4.4.1 : the first process with PID = 0 is the process that will be created by the OS during boot time and is the only process with no parent at all.

The second process with PID = 1 the first process started by linux kernel. This is the init process and any child process without a parent during runtimes will be assigned to init and will become the parent of that process.

The third process with PID = 9 is the –bash command that we've entered to Ubuntu

The last process with PID = 22 and entered by the user is the ps aux command that actually shows us all the running programs on our system.

4.4.3 : the wait(NULL) command will actually block parent processes until any of its children has finished and if the child terminates before parent process reaches wait(NULL) then it will read the exit status, release the process entry in the process table and continue execution until it finishes. Basically parent process will be blocked until child process returns an exit status to the operating system which is then returned to parent process.

Also if parent process finishes first then the child process will be assigned to init as its child and init will wait and release the process entry(memory) in the process table

4.4.4 : Difference between excecv and execl :in execv there exists a **list** of one or more pointers that point to a null terminated string that represent the argument list available to the **executed program,** but in execv there exists an **array** of pointers that point to a null terminated string that represebt the argument list available to the **new program.** Also in the end in execl a list of arguments must be terminated by a NULL pointer but in execv an array of pointers gets terminated by the same NULL pointer.

Difference between execvp, execlp and the other two discussed above is that here they look for possible errors that might occur while running a certain process. The return value is -1, and **errno** is set to indicate the error. Also The **execlp**(), **execvp**() functions duplicate the actions of the shell in searching for an executable file if the specified filename does not contain a slash (/) character.