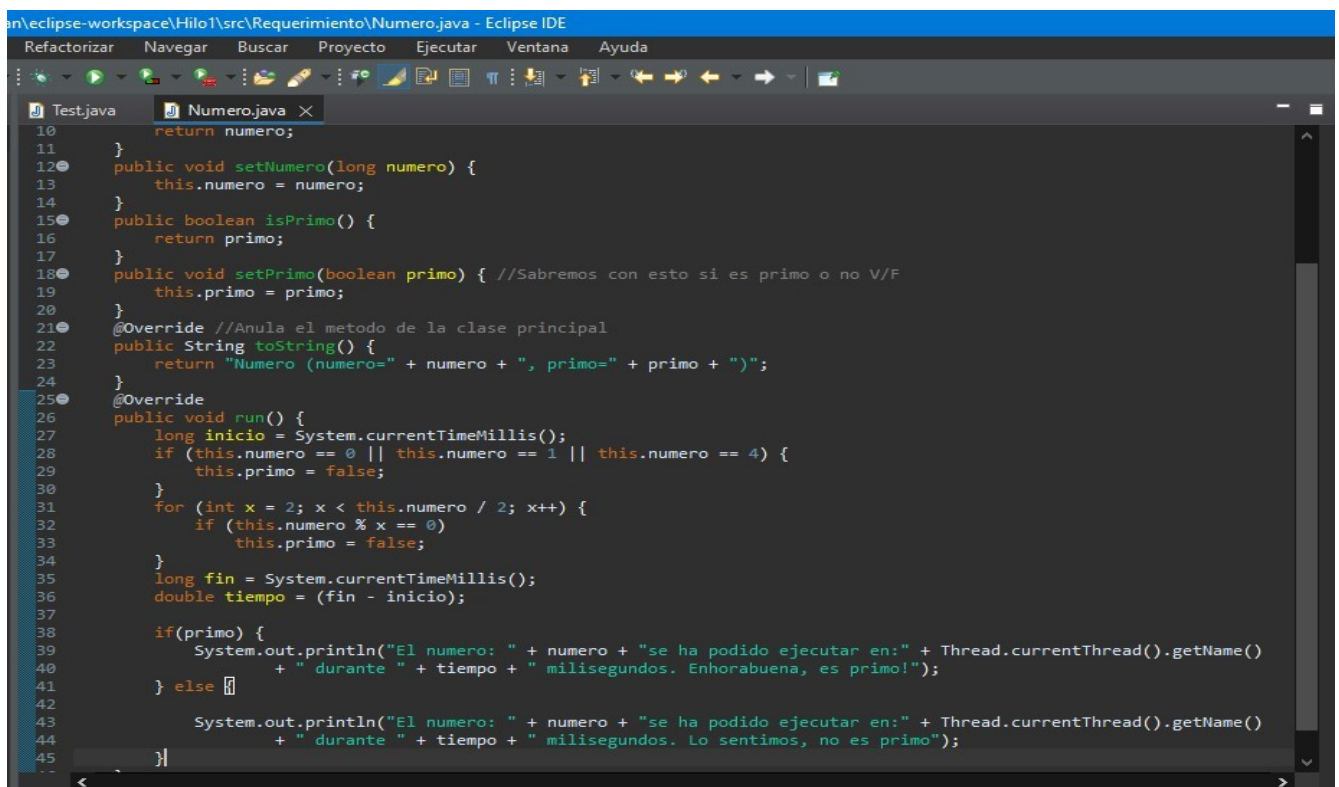


Requerimiento 1

Para este ejercicio he creado dos clases java, una con los números primos (main) y otra con el test para saber si son primos.

La clase main contendrá los números que los iremos creando con una clase principal, después crearemos el operador booleano que nos determinara si es primo o no cuando hagamos el test (verdadero/falso), para después introducir la clase secundaria override para anular el método de la clase principal (booleano), devolviéndonos la salida por consola el texto confirmando si es número primo o no mediante un condicional if si es primo o else si no lo es.



```
10     return numero;
11 }
12 public void setNumero(long numero) {
13     this.numero = numero;
14 }
15 public boolean isPrimo() {
16     return primo;
17 }
18 public void setPrimo(boolean primo) { //Sabremos con esto si es primo o no V/F
19     this.primo = primo;
20 }
21 @Override //Anula el metodo de la clase principal
22 public String toString() {
23     return "Numero (numero=" + numero + ", primo=" + primo + ")";
24 }
25 @Override
26 public void run() {
27     long inicio = System.currentTimeMillis();
28     if (this.numero == 0 || this.numero == 1 || this.numero == 4) {
29         this.primo = false;
30     }
31     for (int x = 2; x < this.numero / 2; x++) {
32         if (this.numero % x == 0)
33             this.primo = false;
34     }
35     long fin = System.currentTimeMillis();
36     double tiempo = (fin - inicio);
37
38     if(primo) {
39         System.out.println("El numero: " + numero + "se ha podido ejecutar en:" + Thread.currentThread().getName()
40             + " durante " + tiempo + " milisegundos. Enhorabuena, es primo!");
41     } else {
42
43         System.out.println("El numero: " + numero + "se ha podido ejecutar en:" + Thread.currentThread().getName()
44             + " durante " + tiempo + " milisegundos. Lo sentimos, no es primo");
45     }
46 }
```

La segunda clase, la que ejecutara el test contendrá dentro del método main la clase scanner, que es por donde introduciremos los datos por consola para saber si el numero que hemos introducido es primo o no, devolviéndonos por el método system.out la indicación de que debemos introducir hasta cuatro números. Después se ejecutara el método sc.close que cerrara el método scanner que iniciamos antes y empezara a ejecutar la operación de los cuatro números.

```
\eclipse-workspace\Hilo1\src\Requerimiento\Test.java - Eclipse IDE
Refactorizar  Navegar  Buscar  Proyecto  Ejecutar  Ventana  Ayuda

Test.java  Numero.java
1 package Requerimiento;
2 import java.util.*; //Hace referencia a importar todas las clases que dependan de ella
3 public class Test {
4     public static void main(String[] args) {
5
6         Scanner sc = new Scanner(System.in);
7         System.out.println("Introduce el primer número:");
8         Numero num1 = new Numero(sc.nextLong());
9
10        System.out.println("Introduce el segundo número:");
11        Numero num2 = new Numero(sc.nextLong());
12
13        System.out.println("Introduce el tercer número:");
14        Numero num3 = new Numero(sc.nextLong());
15
16        System.out.println("Introduce el cuarto número:");
17        Numero num4 = new Numero(sc.nextLong());
18
19        sc.close();
20
21        num1.start();
22        num2.start();
23        num3.start();
24        num4.start();
25    }
26 }
```

Requerimiento 2

Para el siguiente ejercicio crearemos varios archivos de clase, uno por cada proceso que vamos a ejecutar. El primero será la clase *main* donde crearemos varios objetos tales como la cola donde se almacenaran los emails para mandarse, el generador que hará las funciones de remitente de email y el cliente que recibirá los emails.

```
Desktop\Requerimiento 2\Rquerimiento_2\Main.java - Eclipse IDE
Refactorizar  Navegar  Buscar  Proyecto  Ejecutar  Ventana  Ayuda

Main.java  Generador.java  Cliente.java  *Email.java  Generador_Email.java  Cola.java
1 package Rquerimiento_2;
2 public class Main {
3
4     public static void main(String[] args) {
5
6         Cola cola = new Cola();
7
8         Generador p1 = new Generador("Generador 1", cola);
9         Generador p2 = new Generador("Generador 2", cola);
10        Generador p3 = new Generador("Generador 3", cola);
11
12        Cliente c1 = new Cliente("Cliente 1", cola);
13        Cliente c2 = new Cliente("Cliente 1", cola);
14
15        p1.start();
16        p2.start();
17        p3.start();
18
19        c1.start();
20        c2.start();
21    }
22 }
```

En la clase generador lo convertiremos en una super clase con el método *super ()* y el nombre con el constructor *this.nombre* y *this.colas*. Tras ello generaremos el método *override* para finalizar la superclase que hemos creado, después crearemos una clase de generador de email donde le estableceremos un limite de diez emails para mandar $i < 10$ y al final y mensaje que salga por consola que se ha generado el email.

```
\Desktop\Requerimiento 2\Requerimiento_2\Generador.java - Eclipse IDE
Refactorizar  Navegar  Buscar  Proyecto  Ejecutar  Ventana  Ayuda

Main.java  Generador.java  Cliente.java  *Email.java  Generador_Email.java  Cola.java

1 package Rquerimiento_2;
2 public class Generador extends Thread{
3
4     public String nombre;
5     public Cola cola;
6     public Generador(String nombre, Cola cola) {
7         super();
8         this.nombre = nombre;
9         this.colas = cola;
10    }
11    @Override
12    public void run() {
13        Generador_Email generador = new Generador_Email();
14        for(int i=1; i<=10;i++) {
15            Email email =generador.generarEmail();
16            try {
17                Thread.sleep(500);
18            } catch (InterruptedException e){
19                e.printStackTrace();
20            }
21            System.out.println(nombre + " ha generado el email con el siguiente nombre: "+ email.getId());
22            cola.addMensaje(email);
23        }
24    }
25 }
```

En la clase cliente haremos casi lo mismo creando el objeto *cliente* y en una cadena el nombre y la posición en la cola de emails, tras ellos crearemos una clase *run* que se ejecute cuando se reciba el email con el mensaje por consola de que se ha recibido el email.

```
\Desktop\Requerimiento 2\Requerimiento_2\Cliente.java - Eclipse IDE
Refactorizar  Navegar  Buscar  Proyecto  Ejecutar  Ventana  Ayuda

Main.java  Generador.java  Cliente.java  *Email.java  Generador_Email.java  Cola.java

1 package Rquerimiento_2;
2 public class Cliente extends Thread {
3
4     public String nombre;
5     public Cola cola;
6
7     public Cliente (String nombre, Cola cola) {
8         super();
9         this.nombre = nombre;
10        this.colas = cola;
11    }
12    public void run(){
13        while(true) {
14            Email mensaje = cola.getMensaje();
15            System.out.println(nombre + " ha recibido el siguiente email"+ mensaje);
16        }
17    }
18 }
```

En la clase *email* crearemos una cadena con los diferentes campos a rellenar del email, que luego iremos separando con una cadena publica que devuelva su correspondiente.

```
n\Desktop\Requerimiento 2\Rquerimiento_2\Email.java - Eclipse IDE
Refactorizar  Navegar  Buscar  Proyecto  Ejecutar  Ventana  Ayuda

Main.java  Generador.java  Cliente.java  *Email.java  Generador_Email.java  Cola.java

1  package Rquerimiento_2;
2
3  public class Email {
4
5      public String id;
6      public String destinatario;
7      public String remitente;
8      public String asunto;
9      public String cuerpo;
10     public String getId() {
11         return id;
12     }
13     public void setId(String string) {
14         this.id = string;
15     }
16
17     public String getRemitente() {
18         return remitente;
19     }
20     public void setRemitente(String remitente) {
21         this.remitente = remitente;
22     }
23     public String getDestinatario() {
24         return destinatario;
25     }
26     public void setDestinatario(String destinatario) {
27         this.destinatario = destinatario;
28     }
29     public String getAsunto() {
30         return asunto;
31     }
32     public void setAsunto(String asunto) {
33         this.asunto = asunto;
34     }
35     public String getCuerpo() {
36         return cuerpo;
37     }
38 }
```

En la clase *generador de email* ya crearemos una lista de nombres mediante una cadena *string* que van a enviar los emails, los destinatarios y el resto de campos a rellenar, sin embargo al acabar la lista de remitentes y destinatarios crearemos un dato de tipo entero *int* para que almacene en la memoria hasta los diez emails que queremos que se almacenen donde nos devuelva el numero de los que se han enviado.

```
n\Desktop\Requerimiento 2\Rquerimiento_2\Generador_Email.java - Eclipse IDE
Refactorizar  Navegar  Buscar  Proyecto  Ejecutar  Ventana  Ayuda

Main.java  Generador.java  Cliente.java  *Email.java  Generador_Email.java  Cola.java

1  package Rquerimiento_2;
2
3  import java.util.ArrayList;
4
5
6
7  public class Generador_Email {
8
9      public Email generarEmail() {
10         Email email = new Email();
11         email.setId(generarId());
12         email.setDestinatario(generarDestinatario());
13         email.setRemitente(generarRemitente());
14         email.setAsunto(generarAsunto());
15         email.setCuerpo(generarCuerpo());
16         return email;
17     }
18     public String generarId() {
19         List<String> listaNombres = new ArrayList<String>();
20         listaNombres.add("1");
21         listaNombres.add("2");
22         listaNombres.add("3");
23         listaNombres.add("4");
24         listaNombres.add("5");
25         listaNombres.add("6");
26         listaNombres.add("7");
27         listaNombres.add("8");
28         listaNombres.add("9");
29         listaNombres.add("10");
30
31         int numero = ThreadLocalRandom.current().nextInt(0,9);
32         return listaNombres.get(numero);
33     }
34
35     public String generarDestinatario() {
36         List<String> listaNombres = new ArrayList<String>();
37         listaNombres.add("Oack@gmail.com");
38         listaNombres.add("Jesse@gmail.com");
39     }
40 }
```


Para finalizar crearemos una clase *cola* con una clase publica con un máximo de elementos de cinco y una clase privada en la que aparezca el email que se ha mandado. Después introduciremos un condicional *if/else* que nos avisara con un mensaje de error cuando se hayan mandado los cinco emails como máximo. Después con el método *sincronizado* devolverá un valor nulo cuando tras ejecutarse el método *while* haya llegado al máximo de emails tras comprobar con el método *try/catch* que se ha llegado al máximo.

```

1 package Rquerimiento_2;
2
3 import java.util.LinkedList;
4
5
6 public class Cola {
7     public final static int MAX_ELEMENTOS = 5;
8     private Queue<Email> cola = new LinkedList<>();
9     public synchronized void addMensaje(Email email) {
10
11         if(email.getDestinatario()!="pikachu@gmail.com")) {
12             while(cola.size() == MAX_ELEMENTOS){
13                 try {
14                     wait();
15                 } catch (InterruptedException e) {
16                     e.printStackTrace();
17                 }
18             }
19             cola.offer(email);
20             notify();
21         } else {
22             System.out.println("El pokemon ha escapado, has perdido una masterball ");
23         }
24     }
25
26     public synchronized Email getMensaje() {
27         Email s=null;
28         while(cola.size()==0) {
29             try {
30                 wait();
31             } catch (InterruptedException e) {
32                 e.printStackTrace();
33             }
34         }
35         s=cola.poll();
36         notify();
37         return s;

```

Para finalizar ejecutaremos el programa para ver que esta todo correctamente.

```

1 package Rquerimiento_2;
2 public class Main {

```

Console Output:

```

Main [Aplicación Java] C:\Users\Adrian\AppData\Local\Programs\Eclipse Foundation\jdk-11.0.12-hotspot\bin\javaw.exe (17 oct. 2021 18:11:11)
Generador 2 ha generado el email con el siguiente nombre: 4
Cliente 1 ha recibido el siguiente email [id=5, destinatario=Ash@gmail.com, remitente=Ocelot@gmail.com, asunto=Contacto, cuerpo=Tramite su pokemon para llevar aqui]
Cliente 1 ha recibido el siguiente email [id=4, destinatario=Charmander@gmail.com, remitente=Samus@gmail.com, asunto=Final de temporada, cuerpo=Su pedido se entregar el proximo dia]
Cliente 1 ha recibido el siguiente email [id=2, destinatario=James@gmail.com, remitente=Lara@gmail.com, asunto=Bienvenida, cuerpo=Hay una reunion hoy en la sala 007]
Generador 1 ha generado el email con el siguiente nombre: 7
Generador 3 ha generado el email con el siguiente nombre: 8
Generador 2 ha generado el email con el siguiente nombre: 9
Cliente 1 ha recibido el siguiente email [id=8, destinatario=Charmander@gmail.com, remitente=Kratos@gmail.com, asunto=Registro, cuerpo=Aviso importante sobre los tiempos]
Cliente 1 ha recibido el siguiente email [id=9, destinatario=Meowth@gmail.com, remitente=Ocelot@gmail.com, asunto=Circular Informativa, cuerpo=Necesitamos comprobar un pedido]
Cliente 1 ha recibido el siguiente email [id=7, destinatario=Misty@gmail.com, remitente=Link@gmail.com, asunto=Promociones, cuerpo=Gracias por registrarse en Pokemon In]
Generador 2 ha generado el email con el siguiente nombre: 7
Generador 1 ha generado el email con el siguiente nombre: 1
Generador 3 ha generado el email con el siguiente nombre: 3
Cliente 1 ha recibido el siguiente email [id=1, destinatario=Meowth@gmail.com, remitente=Ocelot@gmail.com, asunto=Contacto, cuerpo=Hay una reunion hoy en la sala 007]
Cliente 1 ha recibido el siguiente email [id=3, destinatario=Articuno@gmail.com, remitente=Link@gmail.com, asunto=Circular Informativa, cuerpo=Su pedido se entregar el proximo dia]
Cliente 1 ha recibido el siguiente email [id=7, destinatario=Misty@gmail.com, remitente=Ico@gmail.com, asunto=Promociones, cuerpo=Su pedido se entregar el proximo dia]
Generador 3 ha generado el email con el siguiente nombre: 2
Generador 2 ha generado el email con el siguiente nombre: 7
Generador 1 ha generado el email con el siguiente nombre: 7
Cliente 1 ha recibido el siguiente email [id=7, destinatario=Articuno@gmail.com, remitente=Ico@gmail.com, asunto=Contacto, cuerpo=Promocion Pokemon Rojo Fuego]
Cliente 1 ha recibido el siguiente email [id=2, destinatario=Articuno@gmail.com, remitente=Ocelot@gmail.com, asunto=Promociones, cuerpo=Su pedido se entregar el proximo dia]
Cliente 1 ha recibido el siguiente email [id=7, destinatario=Oack@gmail.com, remitente=Arthas@gmail.com, asunto=Promociones, cuerpo=Su pedido se entregar el proximo dia]
Generador 1 ha generado el email con el siguiente nombre: 5
Generador 2 ha generado el email con el siguiente nombre: 2
Generador 3 ha generado el email con el siguiente nombre: 5
Cliente 1 ha recibido el siguiente email [id=5, destinatario=Jesse@gmail.com, remitente=Ico@gmail.com, asunto=Circular Informativa, cuerpo=Promocion Pokemon Rojo Fuego]
Cliente 1 ha recibido el siguiente email [id=5, destinatario=Meowth@gmail.com, remitente=Ico@gmail.com, asunto=Albaran, cuerpo=Tramite su pokemon para llevar aqui]
Cliente 1 ha recibido el siguiente email [id=2, destinatario=Articuno@gmail.com, remitente=Arthas@gmail.com, asunto=Circular Informativa, cuerpo=Aviso importante sobre el pedido]
Generador 2 ha generado el email con el siguiente nombre: 9
Generador 3 ha generado el email con el siguiente nombre: 9
Generador 1 ha generado el email con el siguiente nombre: 7
Cliente 1 ha recibido el siguiente email [id=9, destinatario=Misty@gmail.com, remitente=Samus@gmail.com, asunto=Contacto, cuerpo=Su pedido se entregar el proximo dia]
Cliente 1 ha recibido el siguiente email [id=9, destinatario=Misty@gmail.com, remitente=Link@gmail.com, asunto=Contacto, cuerpo=Su pedido se entregar el proximo dia]
Cliente 1 ha recibido el siguiente email [id=5, destinatario=James@gmail.com, remitente=Arthas@gmail.com, asunto=Pedido, cuerpo=Si tiene alguna duda, por favor contacta]
Generador 2 ha generado el email con el siguiente nombre: 7
Generador 3 ha generado el email con el siguiente nombre: 9
Generador 1 ha generado el email con el siguiente nombre: 7
Cliente 1 ha recibido el siguiente email [id=7, destinatario=Charmander@gmail.com, remitente=Arthas@gmail.com, asunto=Albaran, cuerpo=Su pedido se entregar el proximo dia]
Cliente 1 ha recibido el siguiente email [id=9, destinatario=Charmander@gmail.com, remitente=Lara@gmail.com, asunto=Albaran, cuerpo=Gracias por registrarse en Pokemon In]
Cliente 1 ha recibido el siguiente email [id=7, destinatario=Oack@gmail.com, remitente=Ico@gmail.com, asunto=Albaran, cuerpo=Hay una reunion hoy en la sala 007]

```