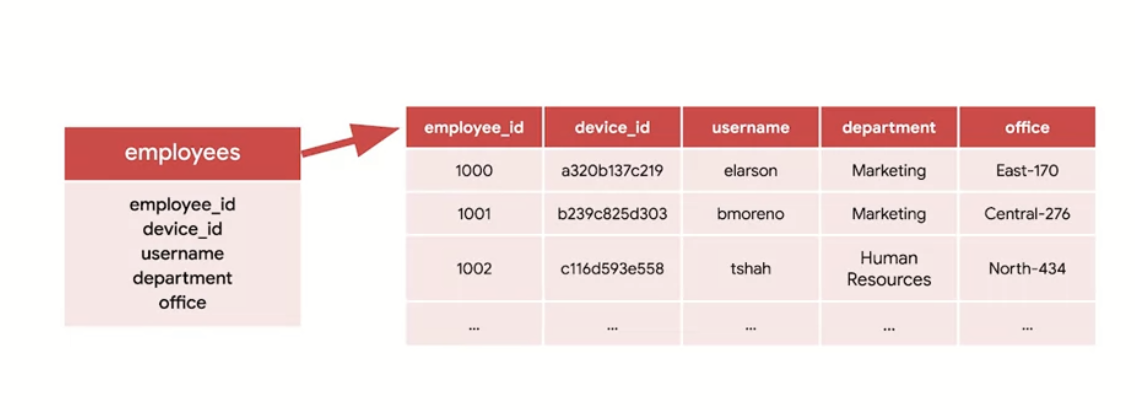
D stands for directory, r = read, w = write, x = execute. First 3 letters is for user, next is for group, next is for other. If any is missing (e.g. other cant execute on the directory), replace the letter with a hyphen (in this case, the last x is replaced with a hyphen)

Root user: user with elevated privileges to modify the system. Only ones that can add users

Sudo: temporary super user command privileges. Can be granted through a sudo configuration file.

SQL  
  
Database: organized collection of data, multiple users, massive amounts of data, perform complex tasks

Relational database: database containing tables that are related to each other



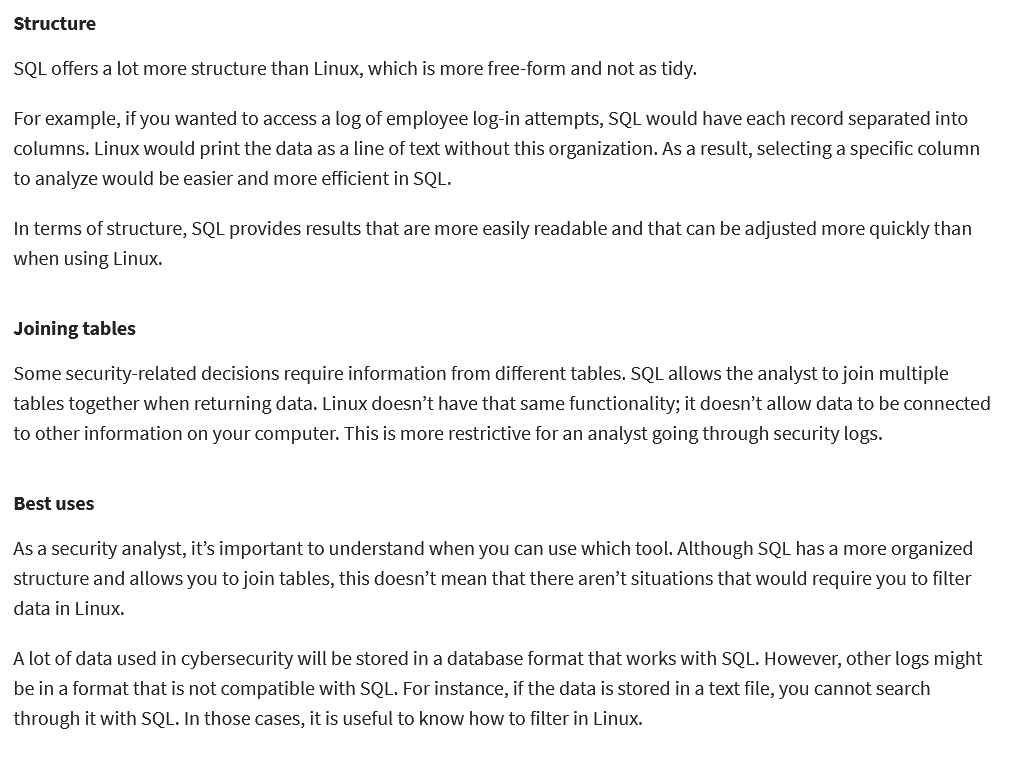
Keys: columns that relate two tables

Primary key: column where every row has unique entry, e.g. employee\_id

Foreign key: column in a table which is a primary key in another table

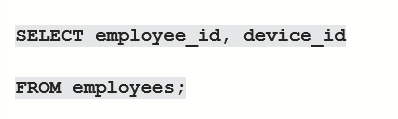
Query: request of data from a database table or a combination of tables

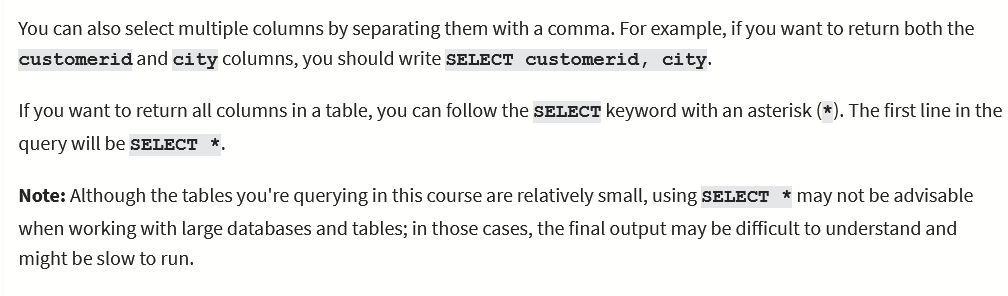
Can access sql through linux CLI: sqlite3 to start up sql lite

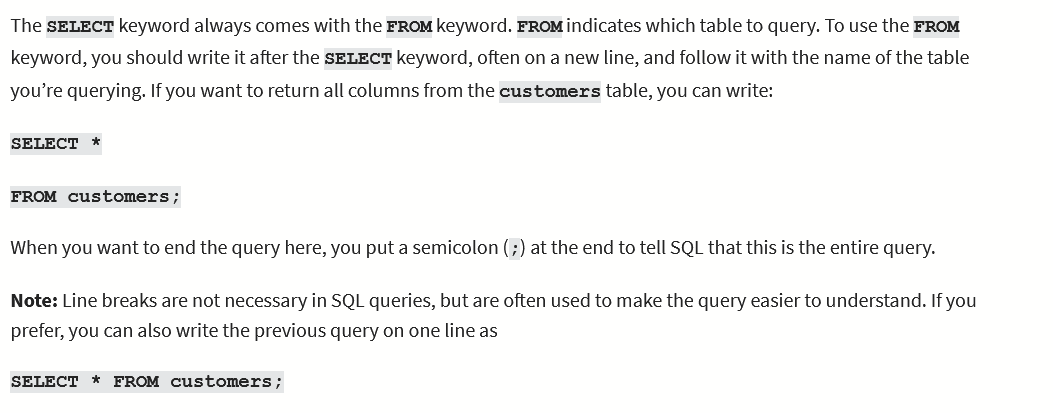


**Sql Commands**

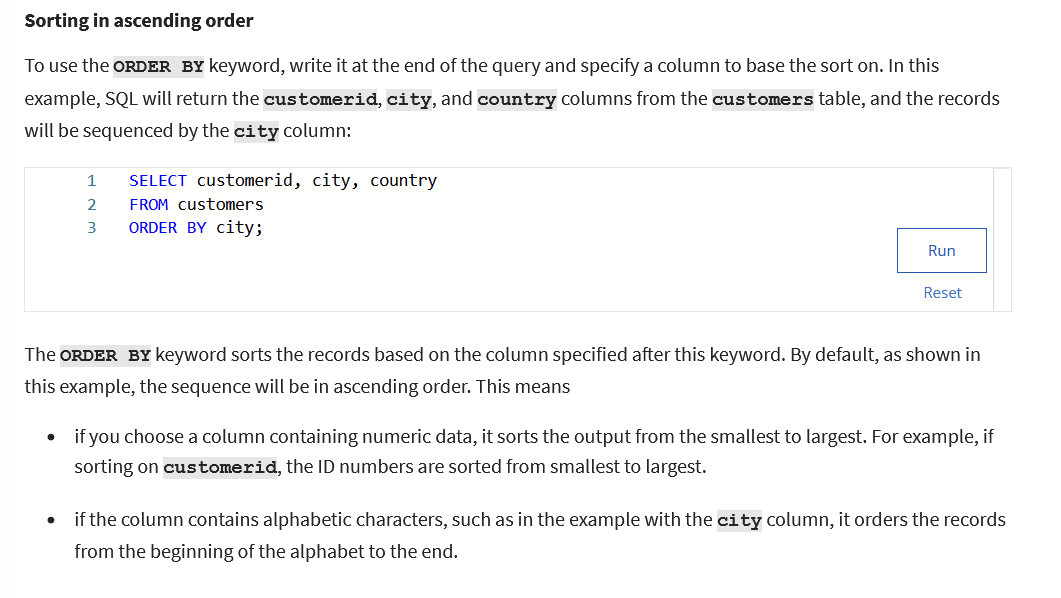
**Select;from: select chooses what data to query, from selects where to get the data from**

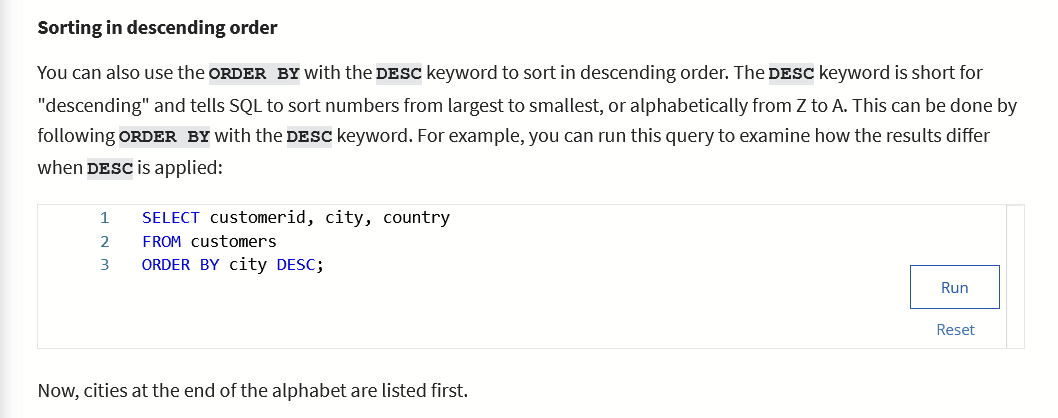
****

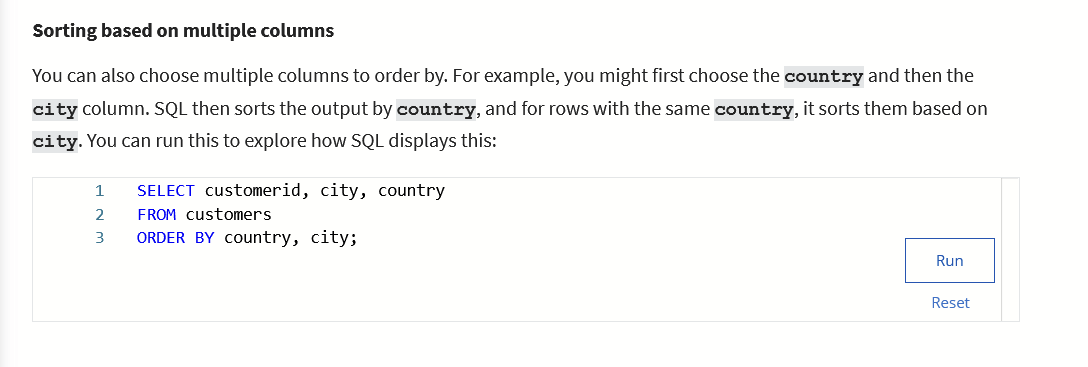
****

****

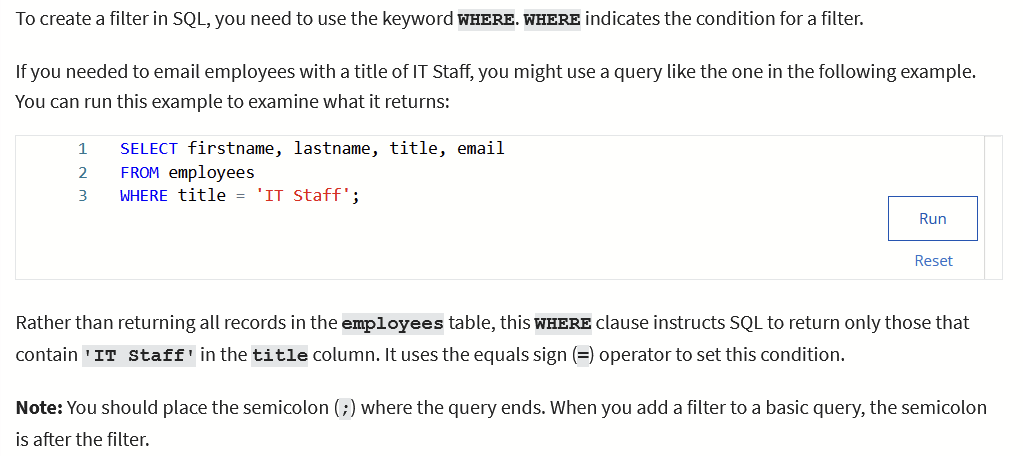
**Order by: orders query by a category**

****

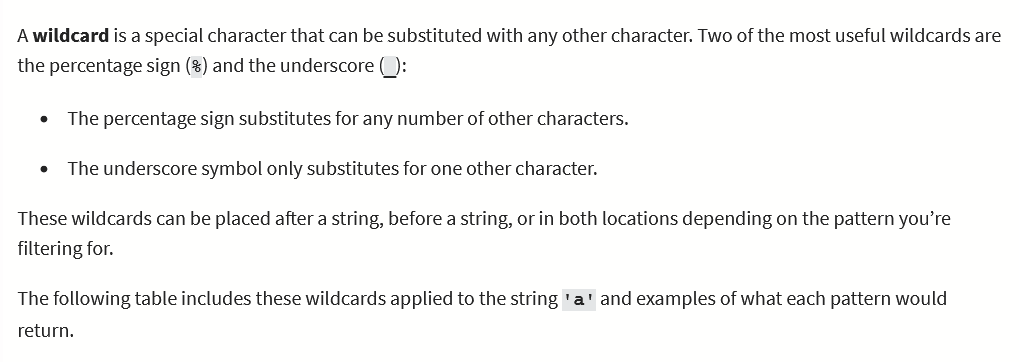
****

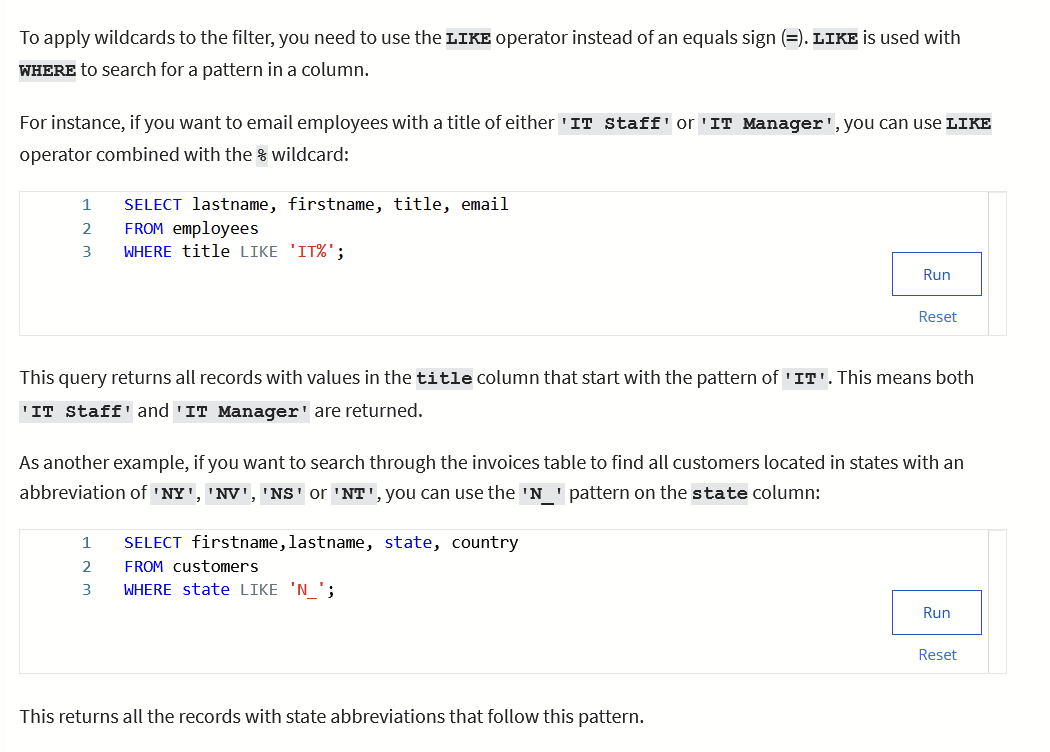
****

**Where: indication for a filter**

****

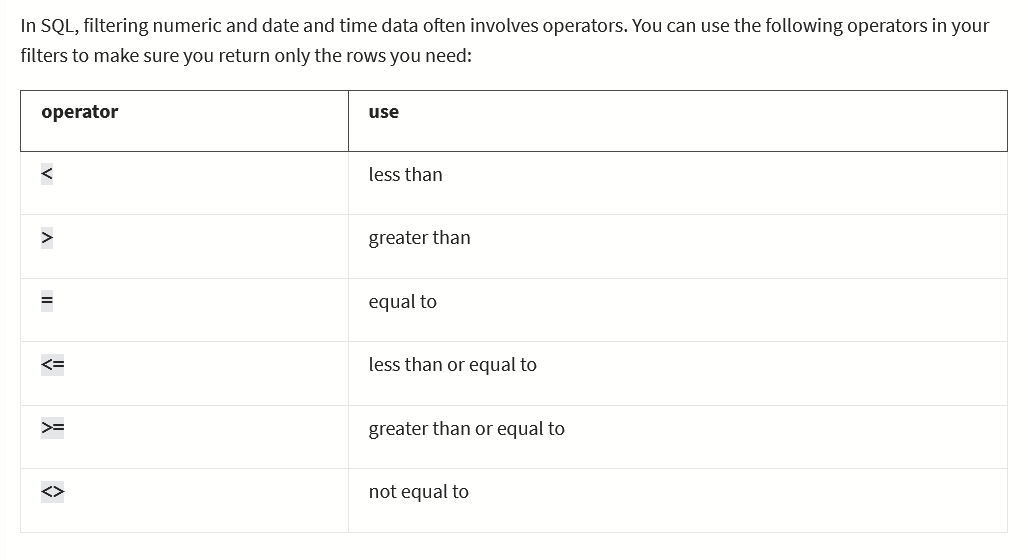
**Like: used with where to search for a pattern in a column**

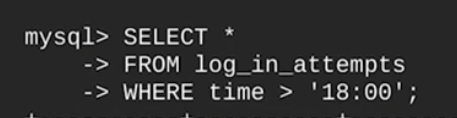
****

****

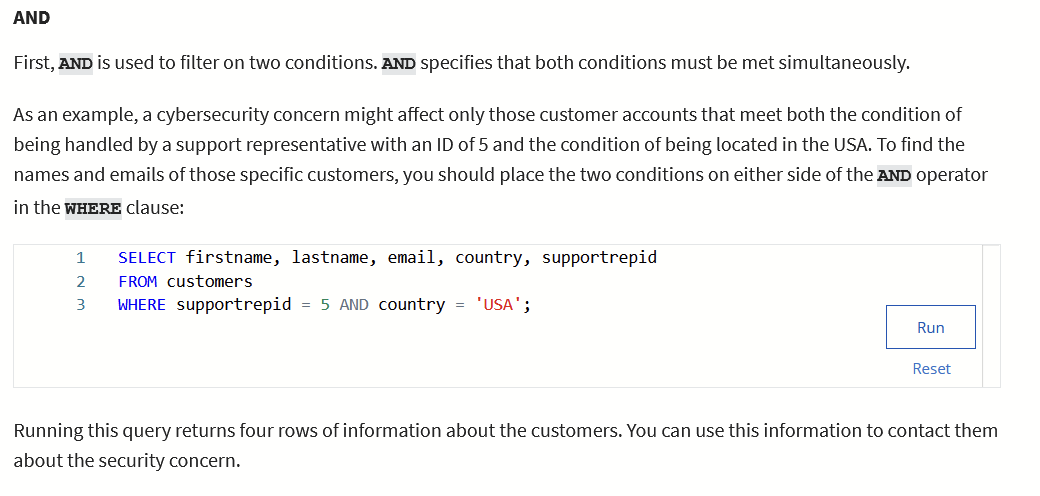
**Between: operator that filters for numbers or dates within a range; used w/ WHERE**

**Operators:**

****

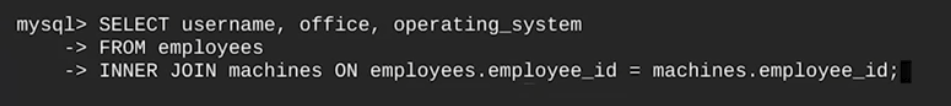


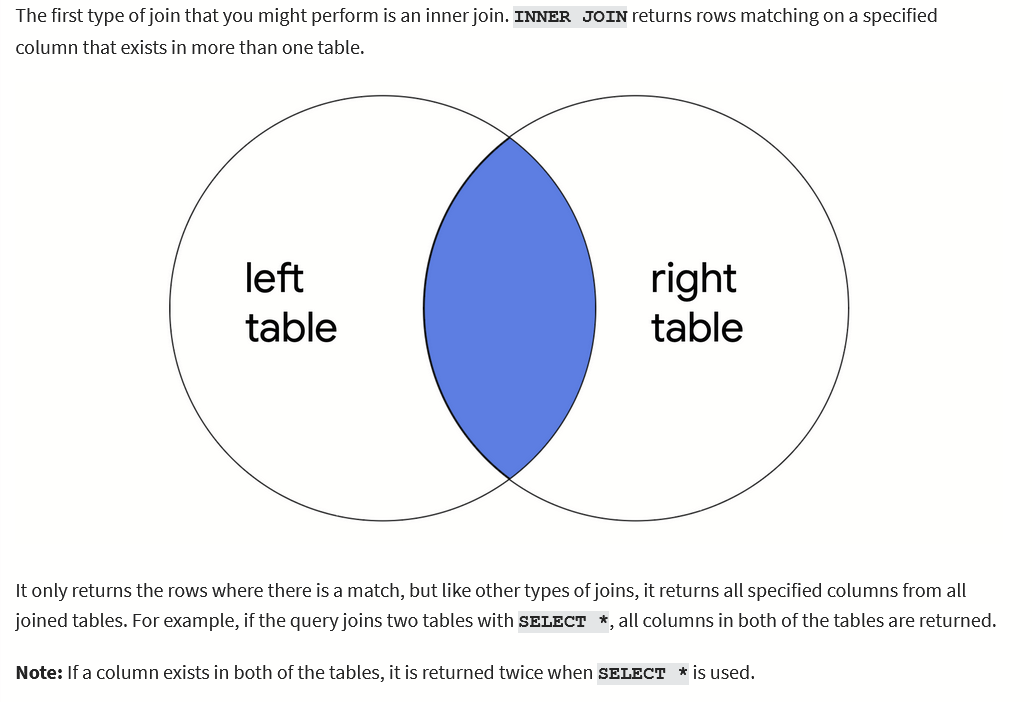
**AND OR NOT**

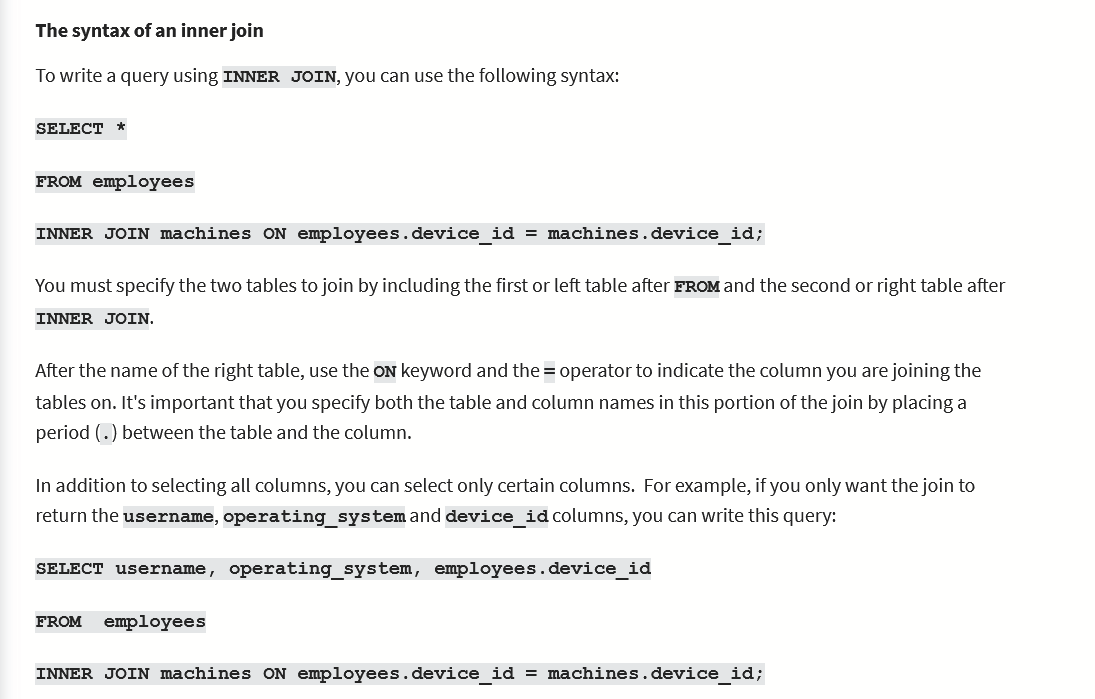
****

**Can do same with “or” and “not”**

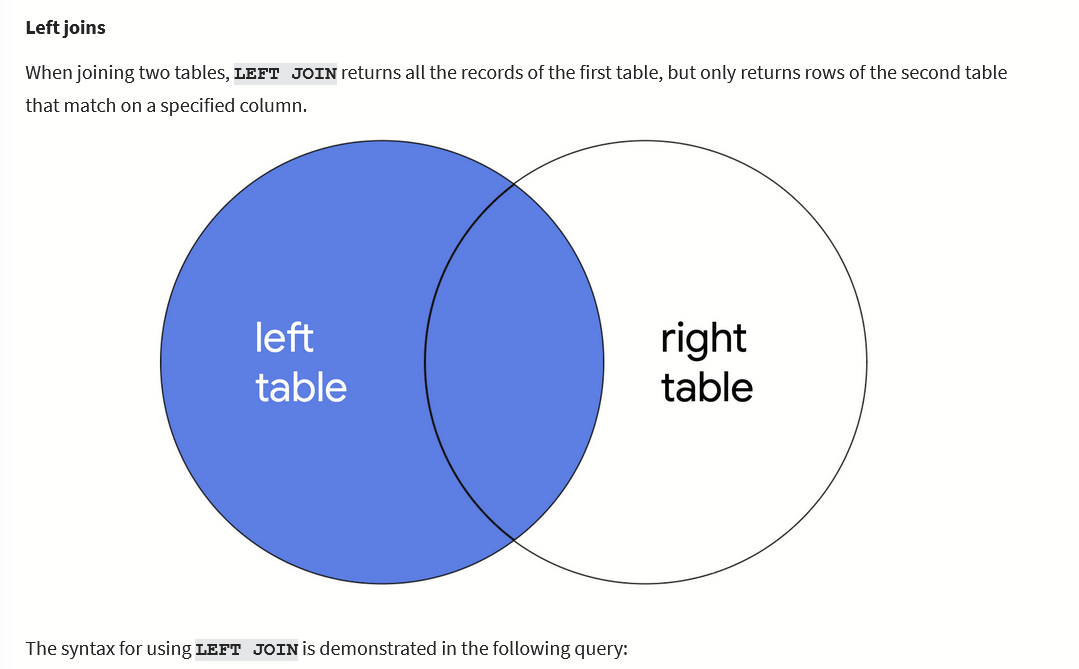
**Inner Join: join two tables using primary key and foreign key**

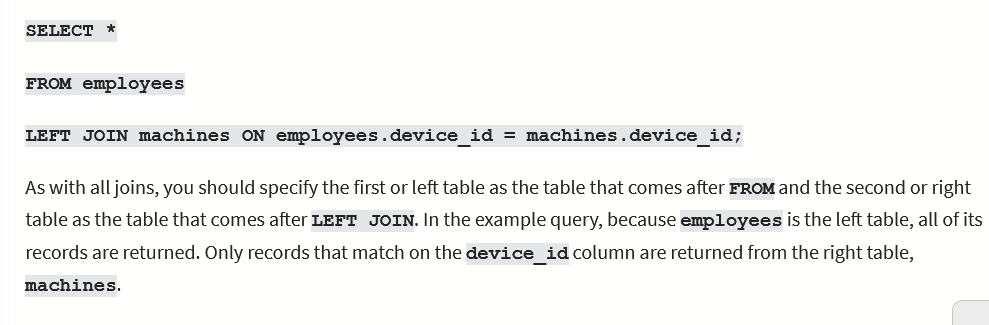
****

****

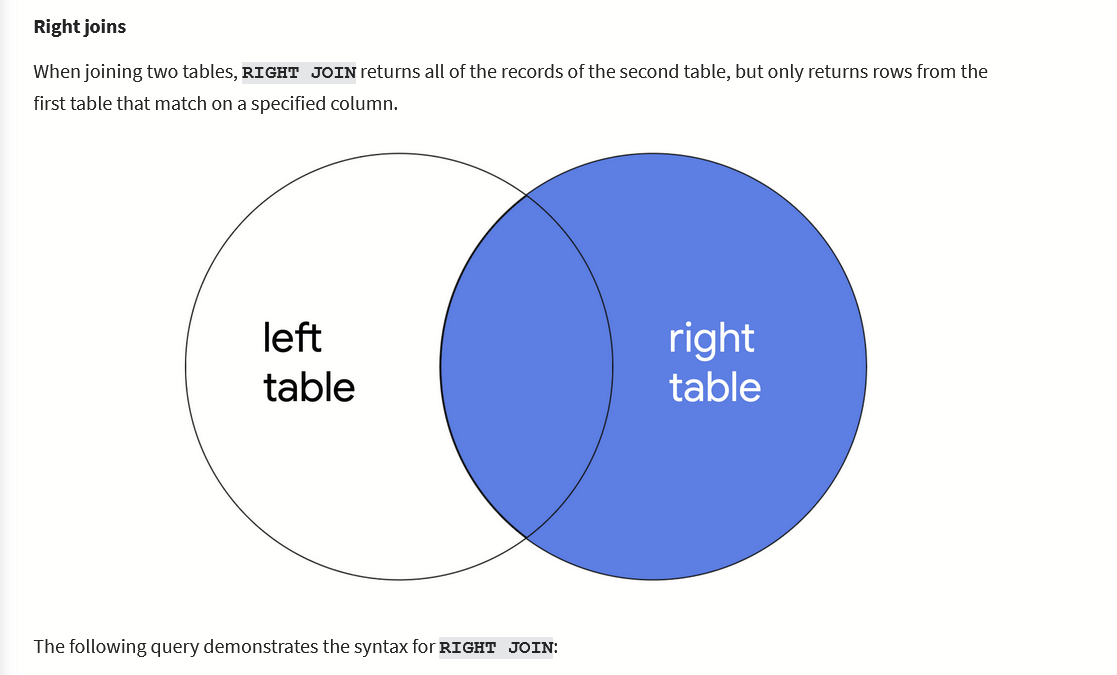
****

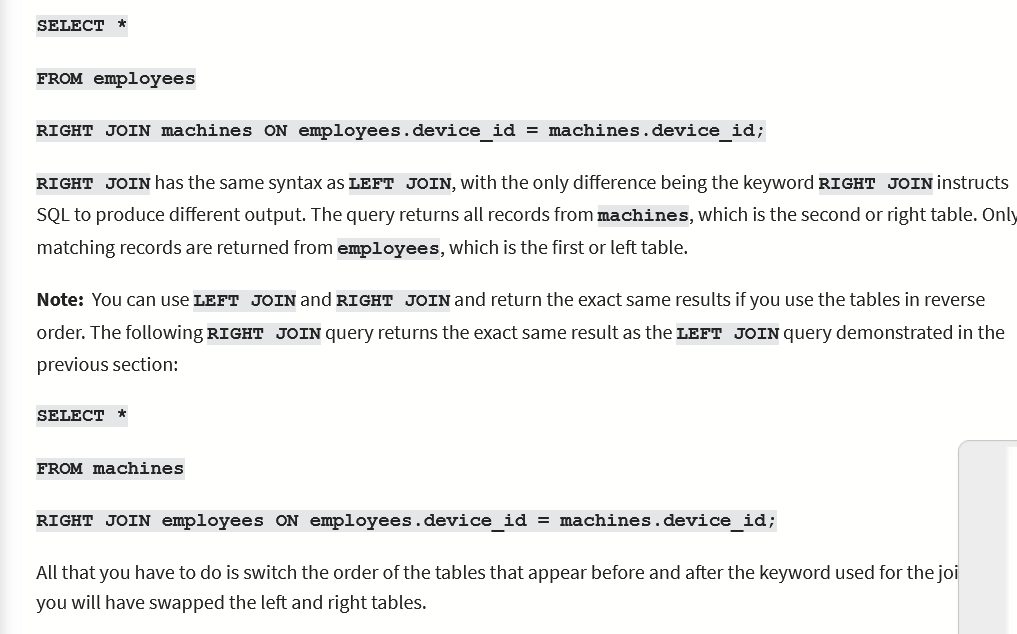
**Left Join**

****

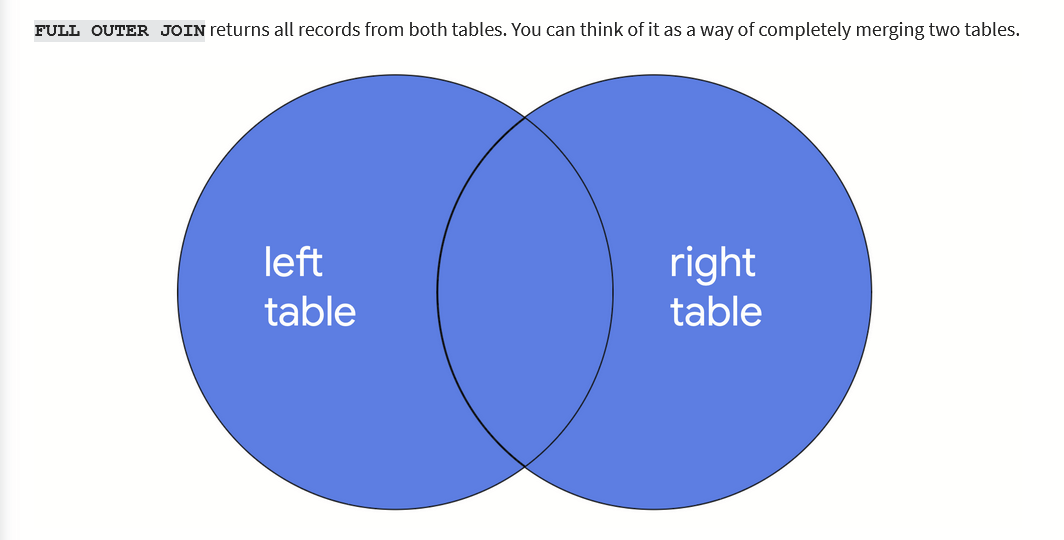
****

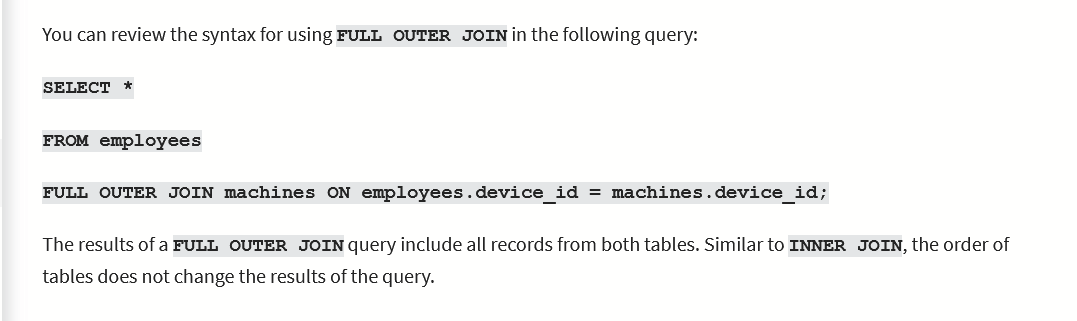
**Right Join**

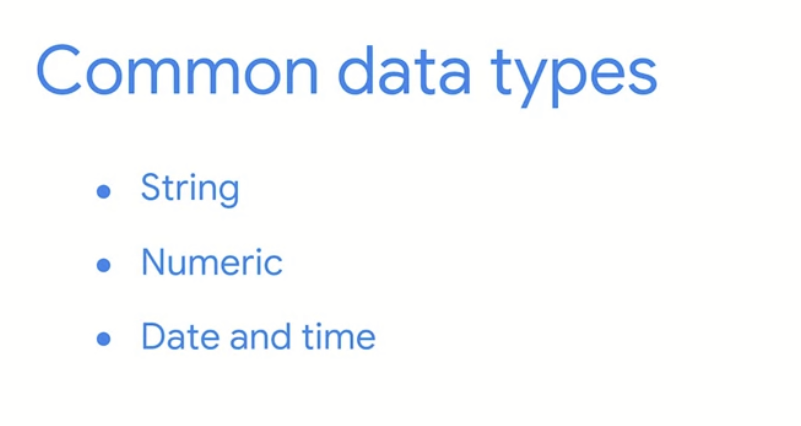
****

****

**Full Outer Joins**

****

****

****

String data: ordered sequence of chars

Numeric: data consisting of numbers

D and T: data representing a date or time

