



UNRN

Universidad Nacional
de Río Negro



| unrionegro

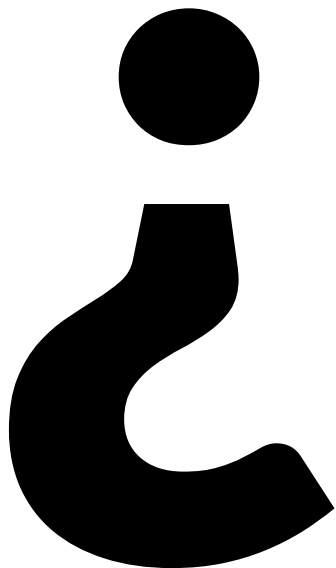
Programación 2

V

UNRN

Universidad Nacional
de **Río Negro**

Martín René Vilugrón
mrvilugron@unrn.edu.ar



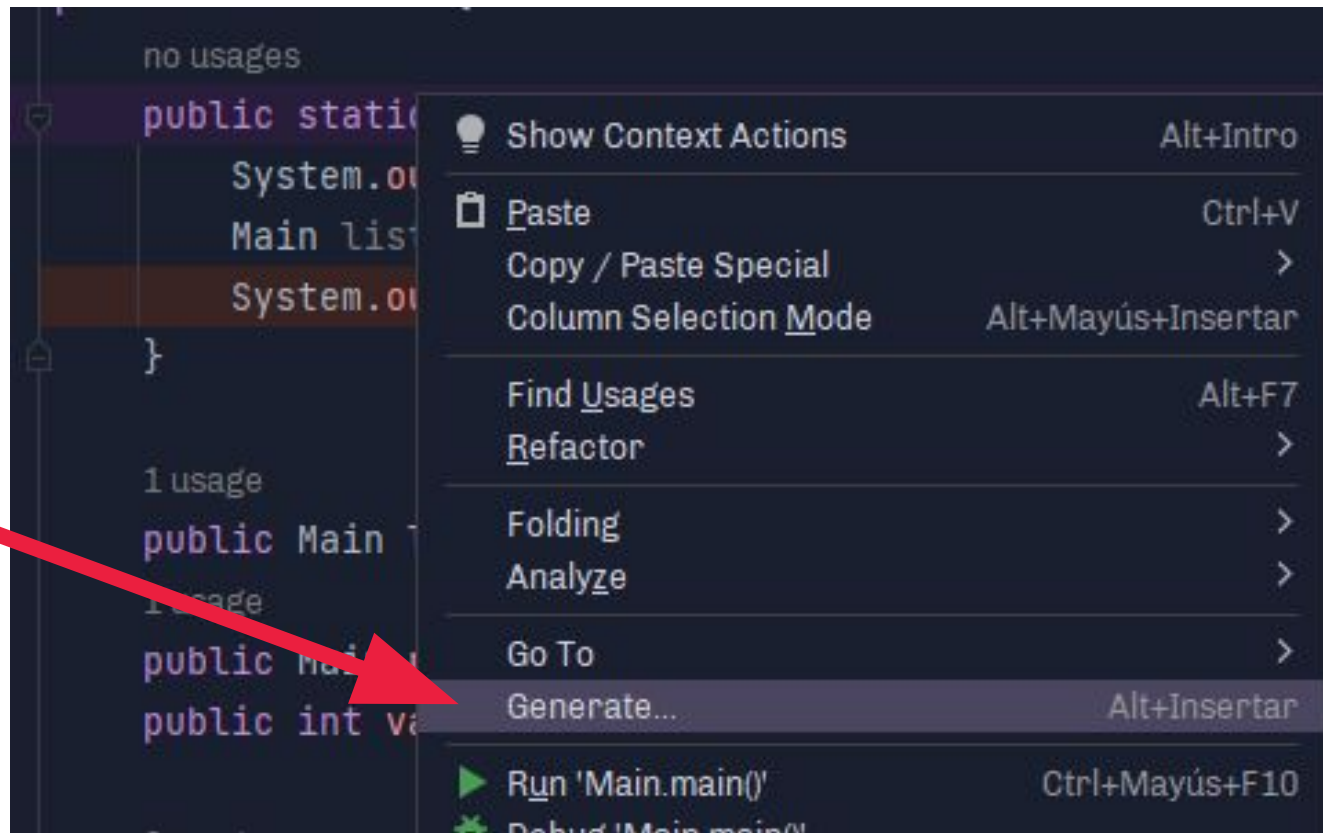
Como armar tests

(intro light a junit)

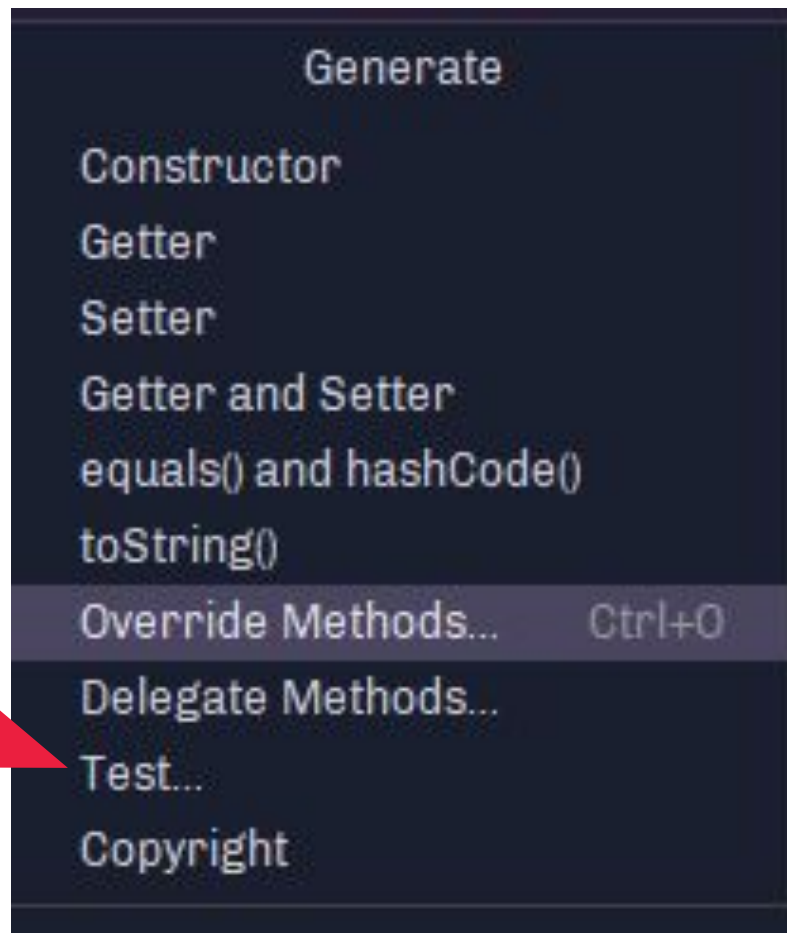
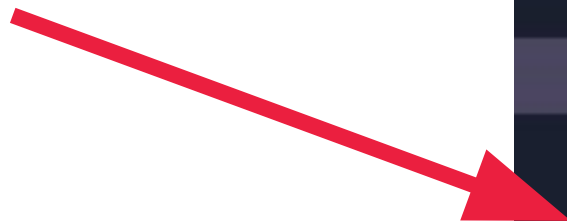


**Esto requiere de un
proyecto con Gradle
con IntelliJ**

1:




2:



3:



 Create Test

Testing library:

JUnit5

Class name:

PideDatosTest

Superclass:

...

Destination package:

unrn.programacion

...

Generate:



☐ setUp/@Before



☐ tearDown/@After

Generate test methods for:

☐ Show inherited methods

Member

☐   main(args:String[]):void

☐   pideInt(msg:String, intentos:int, min:int, max:int):int

?

OK

Cancel

El encabezado

```
package unrn.programacion;
```

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

```
import static org.junit.jupiter.api.Assertions.*;
```


El test en sí

```
class PideDatosTest {  
    @Test  
    void pideIntIgual() {  
        int esperado = 10;  
        int resultado = Main.pideInt();  
        assertEquals(esperado, resultado, "mensaje opcional");  
    }  
}
```



**Que asserts
tenemos**



—

Como regla general
No apilen líneas

assertEquals

```
@Test
public void igualdadTest() {
    String esperado = "UNRN";
    String resultado = "UNRN";

    assertEquals(esperado, resultado, "mensaje");
}
```

assertEquals - con margen de error para decimales

```
@Test
public void igualdadTest() {
    float esperado = 3.1416f;
    float resultado = 3.14f;
    float delta = 2f;
    assertEquals(esperado, resultado, delta);
}
```

assertArrayEquals

```
@Test
public void igualdadArreglos() {
    char[] esperado = {'U','N','R','N',' '};
    char[] resultado = "UNRN".toCharArray();

    assertArrayEquals(esperado, resultado);
}
```

assertArrayEquals

```
@Test
public void igualdadArreglosNulos() {
    int[] esperado = null;
    int[] resultado = null;

    assertEquals(esperado , resultado);
}
```

assertNull / assertNotNull

```
@Test
```

```
public void testNull() {
```

```
    Object cosa = null;
```

```
    assertNull("la cosa debe ser null", null);
```

```
}
```

```
@Test
```

```
public void testNotNull() {
```

```
    Gato michi = new Gato();
```

```
    assertNull("El gato debe de existir!", michi);
```

```
}
```


assertSame / assertNotSame

```
@Test
public void whenAssertingNotSameObject_thenDifferent() {
    Object michi = new Gato();
    Object pichichu = new Perro();

    assertNotSame(michi, pichichu);
}
```

assertTrue / assertFalse

```
@Test
public void testBooleano() {
    assertTrue("5 es mayor que 4", 5 > 4);
    assertFalse("5 no es mayor que 6", 5 > 6);
}
```

fail

```
@Test
public void testDeAsercion() {
    try {
        funcionQueLanzaUnaExcepcion();
        fail("La excepción no fue lanzada :-(");
    } catch (OperacionNoValidaException exc) {
        assertEquals("Operacion No Soportada", exc.getMessage());
    }
}
```



¿Preguntas?

Excepciones II

Ejemplo I

```
public class FailApp{

    public static void main(String[] args) {
        a();
    }
    static void a() {
        b();
    }
    static void b() {
        throw new RuntimeException();
    }
}
```

Ejemplo II

```
public class FailApp{

    public static void main(String[] args) {
        a();
    }
    static void a() {
        try{
            b();
        } catch (RuntimeException exc){
            System.out.println("Ouch");
        }
    }
    static void b() {
        throw new RuntimeException();
    }
}
```

Ejemplos III

```
public class FailApp{

    public static void main(String[] args) {
        funcion();
    }

    static void funcion() {
        try {
            System.out.println("Tambores");
            throw new RuntimeException();
            System.out.println("OK");
        } catch (RuntimeException exc) {
            System.out.println("Ouch");
        }
    }
}
```




¿Preguntas?

Manipulación de archivos y directorios

Gestión

```
File nombreFile = new File("carpeta o archivo");
```

Las operaciones mas interesantes

`createNewFile`

`delete`

`mkdirs`

`getPath`

`getAbsolutePath`

`renameTo`

`exists`

`canWrite`

`canRead`

`isFile`

`isDirectory`

`lastModified`

`length`

Ejemplo I: Crear si no existe

```
if(args.length > 0) {  
    File archivo = new File(args[0]);  
  
    if(!archivo.exists()) {  
        try {  
            archivo.createNewFile();  
        } catch (IOException exc) {  
            exc.printStackTrace();  
        }  
    }  
}
```

Ejemplo II: Un ls casero

```
if(args.length > 0) {  
    File f = new File(args[0]);  
  
    if(f.isDirectory()) {  
        File[] ficheros = f.listFiles();  
  
        System.out.println("Lista de los nombres de ficheros dentro del  
directorio");  
  
        for(File file : ficheros)  
            System.out.println("\t" + file.getName());  
    }  
}
```

Lectura de archivos

FileReader y BufferedReader

```
File f = new File("ejercicio1.txt");

try {
    FileReader fr = new FileReader(f);
    BufferedReader br = new BufferedReader(fr);

    String linea = br.readLine();
    System.out.println();
    while(linea != null) {
        System.out.println(linea);
        linea = br.readLine();
    }

    br.close();
} catch(IOException e) {
    e.printStackTrace();
}
```


Escritura en archivos

FileWriter y PrintWriter

```
try {  
    FileWriter fichero = new FileWriter("ejercicio1.txt");  
    PrintWriter pw = new PrintWriter(fichero);  
    pw.println("Hola Mundo");  
} catch(IOException e) {  
    e.printStackTrace();  
}
```

unrn.edu.ar

UNRN

Universidad Nacional
de **Río Negro**



| **unrionegro**