Problem 1:simple example of multithread

package too1;

class Ex extends Thread {

public void run() {

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

try {

Thread.sleep(500);

} catch (InterruptedException e) {

System.out.println("error");

}

System.out.println(i);

}

}

}

public class Too1 {

public static void main(String[] args) {

Ex t1 = new Ex();

Ex t2 = new Ex();

t1.start();

t2.start();

}

}

Problem 2: multithread example using thread class

package too2;

class Exe extends Thread{

@Override

public void run(){

System.out.println("Run method executed by user child thread");

}

}

public class Too2 {

public static void main(String[] args) {

Exe t1=new Exe();

t1.start();

System.out.println("MAin method executed by main thread");

}

}

Problem 3:example of thread implementing runnable class

package too3;

class fu implements Runnable {

@Override

public void run() {

System.out.println("Run method executed by user child thread");

}

}

public class Too3 {

public static void main(String[] args) {

fu t = new fu();

Thread t1 = new Thread(t);

t1.start();

System.out.println("Main method is execute by MAin thread");

}

}

Problem 4: example of using start method in multithreading

package too4;

class Test extends Thread {

@Override

public void run() {

System.out.println("Run method executed by user child thread");

}

}

public class Too4 {

public static void main(String[] args) {

Test t = new Test();

t.run();

System.out.println("Main method by main Thread");

t.start();

}

}

Problem 5:example of using join method in multi threading (with time)

package too5;

class Test extends Thread {

@Override

public void run() {

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

try {

Thread.sleep(500);

} catch (InterruptedException ex) {

System.out.println("error");

}

System.out.println(i);

}

}

}

public class Too5 {

public static void main(String[] args) {

Test t1 = new Test();

Test t2 = new Test();

Test t3 = new Test();

t1.start();

try {

t1.join(1500);

} catch (InterruptedException ex) {

System.out.println("error again");

}

t2.start();

t3.start();

}

}

Problem 6: example of using join method in multithreading (without time)

package too5;

class Test extends Thread {

@Override

public void run() {

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

try {

Thread.sleep(500);

} catch (InterruptedException ex) {

System.out.println("error");

}

System.out.println(i);

}

}

}

public class Too5 {

public static void main(String[] args) {

Test t1 = new Test();

Test t2 = new Test();

Test t3 = new Test();

t1.start();

try {

t1.join();

} catch (InterruptedException ex) {

System.out.println("error again");

}

t2.start();

t3.start();

}

}