1. Subir a Bitbucket
   1. Añadir p01-netbeans
   2. Pom:
      1. coordenadas:
         1. groupId: org.ppss
         2. artifactId: p01-netbeans
         3. version: 1.0-SNAPSHOT
         4. package: unknown (por defecto es jar)
      2. propiedades
         1. project.build.sourceEncoding: UTF-8 (obligatorio de maven 3 o +)
      3. dependencias (librerías que usamos) (son ficheros .jar)
         1. junit
            1. scope: test (junit solo es necesario durante la ejecución de los tests)
            2. coordenadas: (todos los artefactos se identifican por ellas)

groupId: junit

artifactId: junit

version: 4.12

* + 1. build
       1. plugins (los plugins en <build> nos permiten alterar acciones (goals) que se ejecuten por defecto durante el proceso de ejecución)
          1. surefire (el plugin de surefire está asociado por defecto)

groupId: org.apache.maven.plugins

artifactId: maven-surefire-plugin

version: 2.20

* 1. especificación de triángulo
  2. trianguloTest

**Tabla de casos de prueba**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Identificador del Caso de Prueba | a | b | c | Resultado esperado | Resultado real |
| C1 | 1 | 1 | 1 | "Equilatero" |  |
| C2 | 1 | 1 | 11 | "No es un triángulo" |  |
| C3 | 1 | 2 | 0 | "Valor c fuera del rango permitido" |  |
| C4 | 14 | 10 | 10 | "Isosceles" |  |

* 1. algoritmo con la lógica del programa

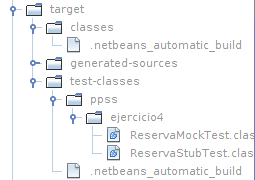
1. compilar y ejecutar los tests
   1. Para poder *ejecutar el código de test*, es imprescindible que al menos el *código* situado en *src/main/java no tenga errores de compilación*, si no, no podremos ejecutar los tests.
   2. Goals
      1. compiler:compile (asociada a la fase compile)
         1. sólo compila el código situado en src/main/java
         2. artefactos generados:

*target/classes/ppss/{Clase}.class*

*target/test-classes/ppss/{ClaseDeTest}.class* *(se supone que no debería compilar las clases en src/main/test)*

* + 1. compiler:testCompile (asociada a la fase test-compile)
       1. sólo compila el código de nuestros tests (situado en src/test/java)
       2. artefactos generados:

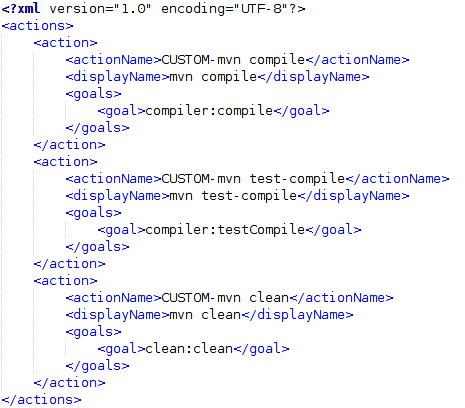
*target/test-classes/ppss/{ClaseDeTest}.class*



* + 1. clean:clean (asociada a la fase clean)
       1. limpia la construcción del proyecto *(borra el directorio target/)*
    2. test (asociada a la fase test)
       1. ejecuta los tests
  1. Fases
     1. *fase compile es anterior a la test-compile*
        1. *desde el terminal (desde el directorio que contiene el fichero pom.xml del proyecto)*
        2. sólo compilar los fuentes de src/main/java ejecutamos: **mvn compile**
        3. compilar fuentes y el código de tests ejecutamos: **mvn test-compile**
  2. Actions
     1. creación: Run Maven
     2. almacenamiento: nbactions.xml
     3. ejemplos (cmd: goal):
        1. mvn compile:compile
        2. mvn test-compile:test-compile
        3. mvn clean:clean
        4. mvn test: test
           1. resultado de la construcción:

Build Failure: el proceso de construcción (mvn test) no se ha completado con éxito puesto que se han detectado problemas en la ejecución de alguna de las fases del proceso

Build Success: proceso de construcción concluido con éxito



1. Matricula.java
   1. Implementar 5 tests "de forma libre"
   2. matriculaTest.java

**Tabla de casos de prueba**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Identificador del Caso de Prueba | Datos de Entrada | | | Resultado esperado | Resultado real |
| Edad | Familia Numerosa | Repetidor |
| C1 |  |  |  |  |  |
| C2 |  |  |  |  |  |
| C3 |  |  |  |  |  |
| C4 |  |  |  |  |  |
| C5 |  |  |  |  |  |

* 1. Pregunta a tus compañeros Xd

1. DataArray.java
   1. implementar tests
   2. Casos de prueba para un método delete()

**Tabla de casos de prueba**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Identificador del Caso de Prueba | Datos de Entrada | | | Resultado esperado | Resultado real |
| Edad | Familia Numerosa | Repetidor |
| C1 |  |  |  |  |  |
| C2 |  |  |  |  |  |
| C3 |  |  |  |  |  |
| C4 |  |  |  |  |  |
| C5 |  |  |  |  |  |

1. implementar método delete()
2. implementar primer test de la tabla de casos de prueba para el delete()
3. ejecutar los tests implementados ÚNICAMENTE en el archivo DataArrayTest
   1. IDE:
      1. en el menú contextual de la clase DataArrayTest, selecciona Test File
   2. Maven:
      1. mvn -Dtest=ppss.DataArrayTest test-compile surefire:test
      2. mvn test-compile + mvn -Dtest=ppss.DataArrayTest surefire:test