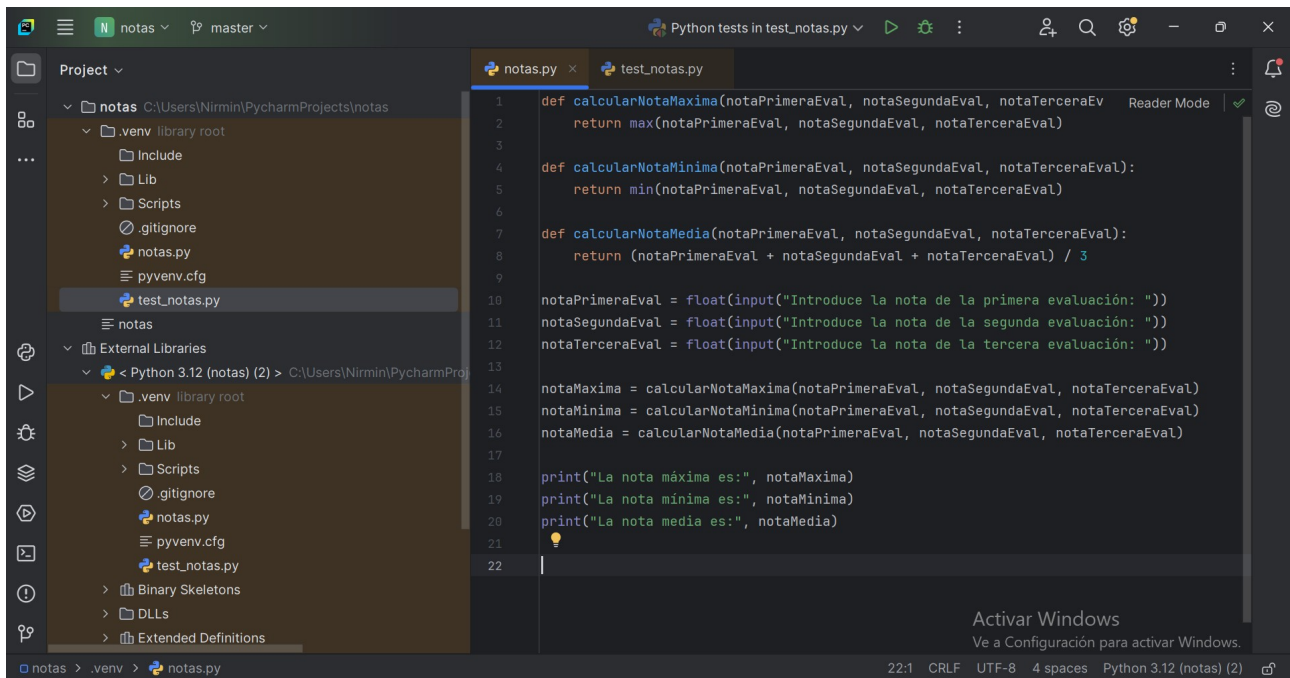
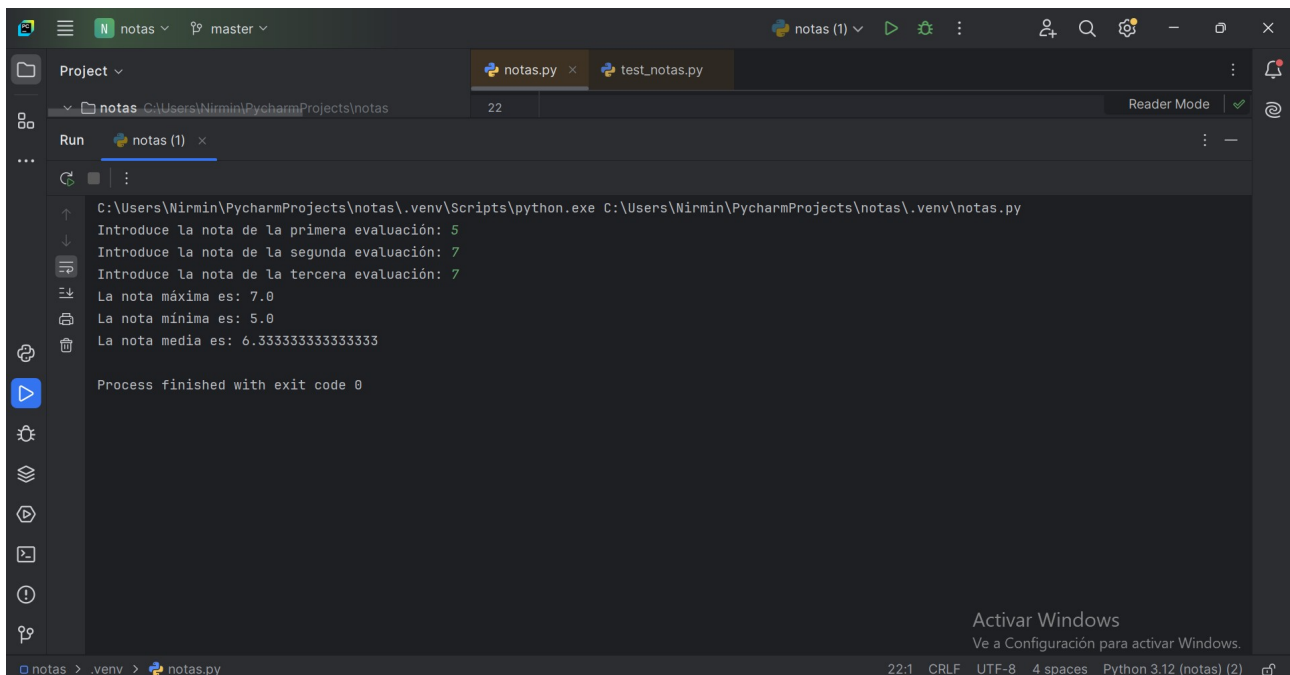


1) Desarrollar una aplicación en Python que pida al alumno (usuario) las tres notas (float) de una asignatura: `float notaPrimeraEval, notaSegundaEval, notaTerceraEval`; La aplicación le devolverá: cuál ha sido la nota máxima de las tres evaluaciones, cuál ha sido la nota mínima obtenida de las tres evaluaciones y cuál ha sido la nota media. Para realizar esos cálculos, la aplicación contará con tres funciones que devuelven un valor float: `calcularNotaMaxima (notaPrimeraEval, notaSegundaEval, notaTerceraEval)` `calcularNotaMinima (notaPrimeraEval, notaSegundaEval, notaTerceraEval)` `calcularNotaMedia (notaPrimeraEval, notaSegundaEval, notaTerceraEval)`



The screenshot shows the PyCharm IDE with the file `notas.py` open. The code defines three functions: `calcularNotaMaxima`, `calcularNotaMinima`, and `calcularNotaMedia`. It then prompts the user for three grades, calculates the maximum, minimum, and average, and prints the results.

```
1 def calcularNotaMaxima(notaPrimeraEval, notaSegundaEval, notaTerceraEval):
2     return max(notaPrimeraEval, notaSegundaEval, notaTerceraEval)
3
4 def calcularNotaMinima(notaPrimeraEval, notaSegundaEval, notaTerceraEval):
5     return min(notaPrimeraEval, notaSegundaEval, notaTerceraEval)
6
7 def calcularNotaMedia(notaPrimeraEval, notaSegundaEval, notaTerceraEval):
8     return (notaPrimeraEval + notaSegundaEval + notaTerceraEval) / 3
9
10 notaPrimeraEval = float(input("Introduce la nota de la primera evaluación: "))
11 notaSegundaEval = float(input("Introduce la nota de la segunda evaluación: "))
12 notaTerceraEval = float(input("Introduce la nota de la tercera evaluación: "))
13
14 notaMaxima = calcularNotaMaxima(notaPrimeraEval, notaSegundaEval, notaTerceraEval)
15 notaMinima = calcularNotaMinima(notaPrimeraEval, notaSegundaEval, notaTerceraEval)
16 notaMedia = calcularNotaMedia(notaPrimeraEval, notaSegundaEval, notaTerceraEval)
17
18 print("La nota máxima es:", notaMaxima)
19 print("La nota mínima es:", notaMinima)
20 print("La nota media es:", notaMedia)
21
22
```

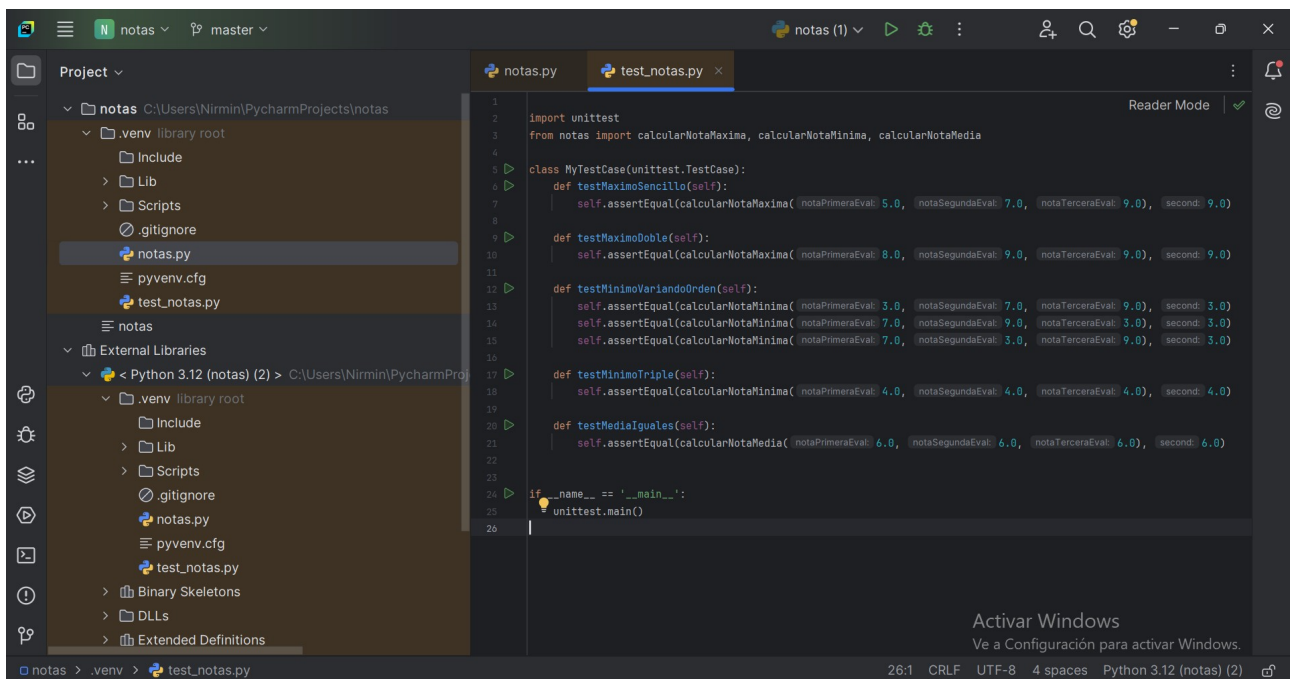


The screenshot shows the PyCharm IDE with the `Run` console open. The output shows the user inputting three grades (5, 7, 7) and the program calculating and printing the maximum (7.0), minimum (5.0), and average (6.333333333333333) grades.

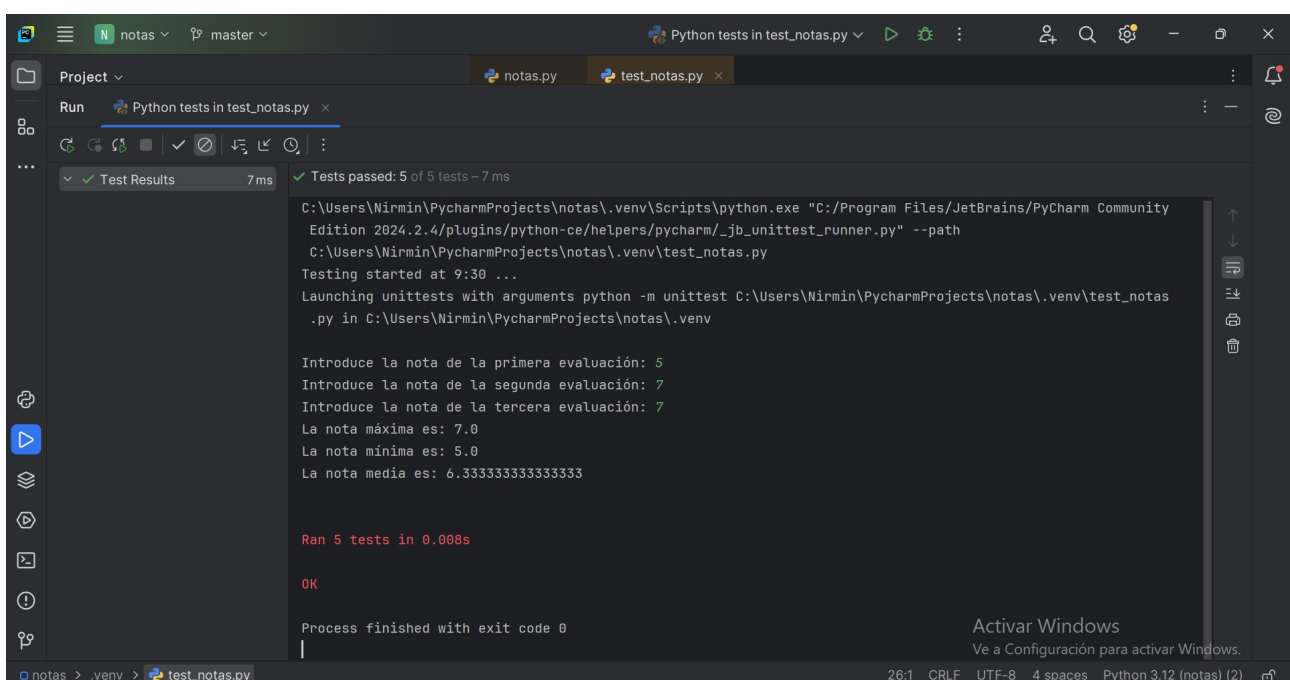
```
Run
C:\Users\Nirmin\PycharmProjects\notas\.venv\Scripts\python.exe C:\Users\Nirmin\PycharmProjects\notas\.venv\notas.py
Introduce la nota de la primera evaluación: 5
Introduce la nota de la segunda evaluación: 7
Introduce la nota de la tercera evaluación: 7
La nota máxima es: 7.0
La nota mínima es: 5.0
La nota media es: 6.333333333333333
Process finished with exit code 0
```

2) Una vez desarrollada la aplicación, se debe crear una clase Test con unittest que permita detectar todos los errores presentes en la aplicación (si los hay), en caso de que los haya genera alguno para que compruebes que los test fallan. Se debe incluir los siguientes casos de prueba mediante una función independiente para cada caso/test:

1. testMaximoSencillo: se cumple si en las notas sólo hay un número máximo y es éste el que se devuelve mediante la función calcularNotaMaxima.
2. testMaximoDoble: se cumple cuando en las notas se repite el número máximo y la función calcularNotaMaxima devuelve ese máximo.
3. testMinimoVariandoOrden: comprueba que la función calcularNotaMinima calcula la nota mínima correcta aunque se varíe el orden de las notas: la nota mínima en la 1ª posición, en la última y en el medio.
4. testMinimoTriple: comprueba que la función calcularNotaMinima devuelve el valor de la nota mínima correcta cuando las tres notas indicadas son iguales.
5. testMedialIguales: verifica el correcto funcionamiento de la función calcularNotaMedia cuando las tres notas indicadas son iguales.



```
1 import unittest
2 from notas import calcularNotaMaxima, calcularNotaMinima, calcularNotaMedia
3
4 class MyTestCase(unittest.TestCase):
5     def testMaximoSencillo(self):
6         self.assertEqual(calcularNotaMaxima( notaPrimeraEval: 5.0, notaSegundaEval: 7.0, notaTerceraEval: 9.0), second: 9.0)
7
8     def testMaximoDoble(self):
9         self.assertEqual(calcularNotaMaxima( notaPrimeraEval: 8.0, notaSegundaEval: 9.0, notaTerceraEval: 9.0), second: 9.0)
10
11     def testMinimoVariandoOrden(self):
12         self.assertEqual(calcularNotaMinima( notaPrimeraEval: 3.0, notaSegundaEval: 7.0, notaTerceraEval: 9.0), second: 3.0)
13         self.assertEqual(calcularNotaMinima( notaPrimeraEval: 7.0, notaSegundaEval: 9.0, notaTerceraEval: 3.0), second: 3.0)
14         self.assertEqual(calcularNotaMinima( notaPrimeraEval: 7.0, notaSegundaEval: 3.0, notaTerceraEval: 9.0), second: 3.0)
15
16     def testMinimoTriple(self):
17         self.assertEqual(calcularNotaMinima( notaPrimeraEval: 4.0, notaSegundaEval: 4.0, notaTerceraEval: 4.0), second: 4.0)
18
19     def testMedialIguales(self):
20         self.assertEqual(calcularNotaMedia( notaPrimeraEval: 6.0, notaSegundaEval: 6.0, notaTerceraEval: 6.0), second: 6.0)
21
22 if __name__ == '__main__':
23     unittest.main()
```



```
Run Python tests in test_notas.py
Test Results 7ms
Tests passed: 5 of 5 tests - 7 ms
C:\Users\Nirmin\PycharmProjects\notas\.venv\Scripts\python.exe "C:/Program Files/JetBrains/PyCharm Community Edition 2024.2.4/plugins/python-ce/helpers/pycharm/_jb_unittest_runner.py" --path C:\Users\Nirmin\PycharmProjects\notas\.venv\test_notas.py
Testing started at 9:30 ...
Launching unittests with arguments python -m unittest C:\Users\Nirmin\PycharmProjects\notas\.venv\test_notas.py in C:\Users\Nirmin\PycharmProjects\notas\.venv
Introduce la nota de la primera evaluación: 5
Introduce la nota de la segunda evaluación: 7
Introduce la nota de la tercera evaluación: 7
La nota máxima es: 7.0
La nota mínima es: 5.0
La nota media es: 6.333333333333333
Ran 5 tests in 0.008s
OK
Process finished with exit code 0
```