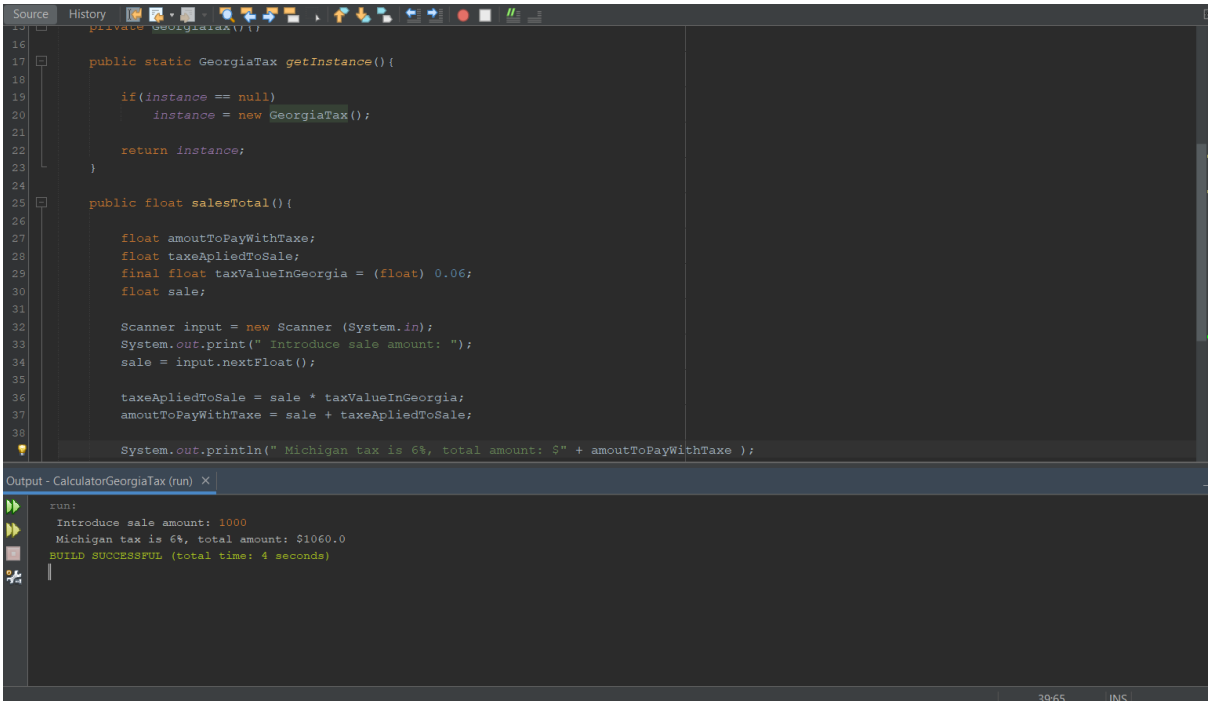
WS34 - SINGLETON PATTERN

DATE: 15th february 2022.

1 ALMACHE LITARDO ANDERSON MOISES

Singleton Pattern is a creational design pattern that allows us to ensure that a class has a single instance, while providing a global access point to that instance.

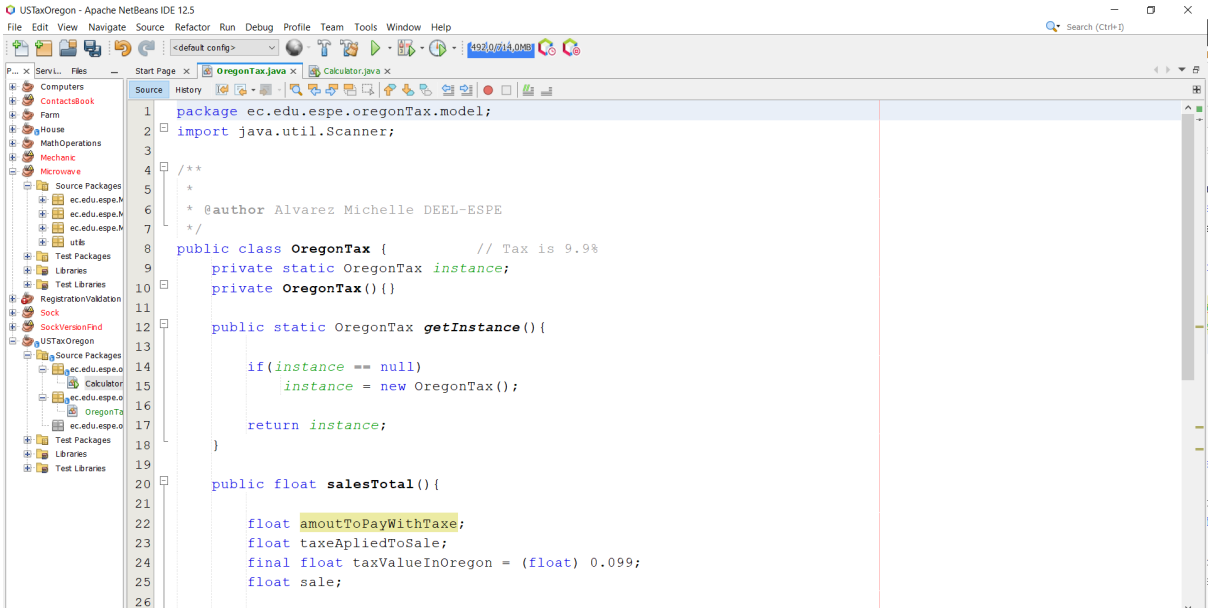


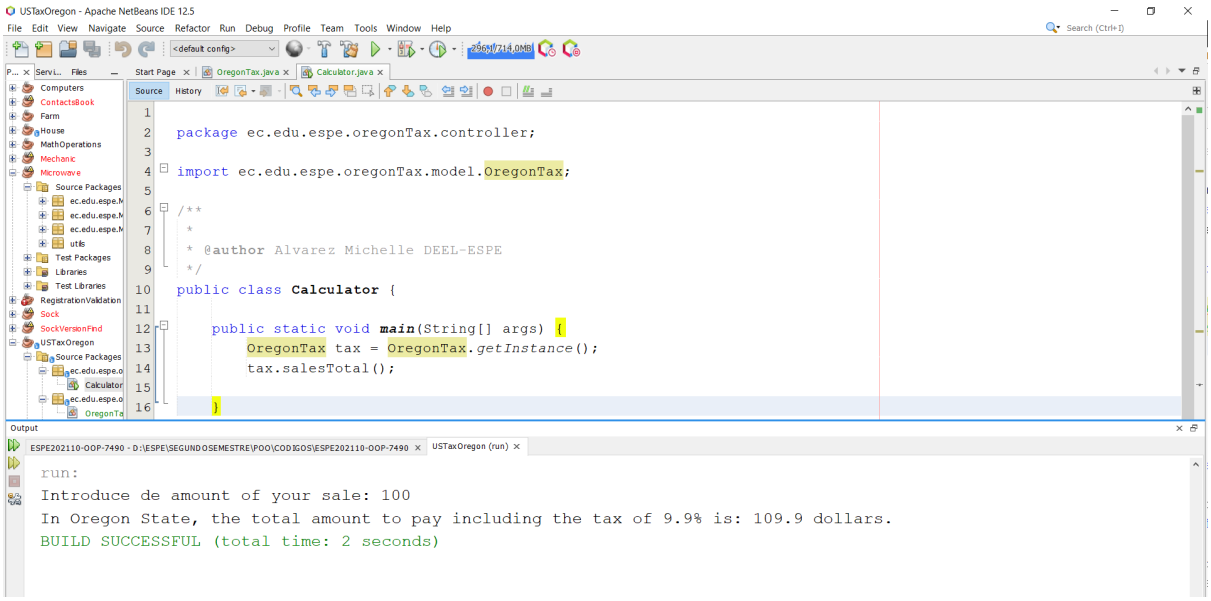
2 ALTAMIRANO BENALCAZAR CRISTHIAN ALEXANDER

The single pattern consists of a class that has only one instance and gets a global access point. There is no more than one object per class, for this the initializer must be the base class and it must have a public method.

3 ALVAREZ RAMIREZ MICHELLE ESTEFANIA

A single Pattern is one of the simplest design patterns in Java. It makes sure that **a class only has one instance**. This class provides a way to access its only object which can be accessed directly without need to instantiate the object of the class.

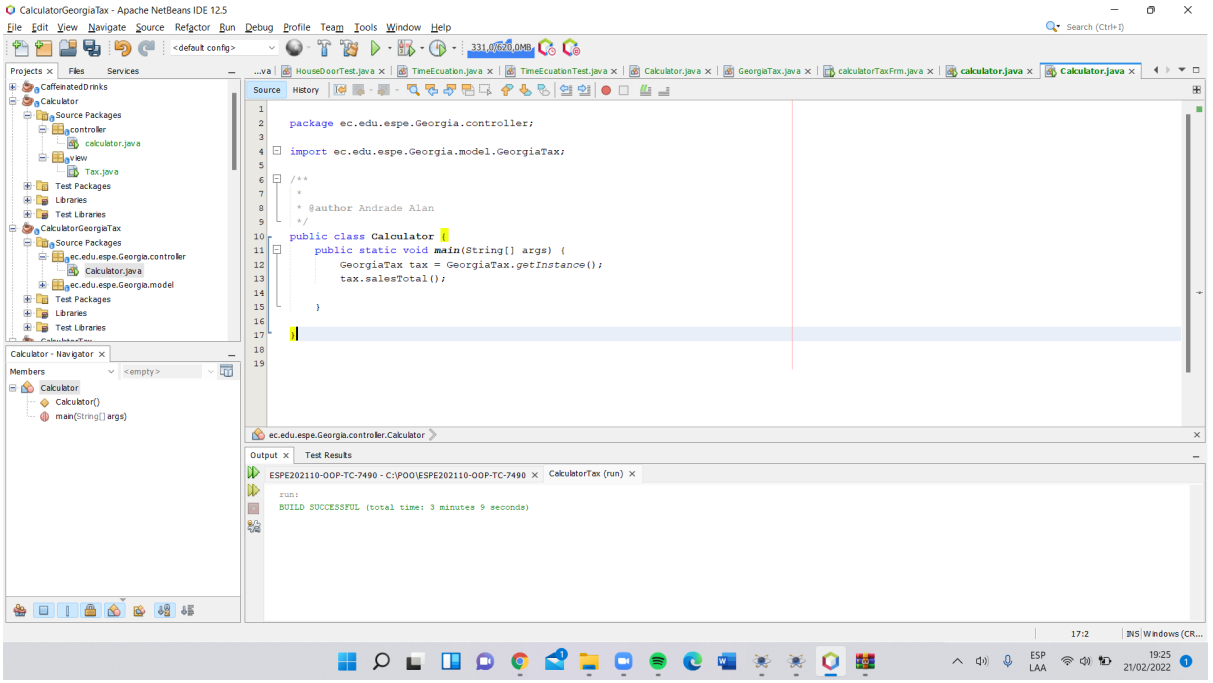


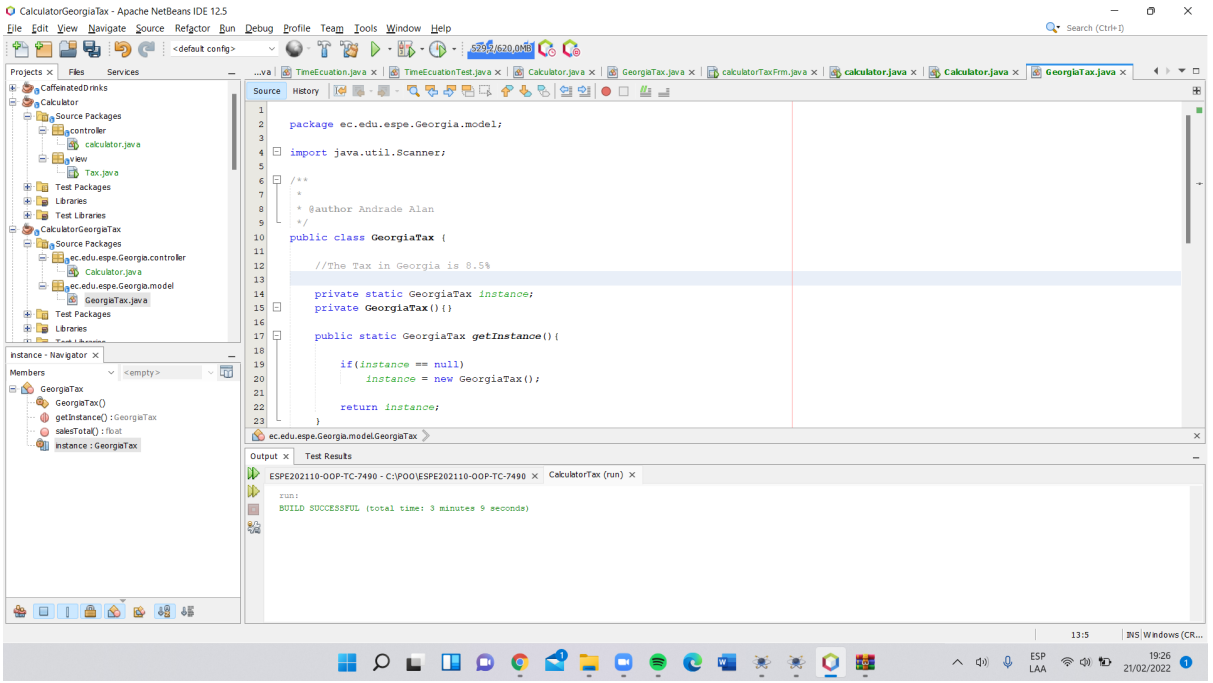


4 ANDRADE CARATE ALAN DAMIAN

The singleton pattern: this pattern helps so that there is only one instance for each class, which means that there is only one access point.

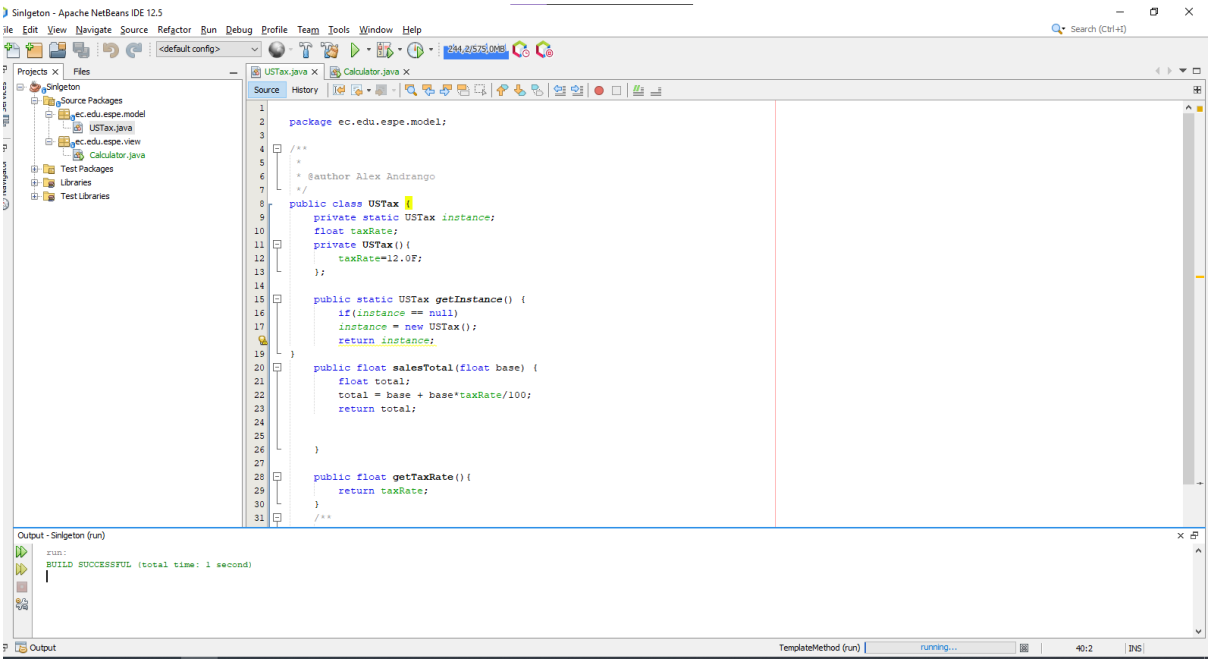
To implement it we must create a method in our class and an instance is created.





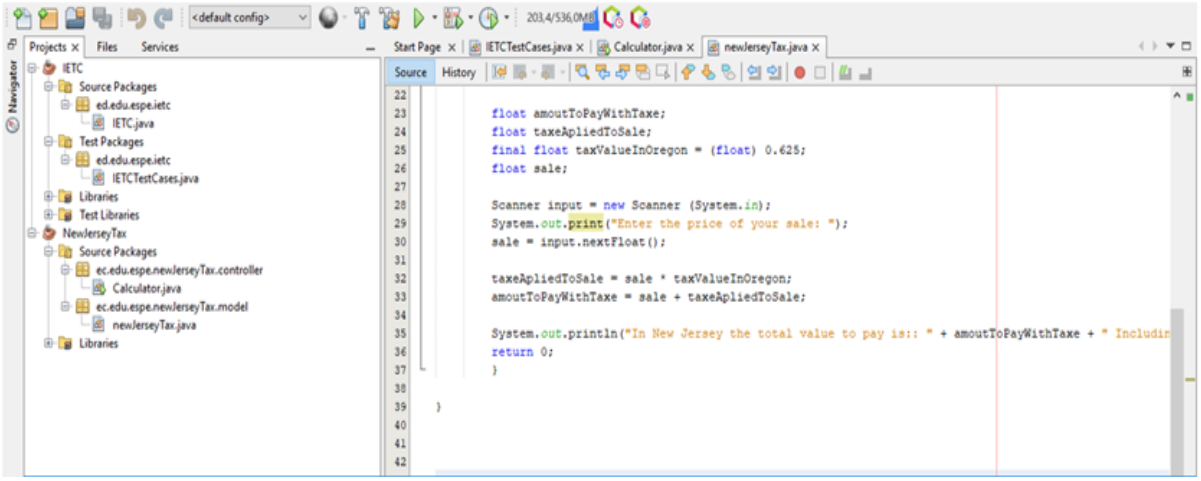
5 ANDRANGO ESPINOSA ALEX PAUL

The singleton pattern is designed to restrict the multiple instantiation of a class to one object. Also the class only has one instance and provides a global point of access to it controlled.



6 ARROBA SOLORZANO CRISTIAN ALEXANDER

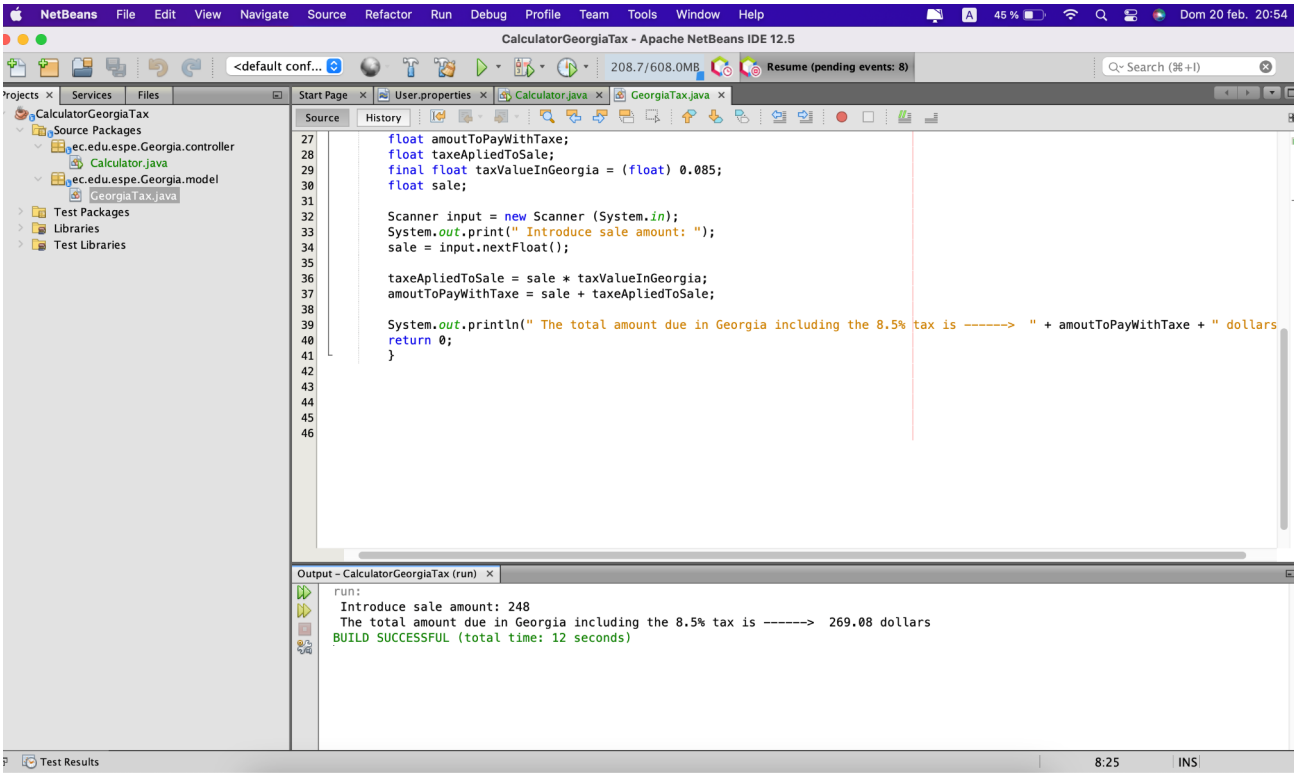
The singleton pattern consists of creating one and only one instance of an object, for our entire application. It would be like a kind of global variable that stores our object. At first this definition may sound very strange. In general, it is always recommended not to use global variables in an application, let alone in object-oriented programming.



7 ASUMAZA GUALOTO DYLAN ALEXANDER

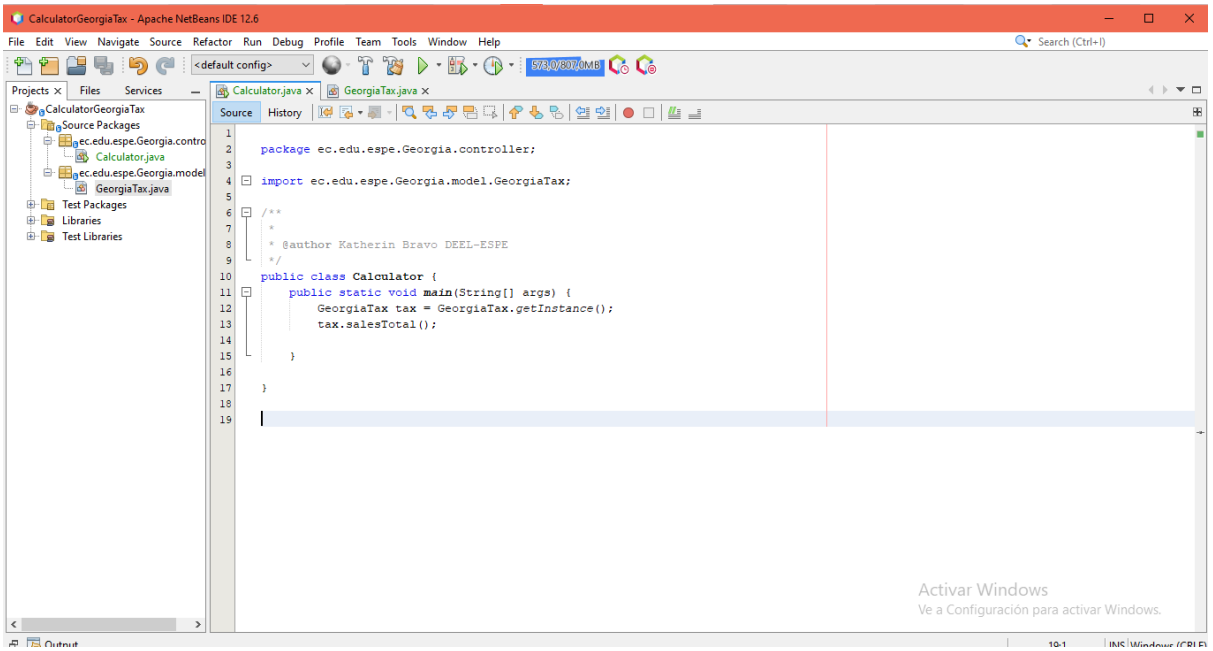
Singleton is a design pattern that allows you to reduce the creation of objects belonging to a class or the value of a type to a single object.

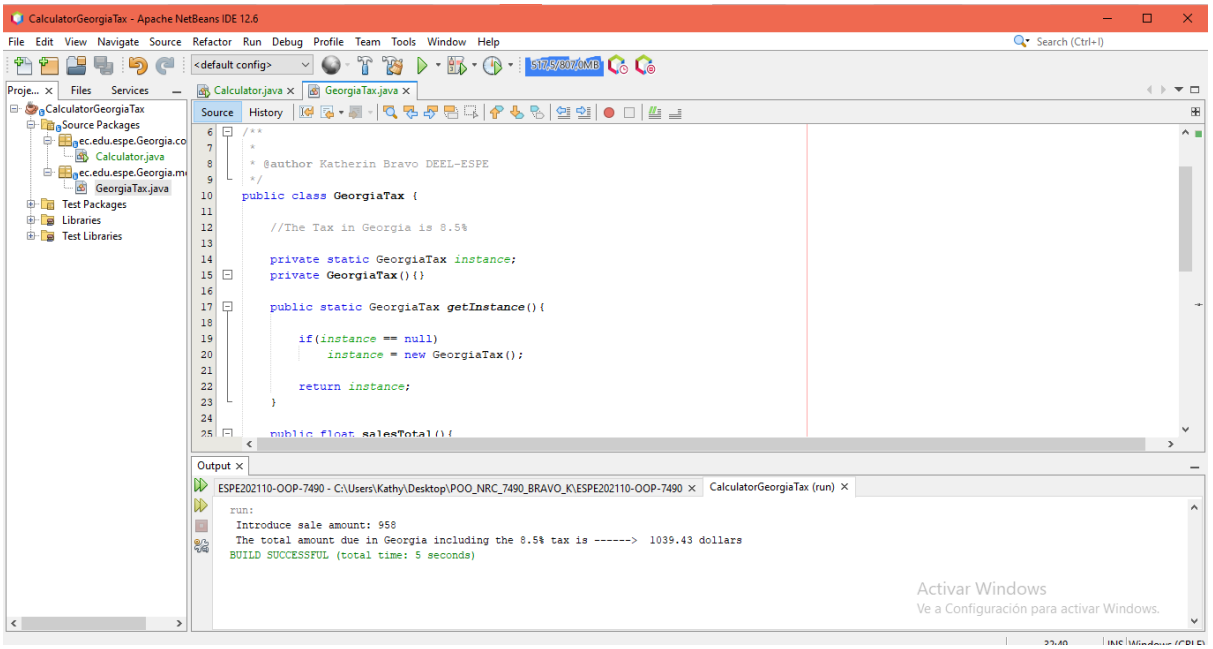
Its intent is to ensure that a class has only one instance and to provide a global access point to it.



8 BRAVO RODRIGUEZ KATHERIN DAYANNE

Singleton Pattern is one of the Gangs of Four Design patterns and it comes under the Creational Design Pattern category, it seems to be a very simple design pattern but when it comes to implementation, it comes with a lot of implementation concerns.
They provide tried and tested templates for solving programming tasks. And it is achieved by creating the desired object in a class and retrieving it as a static instance.



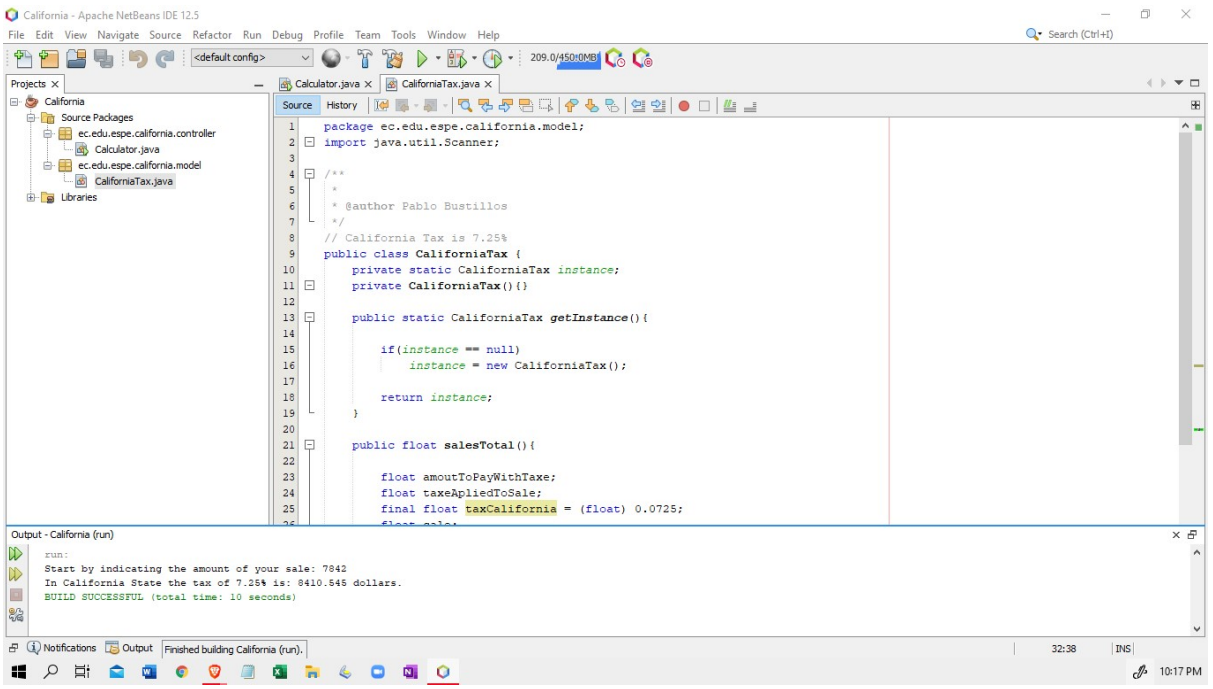


9 BRAVO VILLALOBOS CHRISTIAN DAVID

10 BUSTILLOS MONTENEGRO PABLO SEBASTIAN

Singleton is a creative design pattern that ensures that there is only one object of its kind and provides a single point of access to it for any other code.

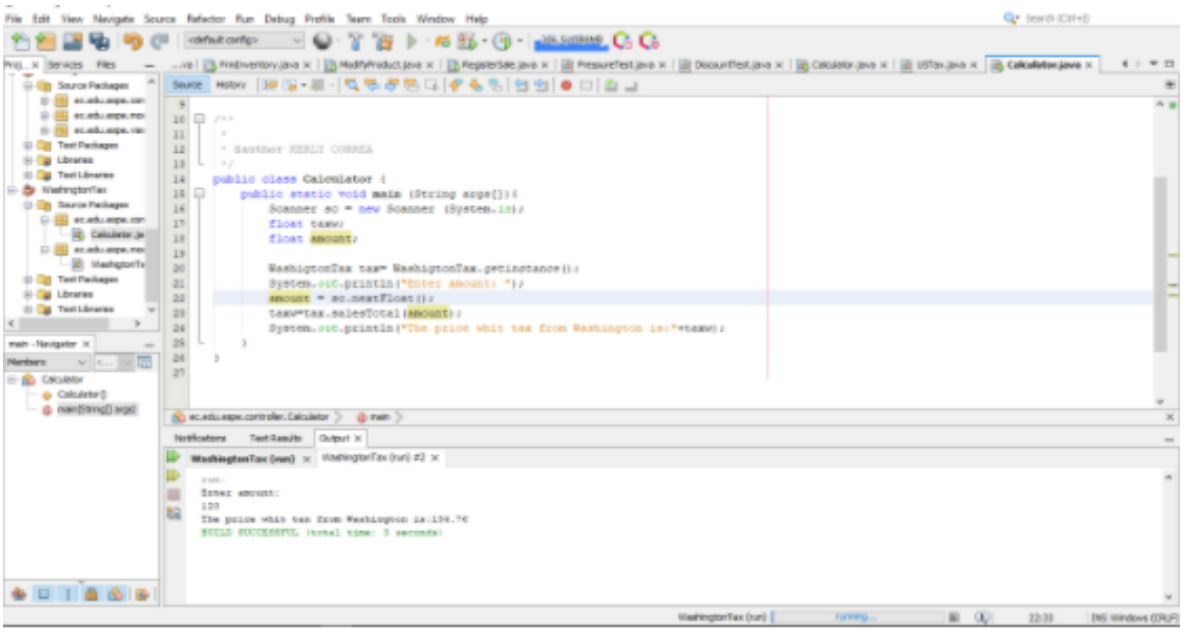
A Singleton only allows for a single instantiation, but many instances of the same object. The Singleton restricts clients from creating multiple objects, after the first object created, it will return instances of itself.



11 CADENA ROMAN BENJAMIN ABEL

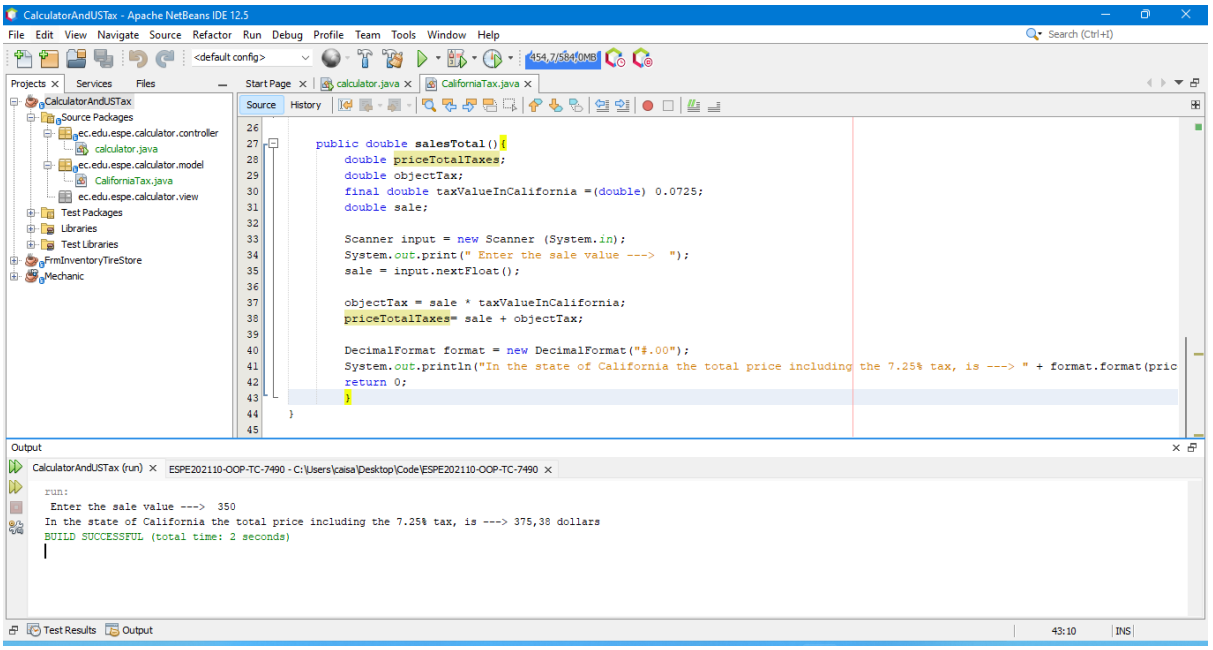
The Singleton design pattern is designed to restrict the creation of objects belonging to a class or the value of a type to a single object. Its intent is to ensure that a class has only one instance and to provide a global access point to it.

This pattern is applicable in systems where you want to be able to guarantee that only one instance of a class exists.



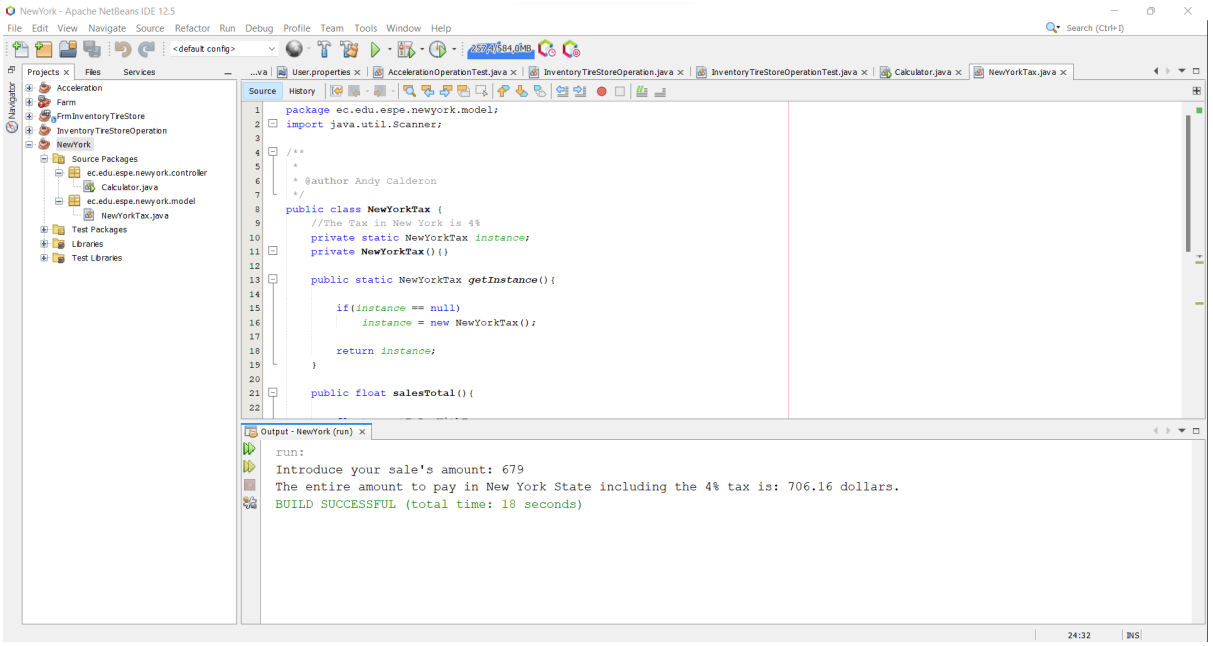
12 CAISATOA RAMIREZ SEBASTIAN BERNARDO

Singleton Pattern is a single instance is a design pattern that allows to restrict the creation of objects belonging to a class or the value of a type to a single object, ensure a class only has one instance, and provides a global point of access to it.



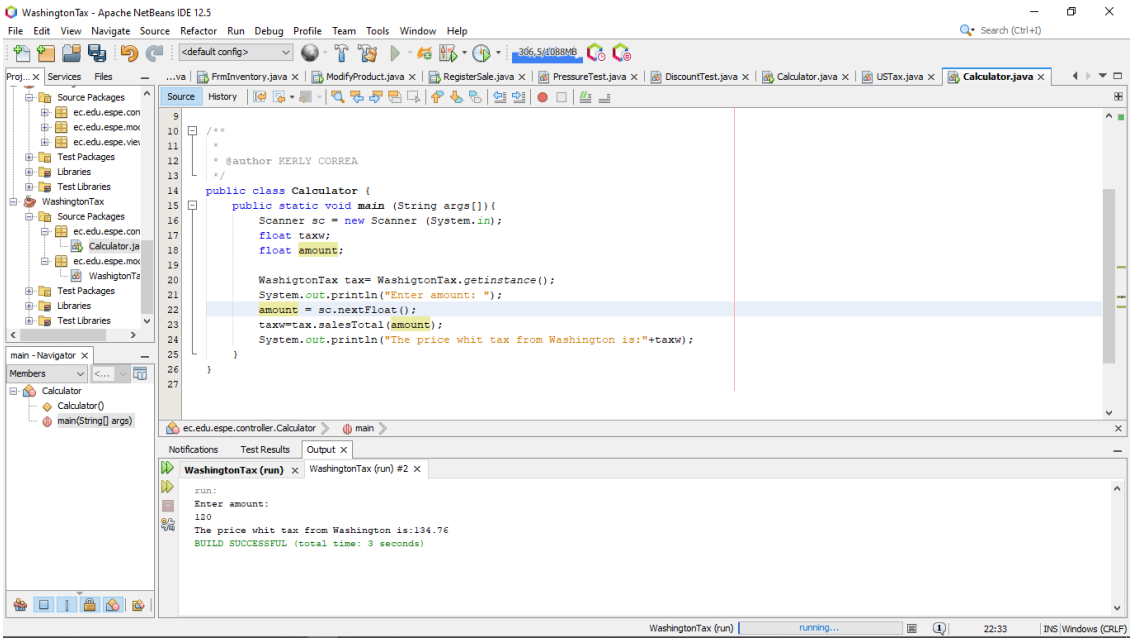
13 CALDERON MERCHAN ANDY JOSUE

Singleton is a creational design pattern that ensures that only one object of its type exists and provides a single point of access to it for all other code. This type of design pattern comes under creational pattern as this pattern provides one of the best ways to create an object.



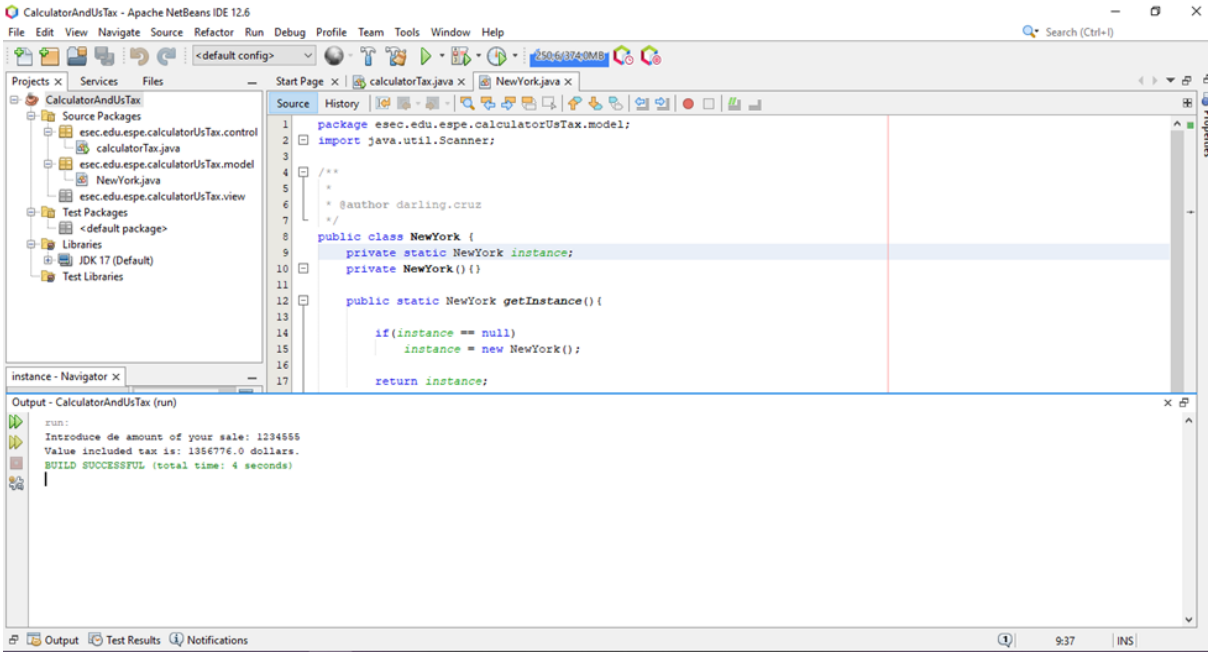
14 CORREA RUIZ KERLY YADIRA

The singleton pattern ensures that a class has a single instance as an encapsulation while providing a single or global access point to it; it allows an object to alter its behavior when its internal state changes and defines a static method of the class.



15 CRUZ PANTOJA DARLING MICAELA

The singleton pattern helps that for each class there is only one instance, this means that there is only one access point, it is implemented by creating a method in our class and an instance is created as long as it did not exist yet.



16 DE LA CRUZ QUINGA ALEJANDRO SEBASTIAN

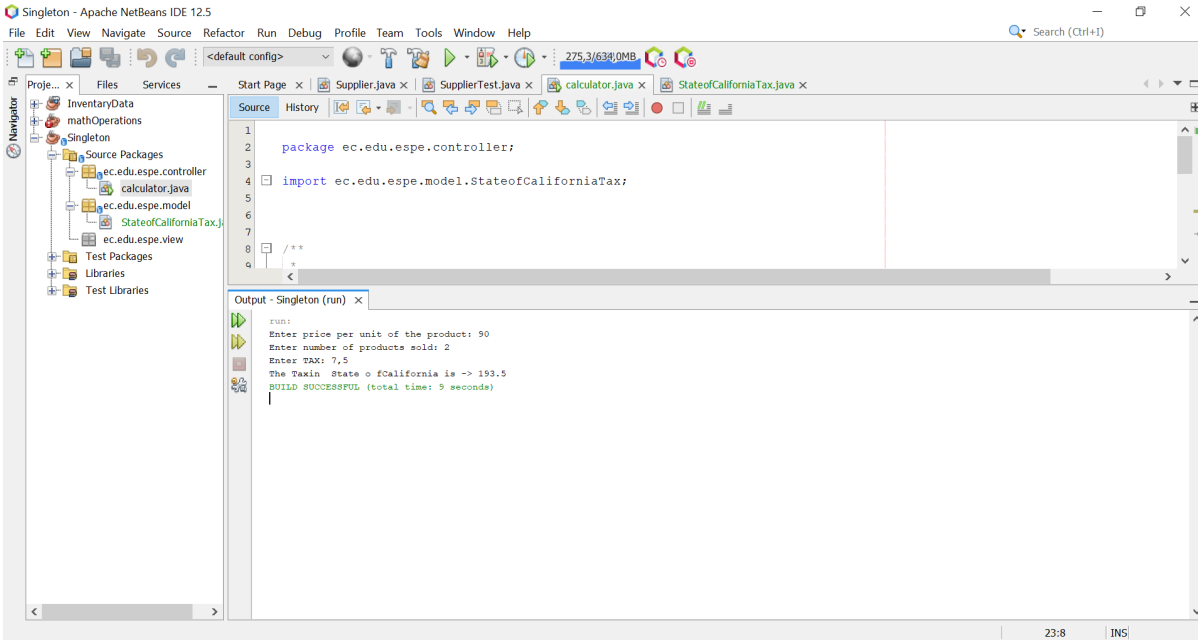
Singleton Pattern es un patrón de creación que garantiza que una clase sólo tenga una instancia y proporciona un punto de acceso global a ella; este patrón rompe el principio de responsabilidad única porque realiza estas dos acciones.

17 EIVAR DAGUA JAIME MAURICIO

The Singleton pattern is one of the simplest designs in Java, it is included in the creation pattern, it has only one class to create an object, which in turn has only one instance.
This class manages that only one instance is created and that the single object can be used.

18 GARCIA BARRETO MAYERLY PRISSILLA

The Singleton includes a class that has only one instance of the problem domain or solution domain viewpoint, and many other elements of the system require entering that instance. In addition, the system requires maintaining control of when the instance is made first or the instance needs to be extensible. Objects need a single, global access point.



19 GOMEZ DIAZ MELISSA MALAYCA

Singleton pattern is one of the simplest design patterns in Java. This type of design pattern comes under creational pattern as this pattern provides one of the best ways to create an object.

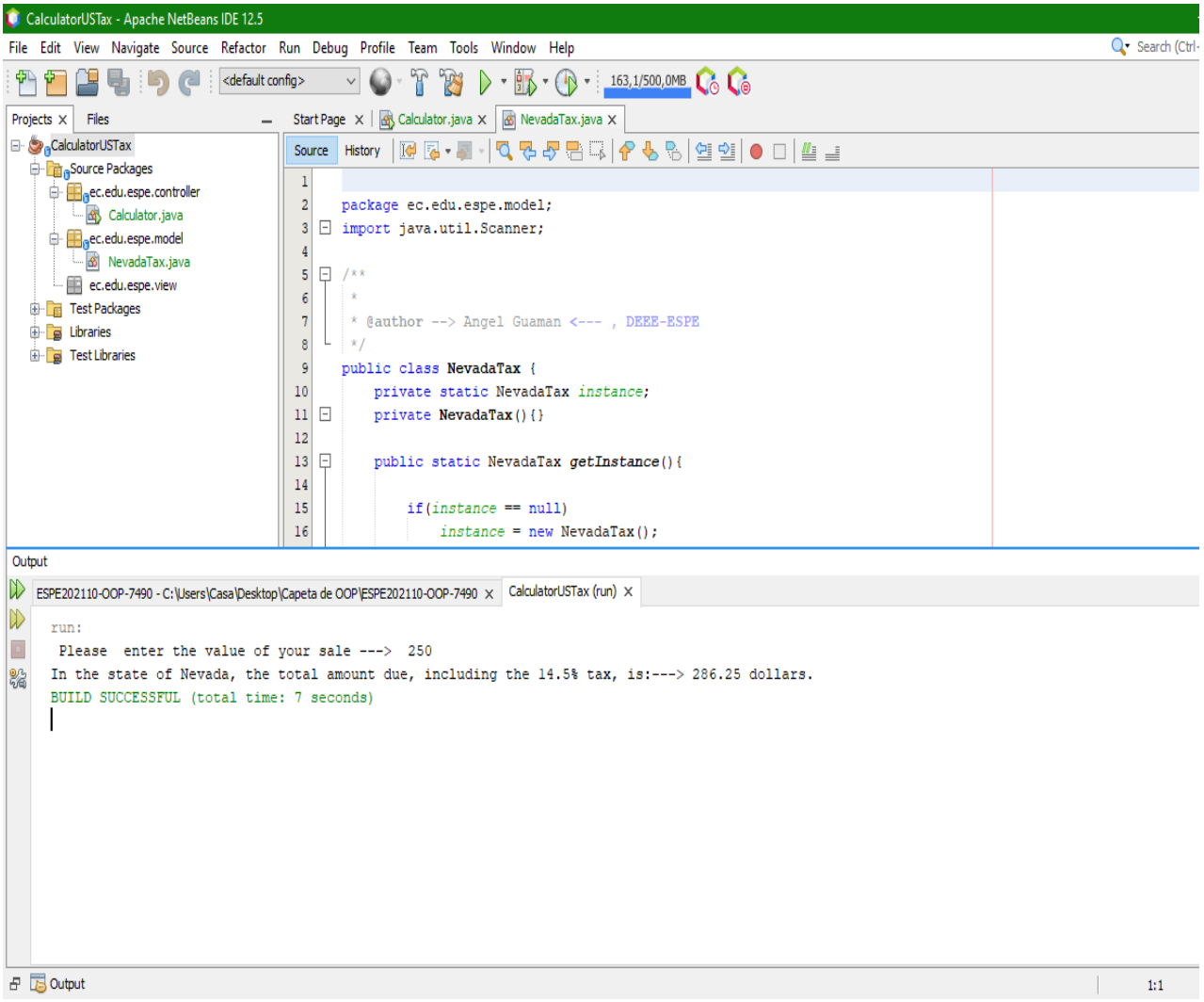
This pattern involves a single class which is responsible to create an object while making sure that only single object gets created. This class provides a way to access its only object which can be accessed directly without need to instantiate the object of the class.

20 GUAMAN VEJARANO ANGEL DAVID

Design Patterns

Design Patterns are working models focused on dividing a problem into parts in order to approach each of them separately to simplify their resolution.

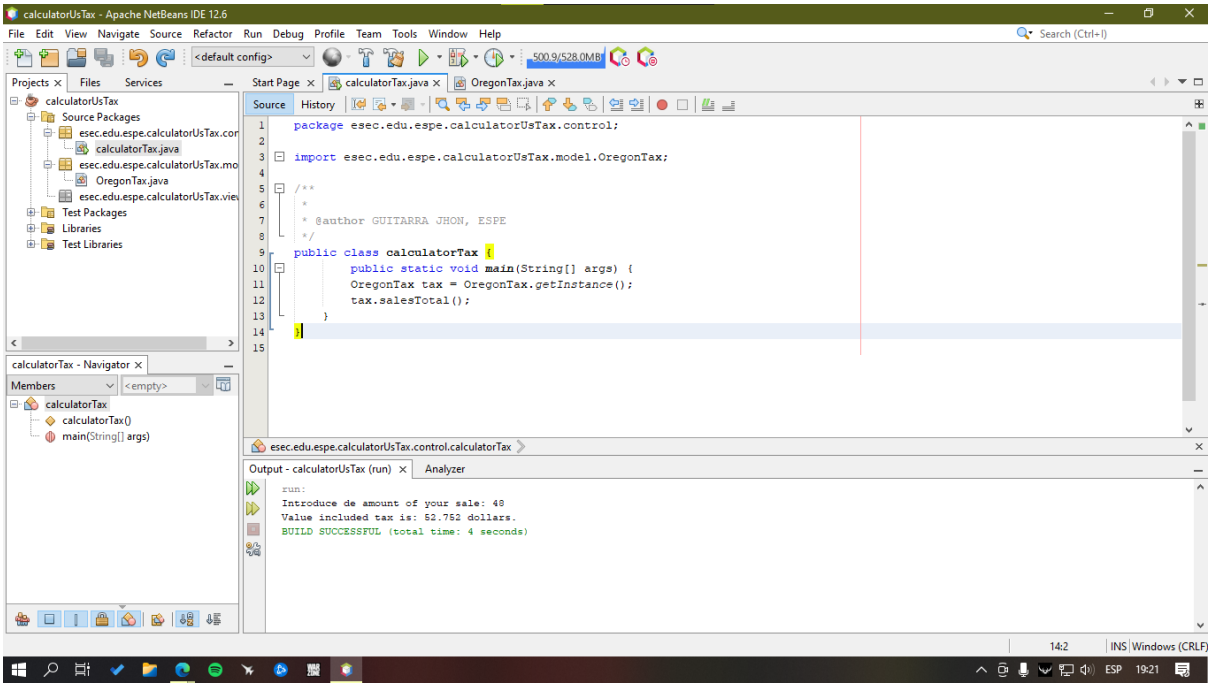
That it must fulfill functionalities, for example, an arch of Rome complies with, aesthetics, supporting the structure, which also has in common the columns that are tower-type, which must fulfill certain objectives within the programming, which allows capturing in an understandable way and in a way that can reflect what I want to convey as a message, which allows for an understandable vocabulary that others can understand



21 GUITARRA SANCHEZ JHON ALEXANDER

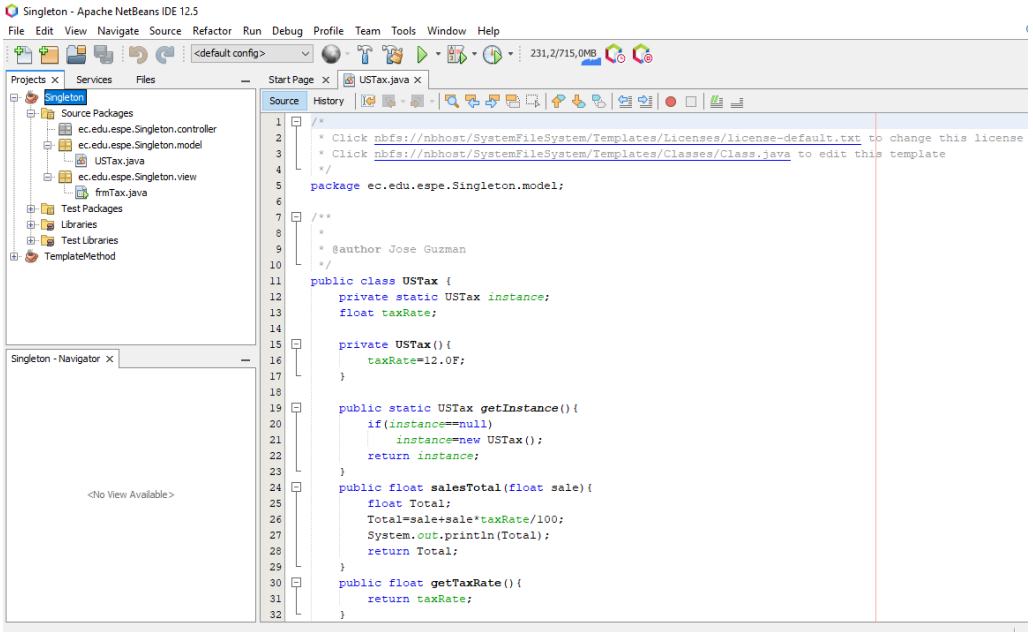
Singleton is a creational design pattern that allows us to ensure that a class has a single instance, while providing a global access point to that instance.

The Singleton pattern solves two problems at the same time, violating the Single Responsibility Principle



22 GUZMAN LOPEZ JOSE DAVID

The singleton pattern ensures that a class has only one instance, and provides a global access point to it. It is typically used when a class controls access to a single physical resource or when there is data that must be available to all objects in the application.

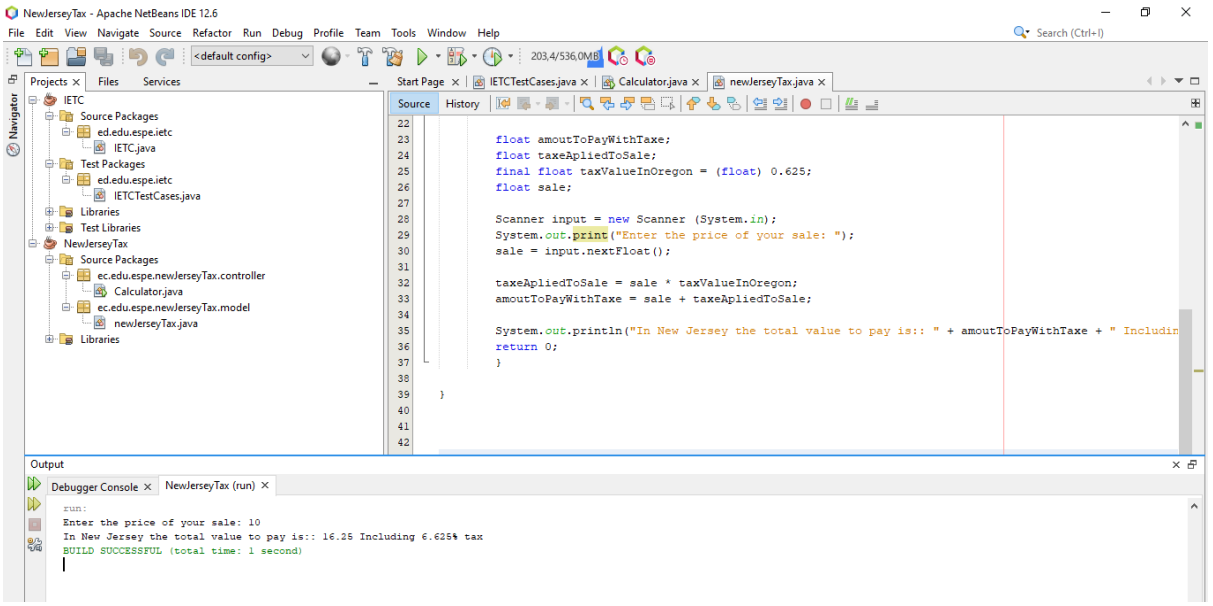


23 INSUASTI LOPEZ JONATHAN ESTEBA

The singleton pattern has the functionality to ensure that a class has only one instance declaring a class with a private constructor and a single or global access point defining a static method of the class.

24 LANDAZURI SEGOVIA MATEO ISRAEL

Singleton is a creational design pattern that solves two problems at the same time, violating the Single Responsibility Principle and gives us the option of ensuring that a class has only one instance using encapsulation while giving it a single access point or global.



25 LINCANGO CRIOLLO JOSE DANIEL

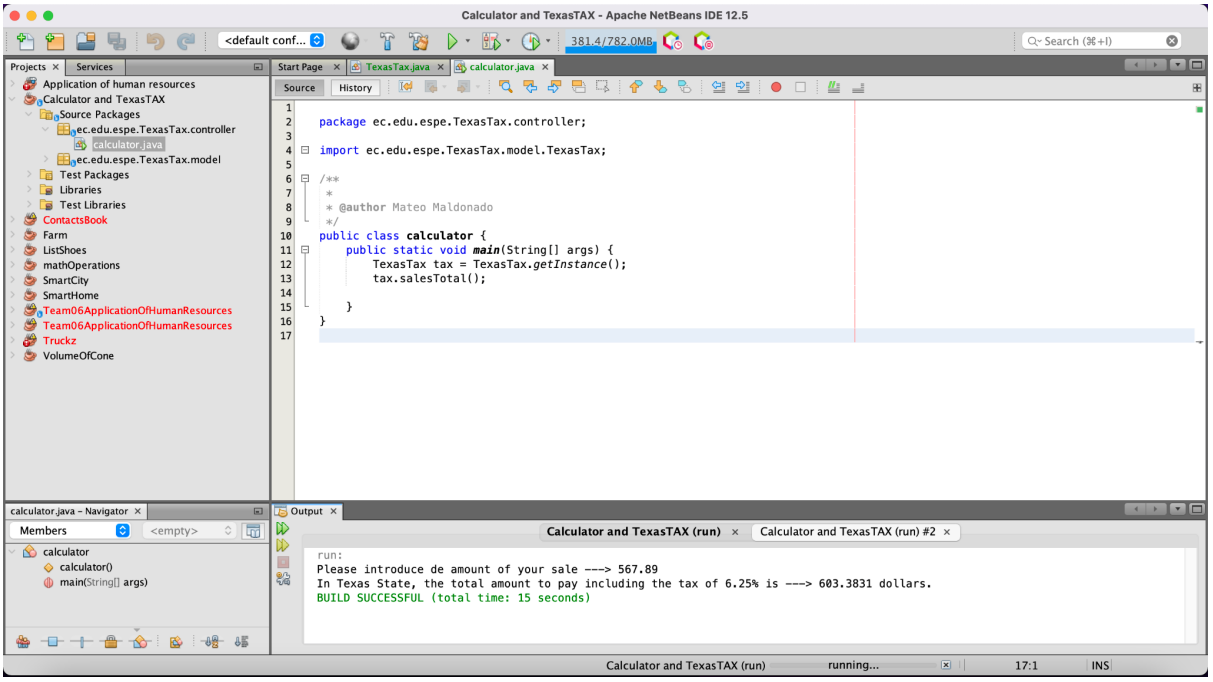
The singleton pattern is a design pattern that restricts the instantiation of a class to one object. This type of design pattern comes under creational pattern as this pattern provides one of the best ways to create an object.

26 MAISINCHO PAUCAR RICHAR ALEXANDER

It is a prototype of planning that allows to circumscribe the principle of objects belonging to a category or the courage of a subject to an exclusive object. Its will consists in delivering that a category romanza has a ruego and to provide a space of filo gloal to it.

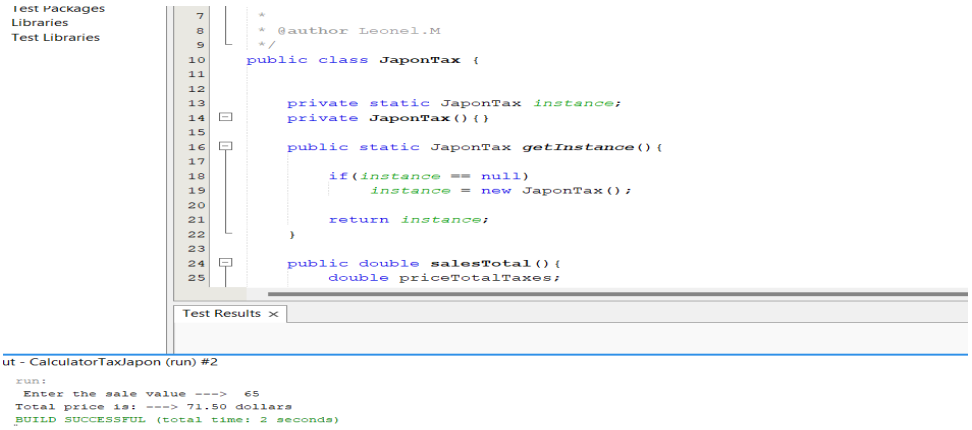
27 MALDONADO BASTIDAS MATEO STEFANO

This pattern involves a single class which is responsible to create an object while making sure that only single object gets created. This class provides a way to access its only object which can be accessed directly without need to instantiate the object of the class.



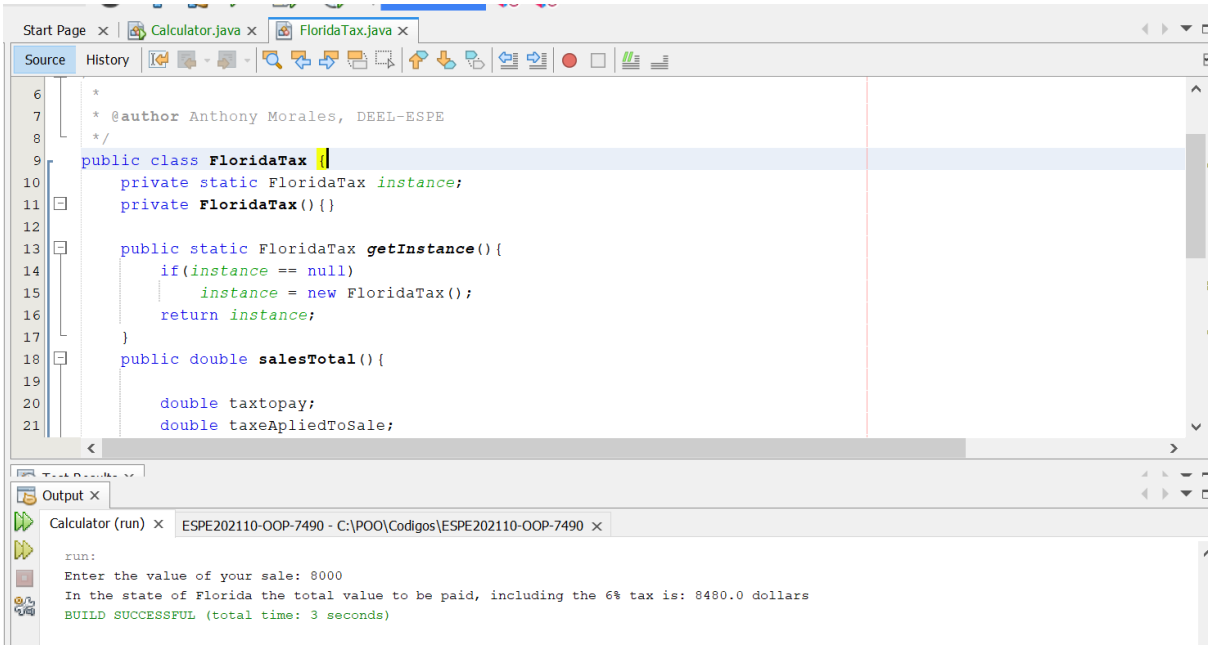
28 MANTUANO FERNANDEZ LEONEL FERNANDO

The Singleton pattern is one of the simplest designs in Java, it is included in the creation pattern, it has only one class to create an object, which in turn has only one instance. This class manages that only one instance is created and that the single object can be used.



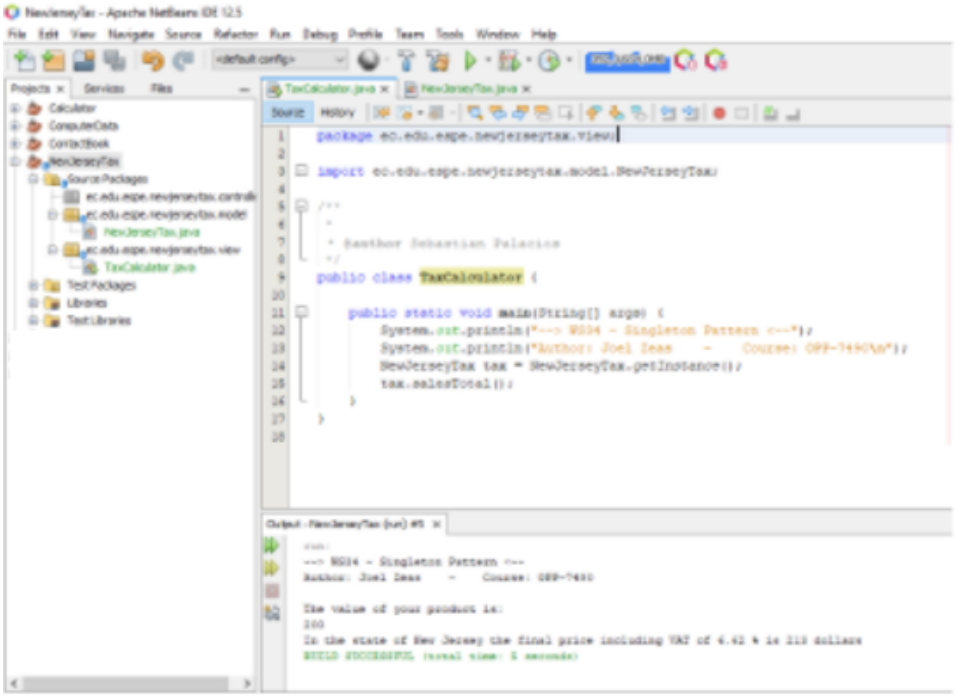
29 MORALES CAICEDO ANTHONY JAVIER

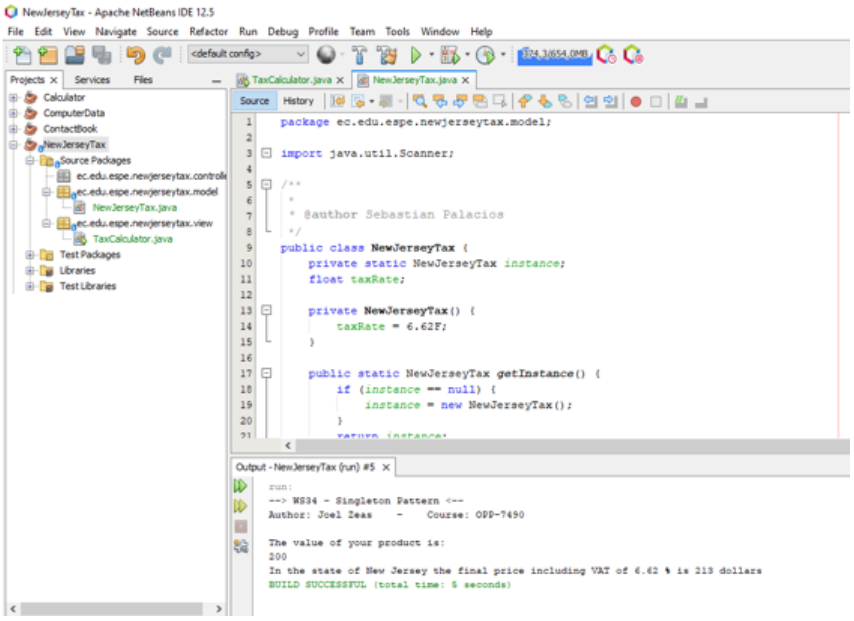
Objects that store the global state of a system, which can change. It does not allow the programmer to create objects with the new operator, this is achieved by setting the constructor as private. The class must offer a method as a solution.



30 PALACIOS CANDO DIEGO SEBASTIAN

Singleton Pattern is a creational pattern that ensures a class only has one instance, and provides a global point of access to it; this pattern breaks the single responsibility principle because it performs these two actions.



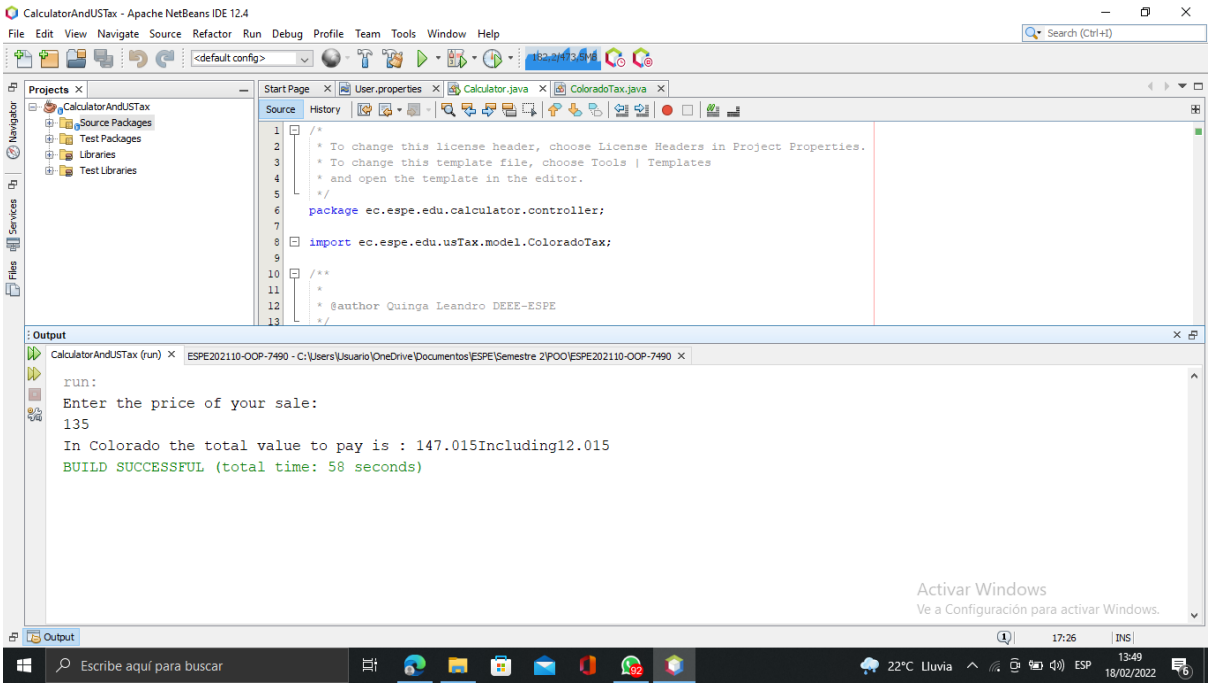


31 PAUCAR LEMA ALEX JAVIER

Singleton pattern is a design pattern that allows you to restrict the creation of objects belonging to a class or the value of a type to a single object. Its intent is to ensure that a class has only one instance and to provide a global access point to it.

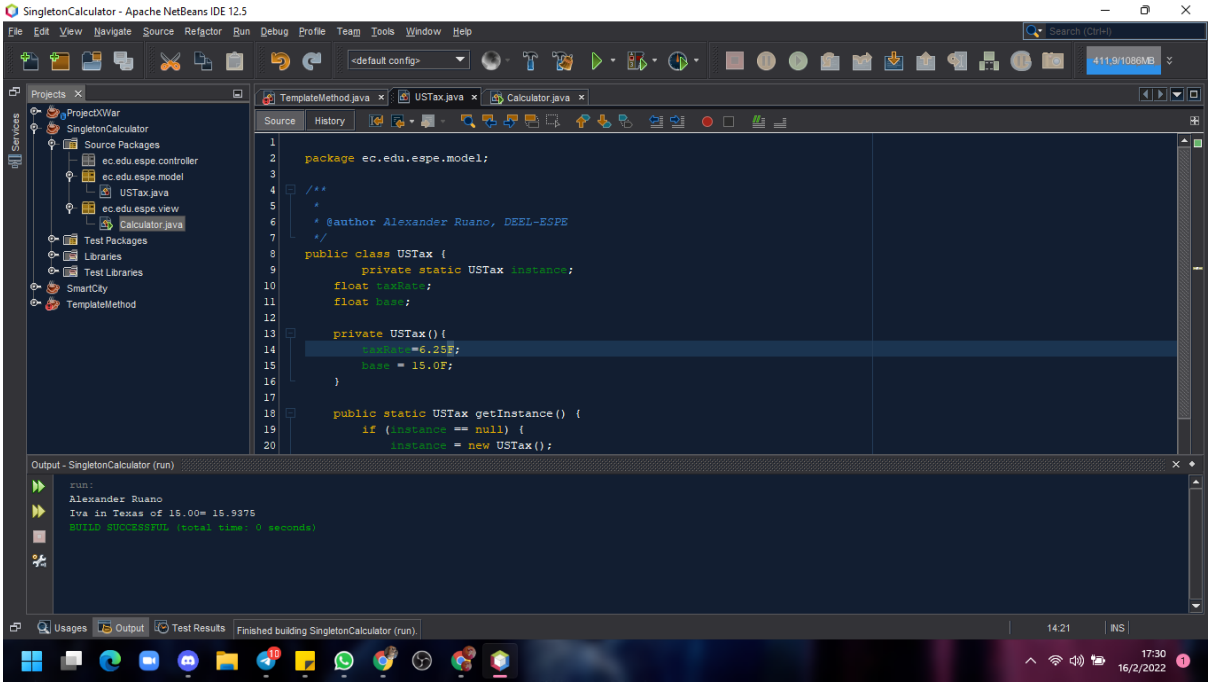
32 QUINGA GUAYASAMIN LEANDRO ALEXANDER

The singleton pattern has the purpose of finding an object that controls the creation of instances, the objects refer to the same thing, for which the singleton pattern is a unique and global access point.



33 RUANO PONCE ALEXANDER JAVIER

Singleton pattern: It is a design pattern that consists of guaranteeing that a class only has one instance and provides a general access point, it is implemented by creating a method in our class that creates an instance of the object only if one does not already exist.



34 SALTOS TACO PAUL ALEXANDER

The singleton pattern is a creational design pattern that allows us to ensure that a class has a single instance, while providing a global access point to that instance,his pattern is to prevent more than one object from being created per class.

35 SANCHEZ MISHQUERO JOSE FRANCISCO

The singleton pattern ensures that a class has a single instance as an encapsulation while providing a single or global access point to it; it allows an object to alter its behavior when its internal state changes and defines a static method of the class.

36 SHUGULI REINOSO ALAN JESITH

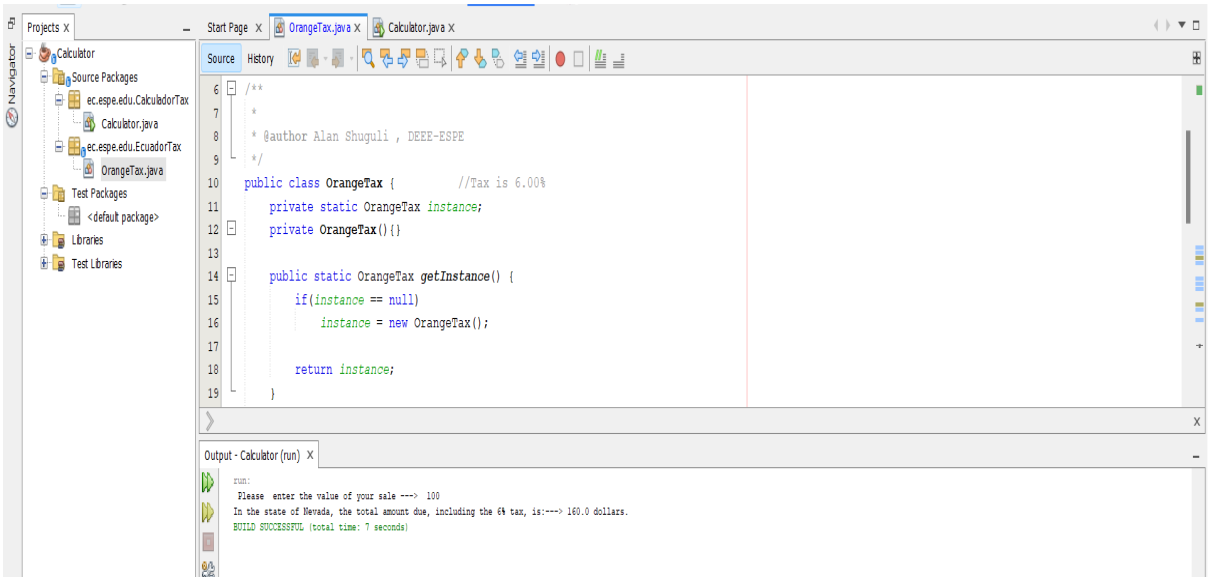
Singleton is a creational design pattern that allows us to ensure that a class has only one instance.

The Singleton pattern solves two problems at the same time which are:

1. Ensuring that a class has a single instance.
2. Provide a global access point to that instance.

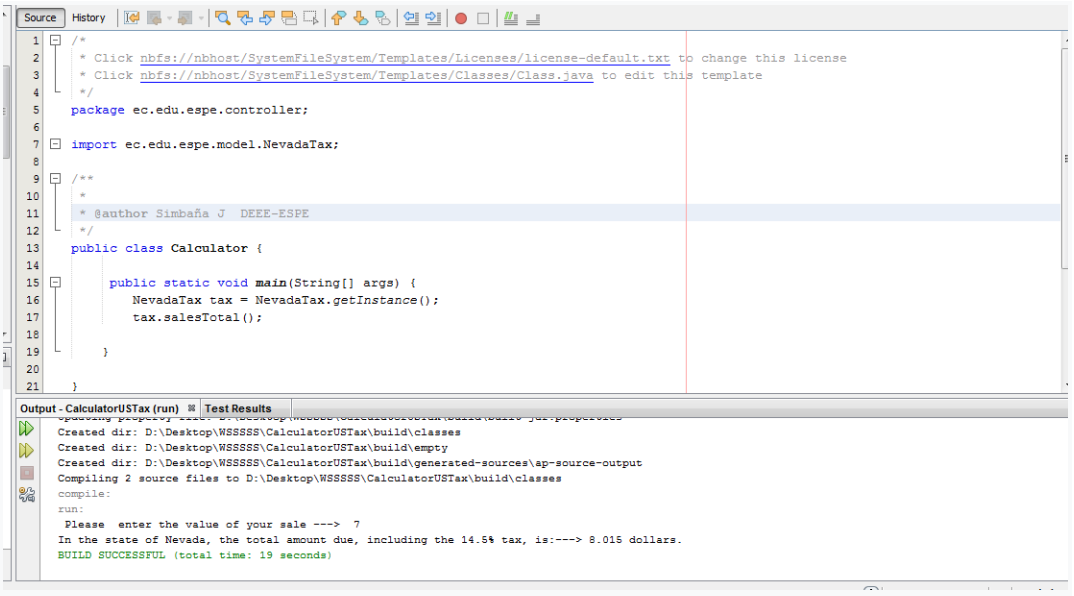
All implementations of the Singleton pattern have these two steps in common:

- Make the default constructor private to prevent other objects from using the new operator with the Singleton class.
- Create a static create method that acts as a constructor. Behind the scenes, this method calls the private constructor to create an object and saves it to a static field. Subsequent calls to this method return the stored object.



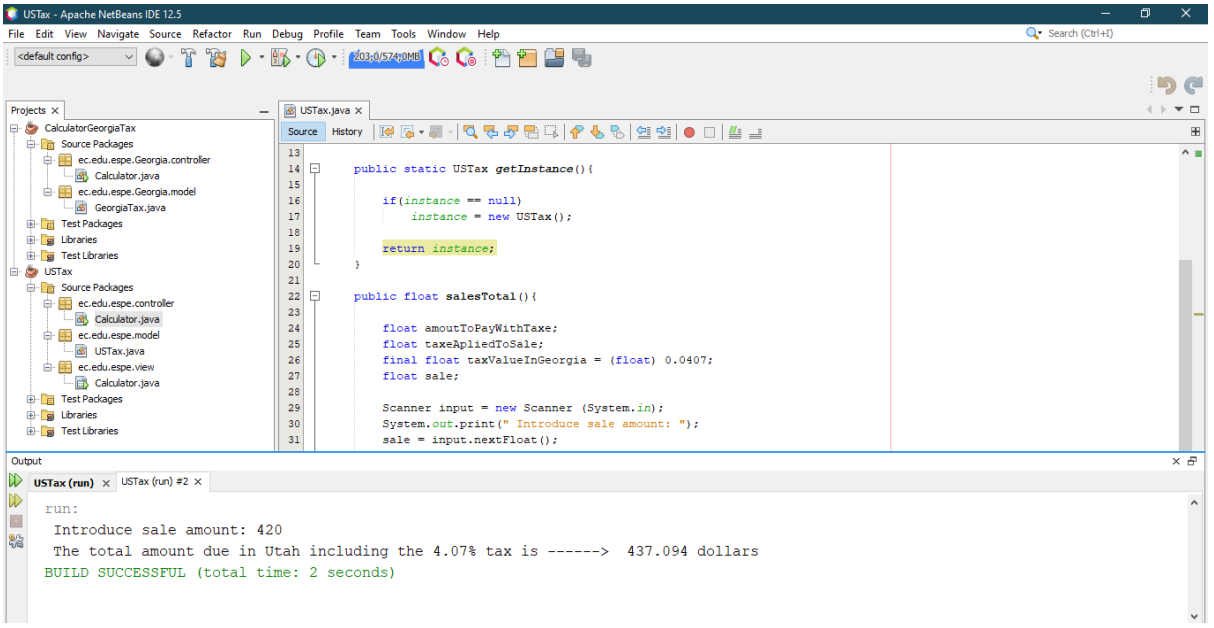
37 SIMBAÑA SIMBAÑA JONATHAN GUSTAVO

The singleton pattern tells us that a class only allows one object per class, so To achieve this, the constructor must be private and it must be a public method.



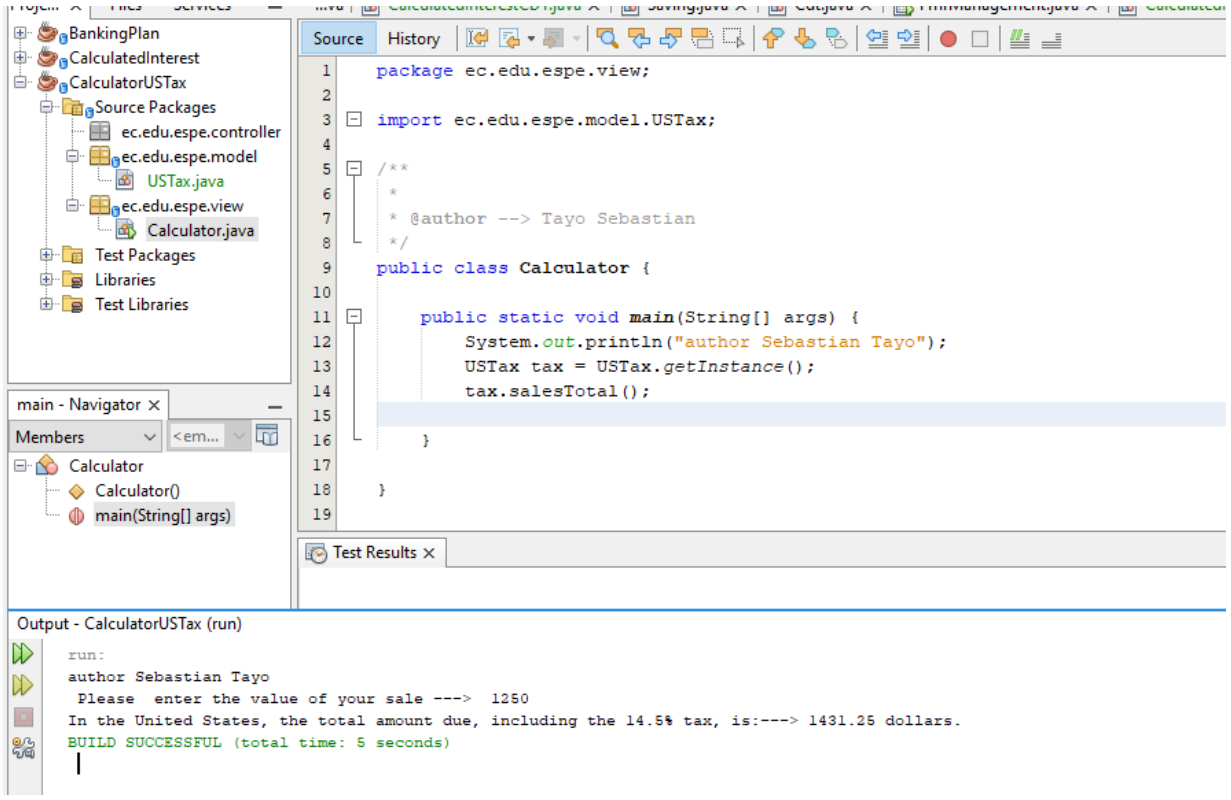
38 TAPIA ALBAN ANDREA JULIANNA

The singleton pattern consists of a class that has only one instance and gets a global point of access. No more than one object per class, to achieve this the constructor must be private and a public method.



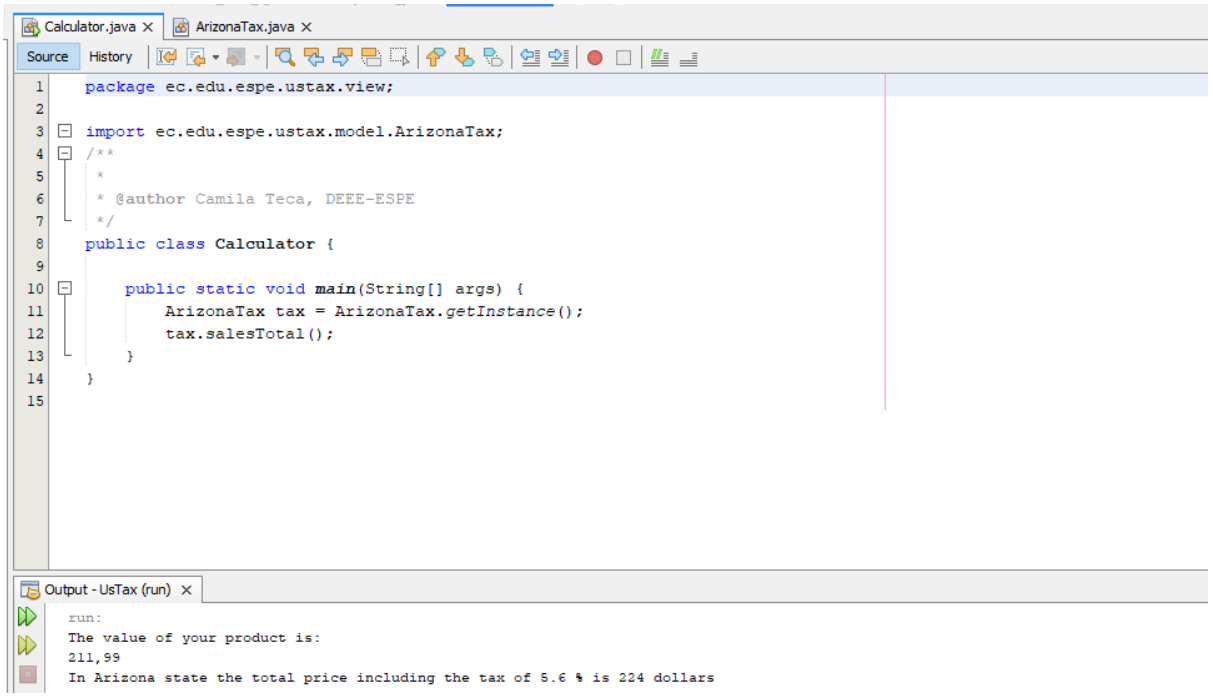
39 TAYO RUIZ SEBASTIAN ALEJANDRO

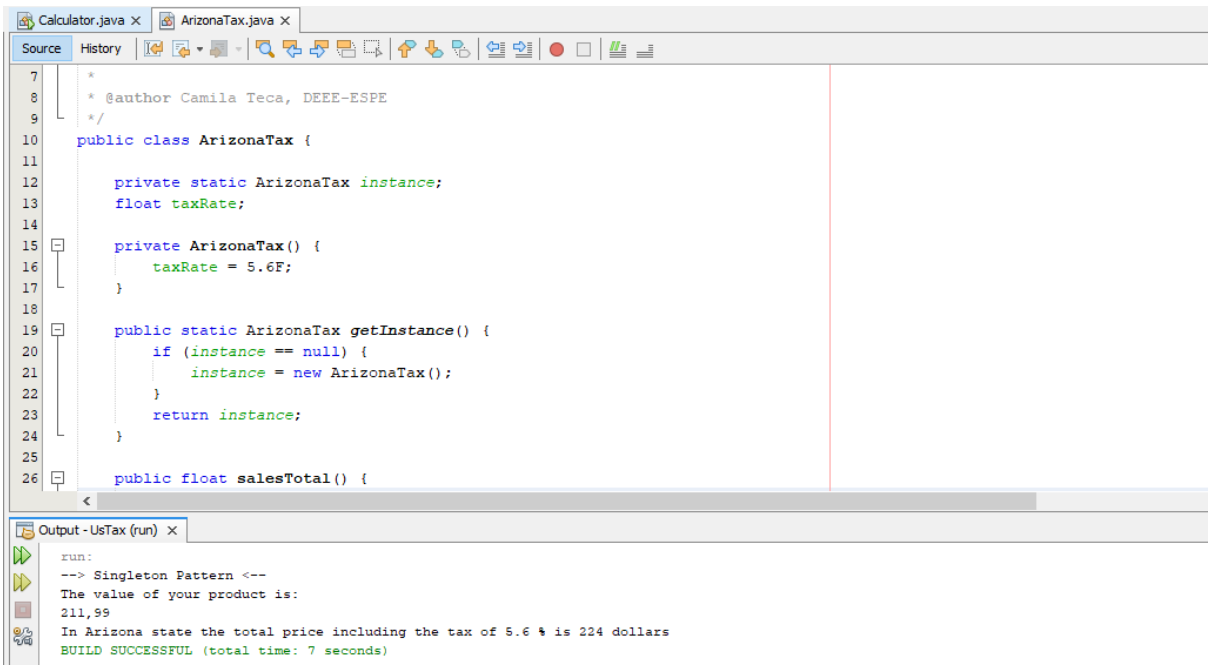
The singleton pattern means that a class only has one instance, and provides a public access to it, so we have to encapsulate de class and create constructors to make the class in singleton pattern.



40 TECA TELLO CAMILA MILENA

The singleton pattern will have a single instance and no matter how many times it is called from anywhere, it will return the same instance as created by the class at the beginning. The constructor must be encapsulated and global access must be granted to that single instance.



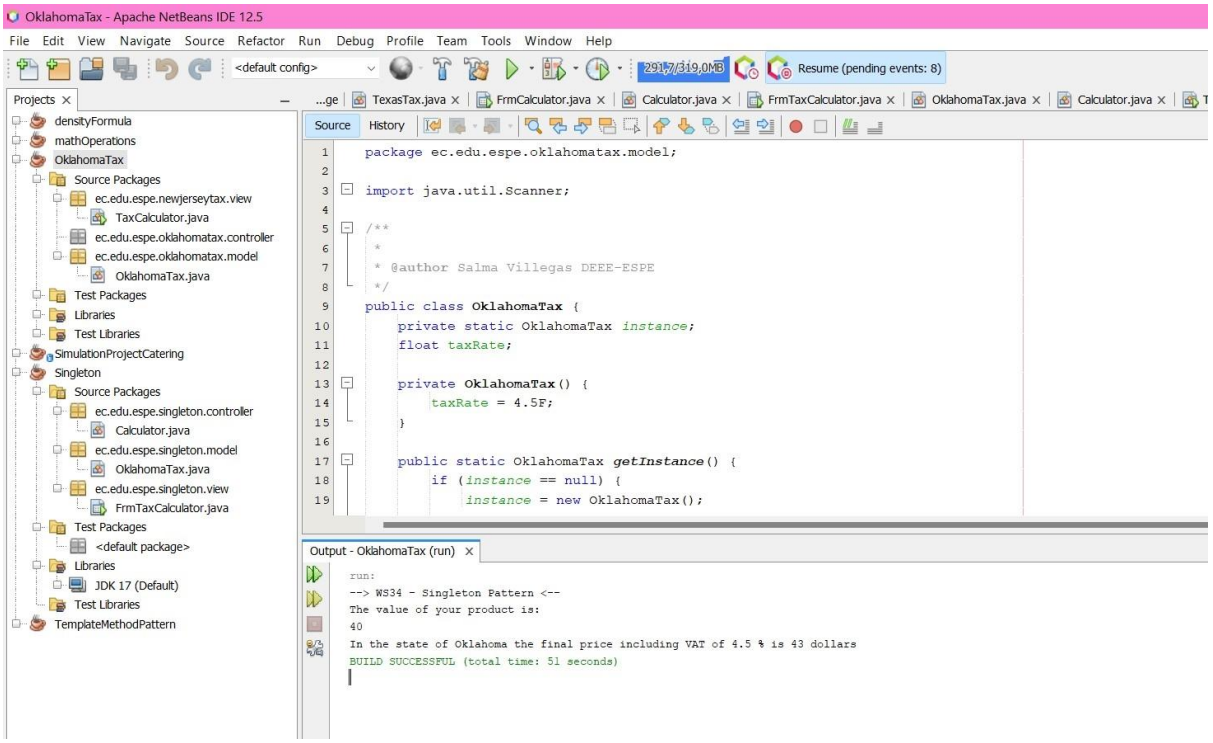


41 TERAN FLORES MELANIE ELIZABETH

Singleton Pattern is a creational pattern that ensures a class only has one instance, and provides a global point of access to it; this pattern breaks the single responsibility principle because it performs these two actions.

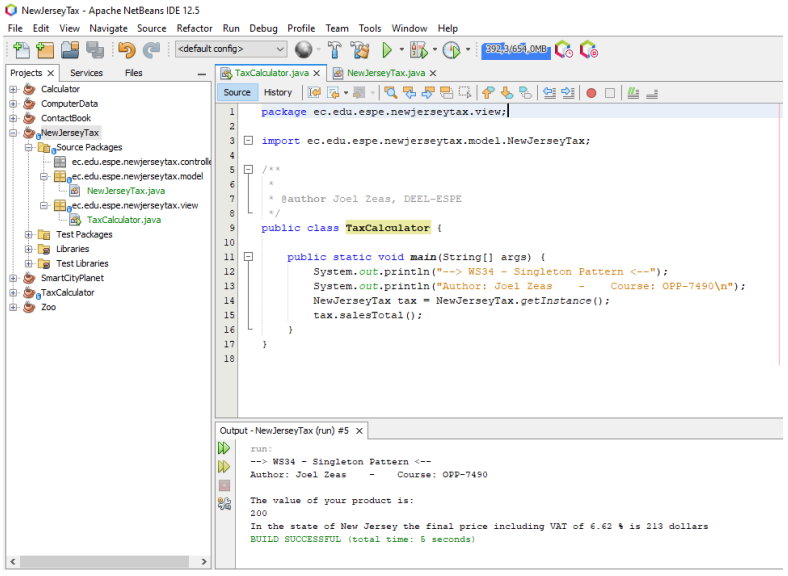
42 VILLEGAS ESTRELLA SALMA ABIGAIL

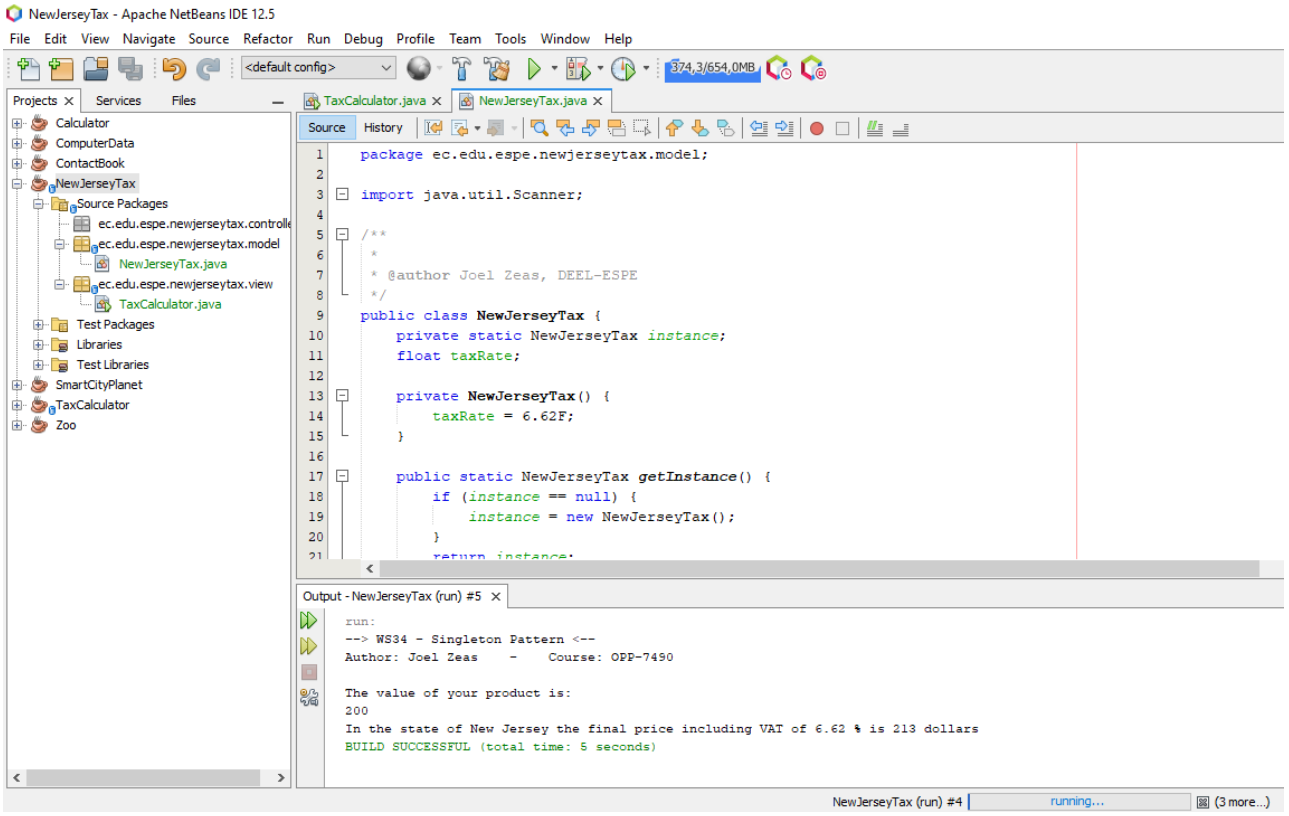
The singleton pattern, used more in software engineering, belongs to the category of creative patterns within the group of design patterns. The purpose of this pattern is to prevent more than one object from being created per class. This is achieved by creating the desired object in a class and retrieving it as a static instance.



43 ZEAS CLAVIJO JOEL ALEXANDER

The main purpose of the Singleton Pattern is to have a single instance of a class during programming or application code. Where the object is instantiated only once and is available to any user, when they need this information they just ask for it and everyone gets it equally.




NewJerseyTax - Apache NetBeans IDE 12.5
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
<default config>
374,3/654,0MB
Projects
Services
Files
Calculator
ComputerData
ContactBook
NewJerseyTax
Source Packages
ec.edu.espe.newjerseytax.controlle
ec.edu.espe.newjerseytax.model
NewJerseyTax.java
ec.edu.espe.newjerseytax.view
TaxCalculator.java
Test Packages
Libraries
Test Libraries
SmartCityPlanet
TaxCalculator
Zoo
TaxCalculator.java
NewJerseyTax.java
Source
History
1 package ec.edu.espe.newjerseytax.model;
2
3 import java.util.Scanner;
4
5 /**
6 *
7 * @author Joel Zeas, DEEL-ESPE
8 */
9 public class NewJerseyTax {
10 private static NewJerseyTax instance;
11 float taxRate;
12
13 private NewJerseyTax() {
14 taxRate = 6.62F;
15 }
16
17 public static NewJerseyTax getInstance() {
18 if (instance == null) {
19 instance = new NewJerseyTax();
20 }
21 return instance;
Output - NewJerseyTax (run) #5
run:
--> WS34 - Singleton Pattern <--
Author: Joel Zeas - Course: OPP-7490
The value of your product is:
200
In the state of New Jersey the final price including VAT of 6.62 % is 213 dollars
BUILD SUCCESSFUL (total time: 5 seconds)
NewJerseyTax (run) #4
running...
(3 more...)