

Thread

Nama : Muhamad Wahyu Saputra

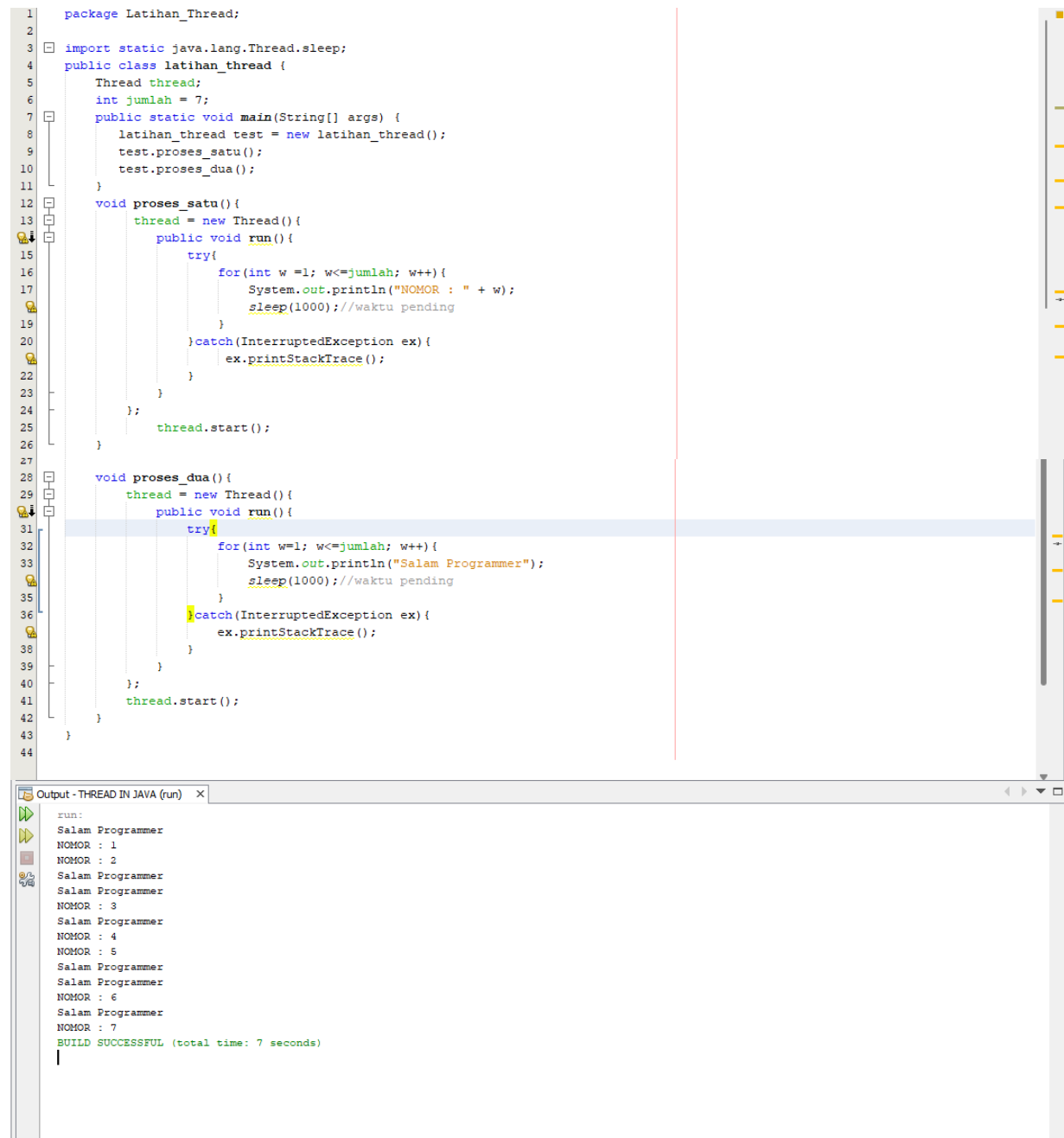
Kelas : XII RPL B

Absen : 34

A.Thread

merupakan kemampuan yang disediakan oleh Java untuk membuat aplikasi yang tangguh, karena thread dalam program memiliki fungsi dan tugas tersendiri.

LATIHAN 1

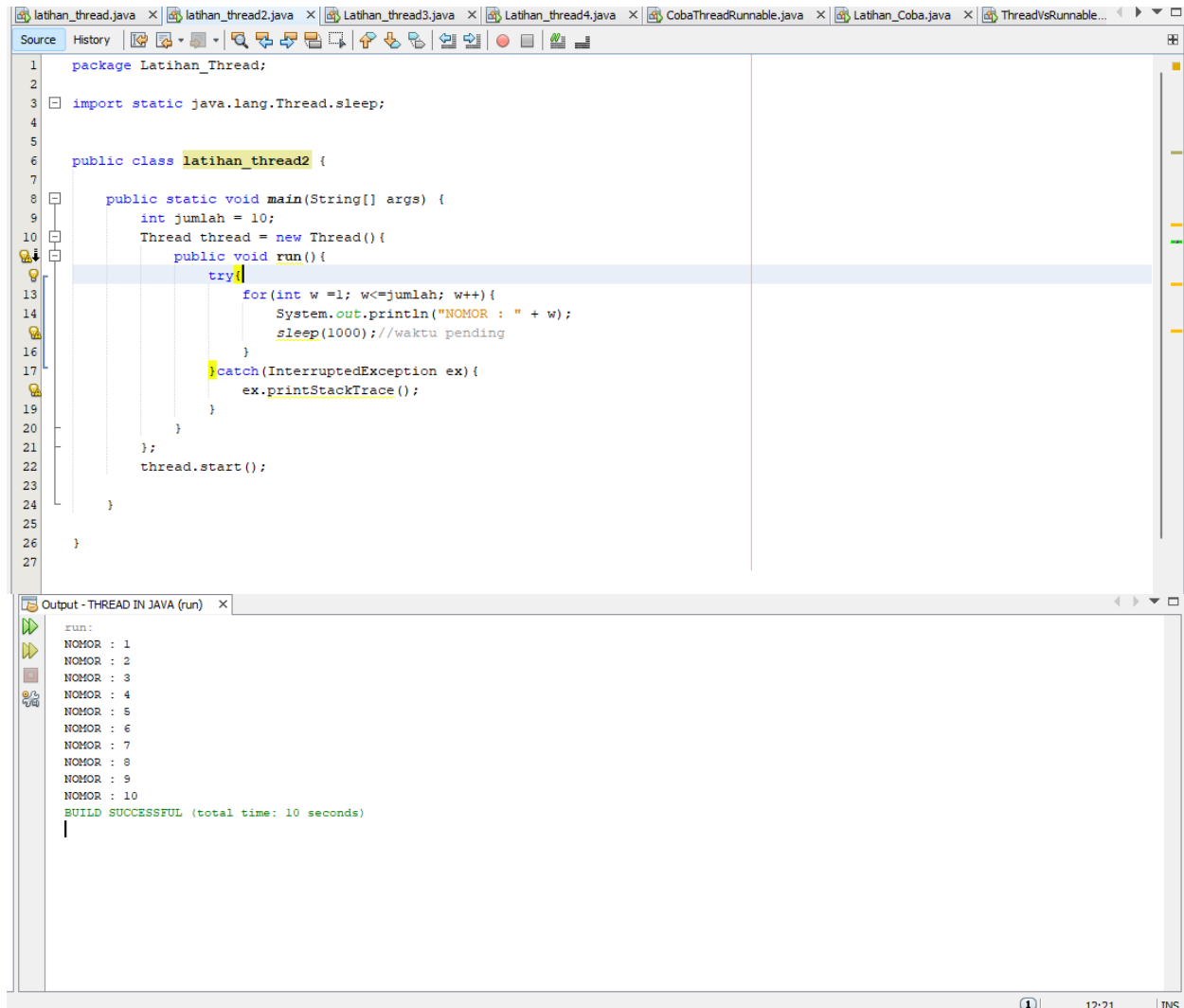


```
1 package Latihan_Thread;
2
3 import static java.lang.Thread.sleep;
4 public class latihan_thread {
5     Thread thread;
6     int jumlah = 7;
7     public static void main(String[] args) {
8         latihan_thread test = new latihan_thread();
9         test.proses_satu();
10        test.proses_dua();
11    }
12    void proses_satu(){
13        thread = new Thread(){
14            public void run(){
15                try{
16                    for(int w=1; w<=jumlah; w++){
17                        System.out.println("NOMOR : " + w);
18                        sleep(1000); //waktu pending
19                    }
20                } catch (InterruptedException ex){
21                    ex.printStackTrace();
22                }
23            }
24        };
25        thread.start();
26    }
27
28    void proses_dua(){
29        thread = new Thread(){
30            public void run(){
31                try{
32                    for(int w=1; w<=jumlah; w++){
33                        System.out.println("Salam Programmer");
34                        sleep(1000); //waktu pending
35                    }
36                } catch (InterruptedException ex){
37                    ex.printStackTrace();
38                }
39            }
40        };
41        thread.start();
42    }
43 }
44
```

Output - THREAD IN JAVA (run)

```
run:
Salam Programmer
NOMOR : 1
NOMOR : 2
Salam Programmer
Salam Programmer
NOMOR : 3
Salam Programmer
NOMOR : 4
NOMOR : 5
Salam Programmer
Salam Programmer
NOMOR : 6
Salam Programmer
NOMOR : 7
BUILD SUCCESSFUL (total time: 7 seconds)
```

LATIHAN 2



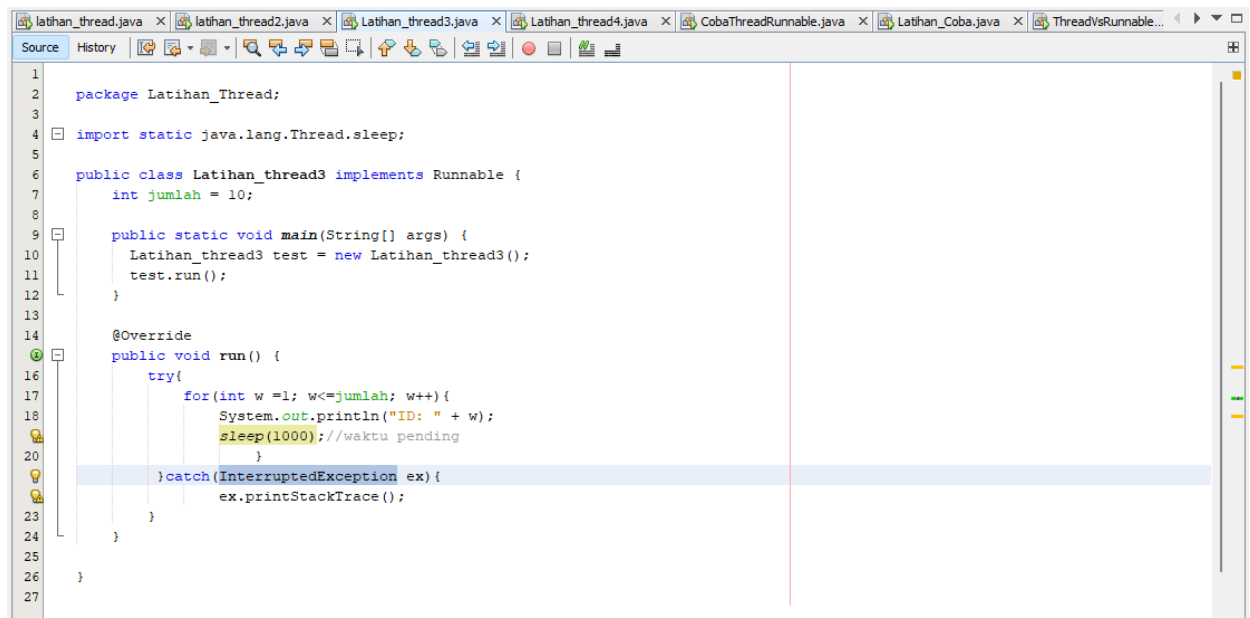
The screenshot shows an IDE with multiple tabs. The active tab is `latihan_thread2.java`. The code defines a package `Latihan_Thread`, imports `java.lang.Thread.sleep`, and defines a class `latihan_thread2`. The `main` method creates a `Thread` object and starts it. The `run` method contains a loop that prints numbers 1 to 10 with a 1-second delay between each print. The output window shows the execution results, confirming the numbers 1 through 10 are printed sequentially over a 10-second period.

```
1 package Latihan_Thread;
2
3 import static java.lang.Thread.sleep;
4
5
6 public class latihan_thread2 {
7
8     public static void main(String[] args) {
9         int jumlah = 10;
10        Thread thread = new Thread(){
11            public void run() {
12                try{
13                    for(int w =1; w<=jumlah; w++){
14                        System.out.println("NOMOR : " + w);
15                        sleep(1000);//waktu pending
16                    }
17                }catch (InterruptedException ex){
18                    ex.printStackTrace();
19                }
20            }
21        };
22        thread.start();
23    }
24 }
25
26
27
```

Output - THREAD IN JAVA (run)

```
run:
NOMOR : 1
NOMOR : 2
NOMOR : 3
NOMOR : 4
NOMOR : 5
NOMOR : 6
NOMOR : 7
NOMOR : 8
NOMOR : 9
NOMOR : 10
BUILD SUCCESSFUL (total time: 10 seconds)
```

LATIHAN 3



The screenshot shows an IDE with multiple tabs. The active tab is `Latihan_thread3.java`. The code defines a package `Latihan_Thread`, imports `java.lang.Thread.sleep`, and defines a class `Latihan_thread3` that implements the `Runnable` interface. The `main` method creates an instance of `Latihan_thread3` and calls its `run` method. The `run` method contains a loop that prints IDs 1 to 10 with a 1-second delay between each print.

```
1 package Latihan_Thread;
2
3 import static java.lang.Thread.sleep;
4
5 public class Latihan_thread3 implements Runnable {
6     int jumlah = 10;
7
8     public static void main(String[] args) {
9         Latihan_thread3 test = new Latihan_thread3();
10        test.run();
11    }
12
13    @Override
14    public void run() {
15        try{
16            for(int w =1; w<=jumlah; w++){
17                System.out.println("ID: " + w);
18                sleep(1000);//waktu pending
19            }
20        }catch (InterruptedException ex){
21            ex.printStackTrace();
22        }
23    }
24 }
25
26
27
```

```
Output - THREAD IN JAVA (run) x
run:
ID: 1
ID: 2
ID: 3
ID: 4
ID: 5
ID: 6
ID: 7
ID: 8
ID: 9
ID: 10
BUILD SUCCESSFUL (total time: 10 seconds)
```

LATIHAN 4

```
Source History
1 package Latihan_Thread;
2 public class Latihan_thread4 {
3     public static void main(String[] args) {
4         Baju baju1 = new Baju("Baju 1");
5         Baju baju2 = new Baju("Baju 2");
6         baju1.start();
7         baju2.start();
8     }
9 }
10 class Baju extends Thread{
11     //konstruktor
12     public Baju (String id){
13         super(id);
14     }
15     //mendefinisikan sendiri run()
16     @Override
17     public void run(){
18         String nama = getName();
19         for (int i =0; i<5; i++){
20             try{
21                 sleep(1000); //tunggu satu detik
22             }catch (InterruptedException ie){
23                 System.out.println("Terinterupsi");
24             }
25             System.out.println("Thread " + nama + ":Posisi" + i);
26         }
27     }
28 }
29

Output - THREAD IN JAVA (run) x
run:
Thread Baju 1:Posisi1
Thread Baju 2:Posisi1
Thread Baju 2:Posisi1
Thread Baju 1:Posisi1
Thread Baju 1:Posisi1
Thread Baju 2:Posisi1
Thread Baju 2:Posisi1
Thread Baju 1:Posisi1
Thread Baju 1:Posisi1
Thread Baju 2:Posisi1
BUILD SUCCESSFUL (total time: 5 seconds)
```

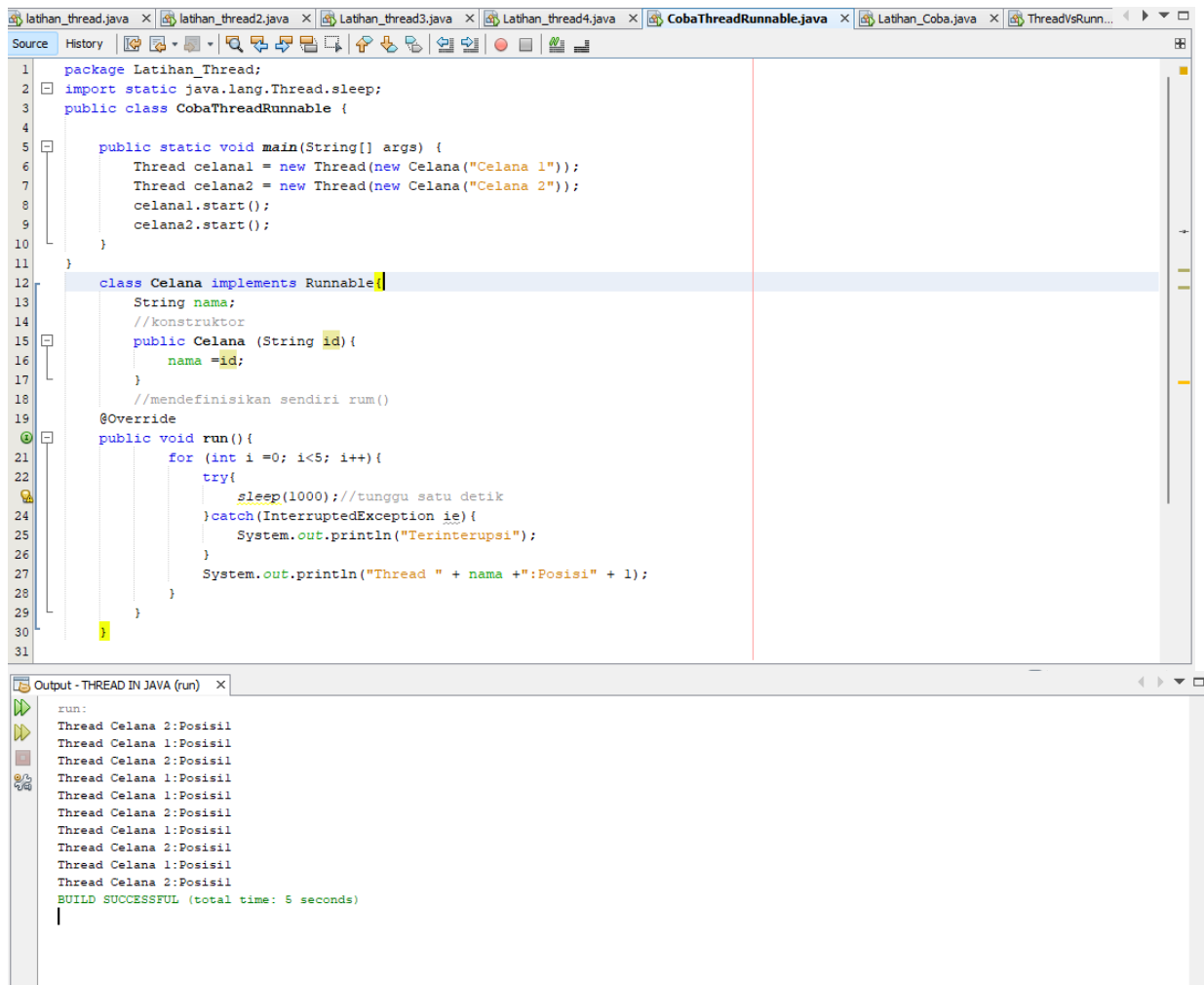
LATIHAN 5

```
1 package Latihan_Thread;
2
3 public class Latihan_Coba {
4
5
6     public static void main(String[] args) {
7         int i ;
8         int a ;
9
10        Thread kopasus = new Thread();
11        kopasus.start();
12        while(true){
13
14            for(i=1; i<=20; i++){
15                System.out.println("Pasukan kopasus Berhasil menyelamatkan ibu hamil ke : " + i);
16
17                try{
18                    kopasus.sleep(1000);
19                }catch(Exception e){
20                    e.printStackTrace();
21                }
22                if(i==20){
23                    System.out.println("Pasukan marinir datang ke TKP ");
24                }
25            }
26            Thread marinir = new Thread();
27            marinir.start();
28
29            for(a=1; a<=20; a++){
30                System.out.println("Pasukan marinir berhasil membunuh teroris ke : " + a);
31                if(a<=15){
32                    System.out.println("Pasukan marinir berhasil menyelamatkan warga manusia ke : " + a);
33                }
34                try{
35                    marinir.sleep(1000);
36                }catch(Exception e){
37                    e.printStackTrace();
38                }
39                if(a==50){
40                    System.out.println("Pasukan infanteri datang ke TKP ");
41                }
42            }
43            break;
44        }
45    }
46 }
47
48
49
50
```

Output - THREAD IN JAVA (run) x

```
run:
Pasukan kopasus Berhasil menyelamatkan ibu hamil ke : 1
Pasukan kopasus Berhasil menyelamatkan ibu hamil ke : 2
Pasukan kopasus Berhasil menyelamatkan ibu hamil ke : 3
Pasukan marinir berhasil membunuh teroris ke : 1
Pasukan marinir berhasil menyelamatkan warga manusia ke : 1
Pasukan marinir berhasil membunuh teroris ke : 2
Pasukan marinir berhasil menyelamatkan warga manusia ke : 2
Pasukan marinir berhasil membunuh teroris ke : 3
Pasukan marinir berhasil menyelamatkan warga manusia ke : 3
Pasukan marinir berhasil membunuh teroris ke : 4
Pasukan marinir berhasil menyelamatkan warga manusia ke : 4
Pasukan marinir berhasil membunuh teroris ke : 5
Pasukan marinir berhasil menyelamatkan warga manusia ke : 5
Pasukan infanteri datang ke TKP
BUILD SUCCESSFUL (total time: 8 seconds)
```

LATIHAN 6



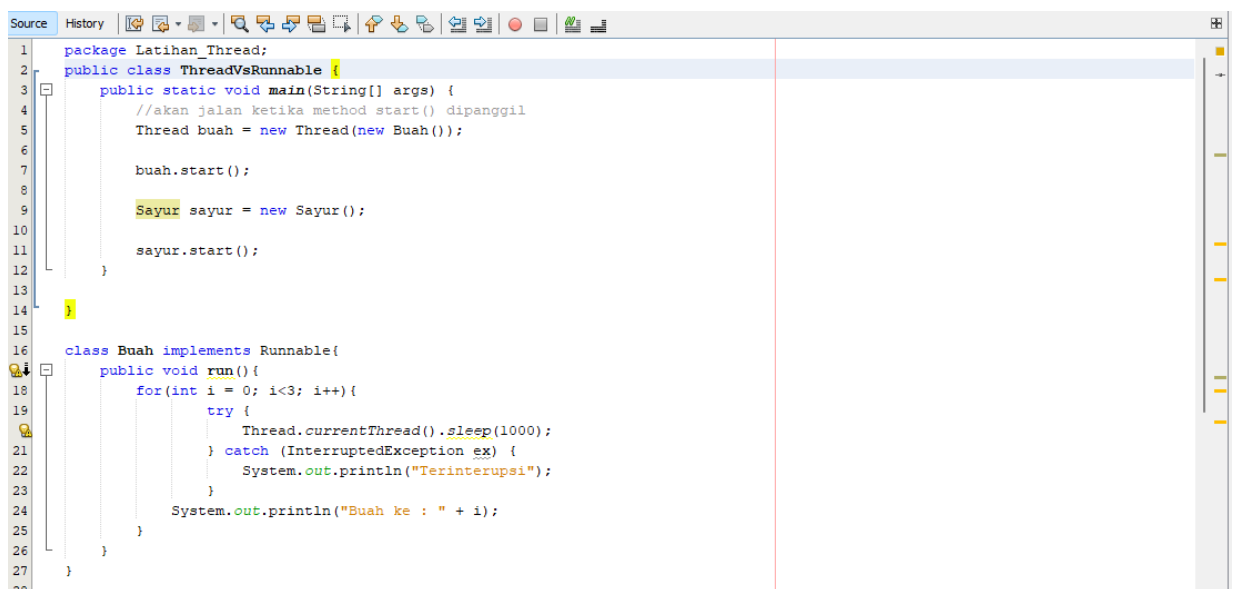
The screenshot shows an IDE with the following code in `CobaThreadRunnable.java`:

```
1 package Latihan_Thread;
2 import static java.lang.Thread.sleep;
3 public class CobaThreadRunnable {
4
5     public static void main(String[] args) {
6         Thread celana1 = new Thread(new Celana("Celana 1"));
7         Thread celana2 = new Thread(new Celana("Celana 2"));
8         celana1.start();
9         celana2.start();
10    }
11 }
12 class Celana implements Runnable{
13     String nama;
14     //konstruktor
15     public Celana (String id){
16         nama =id;
17     }
18     //mendefinisikan sendiri run()
19     @Override
20     public void run(){
21         for (int i =0; i<5; i++){
22             try{
23                 sleep(1000); //tunggu satu detik
24             } catch (InterruptedException ie){
25                 System.out.println("Terinterupsi");
26             }
27             System.out.println("Thread " + nama +" :Posisi" + i);
28         }
29     }
30 }
31 }
```

The output window shows the following text:

```
run:
Thread Celana 2:Posisi1
Thread Celana 1:Posisi1
Thread Celana 2:Posisi1
Thread Celana 1:Posisi1
Thread Celana 1:Posisi1
Thread Celana 2:Posisi1
Thread Celana 1:Posisi1
Thread Celana 2:Posisi1
Thread Celana 1:Posisi1
Thread Celana 2:Posisi1
Thread Celana 1:Posisi1
BUILD SUCCESSFUL (total time: 5 seconds)
```

LATIHAN 7



The screenshot shows an IDE with the following code in `ThreadVsRunnable.java`:

```
1 package Latihan_Thread;
2 public class ThreadVsRunnable {
3     public static void main(String[] args) {
4         //akan jalan ketika method start() dipanggil
5         Thread buah = new Thread(new Buah());
6
7         buah.start();
8
9         Sayur sayur = new Sayur();
10
11         sayur.start();
12    }
13 }
14
15
16 class Buah implements Runnable{
17     public void run(){
18         for(int i = 0; i<3; i++){
19             try {
20                 Thread.currentThread().sleep(1000);
21             } catch (InterruptedException ex) {
22                 System.out.println("Terinterupsi");
23             }
24             System.out.println("Buah ke : " + i);
25         }
26     }
27 }
28 }
```

```
28
29 class Sayur extends Thread{
30     public void run(){
31         for(int i = 0; i<3; i++){
32             try {
33                 Thread.currentThread().sleep(1000);
34             } catch (InterruptedException ex) {
35                 System.out.println("Terinterupsi");
36             }
37             System.out.println("Sayur ke : " + i);
38         }
39     }
40 }
41
```

Output - THREAD IN JAVA (run) x

```
run:
Sayur ke : 0
Buah ke : 0
Buah ke : 1
Sayur ke : 1
Buah ke : 2
Sayur ke : 2
BUILD SUCCESSFUL (total time: 3 seconds)
```

=====

=====()=====()

=====

-----()
