**Booking System App:**

**Code 1:**

**package** synchronization;

**class** synchmethod {

**int** total\_seats=10;

**void** busTicket(**int** seats) {

**if**(total\_seats>=seats)

{

System.***out***.println(seats+ " Seats booked successfully");

total\_seats=total\_seats-seats;

System.***out***.println("seats left "+total\_seats);

}

**else** {

System.***out***.println("Sorry Seats cannot be booked....!!");

System.***out***.println("seats left "+total\_seats);

}}

}

**class** Bus **extends** Thread

{

**static** synchmethod *b*;

**int** seats;

**public** **void** run()

{

*b*. busTicket(seats);

}

**public** **static** **void** main(String args[])

{

*b* =**new** synchmethod();

Bus t1 =**new** Bus();

t1.seats=7;

t1.start();

Bus t2 =**new** Bus();

t2.seats=7;

t2.start();

Bus t3 =**new** Bus();

t3.seats=7;

t3.start();

}

}

**Code 2:**

**package** synchronization;

**class** synchmethod {

**int** total\_seats=10;

**synchronized** **void** busTicket(**int** seats) {

**if**(total\_seats>=seats)

{

System.***out***.println(seats+ " Seats booked successfully");

total\_seats=total\_seats-seats;

System.***out***.println("seats left "+total\_seats);

}

**else** {

System.***out***.println("Sorry Seats cannot be booked....!!");

System.***out***.println("seats left "+total\_seats);

}}

}

**class** Bus **extends** Thread

{

**static** synchmethod *b*;

**int** seats;

**public** **void** run()

{

*b*. busTicket(seats);

}

**public** **static** **void** main(String args[])

{

*b* =**new** synchmethod();

Bus t1 =**new** Bus();

t1.seats=7;

t1.start();

Bus t2 =**new** Bus();

t2.seats=7;

t2.start();

Bus t3 =**new** Bus();

t3.seats=7;

t3.start();

}

}

**Code 3:**

//synchronized block

**package** synchronization;

**public** **class** Synchronizedblock

{

**int** total\_seats=10;

**void** busTicket(**int** seats) {

**synchronized**(**this**) {

**if**(total\_seats>=seats)

{

System.***out***.println(seats+ " Seats booked successfully");

total\_seats=total\_seats-seats;

System.***out***.println("seats left "+total\_seats);

}

**else** {

System.***out***.println("Sorry Seats cannot be booked....!!");

System.***out***.println("seats left "+total\_seats);

}

}

}

}

**class** Busseats **extends** Thread

{

**static** Synchronizedblock *b*;

**int** seats;

**public** **void** run()

{

*b*. busTicket(seats);

}

**public** **static** **void** main(String args[])

{

*b* =**new** Synchronizedblock ();

Busseats t1 =**new** Busseats ();

t1.seats=7;

t1.start();

Busseats t2 =**new** Busseats ();

t2.seats=7;

t2.start();

}

}

**Code 4:**

**package** synchronization;

**public** **class** BusApp {

**public** **static** **void** main(String args[]){

Totalearnings te=**new** Totalearnings();

te.start();

System.***out***.println("Total earnings:" +te.total+"rs");

}}

**class** Totalearnings **extends** Thread{

**int** total=0;

**public** **void** run()

{

**for**(**int** i=1; i<=10; i++)

{

total=total+100;

}

}

}

**Code 5:**

**package** synchronization;

**public** **class** Busappsol {

**public** **static** **void** main(String args[]) **throws** InterruptedException{

Totalearning te=**new** Totalearning();

te.start();

**synchronized**(te)

{

System.***out***.println("Total earnings:" +te.total+"rs");

te.wait(1200);

System.***out***.println("Total earnings:" +te.total+"rs");

}

}}

**class** Totalearning **extends** Thread{

**int** total=0;

**public** **void** run()

{

**synchronized**(**this**) {

**for**(**int** i=1; i<=10; i++)

{

total=total+100;

}

**this**.notify(); }

}

}

Thread priority method:

**package** projectname;

**class** threadpriority **extends** Thread{

**public** **void** run(){

System.***out***.println("running thread name is:"+Thread.*currentThread*().getName());

System.***out***.println("running thread priority is:"+Thread.*currentThread*().getPriority());

}

**public** **static** **void** main(String args[]){

threadpriority m3=**new** threadpriority();

threadpriority m1=**new** threadpriority();

threadpriority m2=**new** threadpriority();

System.***out***.println("running thread priority is:"+Thread.*currentThread*().getPriority());

m3.setPriority(4);

m3.start();

m1.setPriority(3);

m1.start();

m2.start();

m2.~~stop~~();

}

}

Thread naming method:

**package** projectname;

**class** namingthread2 **extends** threadpriority {

**public** **void** run() {

// Thread.currentThread().setName("THREADS IN JAVA");

System.***out***.println("thread task is executed by"+ Thread.*currentThread*().getName());

}

**public** **static** **void** main(String args[]){

System.***out***.println("hello is printed by "+ Thread.*currentThread*().getName());

Thread.*currentThread*().setName("sec D");

System.***out***.println("hello is printed by "+ Thread.*currentThread*().getName());

namingthread2 t=**new** namingthread2();

System.***out***.println("Name of t1:"+t.getName());

t.start();

Thread.*currentThread*().setName("developer created thread");

System.***out***.println("hello is printed by "+ Thread.*currentThread*().getName());

namingthread2 t1=**new** namingthread2();

t1.start();

System.***out***.println("hello");

//System.out.println(t.isAlive());

//System.out.println(currentThread().currentThread().isAlive()); //true

//System.out.println(t1.isAlive());

}

}

Thread Naming code 2:

**package** projectname;

**public** **class** threadnaming4 {

**public** **void** run() {

System.***out***.println("thread task is executed by "+ Thread.*currentThread*().getName());

}

**public** **static** **void** main(String args[]){

System.***out***.println("hello is printed by "+ Thread.*currentThread*().getName());

Thread.*currentThread*().setName("new main thread");

System.***out***.println("hello is printed by "+ Thread.*currentThread*().getName());

namingthread2 t=**new** namingthread2();

t.start();

t.setName("developer");

//namingthread2 t1=new namingthread2();

//t1.start();

//System.out.println(t.isAlive());

//System.out.println(currentThread().currentThread().isAlive()); //true

//System.out.println(t1.isAlive());

}

}

Sleep method in java:

**package** projectname;

**class** sleepthreadmethod **extends** Thread{

**public** **void** run(){

**for**(**int** i=1;i<5;i++){

**try**{Thread.*sleep*(5000);}**catch**(InterruptedException e){System.***out***.println(e);}

System.***out***.println(i);

}

}

**public** **static** **void** main(String args[]){

sleepthreadmethod t1=**new** sleepthreadmethod();

sleepthreadmethod t2=**new** sleepthreadmethod();

t1.start();

t2.start();

}

}