#include <stdlib.h> // 문자열 반환 위함.

#include <stdio.h> // main() 등의 함수를 쓰기위함.

#include <unistd.h> // getpid() 등의 함수를 쓰기위함.

#include <sys/types.h> // wait() 함수를 쓰기위함.

#include <time.h> // time() 함수를 쓰기위함.

int main(){ // main 함수 호출

pid\_t pid;

pid\_t ppid; //pid, ppid 값을 pid, ppid 식별자를 이용해서 저장하기위함.

int status; //wait(&status)의 값을 반환하기 위한 선언

int count = 1;

time\_t crtTime; //시간 식별자

time(&crtTime); //현재 시간 반환

ppid = getpid();

for(int i = 0; i<10 ; i++){ //자식 프로세스를 만들기 위한 loop

if(ppid==getpid()) {

pid = fork();// 자식 프로세스 생성

}

}

if(pid < 0) {

exit(-1);

}

else if(pid == 0){//pid 값이 0이면 자식이므로 이때 과제에서 원한 결과 값을 출력하기위한 for루프를 추가.

for(int j = 0 ; j<100 ; j++){

printf("[PID = %d, Count = #%d, PPID = %d] datetime = %s", getpid(), count, getppid(), ctime(&crtTime));

count = count +1;

}

}

else{

for(int k = 0 ; k<10 ; k++){//자식프로세스가 종료되었을 때 부모프로세스의 ppid값을 반환하기위한 for loop 10개의 자식프로세스가 종료되어야 하므로 10번 반복한다.

printf("child has finished: PID = %d\n ", wait(&status) );

}

}

return 0; //main 함수의 반환값.

}