Named Pipes

Demo

Syntax

```
#include<sys/types.h>
#include<sys/stat.h>
int mkfifo(const char *pathname, mode_t mode);
```

FIFO NAMED PIPE WORKS IN BLOCKED MODE

BOTH READER AND RECEIVER MUST BE PRESENT

How it works

STEP ONE	STEP TWO	STEP THREE
Create a named PIPE	CREATE A SENDER PROCESS (WRITE)	Create a Reader PROCESS (READ)

1. Create a Named PIPE

```c
#include<stdio.h>
#include<sys/types.h>
#include<sys/stat.h>
int main()
{
 int res;
 res=mkfifo("fifo1",0777);//creates a named pipe
 printf("named piped created\n");
}
```

2. Create a SENDER Process

```c
#include<unistd.h>
#include<stdio.h>
#include<fcntl.h>
int main()
{
 int res, n;
 res=open("fifo1",O_WRONLY);
 write(res,"Message",7);
 printf("Sender Process with pid %d send a message",getpid());
}
```

3. Create a RECEIVER Process

```c
#include<unistd.h>
#include<stdio.h>
#include<fcntl.h>
int main()
{
 int res,n;
 char buffer[100];
 res=open("fifo1",O_RDONLY);
 n=read(res,buffer,100);
 printf("Reader Process having pid %d has started",getpid());
 printf("Data Recieved by Reciever %d is %s",getpid(),buffer);
}
```

Running both in BLOCK Mode

```
./sender.out & ./rece.out
```