

ENTORNOS DE DESARROLLO

UD 10: Mi primer TDD

Francisco Salar Pérez 1º S

Reproduce los siguientes vídeos y sigue con atención la explicación.

<https://www.youtube.com/watch?v=zJa62-3qjfs> (4:46 minutos)

<https://www.youtube.com/watch?v=OQVVNeXPcOM> (6:07 minutos)

A continuación, replica fielmente el contenido de los vídeos. El entorno de desarrollo utilizado es IntelliJ Idea y el plugin Junit 5.

Inicializa el control de versiones y ve realizando commits mientras replicas el contenido de los vídeos según estimes oportuno.

Crea una nueva rama llamada "Refactorizado". Sube el proyecto a esta nueva rama refactorizando todos los métodos cambiándoles el nombre por "nombremetodo_tu_nombre", es decir, agregando tu nombre a cada método. Cuando tengas replicado el ejemplo, hayas creado la rama "Refactorizado", y acabado de refactorizar todo, sube las dos ramas a gitHub si no lo has ido haciendo con anterioridad.

Realiza una memoria de los pasos que vas realizando conforme avanzas en la práctica. Esta memoria debe incluir pantallazos apoyados con texto de los pasos que vas realizando, botones que pulsas, opciones de menús, etc. etc. etc.

Entrega de la práctica.

Copia un enlace al repositorio gitHub donde has subido la práctica.

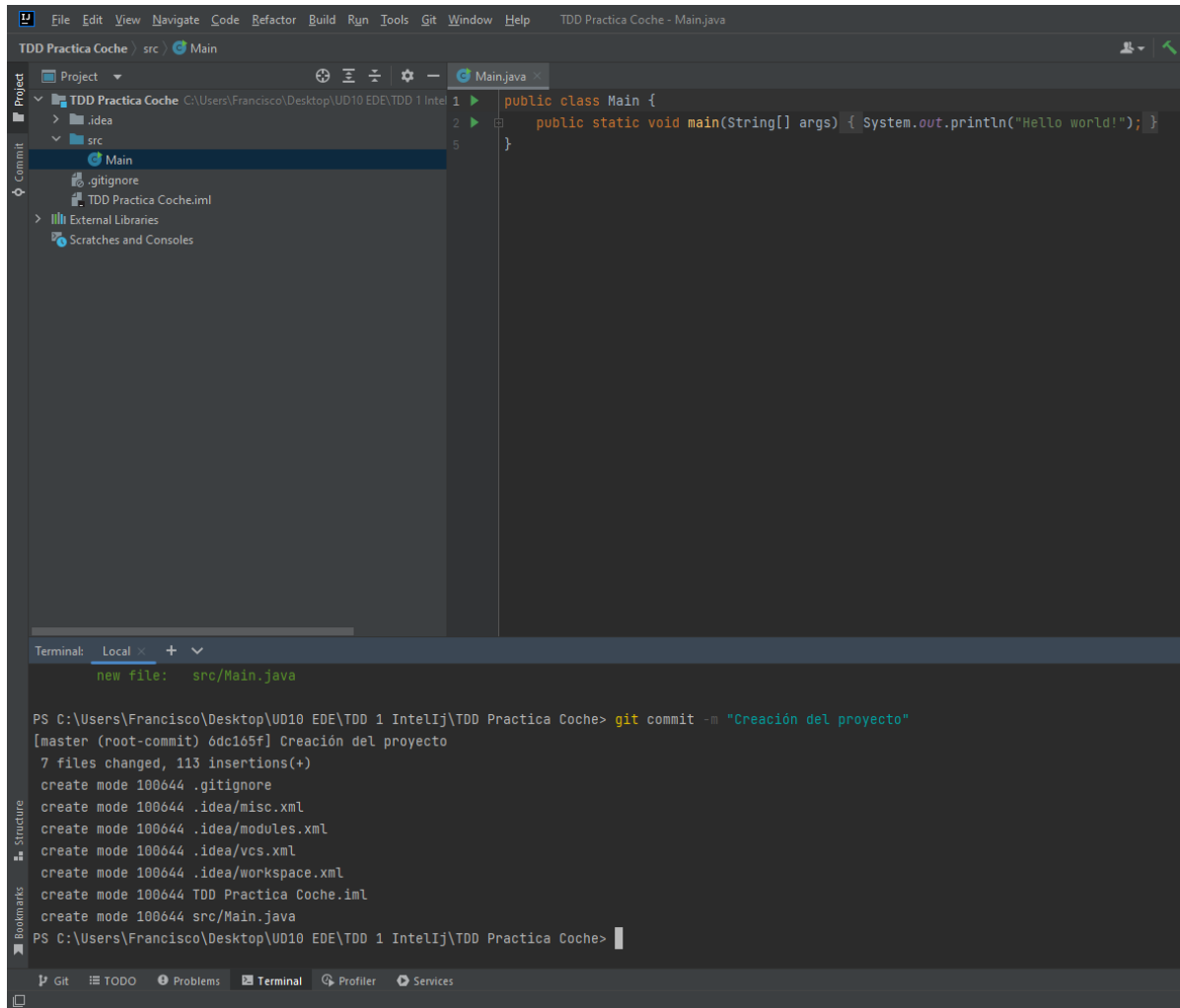
Asegúrate de que cada rama contiene lo que se pide.

Crea una rama que se llame "Memoria" y sube a esta rama la memoria realizada. *** FORMATO PDF O WORD ***

Tiempo estimado para realizar la práctica: 1h:45 minutos.

Solución:

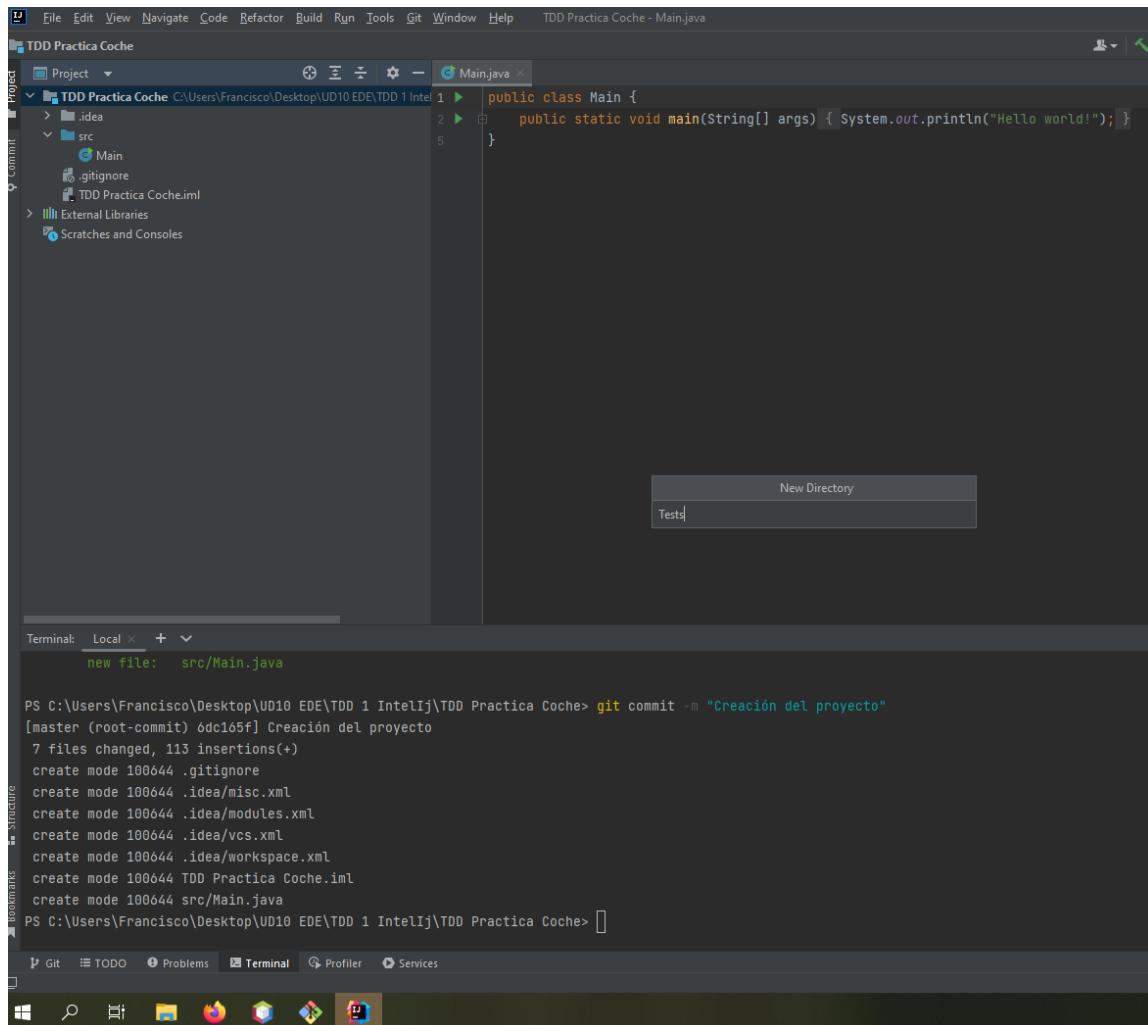
Primero creamos el proyecto de Java, clicando en la pestaña “New Project” seleccionando un nuevo proyecto en java nombrándolo: TDD Practica Coche. Obteniendo una vista en el IDE IntelliJ Idea como se ve en la siguiente captura:



En este punto realizamos nuestro primer “commit”, ya que acabamos de crear nuestro proyecto y no hemos empezado a trabajar en él, de esta forma, siempre podremos volver al estado inicial.

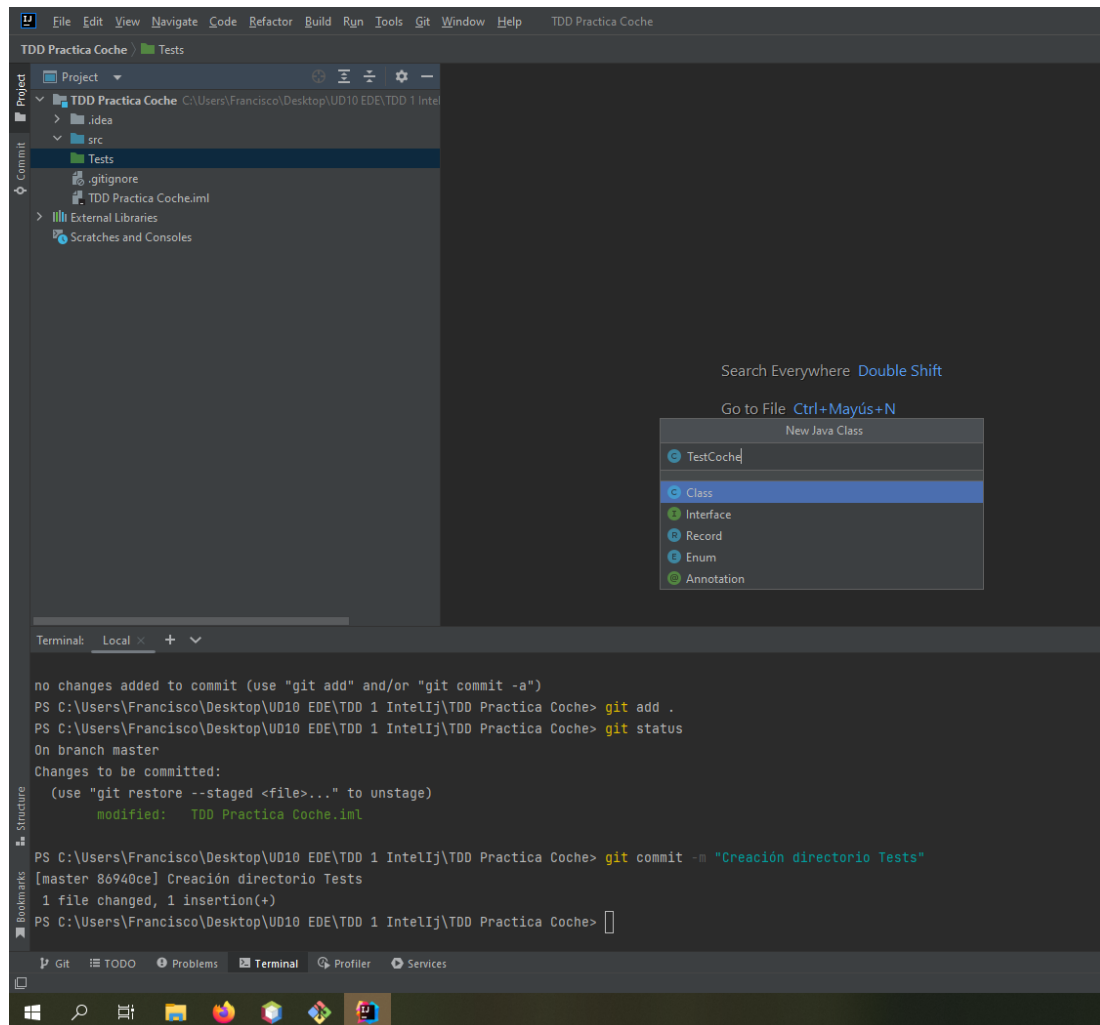
Mencionar que el IDE ya nos ha inicializado el repositorio Git Local en la carpeta que le hemos seleccionado para tal efecto, también va añadiendo a él los directorios y “Java Class” que vamos creando.

Posteriormente, hacemos click en el botón secundario del ratón en la carpeta de nuestro proyecto para acceder al menú desplegable y seleccionamos crear nuevo directorio y llamándolo Tests.

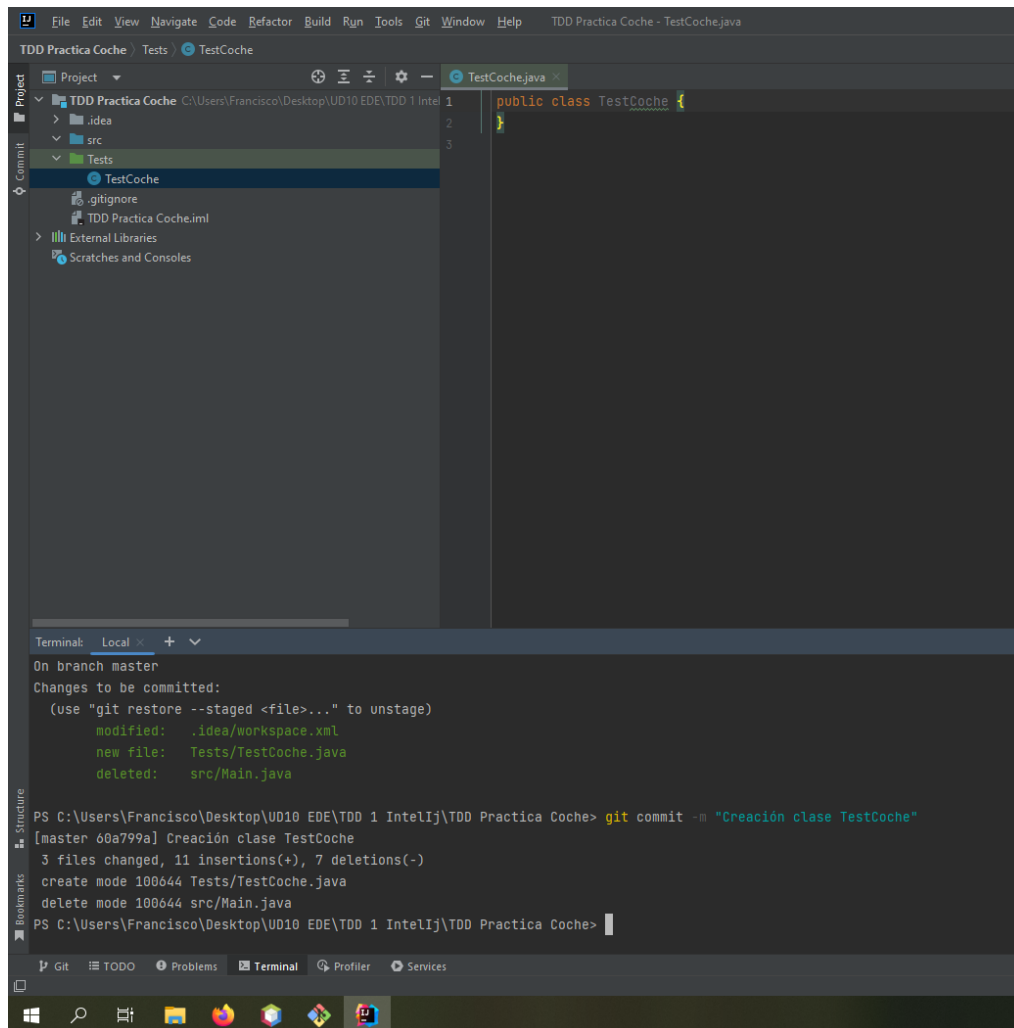


En este punto hemos realizado nuestro segundo “commit”, justo después de crear el directorio Tests y habiéndolo marcado como “Test”.

Una vez creada el directorio lo marcamos como un directorio de “Test Sources Root” y después haciendo click secundario sobre él, creamos una “Java Class” con el nombre de TestCoche.

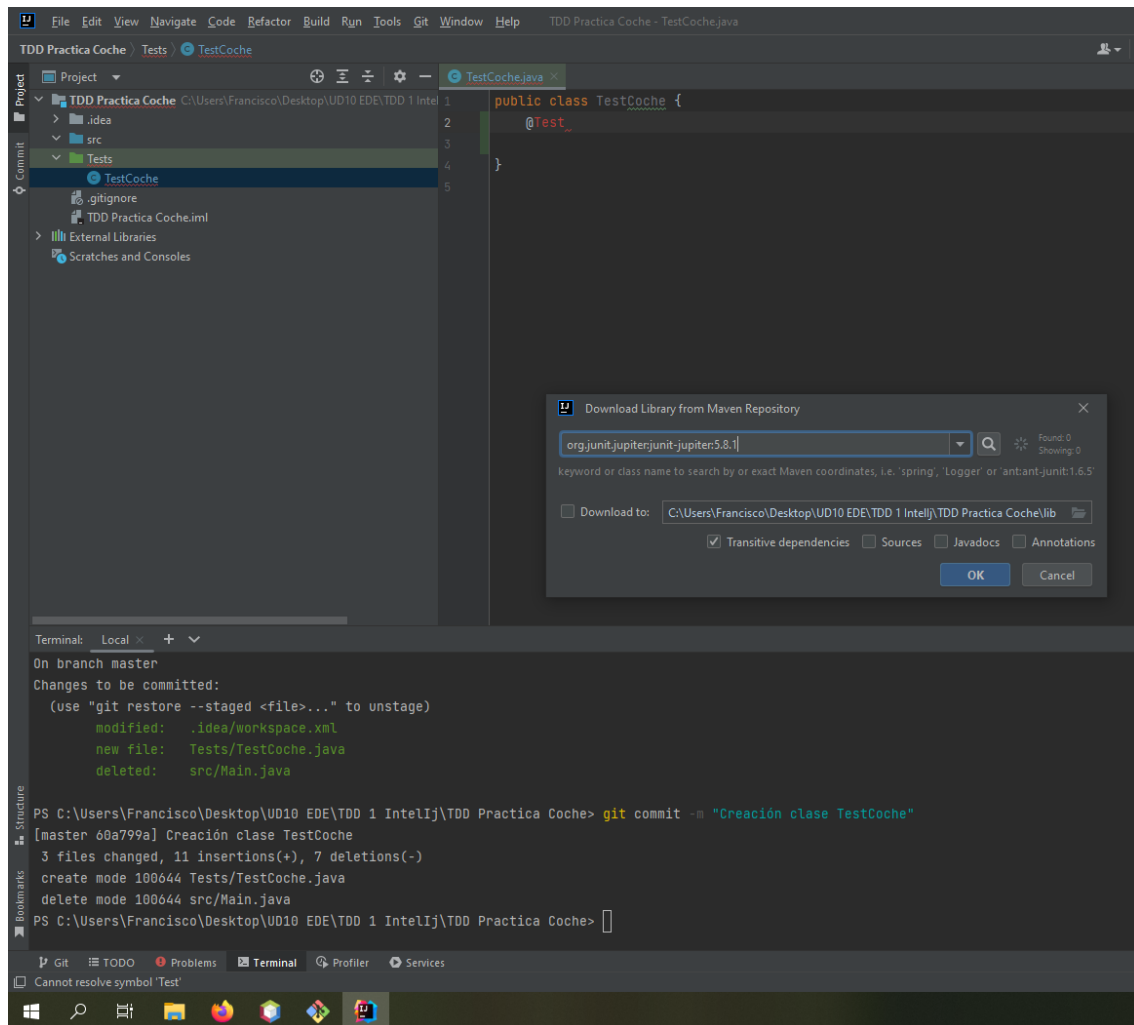


Una vez hecho lo anterior obtendríamos algo así:

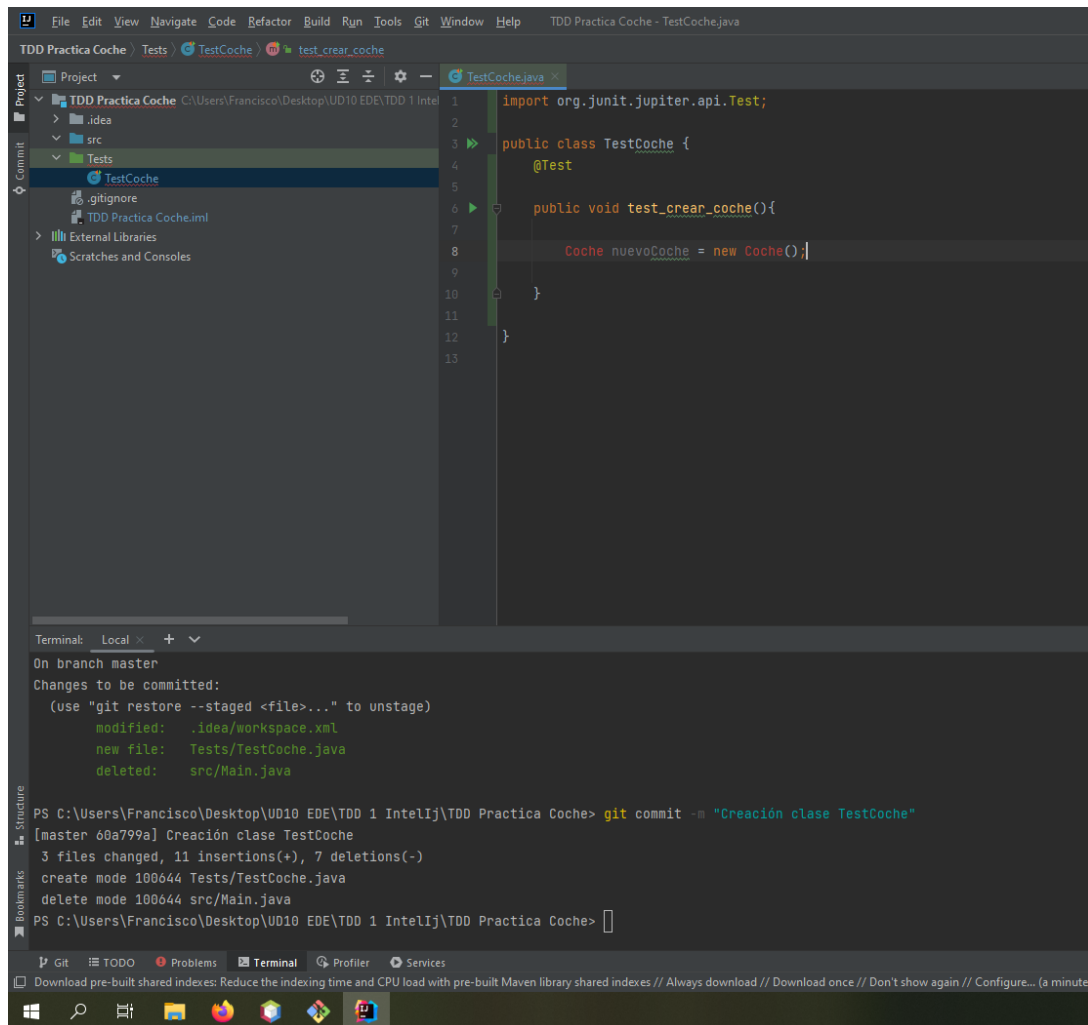


Hacemos entonces un “commit” con los cambios efectuados: creación de la clase TestCoche, como se puede ver en la imagen de arriba.

Al proceder con la realización del primer `@Test`, el IDE nos da un error y nos indica que tenemos que importar desde la “classpath” Junit 5 pulsando “Alt+Enter” y seleccionando la opción para poder empezar a trabajar. En la imagen de abajo podemos ver como se haría:

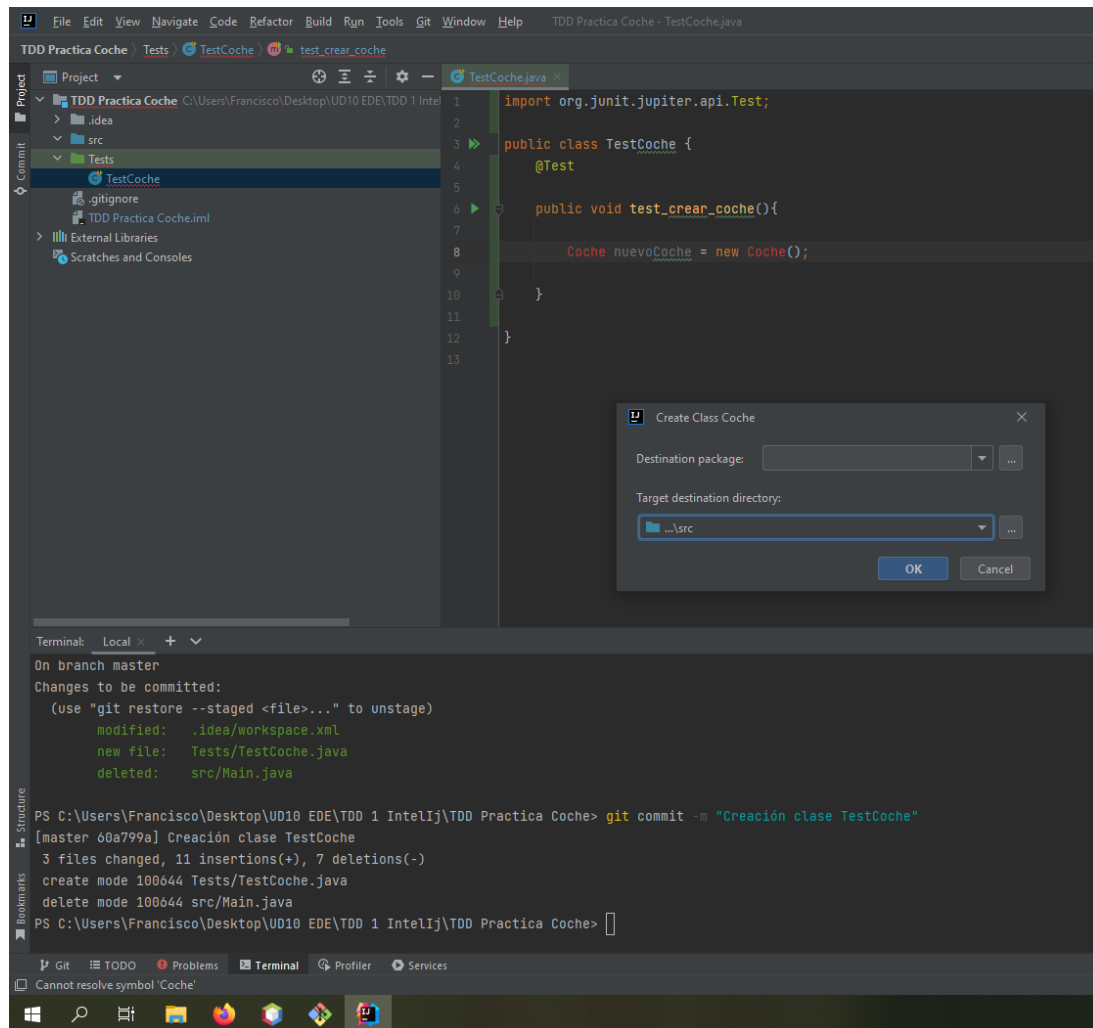


Comenzamos entonces con la creación de nuestro primer Test:

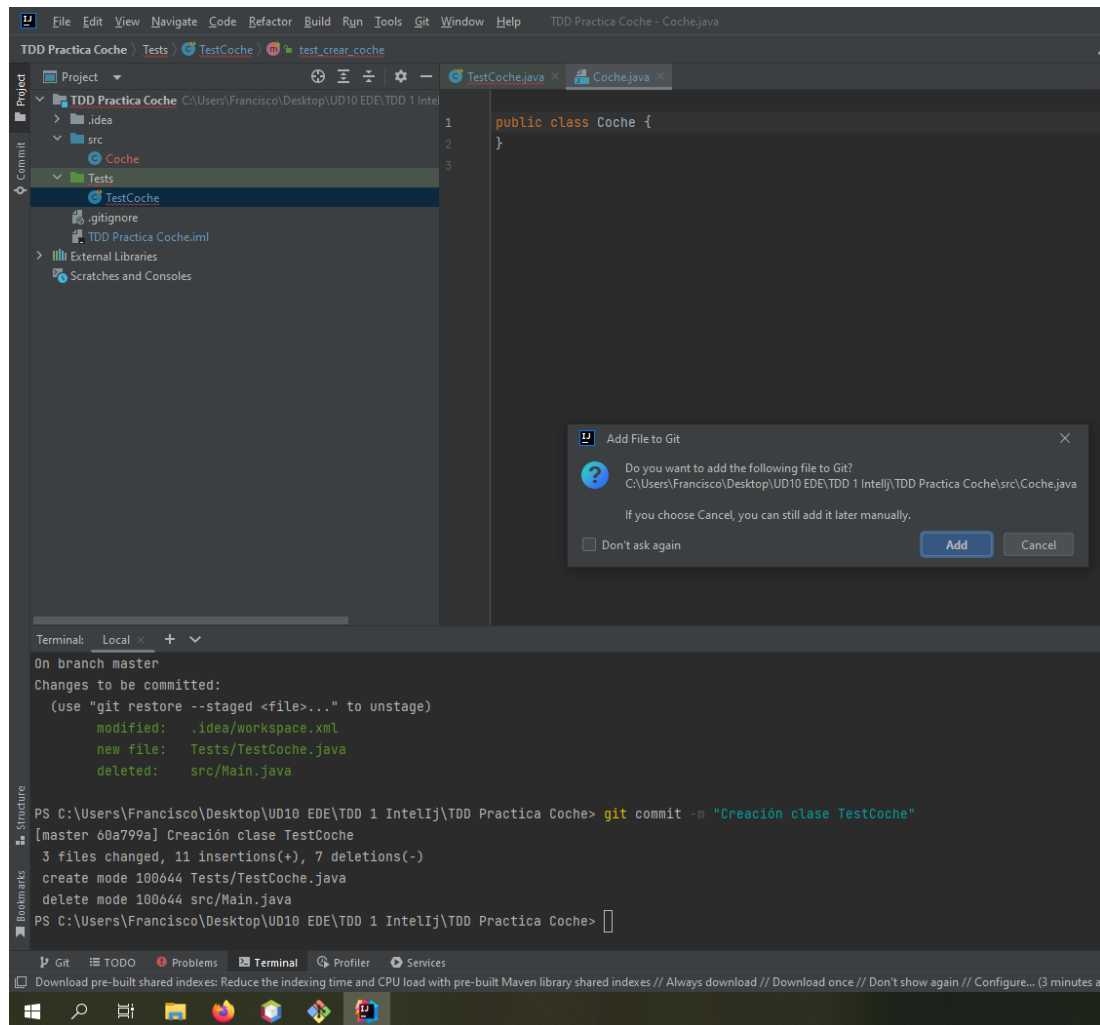


Podemos observar que obtenemos un error, y esto se debe a que debemos crear la “clase coche” para poder continuar, ya que la hemos definido sin haberla creado.

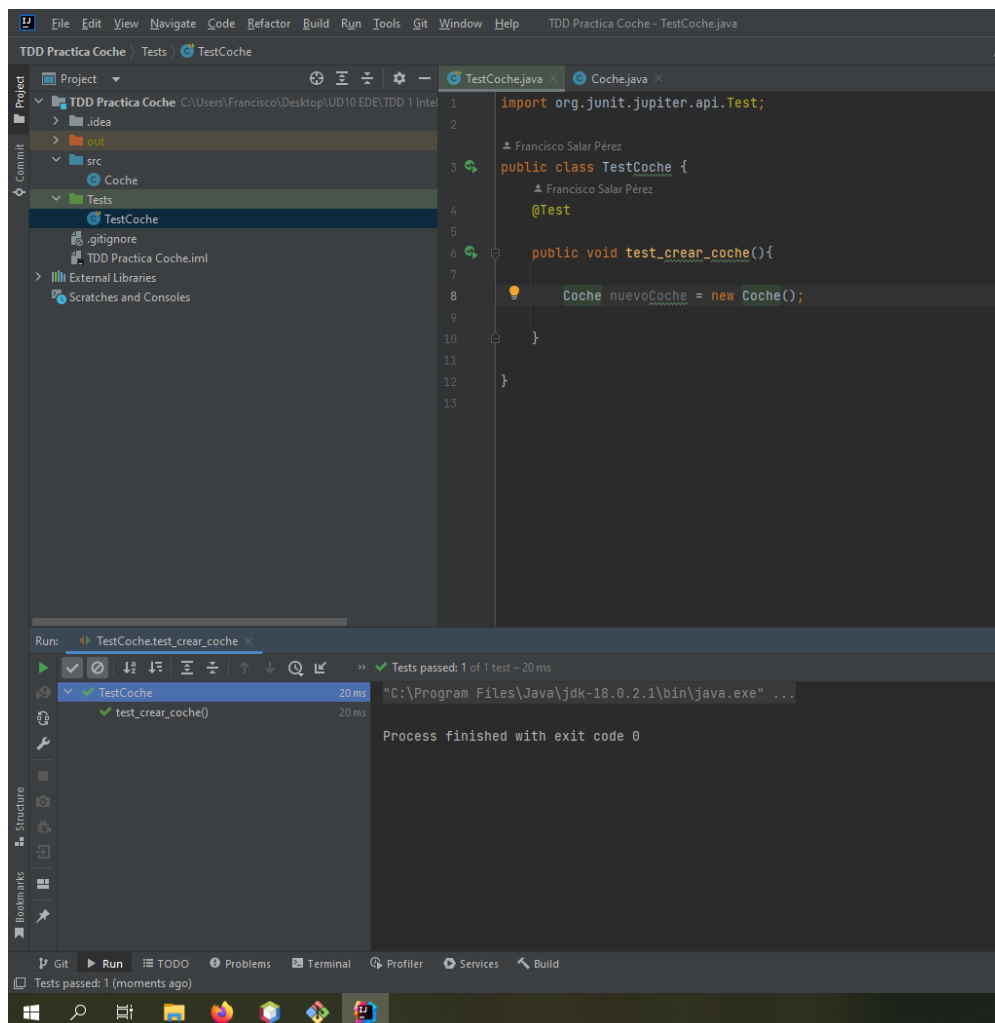
Para ello, el IDE nos permite crearlo automáticamente como se puede apreciar en la imagen siguiente:



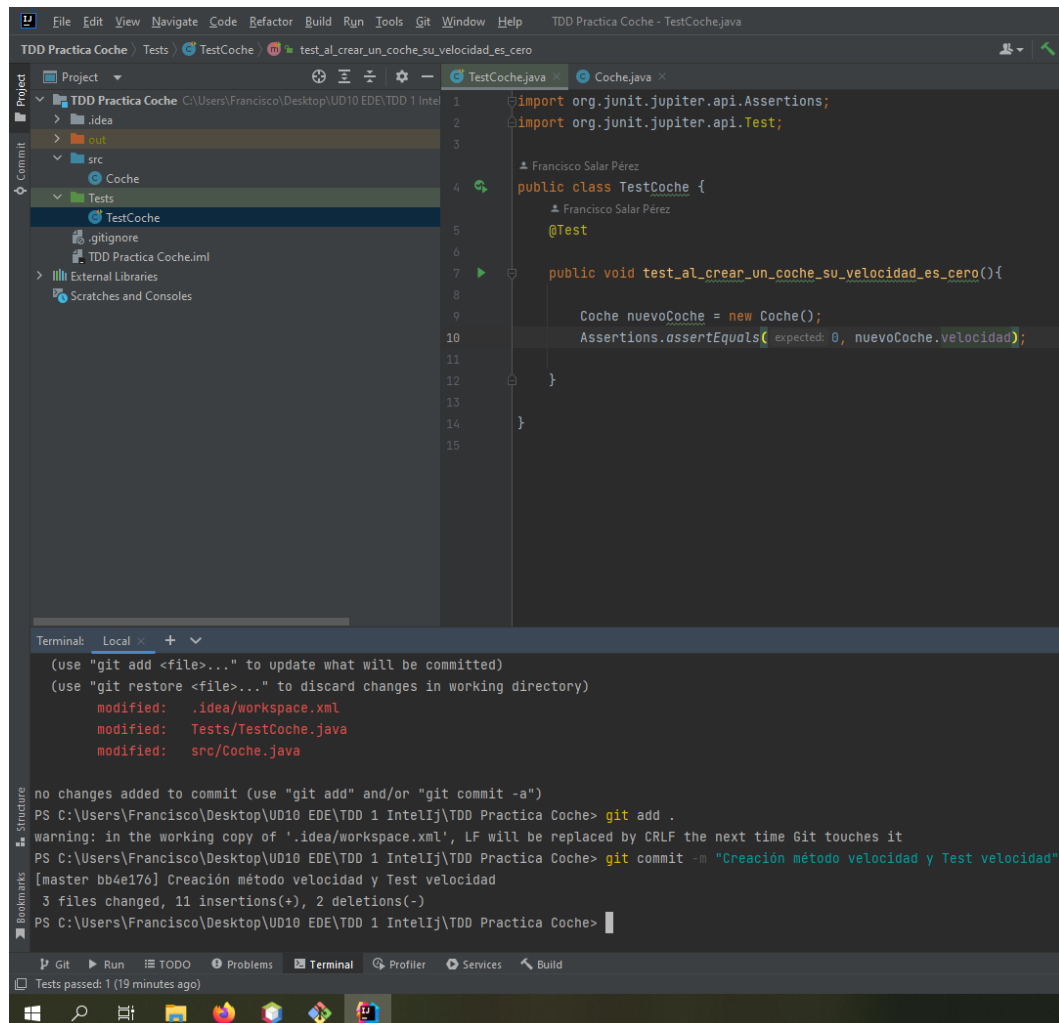
En la siguiente imagen podemos ver creada la clase Coche y como el IDE nos avisa de si queremos incorporarla al repositorio local de Git. Le damos a “Add” y ya lo tendríamos añadido. Decir qué si no lo hacemos en este punto, lo podemos hacer más adelante manualmente.



Una vez hecho lo anterior, podemos ejecutar el test creado, y vemos como se ha superado perfectamente sin arrojar ningún error. También es cierto que este primer test es muy simple.



Para hacerlo más completo, procedemos a cambiar el primer test añadiendo una “Assertions” para poder crear un coche, en este caso, y que su velocidad sea cero una vez se cree.



The screenshot shows an IDE with a project named "TDD Practica Coche". The left sidebar shows the project structure with folders for "src" and "Tests". The "Tests" folder is expanded, showing "TestCoche". The main editor displays the "TestCoche.java" file with the following code:

```
1 import org.junit.jupiter.api.Assertions;
2 import org.junit.jupiter.api.Test;
3
4 public class TestCoche {
5     @Test
6
7     public void test_al_crear_un_coche_su_velocidad_es_cero(){
8
9         Coche nuevoCoche = new Coche();
10        Assertions.assertEquals( expected: 0, nuevoCoche.velocidad);
11    }
12 }
13
14 }
15
```

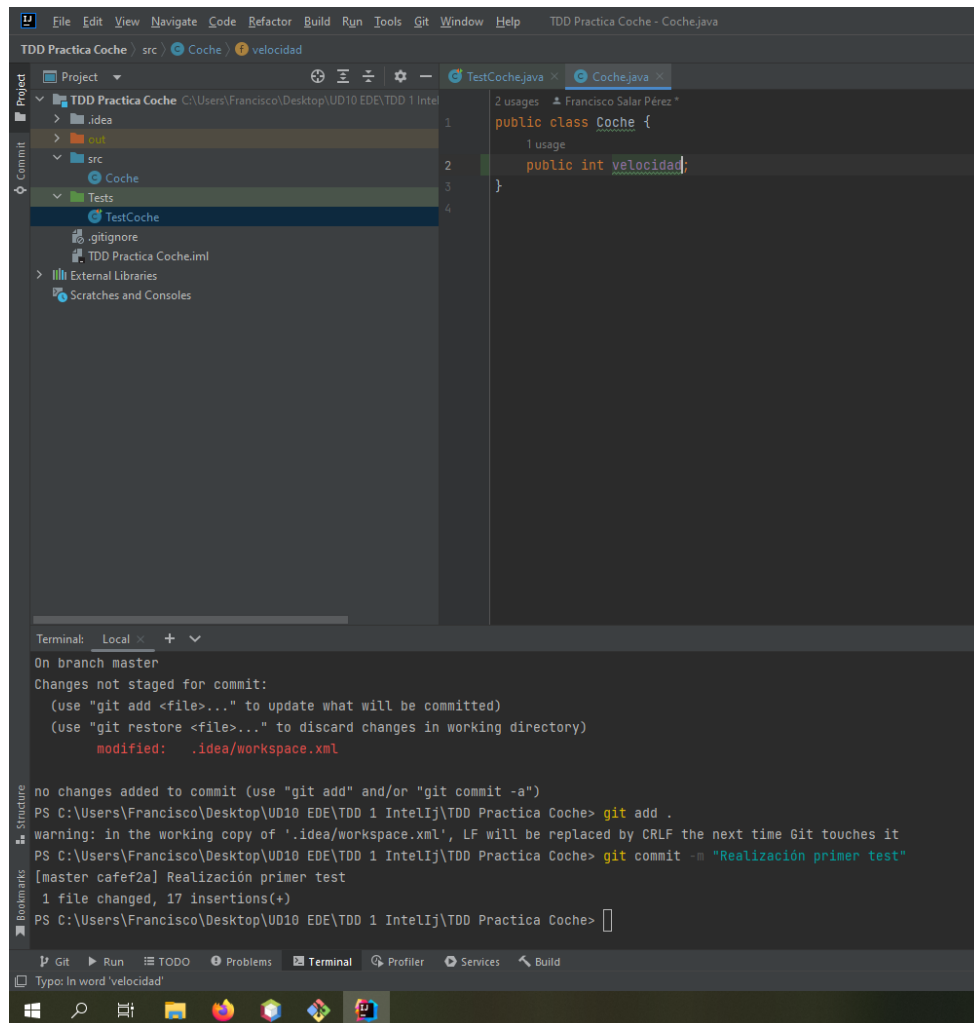
The bottom panel shows a terminal window with the following output:

```
(use "git add <file>..." to update what will be committed)
(use "git restore <file>..." to discard changes in working directory)
modified:   .idea/workspace.xml
modified:   Tests/TestCoche.java
modified:   src/Coche.java

no changes added to commit (use "git add" and/or "git commit -a")
PS C:\Users\Francisco\Desktop\UD10 EDE\TDD 1 IntelJ\TDD Practica Coche> git add .
warning: in the working copy of '.idea/workspace.xml', LF will be replaced by CRLF the next time Git touches it
PS C:\Users\Francisco\Desktop\UD10 EDE\TDD 1 IntelJ\TDD Practica Coche> git commit -m "Creación método velocidad y Test velocidad"
[master bb4e176] Creación método velocidad y Test velocidad
3 files changed, 11 insertions(+), 2 deletions(-)
PS C:\Users\Francisco\Desktop\UD10 EDE\TDD 1 IntelJ\TDD Practica Coche>
```

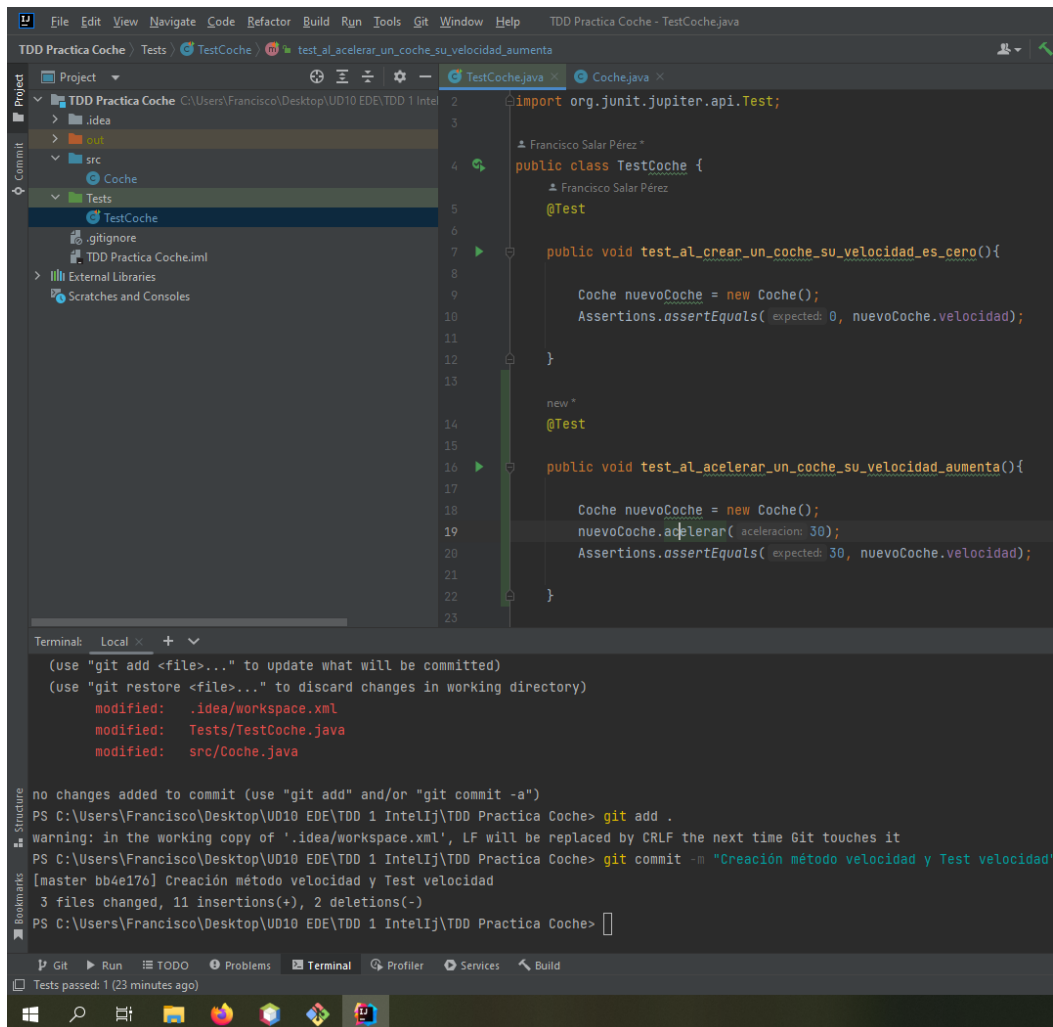
The bottom status bar shows "Tests passed: 1 (19 minutes ago)".

Para ello, utilizamos el método `velocidad` que, al no estar creado, nos da un error que subsanaremos creándolo en la clase `Coche` automáticamente como vemos a continuación.



Una vez realizado, hacemos otro “commit” con los cambios hechos: modificación test y creación método `velocidad`.

Ahora procedemos a crear nuestro segundo test, el de acelerar aumenta la velocidad. Para ello, hacemos un copia y pega del test anterior y lo modificamos añadiendo un nuevo método acelerar, pasándole por parámetro 30.



The screenshot shows an IDE window titled "TDD Practica Coche - TestCoche.java". The editor displays the following Java code:

```
import org.junit.jupiter.api.Test;

public class TestCoche {

    @Test
    public void test_al_crear_un_coche_su_velocidad_es_cero(){
        Coche nuevoCoche = new Coche();
        Assertions.assertEquals( expected: 0, nuevoCoche.velocidad);
    }

    new *
    @Test
    public void test_al_acelerar_un_coche_su_velocidad_aumenta(){
        Coche nuevoCoche = new Coche();
        nuevoCoche.acelerar( aceleracion: 30);
        Assertions.assertEquals( expected: 30, nuevoCoche.velocidad);
    }
}
```

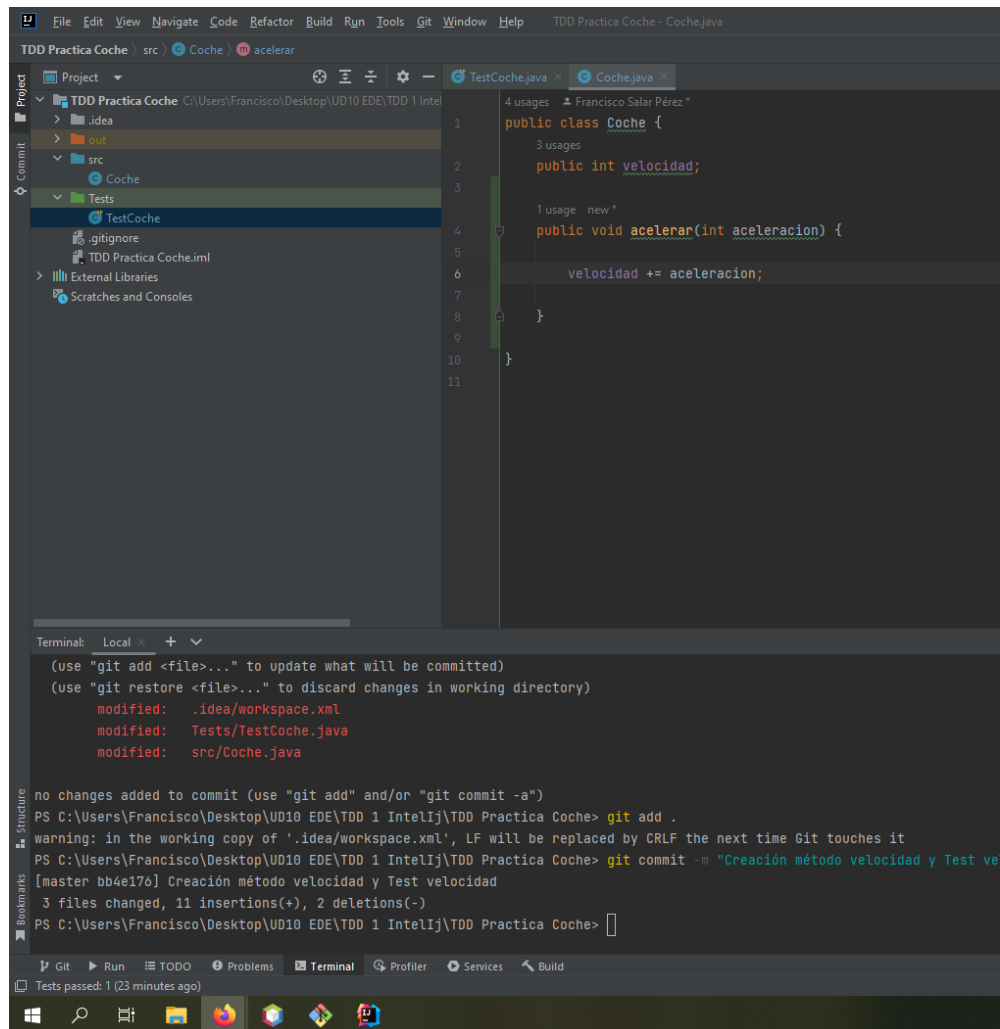
The left sidebar shows the project structure with "Tests" expanded, containing "TestCoche". The bottom terminal window shows the following output:

```
(use "git add <file>..." to update what will be committed)
(use "git restore <file>..." to discard changes in working directory)
modified:   .idea/workspace.xml
modified:   Tests/TestCoche.java
modified:   src/Coche.java

no changes added to commit (use "git add" and/or "git commit -a")
PS C:\Users\Francisco\Desktop\UD10 EDE\TDD 1 IntelIj\TDD Practica Coche> git add .
warning: in the working copy of '.idea/workspace.xml', LF will be replaced by CRLF the next time Git touches it
PS C:\Users\Francisco\Desktop\UD10 EDE\TDD 1 IntelIj\TDD Practica Coche> git commit -m "Creación método velocidad y Test velocidad"
[master bb4e176] Creación método velocidad y Test velocidad
3 files changed, 11 insertions(+), 2 deletions(-)
PS C:\Users\Francisco\Desktop\UD10 EDE\TDD 1 IntelIj\TDD Practica Coche>
```

The status bar at the bottom indicates "Tests passed: 1 (23 minutes ago)".

También en este caso nos da un error y es porque no hemos creado el método acelerar en la clase Coche, pero el IDE nos da la opción de crearlo automáticamente. Una vez creado le decimos que la velocidad aumente con la instrucción que se ve dentro del método.



The screenshot shows an IDE with the following components:

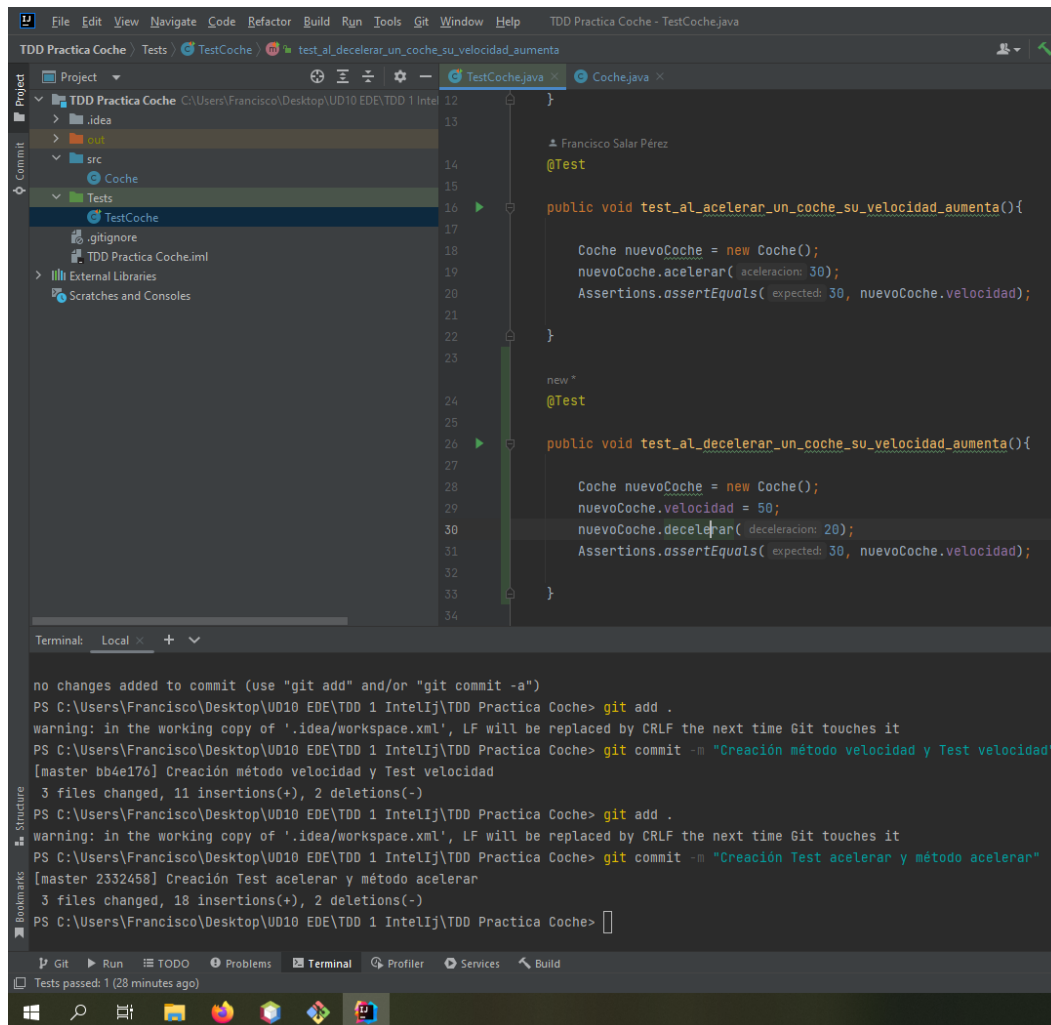
- Project View:** Shows the project structure with folders for `src` (containing `Coche`) and `Tests` (containing `TestCoche`).
- Code Editor:** Displays the `Coche.java` file with the following code:

```
1 public class Coche {  
2     public int velocidad;  
3  
4     public void acelerar(int aceleracion) {  
5         velocidad += aceleracion;  
6     }  
7  
8 }  
9  
10  
11
```
- Terminal:** Shows the output of git commands:

```
(use "git add <file>..." to update what will be committed)  
(use "git restore <file>..." to discard changes in working directory)  
modified:   .idea/workspace.xml  
modified:   Tests/TestCoche.java  
modified:   src/Coche.java  
  
no changes added to commit (use "git add" and/or "git commit -a")  
PS C:\Users\Francisco\Desktop\UD10 EDE\TDD 1 IntelIj\TDD Practica Coche> git add .  
warning: in the working copy of '.idea/workspace.xml', LF will be replaced by CRLF the next time Git touches it  
PS C:\Users\Francisco\Desktop\UD10 EDE\TDD 1 IntelIj\TDD Practica Coche> git commit -m "Creación método velocidad y Test ve  
[master bb4e176] Creación método velocidad y Test velocidad  
3 files changed, 11 insertions(+), 2 deletions(-)  
PS C:\Users\Francisco\Desktop\UD10 EDE\TDD 1 IntelIj\TDD Practica Coche> 
```
- Status Bar:** Shows "Tests passed: 1 (23 minutes ago)".

Una vez comprobado que el test se ejecuta correctamente, procedemos a realizar un nuevo "commit" con los cambios realizados: creación de un nuevo método y un nuevo test.

Todo seguido, creamos un nuevo test, esta vez decelerando al coche y disminuyendo su velocidad, para ello debemos hacer que el coche ya tenga una velocidad, en este caso de 50, y posteriormente, decelerarlo mediante el método decelerar unos 20.



The screenshot shows an IDE with a project named "TDD Practica Coche". The left sidebar shows the project structure with folders for ".idea", "src", "Tests", and "External Libraries". The "Tests" folder is expanded, showing "TestCoche". The main editor displays the "TestCoche.java" file, which contains two test methods:

```
12 }
13
14 Francisco Salar Pérez
15 @Test
16 public void test_al_acelerar_un_coche_su_velocidad_aumenta(){
17
18     Coche nuevoCoche = new Coche();
19     nuevoCoche.acelerar( aceleracion: 30);
20     Assertions.assertEquals( expected: 30, nuevoCoche.velocidad);
21 }
22
23 new *
24 @Test
25
26 public void test_al_decelerar_un_coche_su_velocidad_aumenta(){
27
28     Coche nuevoCoche = new Coche();
29     nuevoCoche.velocidad = 50;
30     nuevoCoche.decelerar( deceleracion: 20);
31     Assertions.assertEquals( expected: 30, nuevoCoche.velocidad);
32 }
33
34 }
```

The bottom panel shows the Terminal window with the following output:

```
no changes added to commit (use "git add" and/or "git commit -a")
PS C:\Users\Francisco\Desktop\UD10 EDE\TDD 1 IntelIj\TDD Practica Coche> git add .
warning: in the working copy of '.idea/workspace.xml', LF will be replaced by CRLF the next time Git touches it
PS C:\Users\Francisco\Desktop\UD10 EDE\TDD 1 IntelIj\TDD Practica Coche> git commit -m "Creación método velocidad y Test velocidad"
[master bb4e176] Creación método velocidad y Test velocidad
3 files changed, 11 insertions(+), 2 deletions(-)
PS C:\Users\Francisco\Desktop\UD10 EDE\TDD 1 IntelIj\TDD Practica Coche> git add .
warning: in the working copy of '.idea/workspace.xml', LF will be replaced by CRLF the next time Git touches it
PS C:\Users\Francisco\Desktop\UD10 EDE\TDD 1 IntelIj\TDD Practica Coche> git commit -m "Creación Test acelerar y método acelerar"
[master 2332458] Creación Test acelerar y método acelerar
3 files changed, 18 insertions(+), 2 deletions(-)
PS C:\Users\Francisco\Desktop\UD10 EDE\TDD 1 IntelIj\TDD Practica Coche> 
```

The status bar at the bottom indicates "Tests passed: 1 (28 minutes ago)".

Como viene ya siendo costumbre, como no hemos creado el método antes de utilizarlo, nos da un error, que como anteriormente subsanamos creándolo automáticamente con su correspondiente opción. Dentro del método, incluimos la instrucción que nos permitirá que el coche pueda disminuir la velocidad.

The screenshot shows an IDE window titled 'TDD Practica Coche - Coche.java'. The left sidebar shows the project structure with 'Coche' and 'Tests' folders. The main editor displays the 'Coche.java' file with the following code:

```

1  public class Coche {
2      public int velocidad;
3
4      public void acelerar(int aceleracion) {
5          velocidad += aceleracion;
6      }
7
8      public void decelerar(int deceleracion) {
9          velocidad -= deceleracion;
10     }
11 }

```

The terminal at the bottom shows the following commands and output:

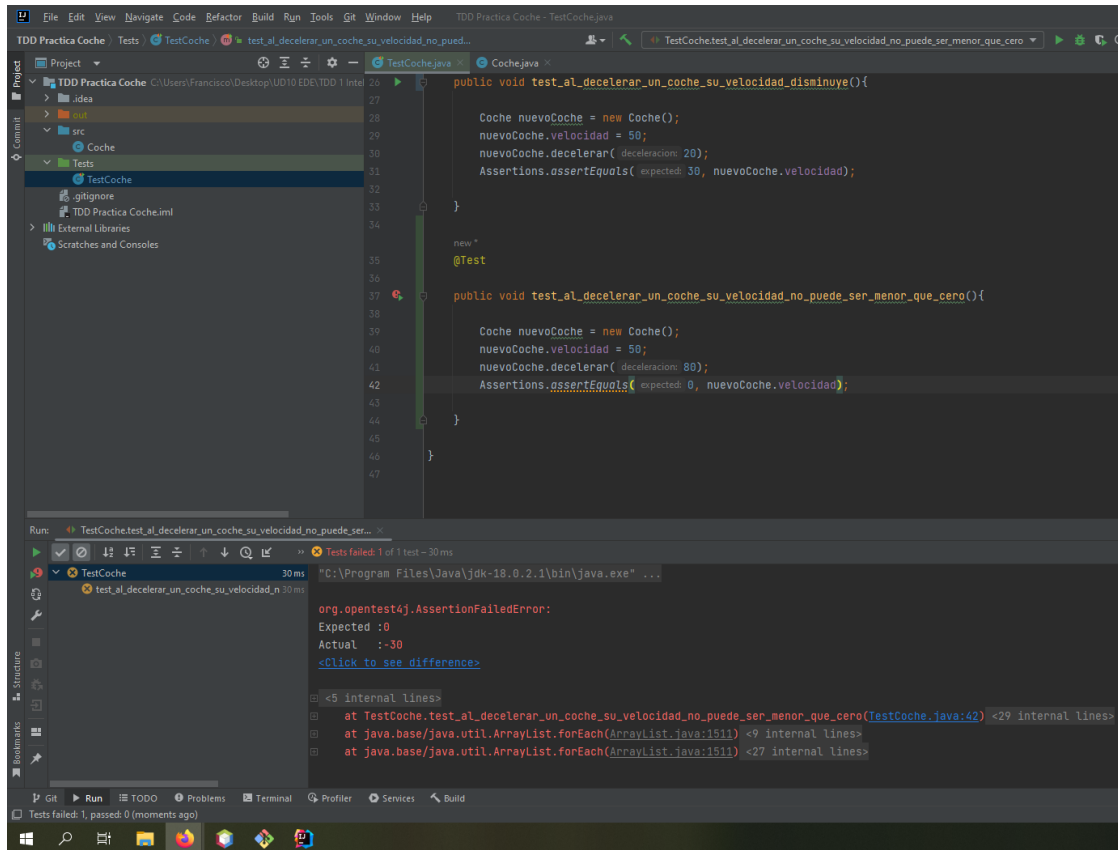
```

no changes added to commit (use "git add" and/or "git commit -a")
PS C:\Users\Francisco\Desktop\UD10 EDE\TDD 1 Intelij\TDD Practica Coche> git add .
warning: in the working copy of '.idea/workspace.xml', LF will be replaced by CRLF the next time Git touches it
PS C:\Users\Francisco\Desktop\UD10 EDE\TDD 1 Intelij\TDD Practica Coche> git commit -m "Creación método velocidad y Test velocidad"
[master bb4e176] Creación método velocidad y Test velocidad
3 files changed, 11 insertions(+), 2 deletions(-)
PS C:\Users\Francisco\Desktop\UD10 EDE\TDD 1 Intelij\TDD Practica Coche> git add .
warning: in the working copy of '.idea/workspace.xml', LF will be replaced by CRLF the next time Git touches it
PS C:\Users\Francisco\Desktop\UD10 EDE\TDD 1 Intelij\TDD Practica Coche> git commit -m "Creación Test acelerar y método acelerar"
[master 2332458] Creación Test acelerar y método acelerar
3 files changed, 18 insertions(+), 2 deletions(-)
PS C:\Users\Francisco\Desktop\UD10 EDE\TDD 1 Intelij\TDD Practica Coche>

```

Posteriormente, realizamos el consiguiente “commit” con los cambios realizados. Creación nuevo test decelerar y método decelerar.

Pasamos ahora a crear otro test, esta vez para que cuando el coche decelere su velocidad no pueda ser inferior a cero. Para ello modificamos el “Assertions” con la velocidad “0” esperada y procedemos a su ejecución.



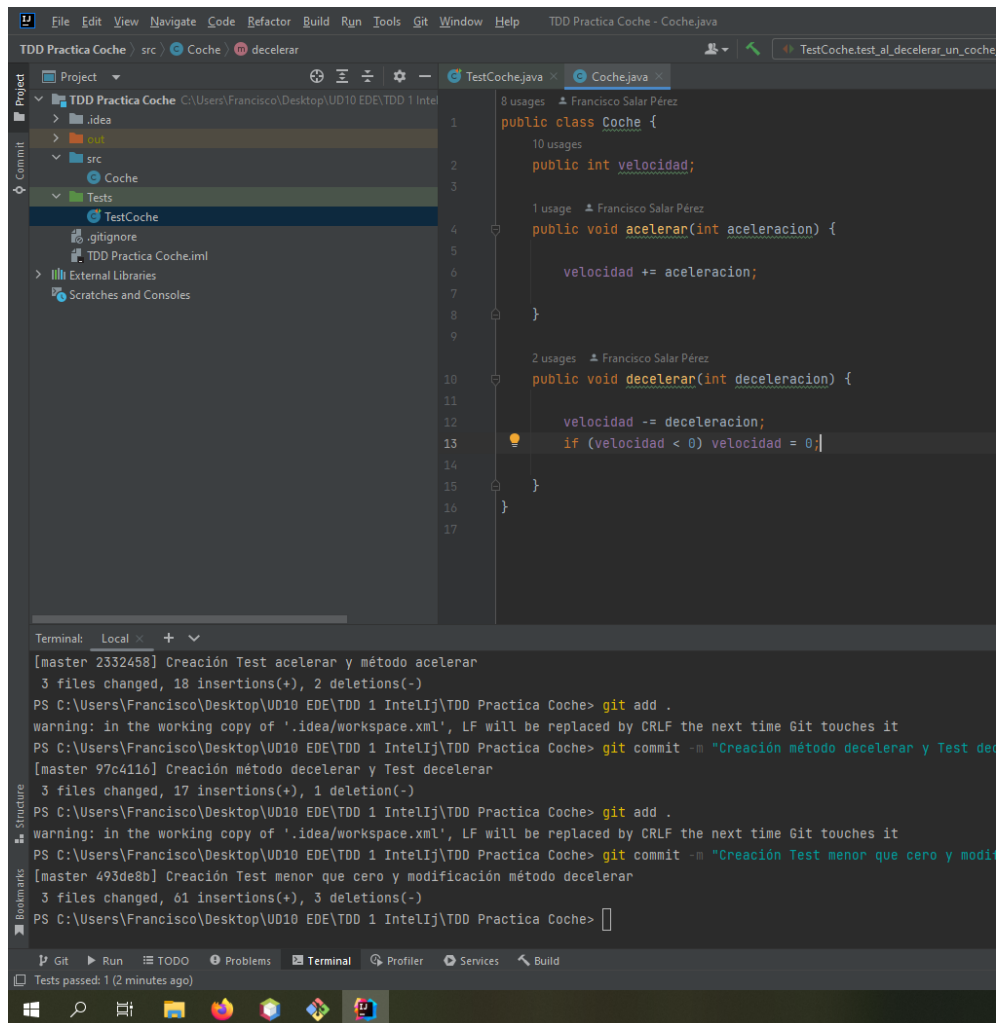
The screenshot shows an IDE with a project named "TDD Practica Coche". The left sidebar shows the project structure with folders for "src" and "Tests". The "Tests" folder is expanded, showing "TestCoche". The main editor displays the code for "TestCoche.java". The code includes two test methods: "test_al_decelerar_un_coche_su_velocidad_disminuye()" and "test_al_decelerar_un_coche_su_velocidad_no_puede_ser_menor_que_cero()". The second test method is highlighted, showing an assertion for a velocity of 0. The bottom panel shows the "Run" output, indicating that the test failed. The error message is "org.opentest4j.AssertionFailedError: Expected :0 Actual :30". The stack trace shows the error occurred in "TestCoche.java:42".

```
public void test_al_decelerar_un_coche_su_velocidad_disminuye(){  
    Coche nuevoCoche = new Coche();  
    nuevoCoche.velocidad = 50;  
    nuevoCoche.decelerar( 20);  
    Assertions.assertEquals( expected: 30, nuevoCoche.velocidad);  
}  
  
@Test  
public void test_al_decelerar_un_coche_su_velocidad_no_puede_ser_menor_que_cero(){  
    Coche nuevoCoche = new Coche();  
    nuevoCoche.velocidad = 50;  
    nuevoCoche.decelerar( deceleracion: 80);  
    Assertions.assertEquals( expected: 0, nuevoCoche.velocidad);  
}  
}
```

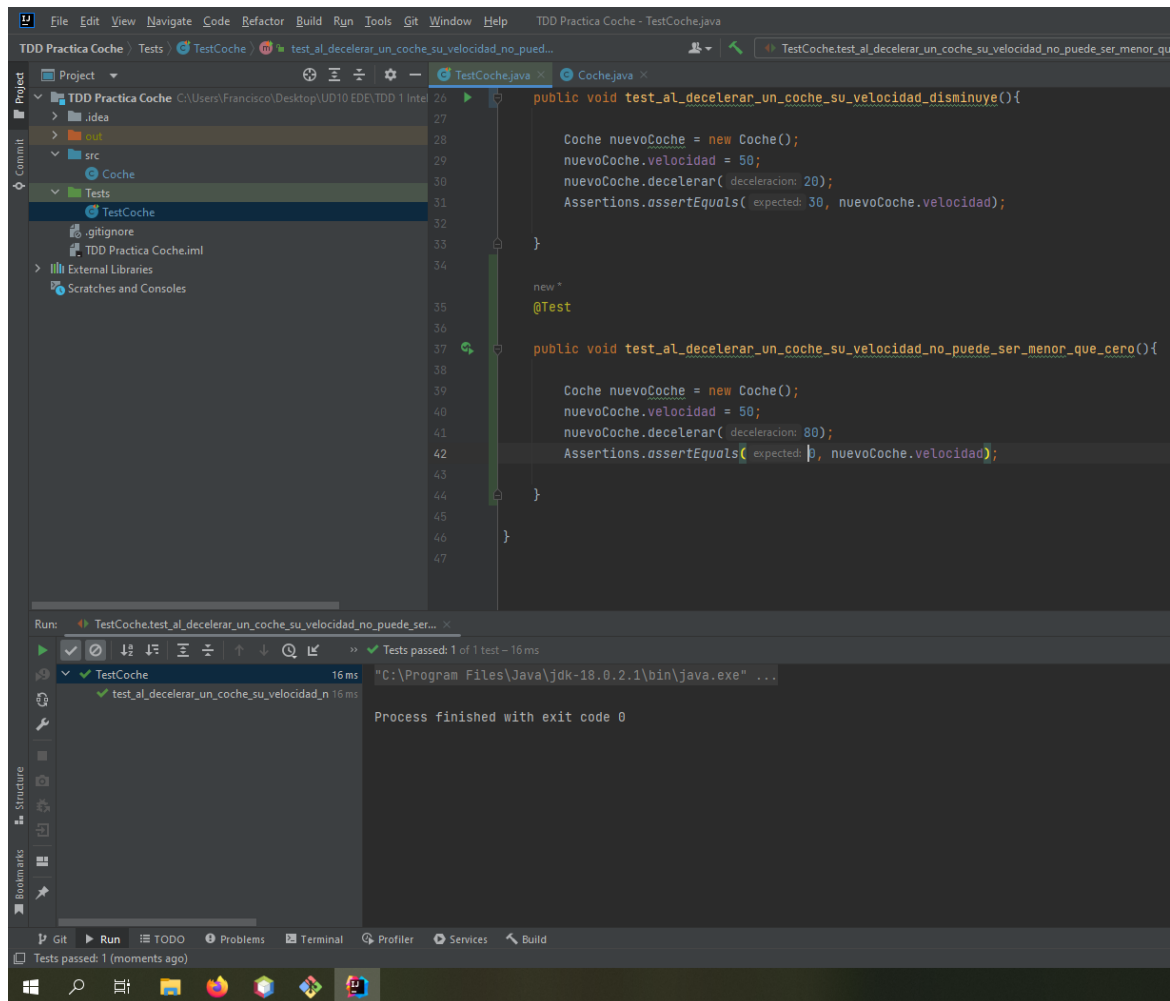
Run: TestCoche.test_al_decelerar_un_coche_su_velocidad_no_puede_ser...
Tests failed: 1 of 1 test - 30 ms
C:\Program Files\Java\jdk-18.0.2.1\bin\java.exe ...
test_al_decelerar_un_coche_su_velocidad_n 30 ms
org.opentest4j.AssertionFailedError:
Expected :0
Actual :30
<Click to see difference>
<5 internal lines>
at TestCoche.test_al_decelerar_un_coche_su_velocidad_no_puede_ser_menor_que_cero(TestCoche.java:42) <29 internal lines>
at java.base/java.util.ArrayList.forEach(ArrayList.java:1511) <9 internal lines>
at java.base/java.util.ArrayList.forEach(ArrayList.java:1511) <27 internal lines>

Vemos que nos da un error, y este se debe a que no hemos modificado el método decelerar para que en caso de obtener valores negativos, como es el caso, el resultado igualmente sea “0”.

Para ello, modificamos el código dentro de nuestro método como se ve en la siguiente imagen:



Y como se puede ver, el test ya no arroja ningún error:



Una vez hecho lo anterior, procedemos a subir a nuestro repositorio remoto el proyecto con los comandos pertinentes, una vez habiendo creado en GitHub nuestro repositorio y habiendo copiado la url para poder hacer un “push a origen”.

```
File Edit View Navigate Code Refactor Build Run Tools Git Window Help TDD Practica Coche - Coche.java
TDD Practica Coche / src / Coche / decelerar TestCoche.java Coche.java TestCoche.test_al_decelerar

Project
  TDD Practica Coche C:\Users\Francisco\Desktop\UD10 EDE\TDD 1 IntelliJ\TDD Practica Coche
    .idea
    out
    src
    Tests
      TestCoche
      .gitignore
      TDD Practica Coche.iml
    External Libraries
    Scratches and Consoles

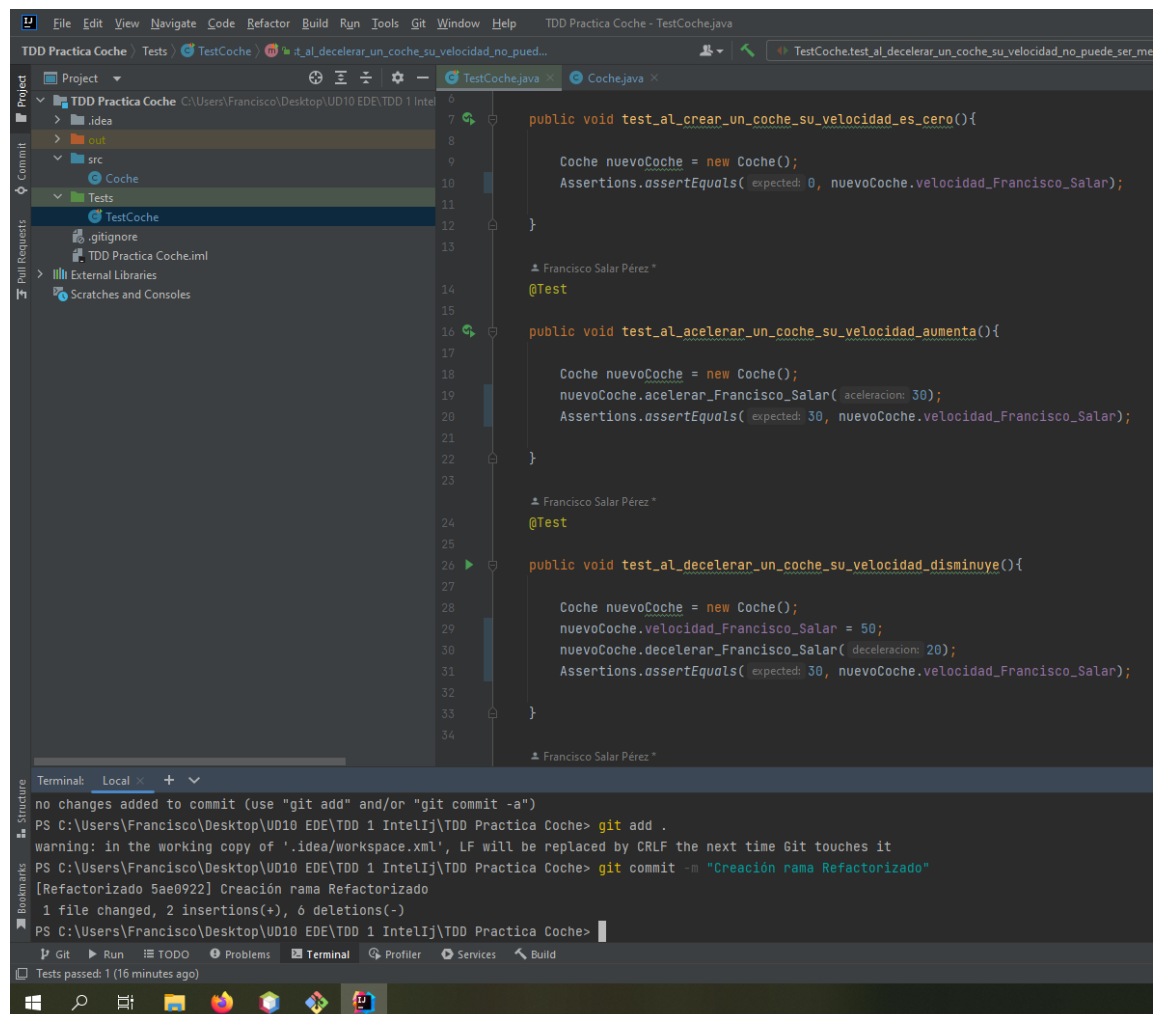
Commit
Pull Requests

Terminal: Local x + v
To https://github.com/CorazaNegra/TDD-1-IntelliJ.git
* [new branch] master -> master
PS C:\Users\Francisco\Desktop\UD10 EDE\TDD 1 IntelliJ\TDD Practica Coche> git branch Refactorizado
PS C:\Users\Francisco\Desktop\UD10 EDE\TDD 1 IntelliJ\TDD Practica Coche> git branch
Refactorizado
* master
PS C:\Users\Francisco\Desktop\UD10 EDE\TDD 1 IntelliJ\TDD Practica Coche> git checkout Refactorizado
Switched to branch 'Refactorizado'
M .idea/workspace.xml
PS C:\Users\Francisco\Desktop\UD10 EDE\TDD 1 IntelliJ\TDD Practica Coche> git branch
* Refactorizado
master
PS C:\Users\Francisco\Desktop\UD10 EDE\TDD 1 IntelliJ\TDD Practica Coche>

Tests passed: 1 (11 minutes ago)
```

Ahora creamos una nueva rama donde refactorizaremos los nombres de los métodos añadiendo nuestro nombre a cada uno de ellos.

Para ello mediante “git Branch Refactorizado”, creamos la nueva rama, y con “git checkout Refactorizado” nos posicionamos en ella para realizar el refactorizado de los métodos.

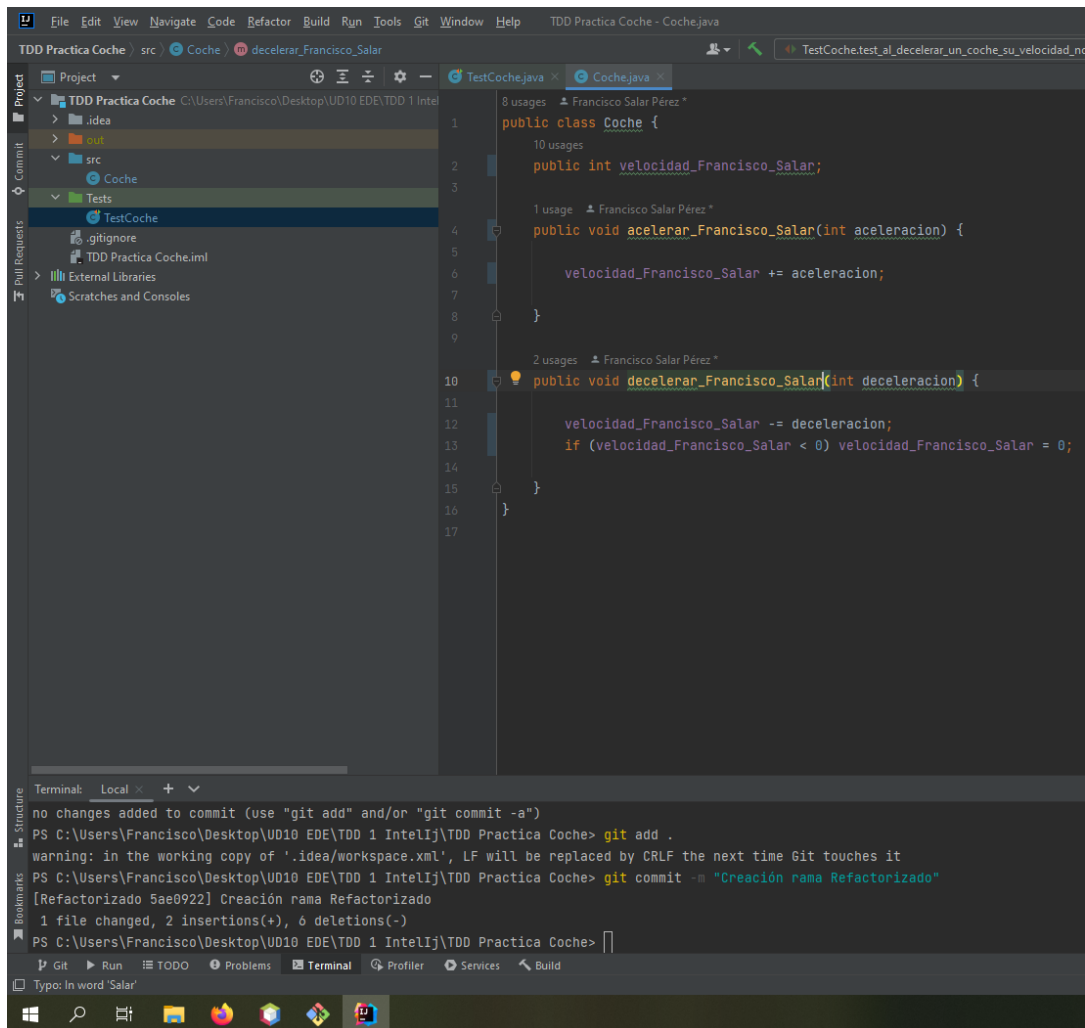


The screenshot shows an IDE with a project named "TDD Practica Coche". The left sidebar shows the project structure with folders like "src" and "Tests". The main editor displays a Java file "TestCoche.java" with three test methods: `test_al_crear_un_coche_su_velocidad_es_cero()`, `test_al_acelerar_un_coche_su_velocidad_aumenta()`, and `test_al_decelerar_un_coche_su_velocidad_disminuye()`. Each method creates a `Coche` object and asserts its velocity. The terminal at the bottom shows the following commands and output:

```
no changes added to commit (use "git add" and/or "git commit -a")
PS C:\Users\Francisco\Desktop\UD10 EDE\TDD 1 IntelIj\TDD Practica Coche> git add .
warning: in the working copy of '.idea/workspace.xml', LF will be replaced by CRLF the next time Git touches it
PS C:\Users\Francisco\Desktop\UD10 EDE\TDD 1 IntelIj\TDD Practica Coche> git commit -m "Creación rama Refactorizado"
[Refactorizado 5ae0922] Creación rama Refactorizado
1 file changed, 2 insertions(+), 6 deletions(-)
PS C:\Users\Francisco\Desktop\UD10 EDE\TDD 1 IntelIj\TDD Practica Coche>
```

Hacemos entonces un “commit” con los cambios hechos: creación de la nueva rama “Refactorizado”.

En la anterior imagen junto con la siguiente podemos ver como se han modificado los métodos con la adición de mi nombre al final de los mismos.



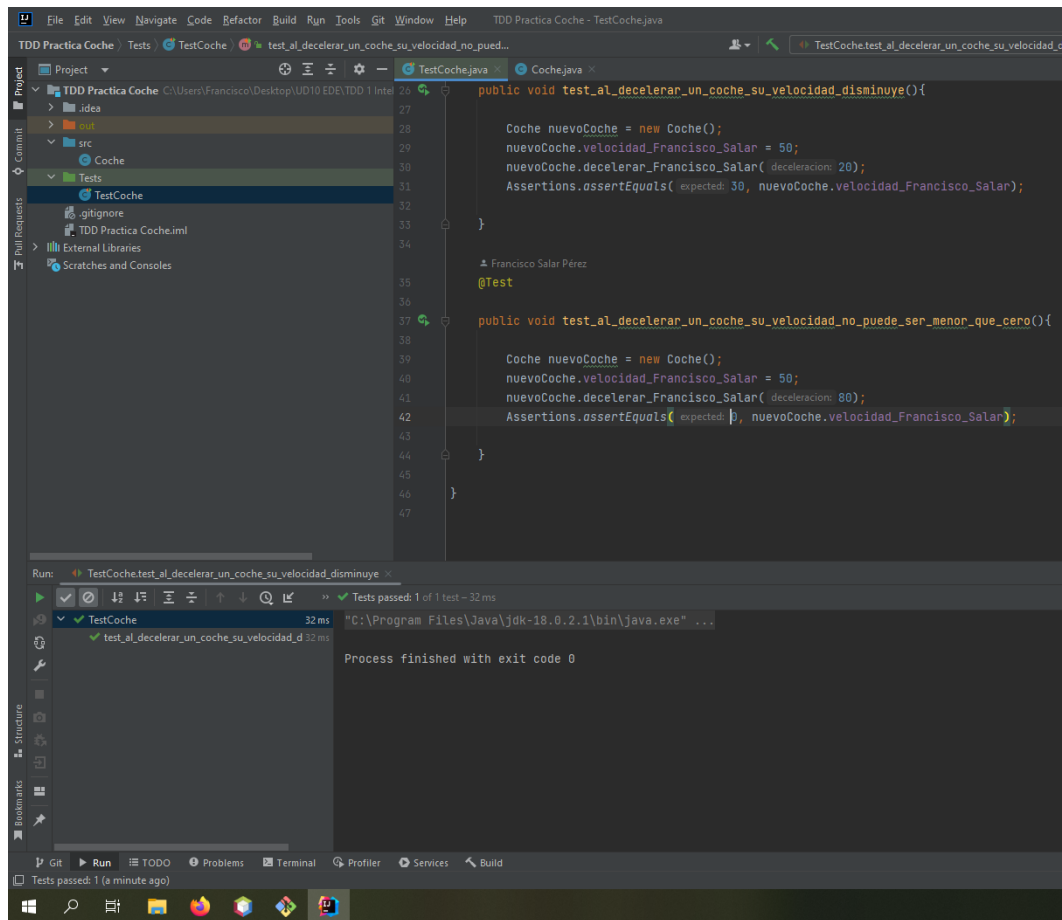
The screenshot shows an IDE with the following components:

- Project Structure:** TDD Practica Coche (src, out, Tests, TestCoche, .gitignore, TDD Practica Coche.iml, External Libraries, Scratches and Consoles).
- Code Editor:** Shows the `Coche.java` file with the following code:

```
1 public class Coche {  
2     public int velocidad_Francisco_Salar;  
3  
4     public void acelerar_Francisco_Salar(int aceleracion) {  
5  
6         velocidad_Francisco_Salar += aceleracion;  
7     }  
8  
9  
10    public void decelerar_Francisco_Salar(int deceleracion) {  
11  
12        velocidad_Francisco_Salar -= deceleracion;  
13        if (velocidad_Francisco_Salar < 0) velocidad_Francisco_Salar = 0;  
14    }  
15  
16 }  
17
```
- Terminal:** Shows the following commands and output:

```
no changes added to commit (use "git add" and/or "git commit -a")  
PS C:\Users\Francisco\Desktop\UD10 EDE\TDD 1 IntelIj\TDD Practica Coche> git add .  
warning: in the working copy of '.idea/workspace.xml', LF will be replaced by CRLF the next time Git touches it  
PS C:\Users\Francisco\Desktop\UD10 EDE\TDD 1 IntelIj\TDD Practica Coche> git commit -m "Creación rama Refactorizado"  
[Refactorizado 5ae0922] Creación rama Refactorizado  
1 file changed, 2 insertions(+), 0 deletions(-)  
PS C:\Users\Francisco\Desktop\UD10 EDE\TDD 1 IntelIj\TDD Practica Coche>
```

Hacemos una última comprobación para ver que los test sigan funcionando:



A continuación hacemos un “commit” a nuestro repositorio local con los cambios realizados en la rama “Refactorizado” y posteriormente con los comandos correspondientes subimos a nuestro repositorio remoto la nueva rama con sus cambios “git push origin Refactorizado”.

The screenshot shows an IDE with a project named "TDD Practica Coche". The left sidebar shows the project structure with folders like ".idea", "src", and "Tests". The main editor displays two Java test methods in "TestCoche.java":

```

25
26 public void test_al_decelerar_un_coche_su_velocidad_disminuye(){
27
28     Coche nuevoCoche = new Coche();
29     nuevoCoche.velocidad_Francisco_Salar = 50;
30     nuevoCoche.decelerar_Francisco_Salar( deceleracion: 20);
31     Assertions.assertEquals( expected: 30, nuevoCoche.velocidad_Francisco_Salar);
32 }
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100

```

The terminal window shows the following commands and output:

```

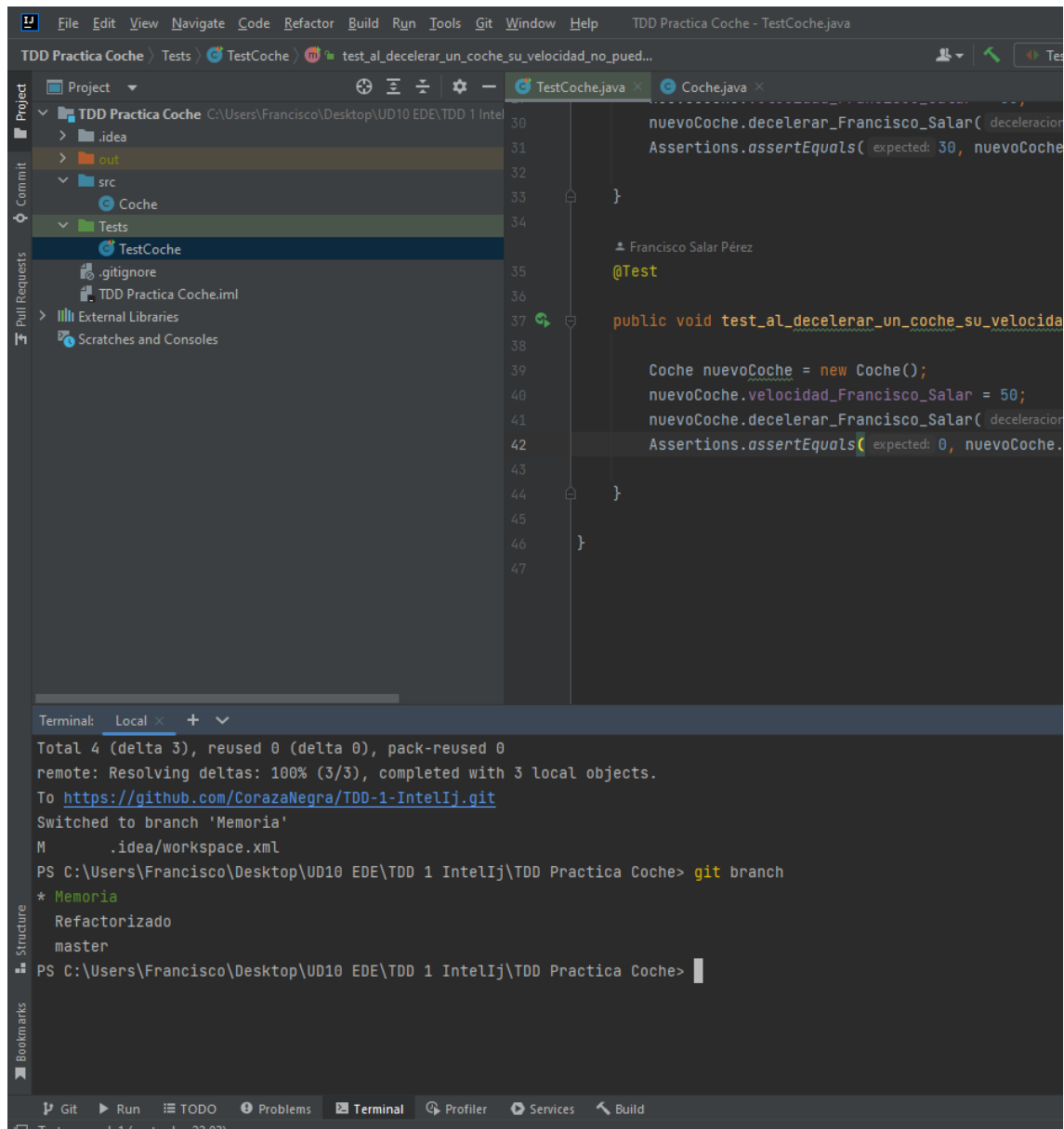
PS C:\Users\Francisco\Desktop\UD10 EDE\TDD 1 IntelIj\TDD Practica Coche> git status
On branch Refactorizado
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
       modified:   .idea/workspace.xml
       modified:   src/Coche.java

PS C:\Users\Francisco\Desktop\UD10 EDE\TDD 1 IntelIj\TDD Practica Coche> git add .
remote: Create a pull request for 'Refactorizado' on GitHub by visiting:
remote:   https://github.com/CorazaNegra/TDD-1-IntelIj/pull/new/Refactorizado
remote:
To https://github.com/CorazaNegra/TDD-1-IntelIj.git
 * [new branch]   Refactorizado -> Refactorizado
PS C:\Users\Francisco\Desktop\UD10 EDE\TDD 1 IntelIj\TDD Practica Coche>

```

The bottom status bar indicates "Tests passed: 1 (3 minutes ago)".

Por último, creamos una rama llamada “Memoria” donde subiremos el pdf con la memoria realizada de esta práctica.



The screenshot shows an IDE with a project named "TDD Practica Coche". The left sidebar shows the project structure with folders for "out", "src", "Coche", and "Tests". The "Tests" folder is expanded, showing "TestCoche". The main editor displays the "TestCoche.java" file. The code in the editor is as follows:

```
30 nuevoCoche.decelerar_Francisco_Salar( deceleracion
31 Assertions.assertEquals( expected: 30, nuevoCoche
32
33 }
34
35
36 @Test
37
38 public void test_al_decelerar_un_coche_su_velocida
39
40 Coche nuevoCoche = new Coche();
41 nuevoCoche.velocidad_Francisco_Salar = 50;
42 nuevoCoche.decelerar_Francisco_Salar( deceleracion
43 Assertions.assertEquals( expected: 0, nuevoCoche.
44
45 }
46
47 }
```

The terminal window at the bottom shows the following output:

```
Terminal: Local x + v
Total 4 (delta 3), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (3/3), completed with 3 local objects.
To https://github.com/CorazaNegra/TDD-1-IntelliJ.git
Switched to branch 'Memoria'
M .idea/workspace.xml
PS C:\Users\Francisco\Desktop\UD10 EDE\TDD 1 IntelliJ\TDD Practica Coche> git branch
* Memoria
  Refactorizado
  master
PS C:\Users\Francisco\Desktop\UD10 EDE\TDD 1 IntelliJ\TDD Practica Coche>
```

Haciendo el último “commit” con los cambios realizados: creación nueva rama “Memoria” y adición pdf de la memoria. Posterior mente con los comandos que hemos visto anteriormente hacemos el push correspondiente a nuestro repositorio remoto.