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

sem_t readCountAccess;
sem_t databaseAccess;
int readCount=0;

void *Reader(void *arg);
void *Writer(void *arg);

int main()
{
int i=0,NumberofReaderThread=0,NumberofWriterThread;
sem_init(&readCountAccess,0,1);
sem_init(&databaseAccess,0,1);

pthread_t Readers_thr[100],Writer_thr[100];
printf("\nEnter number of Readers thread(MAX 10)");
scanf("%d",&NumberofReaderThread);
printf("\nEnter number of Writers thread(MAX 10)");
scanf("%d",&NumberofWriterThread);

for(i=0;i<numberofreaderthread;i++)
{
pthread_create(&Readers_thr[i],NULL,Reader,(void *)i);
}
for(i=0;i<numberofwriterthread;i++)
{
pthread_create(&Writer_thr[i],NULL,Writer,(void *)i);
}
for(i=0;i<NumberofWriterThread;i++)
{
pthread_join(Writer_thr[i],NULL);
}

for(i=0;i<NumberofReaderThread;i++)
{
pthread_join(Readers_thr[i],NULL);
}
sem_destroy(&databaseAccess);
sem_destroy(&readCountAccess);
return 0;
}

void * Writer(void *arg)
{

sleep(1);
int temp=(int)arg;
printf("\nWriter %d is trying to enter into database for modifying the data",temp);
sem_wait(&databaseAccess);
printf("\nWriter %d is writting into the database",temp);
printf("\nWriter %d is leaving the database");
```

```
sem_post(&databaseAccess);
}

void *Reader(void *arg)
{
sleep(1);
int temp=(int)arg;
printf("\nReader %d is trying to enter into the Database for reading the data",temp);
sem_wait(&readCountAccess);
readCount++;
if(readCount==1)
{
sem_wait(&databaseAccess);
printf("\nReader %d is reading the database",temp);
}
sem_post(&readCountAccess);
sem_wait(&readCountAccess);
readCount–;
if(readCount==0)
{
printf("\nReader %d is leaving the database",temp);
sem_post(&databaseAccess);
}
sem_post(&readCountAccess);
}
```