**ExceptionsDemo.java:-**

//Exception:1) checked Exception(syntact) (or) compile time Exception

//2) Unchecked Exceptions (logical) (or) run time Exception.

/\*Ex:checked Exception

public class ExceptionsDemo {

public static void main(String args[]){

System.out.println("JAVA")

}

}

\*/

/\*\*

\* **@author** Lakshman

\*

\*/

**public** **class** ExceptionsDemo {

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

**try**{

System.*out*.println("open file");

**int** b[]={10,20,30};

b[2]=100; //change index no of b[]

**int** n= args.length;

System.*out*.println("n="+n);

**int** a=40/4; //change a=40/n;

System.*out*.println("a="+a);

}

**catch**(ArithmeticException ae){

System.*out*.println(ae);

}

**catch**(ArrayIndexOutOfBoundsException aoe){

System.*out*.println(aoe);

}

**finally**

{

System.*out*.println("close file");

}

}

}

**userexception.java:-**

/\*\*

\* **@author** Lakshman

\*

\*/

**public** **class** userdefinedexception **extends** Exception{

//store a/c info

**private** **static** **int** *acno*[]={101,102,103,104,105};

**private** **static** String *name*[]={"lakshman","lakshmi harika","Saraswathi","swathi","sailu"};

**private** **static** **double** *bal*[]={10000,20000,30000,9999,11000}; //change 4th value as abow 10000...

//default constructor

userdefinedexception()

{}

userdefinedexception(String str)

{

**super**(str);

}

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

{

// **TODO** Auto-generated method stub

**try**{

System.*out*.println("acno"+"\t"+"customer"+"\t"+"balance");

**for** (**int** i = 0; i < 5; i++) {

System.*out*.println(*acno*[i]+"\t"+*name*[i]+"\t"+*bal*[i]);

**if**(*bal*[i]<10000){

userdefinedexception ude=**new** userdefinedexception("bal amount is lesss");

**throw** ude;

}

}

}

**catch** (userdefinedexception ude) {

ude.printStackTrace();

}

}

}

**Rethrowingexecption.java:-**

**public** **class** Rethrowingexecption {

/\*\*

\* **@author** Lakshman

\*

\*/

**void** method1(){

**try**{

String str="hello";

**char** ch=str.charAt(5); //change index no's

System.*out*.println(ch);

}

**catch**(StringIndexOutOfBoundsException sie){

System.*out*.println("plz see the index is wiyhin the range");

**throw** sie;

}

}

**public** **static** **class** Rethrowingexecption1{

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

// **TODO** Auto-generated method stub

Rethrowingexecption e2=**new** Rethrowingexecption();

**try**{

e2.method1();

}

**catch** (StringIndexOutOfBoundsException sie) {

System.*out*.println("i caught rethrown exception");

// **TODO**: handle exception

}

}

}

}

//throw :To throw an Exception Explicity and catch it.

**class** Sample1{

/\*\*

\* **@author** Lakshman

\*

\*/

**static** **void** demo(){

**try**{

System.*out*.println("inside demo(): ");

**throw** **new** NullPointerException("Exception data ");

}

**catch** (NullPointerException ne) {

// **TODO**: handle exception

System.*out*.println(ne);

}

}

}

**public** **class** Throw\_Demo {

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

{

Sample1.*demo*();

}

}

//throws keyword

**import** java.io.\*;

**class** Sample{

/\*\*

\* **@author** Lakshman

\*

\*/

**private** String name;

**void** accept() **throws** IOException// put comment to this...throws IOException

{

BufferedReader br=**new** BufferedReader(**new** InputStreamReader(System.*in*));

System.*out*.println("Enter name: ");

name=br.readLine();

}

**void** disp(){

System.*out*.println("name is:"+name);

}

}

**public** **class** ThrowsDemo {

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

// put comment to this...throws IOException

{

Sample s=**new** Sample();

s.accept();

s.disp();

}

}

**currentthread.java:-**

**package** threads;

**public** **class** currentthread {

/\*\*

\* **@author** Lakshman

\*

\*/

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

// **TODO** Auto-generated method stub

System.*out*.println("let us find the current thread");

Thread t=Thread.*currentThread*();

System.*out*.println("current thread="+t);

System.*out*.println("its name="+t.getName());

}

}

**Runthread1.java:-**

**package** threads;

**class** runthread **extends** Thread {

/\*\*

\* **@author** Lakshman

\*

\*/

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

**for**(**int** i=1;i<=20;i++){

System.*out*.println(i);

}

}

**public** **static** **class** runthread1{

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

// **TODO** Auto-generated method stub

runthread obj=**new** runthread();

Thread t=**new** Thread(obj);

t.start();

}

}

}

**singlethread.java:-**

**package** threads;

**class** single **implements** Runnable{

/\*\*

\* **@author** Lakshman

\*

\*/

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

task1();

task2();

task3();

}

**void** task1(){

System.*out*.println("this is task1");

}

**void** task2(){

System.*out*.println("this is task2");

}

**void** task3(){

System.*out*.println("this is task3");

}}

**class** singlethread{

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

// **TODO** Auto-generated method stub

single obj=**new** single();

Thread t1=**new** Thread(obj);

t1.start();

}

}

**multithread:-**

**package** threads;

**class** multithread1 **implements** Runnable {

/\*\*

\* **@author** Lakshman

\*

\*/

String str;

**public** multithread1(String str) {

// **TODO** Auto-generated constructor stub

**this**.str=str;

}

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

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

System.*out*.println(str+ " : " +i);

**try**{

Thread.*sleep*(2000);

}

**catch** (InterruptedException ie) {

ie.printStackTrace();

// **TODO**: handle exception

}

}

}

}

**class** multithread{

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

// **TODO** Auto-generated method stub

multithread1 obj1=**new** multithread1("cut the ticket");

multithread1 obj2=**new** multithread1("show the seat");

Thread t1=**new** Thread(obj1);

Thread t2=**new** Thread(obj2);

t1.start();

t2.start();

}

}

**multiplethread.java:-**

**package** threads;

**class** singleobj **implements** Runnable {

/\*\*

\* **@author** Lakshman

\*

\*/

**int** available=1;

**int** wanted;

singleobj(**int** i){

wanted=i;

}

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

System.*out*.println("available berths=" + available);

**if**(available>=wanted){

String name=Thread.*currentThread*().getName();

System.*out*.println(wanted+":berth reserved for " + name);

**try**{

Thread.*sleep*(1500);

available=available-wanted;

}**catch** (InterruptedException ie) {

// **TODO**: handle exception

}

}

**else** System.*out*.println("sorry no berths");

}

}

**class** multiplethread{

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

// **TODO** Auto-generated method stub

singleobj obj=**new** singleobj(1);

Thread t1=**new** Thread(obj);

Thread t2=**new** Thread(obj);

t1.setName("first person");

t2.setName("second person");

t1.start();

t2.start();

}

}

**terthread1.java:-**

**package** threads;

**import** java.io.IOException;

**class** terthread1 **extends** Thread {

/\*\*

\* **@author** Lakshman

\*

\*/

**boolean** stop=**false**;

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

**for**(**int** i=1;i<=100000;i++){

System.*out*.println(i);

**if**(stop)

**return**;

}

} }

**class** terthread{

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

// **TODO** Auto-generated method stub

terthread1 obj=**new** terthread1();

Thread t=**new** Thread(obj);

t.start();

System.*in*.read();//wait till enter key press

obj.stop=**true**;

}

}

**synchron.java:-**

**package** threads;

**class** reserve **implements** Runnable {

/\*\*

\* **@author** Lakshman

\*

\*/

**int** available=1;

**int** wanted;

reserve(**int** i){

wanted=i;

}

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

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

System.*out*.println("available berths=" + available);

**if**(available>=wanted){

String name=Thread.*currentThread*().getName();

System.*out*.println(wanted+":berth reserved for " + name);

**try**{

Thread.*sleep*(1500);

available=available-wanted;

}**catch** (InterruptedException ie) {

// **TODO**: handle exception

}

}

**else** System.*out*.println("sorry no berths");

}

}

}

**class** synchron{

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

// **TODO** Auto-generated method stub

reserve obj=**new** reserve(1);

Thread t1=**new** Thread(obj);

Thread t2=**new** Thread(obj);

t1.setName("first person");

t2.setName("second person");

t1.start();

t2.start();

}

}

**deadlock.java:-**

**package** threads;

**import** threads.bookticket.cancelticket;

/\*\*

\* **@author** Lakshman

\*

\*/

**class** bookticket **extends** Thread

{

Object train,comp;

bookticket(Object train,Object comp)

{

**this**.train=train;

**this**.comp=comp;

}

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

{

**synchronized**(train)

{

System.*out*.println("bookticket locked on train");

**try**

{

Thread.*sleep*(150);

}

**catch**(InterruptedException e)

{

}

System.*out*.println("bookticket now waiting to lock on compartment..");

}

**synchronized**(comp)

{

System.*out*.println("bookticket locked on compartment");

}

}

**public** **static** **class** cancelticket **extends** Thread

{

Object train,comp;

cancelticket(Object train,Object comp)

{

**this**.train=train;

**this**.comp=comp;

}

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

{

**synchronized**(comp)

{

System.*out*.println("cancelticket locked on compartment");

**try**

{

Thread.*sleep*(200);

}

**catch**(InterruptedException e)

{

e.printStackTrace();

}

System.*out*.println("cancelticket now waiting to lock on compartment..");

**synchronized**(train)

{

System.*out*.println("cancelticket locked on train");

}

}

}

}}

**class** deadlock

{

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

{

Object train=**new** Object();

Object compartment=**new** Object();

bookticket obj1=**new** bookticket(train,compartment);

cancelticket obj2=**new** cancelticket(train,compartment);

Thread t1=**new** Thread(obj1);

Thread t2=**new** Thread(obj2);

t1.start();

t2.start();

}

}

**Prior.java:-**

**package** threads;

**class** myclass **extends** Thread {

/\*\*

\* **@author** Lakshman

\*

\*/

**int** count=0;

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

{

**for**(**int** i=0;i<=100;i++)

count++;

System.*out*.println("completd thread:"+Thread.*currentThread*().getName());

System.*out*.println("its priority:"+Thread.*currentThread*().getPriority());

}

}

**class** prior

{

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

{

myclass obj=**new** myclass();

Thread t1=**new** Thread(obj,"one");

Thread t2=**new** Thread(obj,"two");

t1.setPriority(2);

t2.setPriority(Thread.*NORM\_PRIORITY*);

t1.start();

t2.start();

}

}

**tgroups.java:-**

**package** threads;

**class** tgroups

{

/\*\*

\* **@author** Lakshman

\*

\*/

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

{

reservation res=**new** reservation();

cancellation can=**new** cancellation();

ThreadGroup tg=**new** ThreadGroup("first group");

Thread t1=**new** Thread(tg,res,"first thread");

Thread t2=**new** Thread(tg,res,"second thread");

ThreadGroup tg1=**new** ThreadGroup("second group");

Thread t3=**new** Thread(tg1,can,"third thread");

Thread t4=**new** Thread(tg1,can,"four thread");

System.*out*.println("parent of t1="+tg1.getParent());

tg1.setMaxPriority(7);

System.*out*.println("thread group of t1="+t1.getThreadGroup());

System.*out*.println("thread group of t3="+t3.getThreadGroup());

t1.start();

t2.start();

t3.start();

t4.start();

System.*out*.println("no of threads active in tg="+tg.activeCount());

}

}

**class** reservation **extends** Thread

{

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

{

System.*out*.println("i am reservation thread");

}

}

**class** cancellation **extends** Thread

{

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

{

System.*out*.println("i am cancellation thread");

}

}