



## -----Slip 9-----

**Q.1) Generate parent process to write unnamed pipe and will read from it [10 Marks ]**

```
#include <stdio.h>

#include <unistd.h>
#include <string.h>

int main() {
    int fd[2];

    char write_msg[] = "Hello from parent";
    char read_msg[100];

    if (pipe(fd) == -1) {
        perror("pipe");
        return 1;
    }

    // Parent writes to pipe
    write(fd[1], write_msg, strlen(write_msg) + 1);

    // Now read from pipe
    read(fd[0], read_msg, sizeof(read_msg));

    printf("Parent read from pipe: %s\n", read_msg);

    //close both ends
    Close(fd[0]);
    Close(fd[1]);

    Return 0;
}
```

**Command to run:**cc s9q1.c

./a.out





**Q2. Write a C program to Identify the type (Directory, character device, Block device, Regular file, FIFO or pipe, symbolic link or socket) of given file using stat() system call. [20 Marks ]**

```
#include <stdio.h>
```

```
#include <sys/stat.h>
```

```
int main(int argc, char *argv[]) {
```

```
    if (argc != 2) {
```

```
        printf("Usage: %s <file_name>\n", argv[0]);
```

```
        return 1;
```

```
    }
```

```
    struct stat fileStat;
```

```
    if (stat(argv[1], &fileStat) < 0) {
```

```
        perror("stat");
```

```
        return 1;
```

```
    }
```

```
    printf("File type of \"%s\": ", argv[1]);
```

```
    if (S_ISREG(fileStat.st_mode))
```

```
        printf("Regular File\n");
```

```
    else if (S_ISDIR(fileStat.st_mode))
```





```
        printf("Directory\n");
    else if (S_ISCHR(fileStat.st_mode))
        printf("Character Device\n");
    else if (S_ISBLK(fileStat.st_mode))
        printf("Block Device\n");
    else if (S_ISFIFO(fileStat.st_mode))
        printf("FIFO/Pipe\n");
    else if (S_ISLNK(fileStat.st_mode))
        printf("Symbolic Link\n");
    else if (S_ISSOCK(fileStat.st_mode))
        printf("Socket\n");
    else
        printf("Unknown Type\n");

    return 0;
}
```

**Command to run : cc s9q2.c**

**./a.out output.txt**

