1.Develop a c program to implement the Process system calls (fork (), exec(), wait(), create process, terminate process)

PROGRAM

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <sys/types.h>
#include <sys/wait.h>

int main() {
    pid_t child_pid;
    int status;
    // Create a child process using fork()
    child_pid = fork();
    if (child_pid == -1) {
        perror("Fork failed");
        exit(1);
    }

    if (child_pid == 0) {
        // This code block will be executed by the child process

        // Replace the child process with a new program using exec()
        execl("/bin/ls", "ls", "-l", NULL);

        // If execl() fails, the code below will be executed
        perror("Exec failed");
        exit(1);
    } else {
        // This code block will be executed by the parent process

        printf("Parent process: Child process ID = %d\n", child_pid);

        // Wait for the child process to terminate using wait()
        wait(&status);

        if (WIFEXITED(status)) {
            printf("Parent process: Child exited with status %d\n", WEXITSTATUS(status));
        } else {
            printf("Parent process: Child did not exit normally\n");
        }
        // You can continue executing code in the parent process here
    }

    return 0;
}
```

OUTPUT:

Parent process: PID=2249, child PID=2250
Child process: PID=2250
a.out p1.c p7.c
parent process exiting

VIVA Questions:

1. How is Exec() different from fork() ?
2. what are the different variants of Exec()?