

LABORATORIO NO.2

ELABORADO POR:
MIGUEL ÁNGEL SALAMANCA
JUAN CAMILO BAZURTO

PRESENTADO A:
PROF. IVAN VIASUS QUINTERO

ESCUELA COLOMBIANA DE INGENIERÍA JULIO GARAVITO
CICLOS DE VIDA DE SOFTWARE
PROGRAMA DE INGENIERÍA DE SISTEMAS
BOGOTÁ D. C.

2022-2

Ingresar a la página de la herramienta y entender:

- Cuál es su mayor utilidad

Facilitar la gestión y construcción de software, posee la capacidad de realizar ciertas tareas claramente definidas, como la compilación del código y su empaquetado. Es decir, hace posible la creación de software con dependencias.

- Fases de maven
 - Validate
 - Compile
 - Test
 - Package
 - Verify
 - Install
 - Deploy
- Ciclo de vida de la construcción
 - Default
 - Clean
 - Site
- Para qué sirven los plugins
 - Los plugins sirven para añadir funcionalidades extra a un programa sin alterarlo.

Los plugins de Maven son usados para:

- Crear archivos jar
 - Crear archivos war
 - Compilar archivos de código
 - Pruebas unitarias
 - Crear la documentación de proyectos
- Qué es y para qué sirve el repositorio central de maven
 - El repositorio central de maven es un repositorio creado por la comunidad y este cumple la función de tener de manera centralizada las librerías comúnmente usadas.

EJERCICIO DE LAS FIGURAS

CREAR UN PROYECTO CON MAVEN

- Buscar cómo se crea un proyecto maven con ayuda de los arquetipos (archetypes).
- Busque cómo ejecutar desde línea de comandos el objetivo "generate" del plugin "archetype", con los siguientes parámetros:
 - Grupo: edu.eci.cvds
 - Id del Artefacto: Patterns
 - Paquete: edu.eci.cvds.patterns
 - archetypeArtifactId: maven-archetype-quickstart

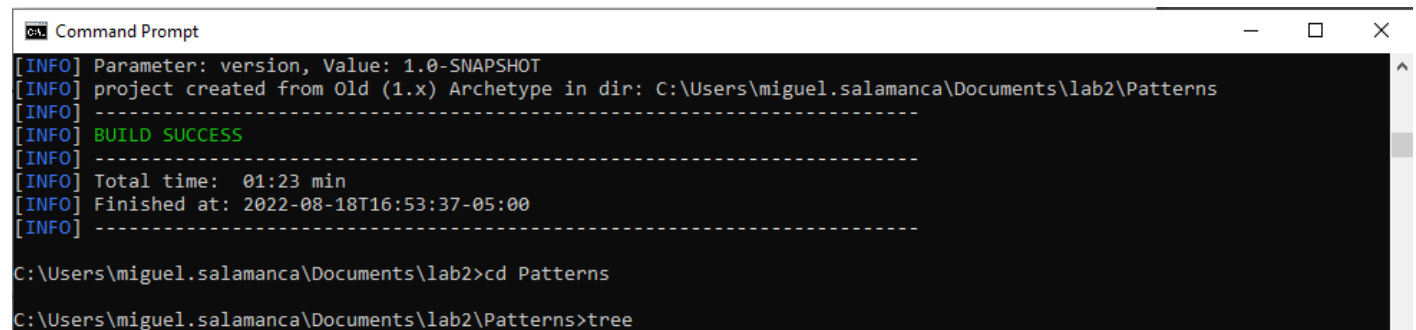
Creamos el proyecto con la ayuda de este arquetipo:

```
mvn archetype:generate -DgroupId=edu.eci.cvds -DartifactId=Patterns -
DarchetypeArtifactId=maven-archetype-quickstart -Dpackage=edu.eci.cvds.patterns
```

C:\Users\miguel.salamanca\Documents\lab2>S

Se debió haber creado en el directorio, un nuevo proyecto Patterns a partir de un modelo o arquetipo, que crea un conjunto de directorios con un conjunto de archivos básicos.

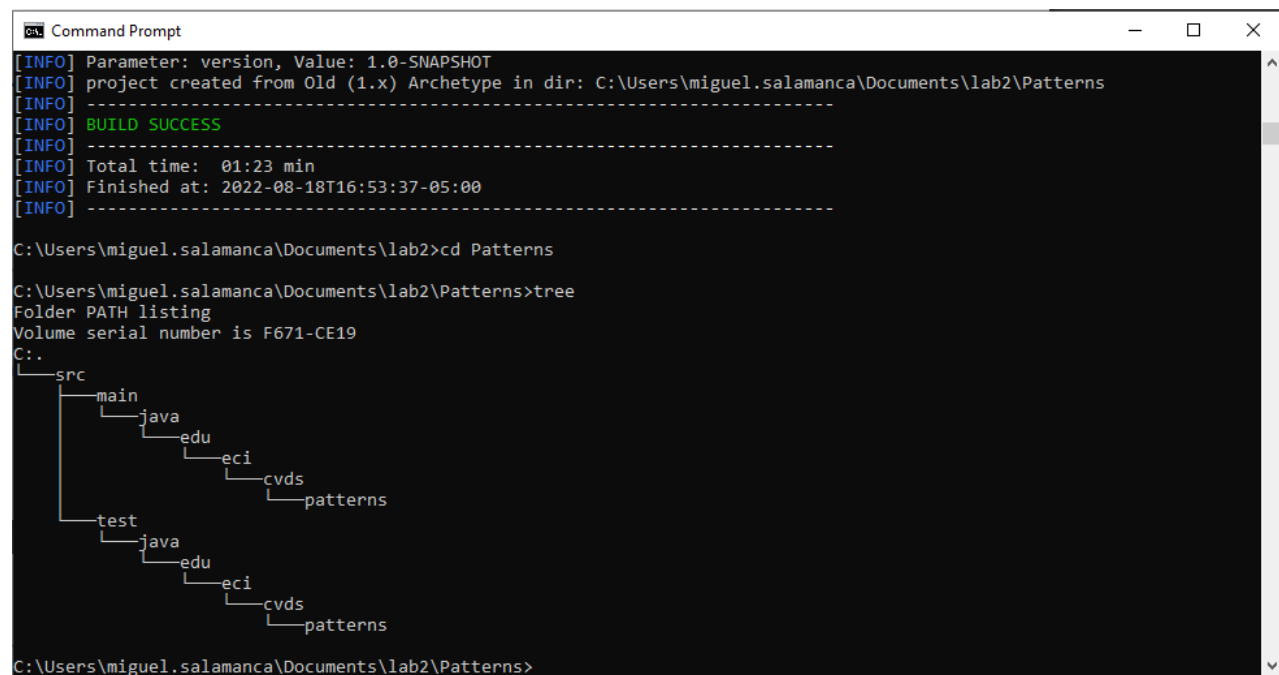
Cambie al directorio Patterns:



```
Command Prompt
[INFO] Parameter: version, Value: 1.0-SNAPSHOT
[INFO] project created from Old (1.x) Archetype in dir: C:\Users\miguel.salamanca\Documents\lab2\Patterns
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 01:23 min
[INFO] Finished at: 2022-08-18T16:53:37-05:00
[INFO] -----

C:\Users\miguel.salamanca\Documents\lab2>cd Patterns
C:\Users\miguel.salamanca\Documents\lab2\Patterns>tree
```

Para ver el conjunto de archivos y directorios creados por el comando mvn ejecute el comando tree.



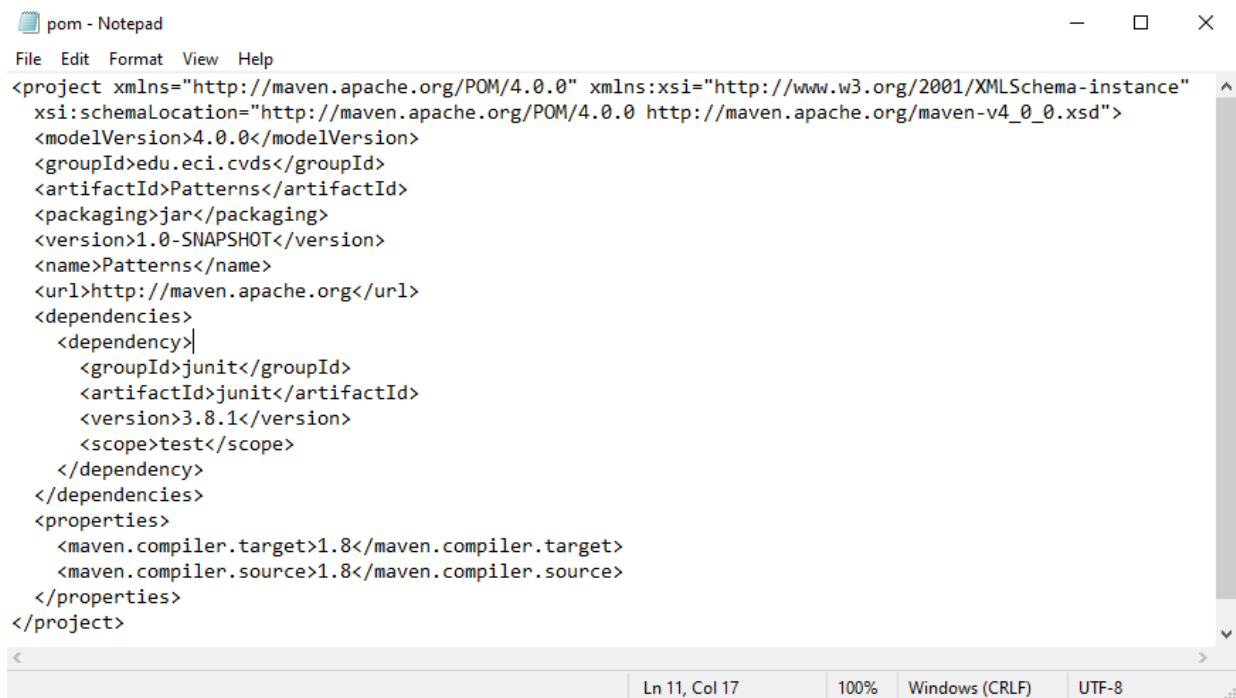
```
Command Prompt
[INFO] Parameter: version, Value: 1.0-SNAPSHOT
[INFO] project created from Old (1.x) Archetype in dir: C:\Users\miguel.salamanca\Documents\lab2\Patterns
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 01:23 min
[INFO] Finished at: 2022-08-18T16:53:37-05:00
[INFO] -----

C:\Users\miguel.salamanca\Documents\lab2>cd Patterns
C:\Users\miguel.salamanca\Documents\lab2\Patterns>tree
Folder PATH listing
Volume serial number is F671-CE19
C:.
├── src
│   ├── main
│   │   ├── java
│   │   │   ├── edu
│   │   │   │   ├── eci
│   │   │   │   │   ├── cvds
│   │   │   │   │   │   └── patterns
│   │   └── test
│   │       ├── java
│   │       │   ├── edu
│   │       │   │   ├── eci
│   │       │   │   │   ├── cvds
│   │       │   │   │   │   └── patterns
└──
```

AJUSTAR ALGUNAS CONFIGURACIONES EN EL PROYECTO

- Edite el archivo pom.xml y realice la siguiente actualización:

- Hay que cambiar la version del compilador de Java a la versión 8, para ello, agregue la sección properties antes de la sección de dependencias:



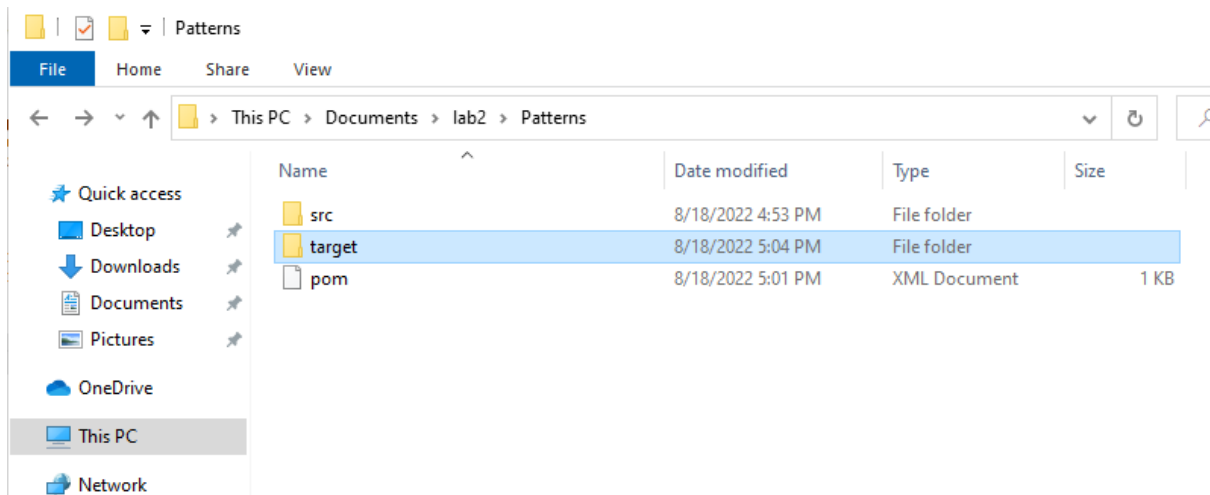
```

pom - Notepad
File Edit Format View Help
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/maven-v4_0_0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <groupId>edu.eci.cvds</groupId>
  <artifactId>Patterns</artifactId>
  <packaging>jar</packaging>
  <version>1.0-SNAPSHOT</version>
  <name>Patterns</name>
  <url>http://maven.apache.org</url>
  <dependencies>
    <dependency>
      <groupId>junit</groupId>
      <artifactId>junit</artifactId>
      <version>3.8.1</version>
      <scope>test</scope>
    </dependency>
  </dependencies>
  <properties>
    <maven.compiler.target>1.8</maven.compiler.target>
    <maven.compiler.source>1.8</maven.compiler.source>
  </properties>
</project>
Ln 11, Col 17 100% Windows (CRLF) UTF-8

```

COMPILAR Y EJECUTAR

- Para compilar ejecute el comando:



- Busque cuál es el objetivo del parámetro "package" y qué otros parámetros se podrían enviar al comando mvn.
 - mvn package: este comando es usado para ejecutar todas las fases de maven hasta la fase de package.
 - mvn -version

- mvn clean install
 - mvn compile
 - mvn test
 - mvn deploy
 - mvn site
- Busque cómo ejecutar desde línea de comandos, un proyecto maven y verifique la salida cuando se ejecuta con la clase App.java como parámetro en "mainClass". Tip:
<https://www.mojohaus.org/exec-maven-plugin/usage.html>

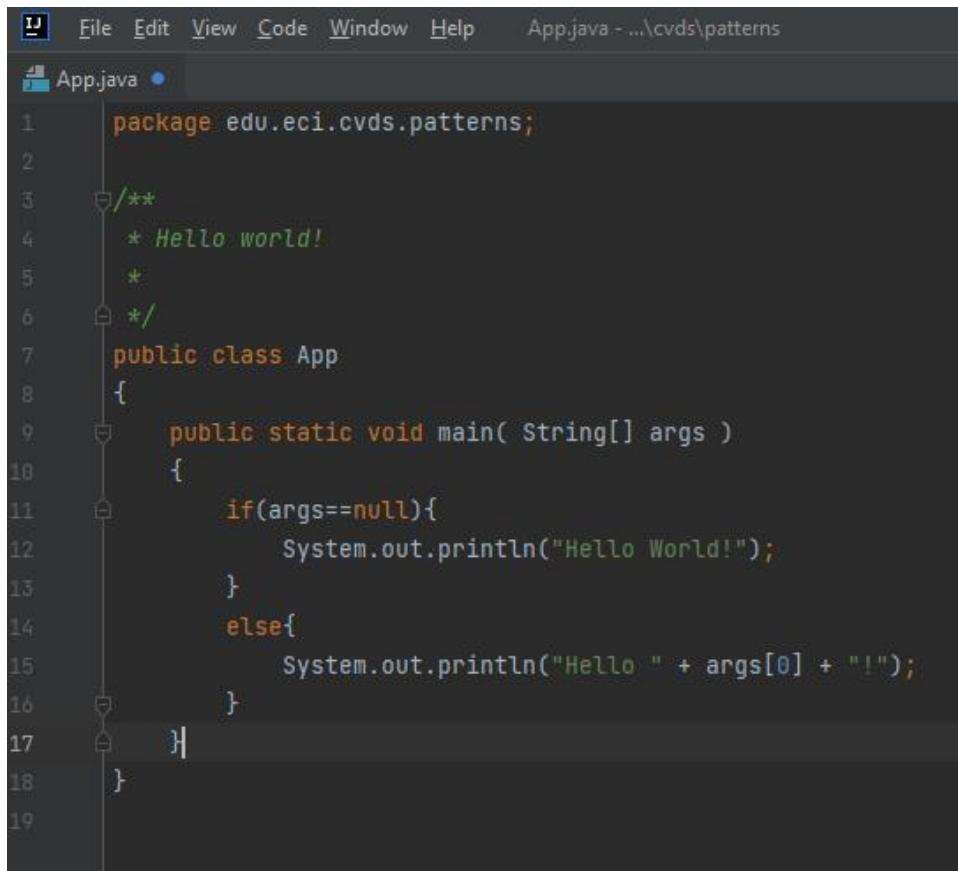
Configuramos este plugin en el pom para la ejecución:

```
<build>
  <plugins>
    <plugin>
      <groupId>org.codehaus.mojo</groupId>
      <artifactId>exec-maven-plugin</artifactId>
      <version>3.1.0</version>
      <configuration>
        <mainClass>edu.eci.cvds.patterns.App</mainClass>
      </configuration>
    </plugin>
  </plugins>
</build>
```

mvn exec:java

```
C:\Users\miguel.salamanca\Documents\lab2\Patterns>mvn exec:java
[INFO] Scanning for projects...
[INFO]
[INFO] -----< edu.eci.cvds:Patterns >-----
[INFO] Building Patterns 1.0-SNAPSHOT
[INFO] -----[ jar ]-----
[INFO]
[INFO] --- exec-maven-plugin:3.1.0:java (default-cli) @ Patterns ---
Hello World!
[INFO]
[INFO] BUILD SUCCESS
[INFO]
[INFO] Total time: 0.281 s
[INFO] Finished at: 2022-08-18T17:29:32-05:00
[INFO]
C:\Users\miguel.salamanca\Documents\lab2\Patterns>
```

- Realice el cambio en la clase App.java para crear un saludo personalizado, basado en los parámetros de entrada a la aplicación. Utilizar la primera posición del parámetro que llega al método "main" para realizar el saludo personalizado, en caso de que no sea posible, se debe mantener el saludo como se encuentra actualmente:

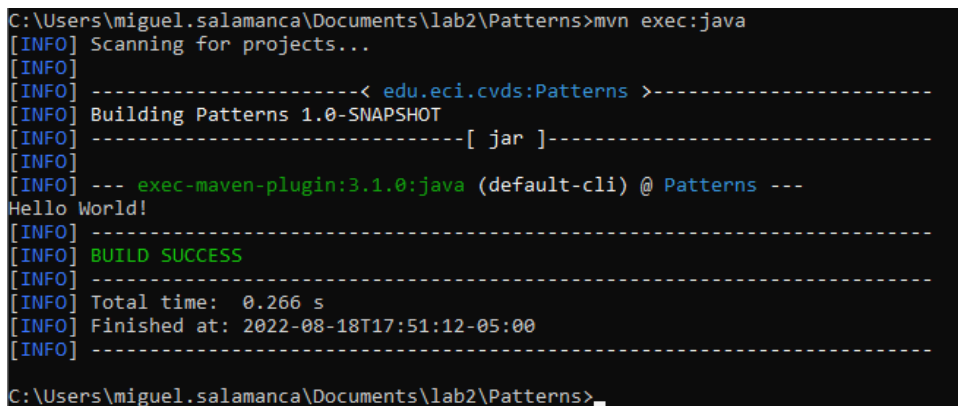


```
1 package edu.eci.cvds.patterns;
2
3 /**
4  * Hello world!
5  *
6  */
7 public class App
8 {
9     public static void main( String[] args )
10    {
11        if(args==null){
12            System.out.println("Hello World!");
13        }
14        else{
15            System.out.println("Hello " + args[0] + "!");
16        }
17    }
18 }
19
```

- Buscar cómo enviar parámetros al plugin "exec".

```
mvn exec:java -Dexec.mainClass="com.example.Main" [-Dexec.args="argument1"] ...
```

- Ejecutar nuevamente la clase desde línea de comandos y verificar la salida: Hello World!



```
C:\Users\miguel.salamanca\Documents\lab2\Patterns>mvn exec:java
[INFO] Scanning for projects...
[INFO] -----< edu.eci.cvds:Patterns >-----
[INFO] Building Patterns 1.0-SNAPSHOT
[INFO] -----[ jar ]-----
[INFO] --- exec-maven-plugin:3.1.0:java (default-cli) @ Patterns ---
Hello World!
[INFO] BUILD SUCCESS
[INFO] Total time: 0.266 s
[INFO] Finished at: 2022-08-18T17:51:12-05:00
[INFO] -----
C:\Users\miguel.salamanca\Documents\lab2\Patterns>
```

- Ejecutar la clase desde línea de comandos enviando su nombre como parámetro y verificar la salida. Ej: Hello Pepito!

```

C:\Users\miguel.salamanca\Documents\lab2\Patterns>mvn exec:java -Dexec.args="miguel"
[INFO] Scanning for projects...
[INFO]
[INFO] -----< edu.eci.cvds:Patterns >-----
[INFO] Building Patterns 1.0-SNAPSHOT
[INFO] -----[ jar ]-----
[INFO]
[INFO] --- exec-maven-plugin:3.1.0:java (default-cli) @ Patterns ---
[INFO] [Ljava.lang.String;@57ffc1e3
Hello miguel!
[INFO]
[INFO] BUILD SUCCESS
[INFO]
[INFO] Total time: 0.266 s
[INFO] Finished at: 2022-08-18T17:59:06-05:00
[INFO]
C:\Users\miguel.salamanca\Documents\lab2\Patterns>

```

- Ejecutar la clase con su nombre y apellido como parámetro. ¿Qué sucedió?

```

C:\Users\miguel.salamanca\Documents\lab2\Patterns>mvn exec:java -Dexec.args="miguel" -Dexec.args="salamanca"
[INFO] Scanning for projects...
[INFO]
[INFO] -----< edu.eci.cvds:Patterns >-----
[INFO] Building Patterns 1.0-SNAPSHOT
[INFO] -----[ jar ]-----
[INFO]
[INFO] --- exec-maven-plugin:3.1.0:java (default-cli) @ Patterns ---
[INFO] Hello salamanca!
[INFO]
[INFO] BUILD SUCCESS
[INFO]
[INFO] Total time: 0.267 s
[INFO] Finished at: 2022-08-18T18:02:23-05:00
[INFO]

```

Solo tomó el último parámetro

- Verifique cómo enviar los parámetros de forma "compuesta" para que el saludo se realice con nombre y apellido.

```

<configuration>
  <mainClass>com.mavericks.App</mainClass>
  <arguments>
    <argument>name</argument>
    <argument>sname</argument>
  </arguments>
</configuration>

```

```
mvn exec:java -Dexec.args="argument1 argument2"
```

- Ejecutar nuevamente y verificar la salida en consola. Ej: Hello Pepito Perez!

```
Command Prompt
T E S T S
-----
Running edu.eci.cvds.patterns.AppTest
Tests run: 1, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.007 sec
Results :
Tests run: 1, Failures: 0, Errors: 0, Skipped: 0

[INFO] --- maven-jar-plugin:2.4:jar (default-jar) @ Patterns ---
[INFO] Building jar: C:\Users\miguel.salamanca\Documents\lab2\Patterns\target\Patterns-1.0-SNAPSHOT.jar
[INFO] BUILD SUCCESS
[INFO] Total time: 1.854 s
[INFO] Finished at: 2022-08-18T18:21:41-05:00
[INFO]
C:\Users\miguel.salamanca\Documents\lab2\Patterns>mvn exec:java -Dexec.args="miguel salamanca"
[INFO] Scanning for projects...
[INFO]
[INFO] -----< edu.eci.cvds:Patterns >-----
[INFO] Building Patterns 1.0-SNAPSHOT
[INFO] [ jar ]-----
[INFO]
[INFO] --- exec-maven-plugin:3.1.0:java (default-cli) @ Patterns ---
Hello miguel salamanca!
[INFO] BUILD SUCCESS
[INFO] Total time: 0.278 s
[INFO] Finished at: 2022-08-18T18:21:54-05:00
[INFO]
C:\Users\miguel.salamanca\Documents\lab2\Patterns>
```

HACER EL ESQUELETO DE LA APLICACION

- Cree el paquete edu.eci.cvds.patterns.shapes y el paquete edu.eci.cvds.patterns.shapes.concrete.
- Cree una interfaz llamada Shape.java en el directorio src/main/java/edu/eci/cvds/patterns/shapes de la siguiente manera:
- Cree una enumeración llamada RegularShapeType.java en el directorio src/main/java/edu/eci/cvds/patterns/shapes así:
- En el directorio src/main/java/edu/eci/cvds/patterns/shapes/concrete cree las diferentes clases (Triangle, Quadrilateral, Pentagon, Hexagon), que implementen la interfaz creada y retornen el número correspondiente de vértices que tiene la figura. Siguiendo el ejemplo del triángulo:
- Cree el archivo ShapeMain.java en el directorio src/main/java/edu/eci/cvds/patterns/shapes con el metodo main:
- Analice y asegúrese de entender cada una de las instrucciones que se encuentran en todas las clases que se crearon anteriormente.
- Cree el archivo ShapeFactory.java en el directorio src/main/java/edu/eci/cvds/patterns/shapes implementando el patrón fábrica, haciendo uso de la instrucción switch-case de Java y usando las enumeraciones.


```
App.java x ShapeMain.java x RegularShapeType.java x ShapeFactory.java x Shape.java x
1 package edu.eci.cvds.patterns.shapes;
2 import edu.eci.cvds.patterns.shapes.concrete.*;
3
4 public class ShapeFactory{
5     public static Shape create(RegularShapeType type)
6     {
7         Shape shape;
8         if (type == null) {
9             return null;
10        }
11        switch (type) {
12            case Triangle:
13                shape = new Triangle();
14                return shape;
15            case Quadrilateral:
16                shape = new Quadrilateral();
17                return shape;
18            case Hexagon:
19                shape = new Pentagon();
20                return shape;
21            case Pentagon:
22                shape = new Hexagon();
23                return shape;
24            default:
25                throw new IllegalArgumentException("Unknown type "+type);
26        }
27    }
28 }
```

- Ejecute múltiples veces la clase ShapeMain, usando el plugin exec de maven con los siguientes parámetros y verifique la salida en consola para cada una:
 - Sin parámetros
 - Parámetro: qwerty
 - Parámetro: pentagon
 - Parámetro Hexagon
- ¿Cuál(es) de las anteriores instrucciones se ejecutan y funcionan correctamente y por qué?
 - Prueba sin parametros, nos señala que se requiere un parámetro tipo RegularShapeType.

```
C:\Users\camil\Documents\2022-2\CVDS\lab2\Patterns>mvn exec:java -Dexec.args=""
[INFO] Scanning for projects...
[INFO]
[INFO] -----< edu.eci.cvds:Patterns >-----
[INFO] Building Patterns 1.0-SNAPSHOT
[INFO] -----[ jar ]-----
[INFO]
[INFO] --- exec-maven-plugin:3.1.0:java (default-cli) @ Patterns ---
[INFO] Parameter of type RegularShapeType is required.
[INFO] BUILD SUCCESS
[INFO]
[INFO] Total time: 0.182 s
[INFO] Finished at: 2022-08-22T19:15:38-05:00
[INFO]
C:\Users\camil\Documents\2022-2\CVDS\lab2\Patterns>
```

- Qwerty, nos señala que no es una RegularShapeType valida.

```
C:\Users\camil\Documents\2022-2\CVDS\lab2\Patterns>mvn exec:java -Dexec.args="qwerty"
[INFO] Scanning for projects...
[INFO] Downloading from central: https://repo.maven.apache.org/maven2/org/codehaus/mojo/exec-maven-plugin/3.1.0/exec-maven-plugin-3.1.0.pom (14 kB at 24 kB/s)
[INFO] Downloaded from central: https://repo.maven.apache.org/maven2/org/codehaus/mojo/mojo-parent/69/mojo-parent-69.pom (35 kB at 183 kB/s)
[INFO] Downloading from central: https://repo.maven.apache.org/maven2/org/junit/junit-bom/5.8.2/junit-bom-5.8.2.pom (5.6 kB at 37 kB/s)
[INFO] Downloaded from central: https://repo.maven.apache.org/maven2/org/codehaus/mojo/exec-maven-plugin/3.1.0/exec-maven-plugin-3.1.0.jar (73 kB at 387 kB/s)
[INFO]
[INFO] -----< edu.eci.cvds:Patterns >-----
[INFO] Building Patterns 1.0-SNAPSHOT
[INFO] -----[ jar ]-----
[INFO]
[INFO] --- exec-maven-plugin:3.1.0:java (default-cli) @ Patterns ---
[INFO] Downloading from central: https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-utils/3.4.2/plexus-utils-3.4.2.pom (8.2 kB at 42 kB/s)
[INFO] Downloaded from central: https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-utils/3.4.2/plexus-utils-3.4.2.pom (8.2 kB at 42 kB/s)
[INFO] Downloading from central: https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus/8/plexus-8.pom (25 kB at 164 kB/s)
[INFO] Downloaded from central: https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-component-annotations/2.1.1/plexus-component-annotations-2.1.1.pom (770 B at 5.1 kB/s)
[INFO] Downloaded from central: https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-component-annotations/2.1.1/plexus-component-annotations-2.1.1.pom (770 B at 5.1 kB/s)
[INFO] Downloading from central: https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-containers/2.1.1/plexus-containers-2.1.1.pom (6.8 kB at 39 kB/s)
[INFO] Downloaded from central: https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-containers/2.1.1/plexus-containers-2.1.1.pom (6.8 kB at 39 kB/s)
[INFO] Downloading from central: https://repo.maven.apache.org/maven2/org/apache/commons/commons-exec/1.3/commons-exec-1.3.pom (11 kB at 70 kB/s)
[INFO] Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/commons/commons-exec/1.3/commons-exec-1.3.pom (11 kB at 70 kB/s)
[INFO] Downloading from central: https://repo.maven.apache.org/maven2/org/apache/commons/commons-parent/35/commons-parent-35.pom (58 kB at 354 kB/s)
[INFO] Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/commons/commons-parent/35/commons-parent-35.pom (58 kB at 354 kB/s)
[INFO] Downloading from central: https://repo.maven.apache.org/maven2/org/apache/apache/15/apache-15.pom (15 kB at 87 kB/s)
[INFO] Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/apache/15/apache-15.pom (15 kB at 87 kB/s)
[INFO] Downloading from central: https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-utils/3.4.2/plexus-utils-3.4.2.jar (267 kB at 1.1 MB/s)
[INFO] Downloaded from central: https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-utils/3.4.2/plexus-utils-3.4.2.jar (267 kB at 1.1 MB/s)
[INFO] Downloading from central: https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-component-annotations/2.1.1/plexus-component-annotations-2.1.1.jar (4.1 kB at 13 kB/s)
[INFO] Downloaded from central: https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-component-annotations/2.1.1/plexus-component-annotations-2.1.1.jar (4.1 kB at 13 kB/s)
[INFO] Parameter 'qwerty' is not a valid RegularShapeType
[INFO] BUILD SUCCESS
[INFO] Total time: 3.320 s
[INFO] Finished at: 2022-08-22T19:15:00-05:00
[INFO]
```

- pentagon, nos señala que no es una RegularShapeType valida.

```
C:\Users\camil\Documents\2022-2\CVDS\lab2\Patterns>mvn exec:java -Dexec.args="pentagon"
[INFO] Scanning for projects...
[INFO]
[INFO] -----< edu.eci.cvds:Patterns >-----
[INFO] Building Patterns 1.0-SNAPSHOT
[INFO] -----[ jar ]-----
[INFO]
[INFO] --- exec-maven-plugin:3.1.0:java (default-cli) @ Patterns ---
[INFO] Parameter 'pentagon' is not a valid RegularShapeType
[INFO] BUILD SUCCESS
[INFO] Total time: 0.190 s
[INFO] Finished at: 2022-08-22T19:17:52-05:00
[INFO]
```

- Hexagon, nos señala que hemos creado un Hexagon con 5 lados.

```
C:\Users\camil\Documents\2022-2\CVDS\lab2\Patterns>mvn exec:java -Dexec.args="Hexagon"
[INFO] Scanning for projects...
[INFO]
[INFO] -----< edu.eci.cvds:Patterns >-----
[INFO] Building Patterns 1.0-SNAPSHOT
[INFO] -----[ jar ]-----
[INFO]
[INFO] --- exec-maven-plugin:3.1.0:java (default-cli) @ Patterns ---
[INFO] Successfully created a Hexagon with 5 sides.
[INFO] BUILD SUCCESS
[INFO] Total time: 0.186 s
[INFO] Finished at: 2022-08-22T19:18:06-05:00
[INFO]
```

Bibliografía

[¿Qué es Maven y para qué se utiliza? - Panama Hitek](#)

[Maven - Plugins \(tutorialspoint.com\)](#)

[Maven Commands | Learn List of Basic To Advanced Maven Commands \(educba.com\)](#)

[https://gustavopeiretti.com/patron-de-diseno-java-factory/](#)