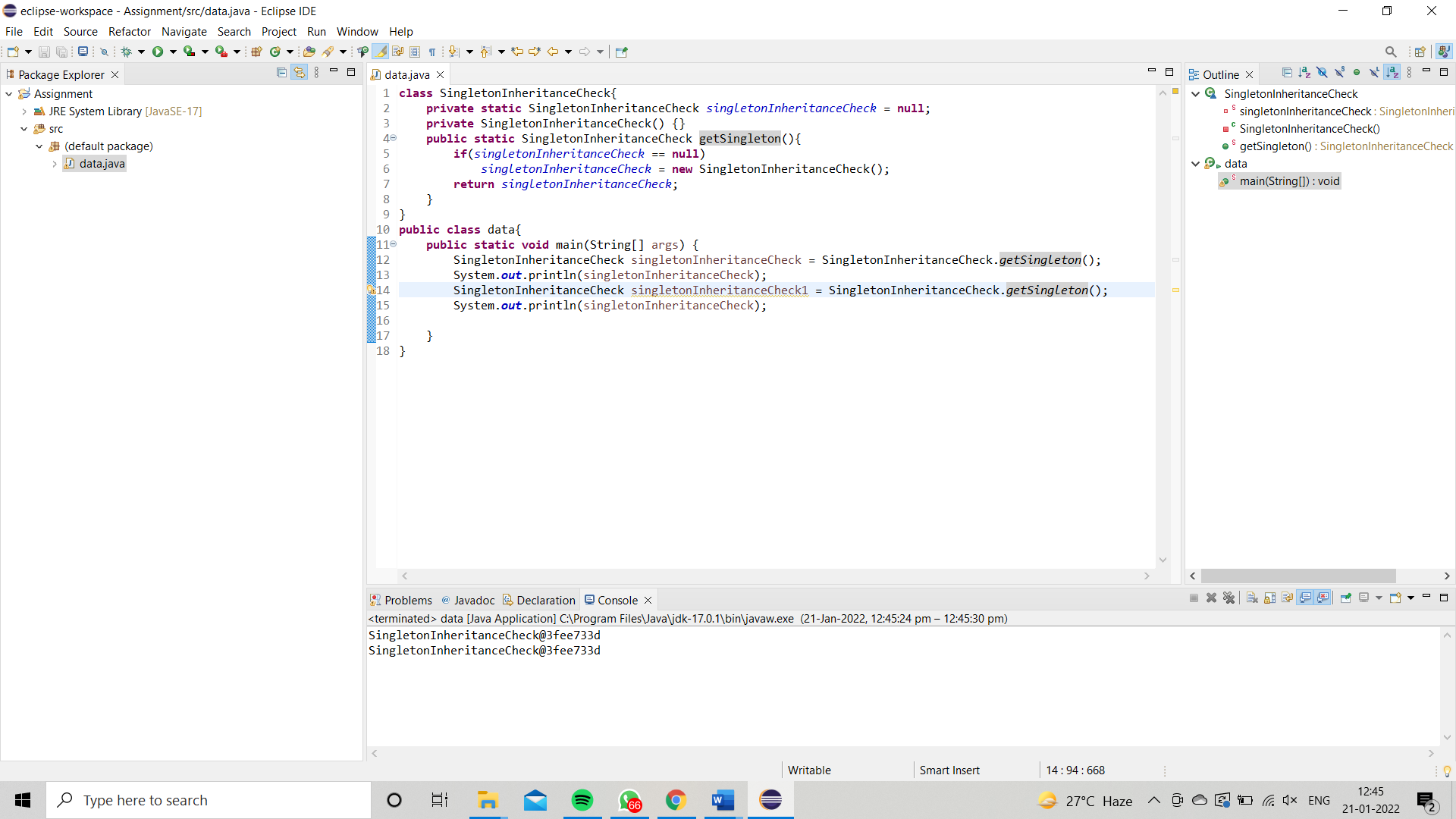
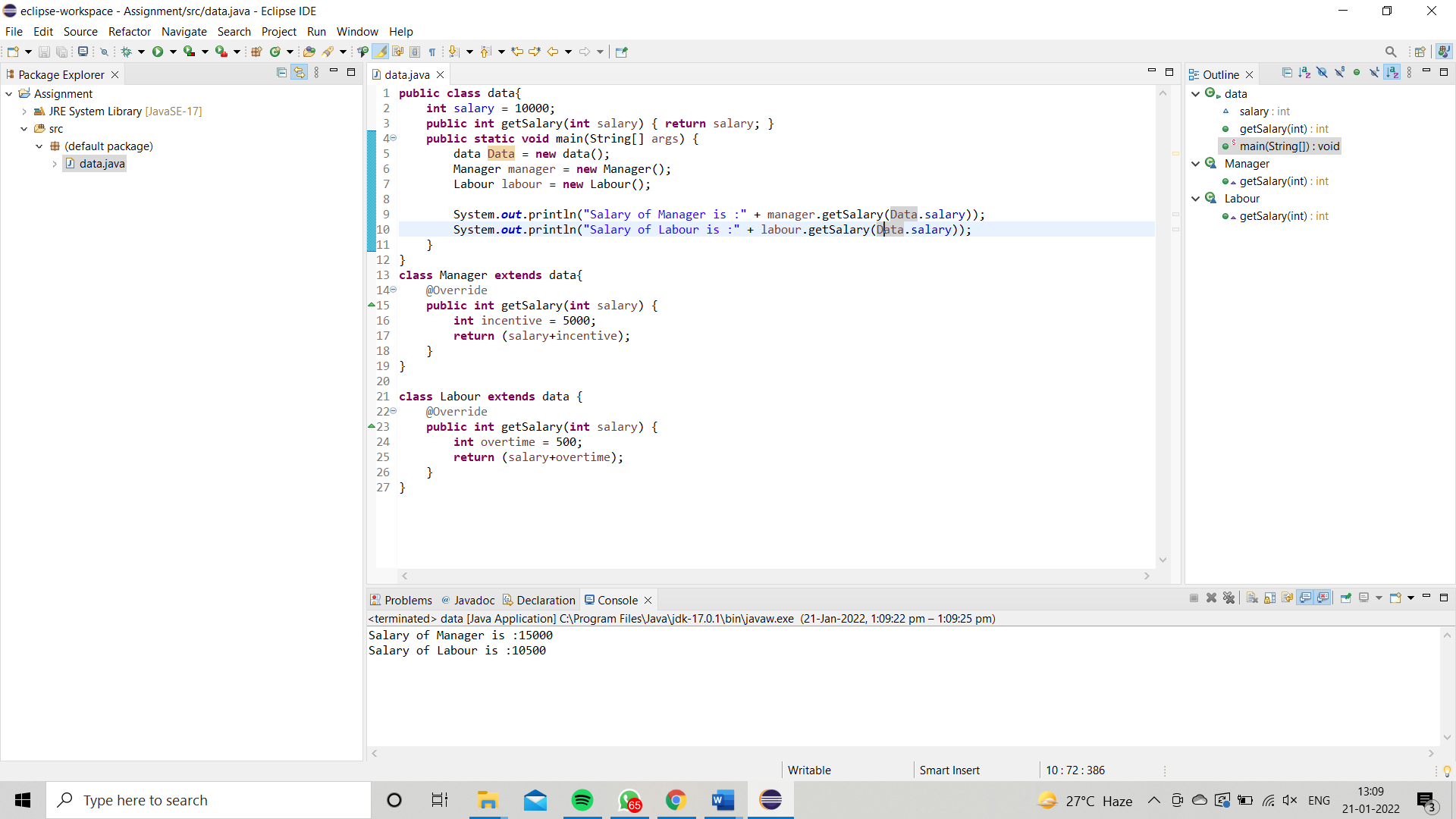
**OOPs Assignment**

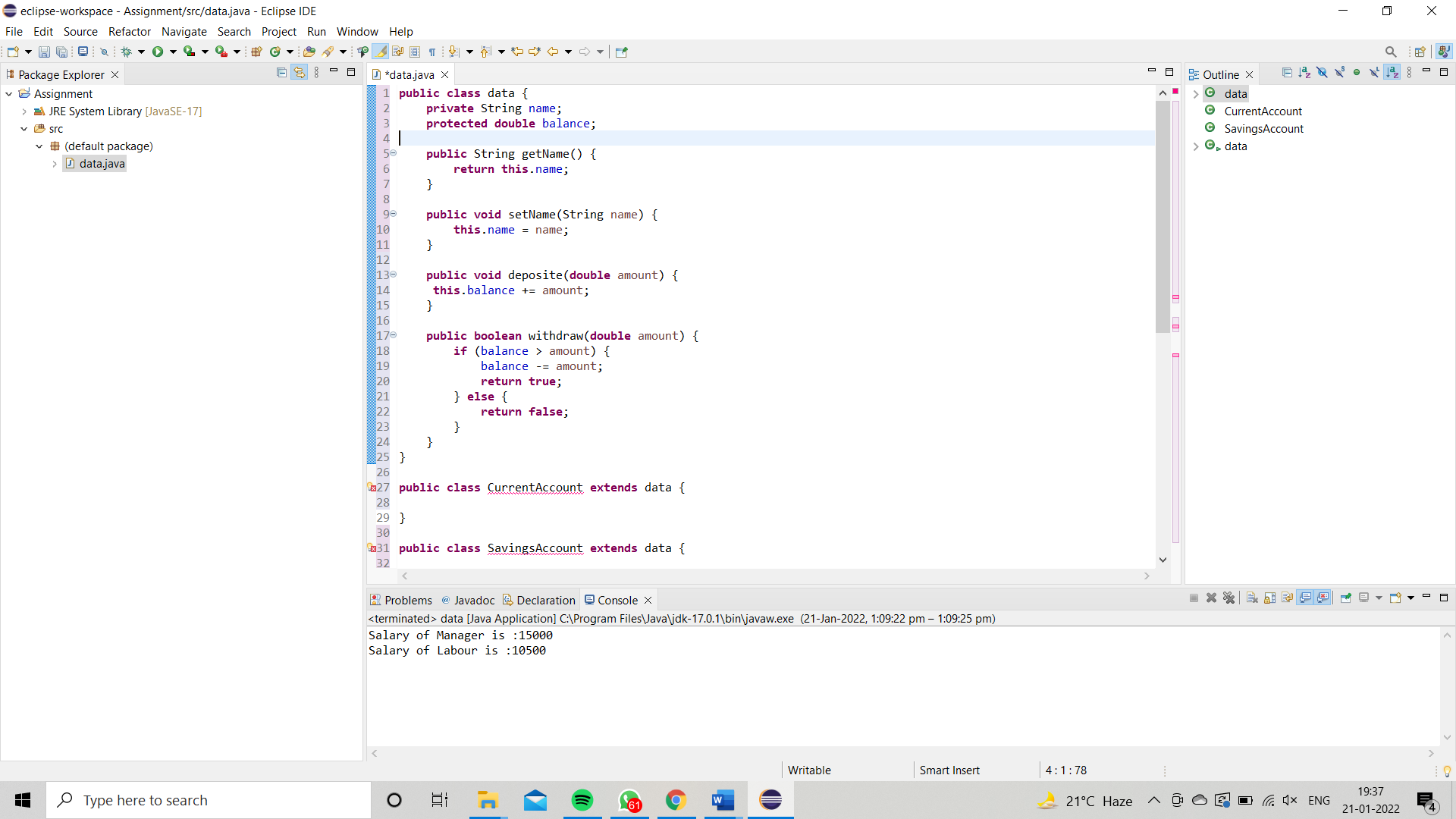
**Q1. Write a singleton class. Confirm that singleton class cannot be inherited.**



[**Q2**](https://adapt.in.capgemini.com/mod/vpl/view.php?id=2260)**. Write a program that describes the hierarchy of an organization. Here we need to write 3 classes Employee, Manager & Labour where Manager & Labour are the sub classes of the Employee. Manager has incentive & Labour has over time. Add the functionality to calculate total salary of all the employees. Use polymorphism i.e. method overriding.**



[**Q3**](https://adapt.in.capgemini.com/mod/vpl/view.php?id=2261)**. Write a program to consider saving & current account in the bank. Saving account holder has ‘Fixed Deposits’ whereas Current account holder has cash credit. Apply polymorphism to find out total cash in the bank.**



**Q4. Test the following principles of an abstract class:**

**• If any class has any of its method abstract then you must declare entire class abstract.**

**• Abstract class cannot be instantiated.**

**• When we extend an abstract class, we must either override all the abstract methods in sub class or declare subclass as abstract.**

**• Abstract class cannot be private.**

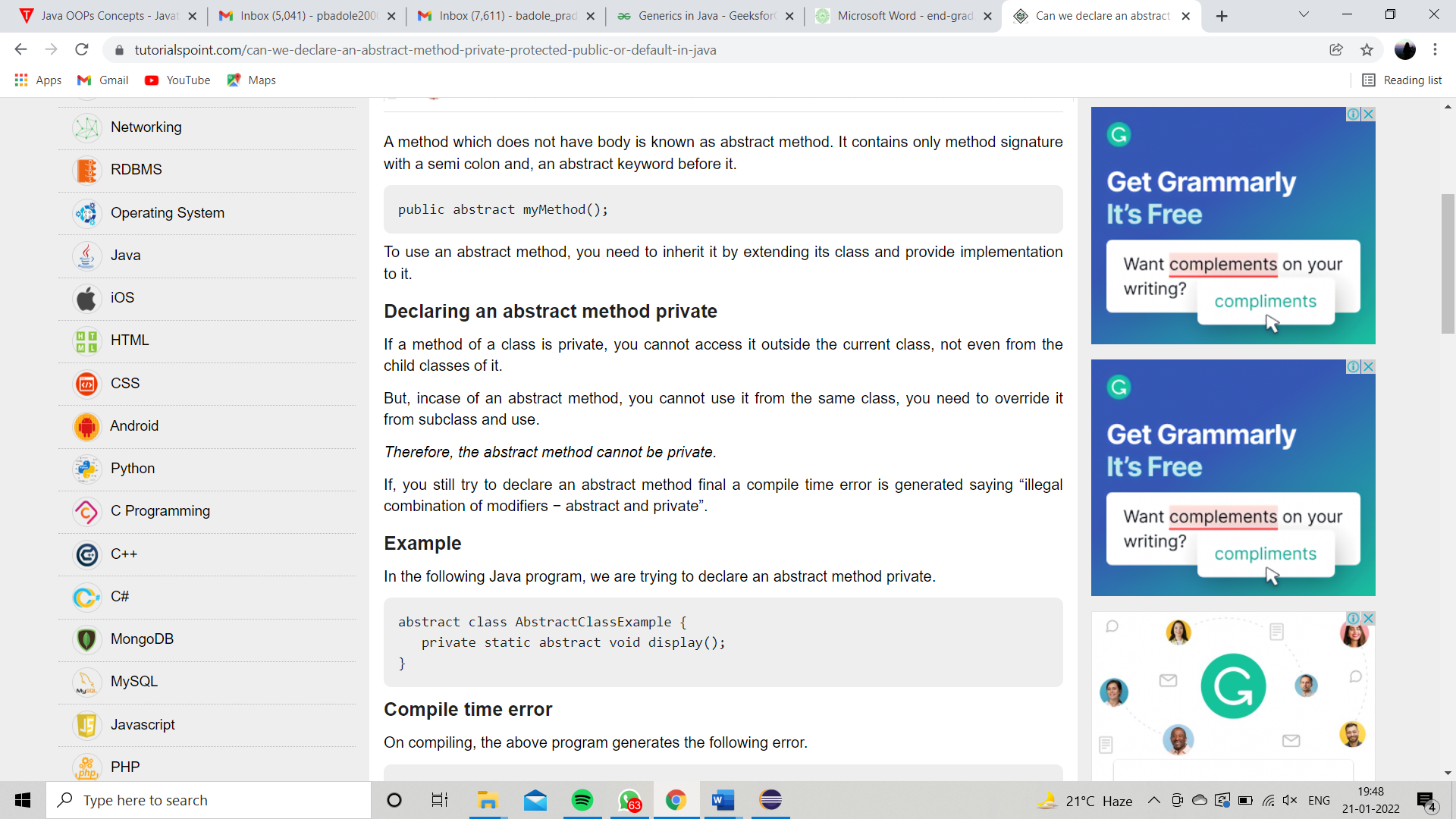
**• Abstract class cannot be final.**

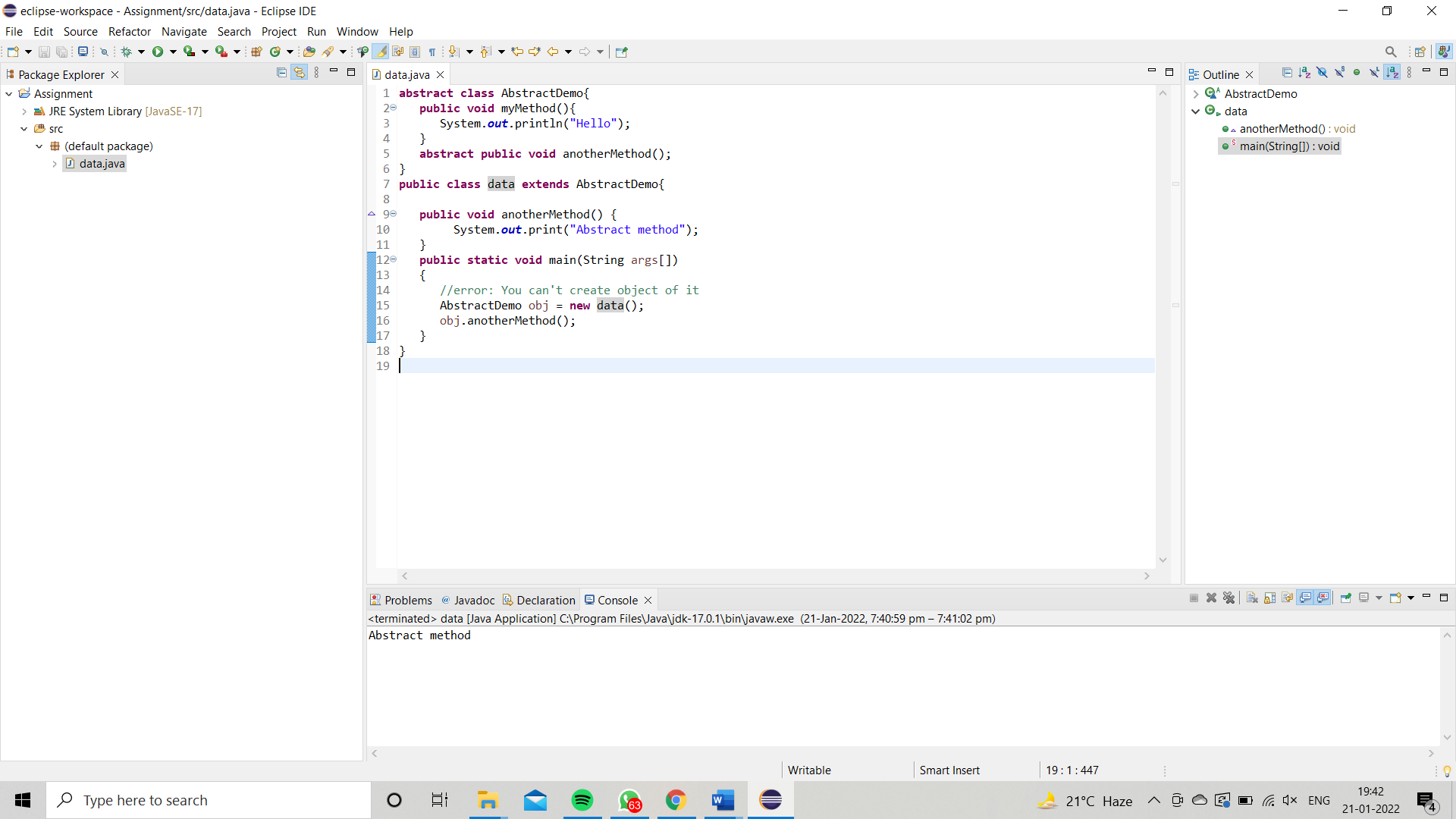
**• You can declare a class abstract without having any abstract method.**

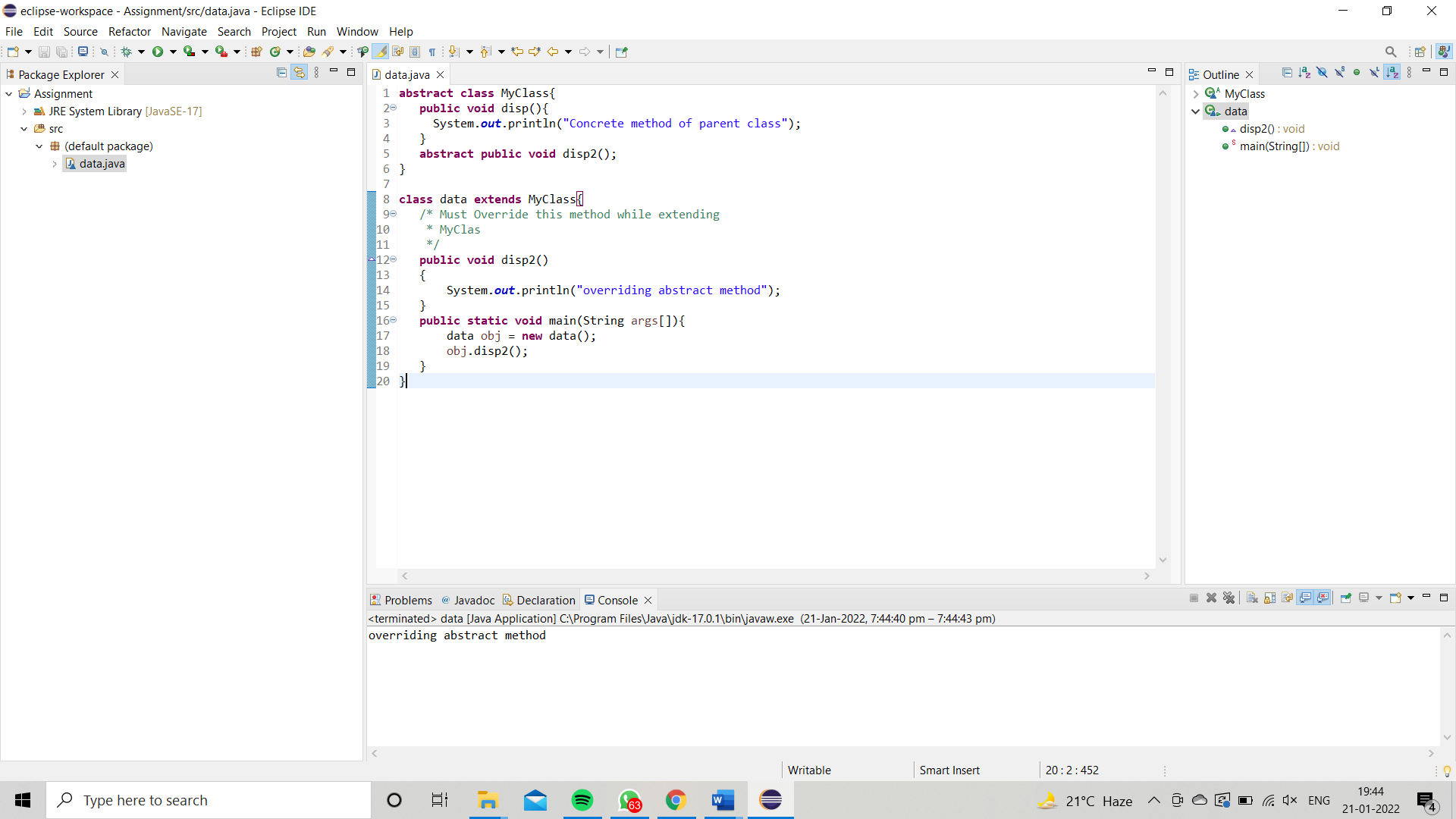
**Note:** A class which contains 0 or more abstract methods is known as abstract class. If it contains at least one abstract method, it must be declared abstract.

And yes, you can declare **abstract** class without defining an abstract method in it. Once you declare a class **abstract** it indicates that the class is incomplete and, you cannot instantiate it.

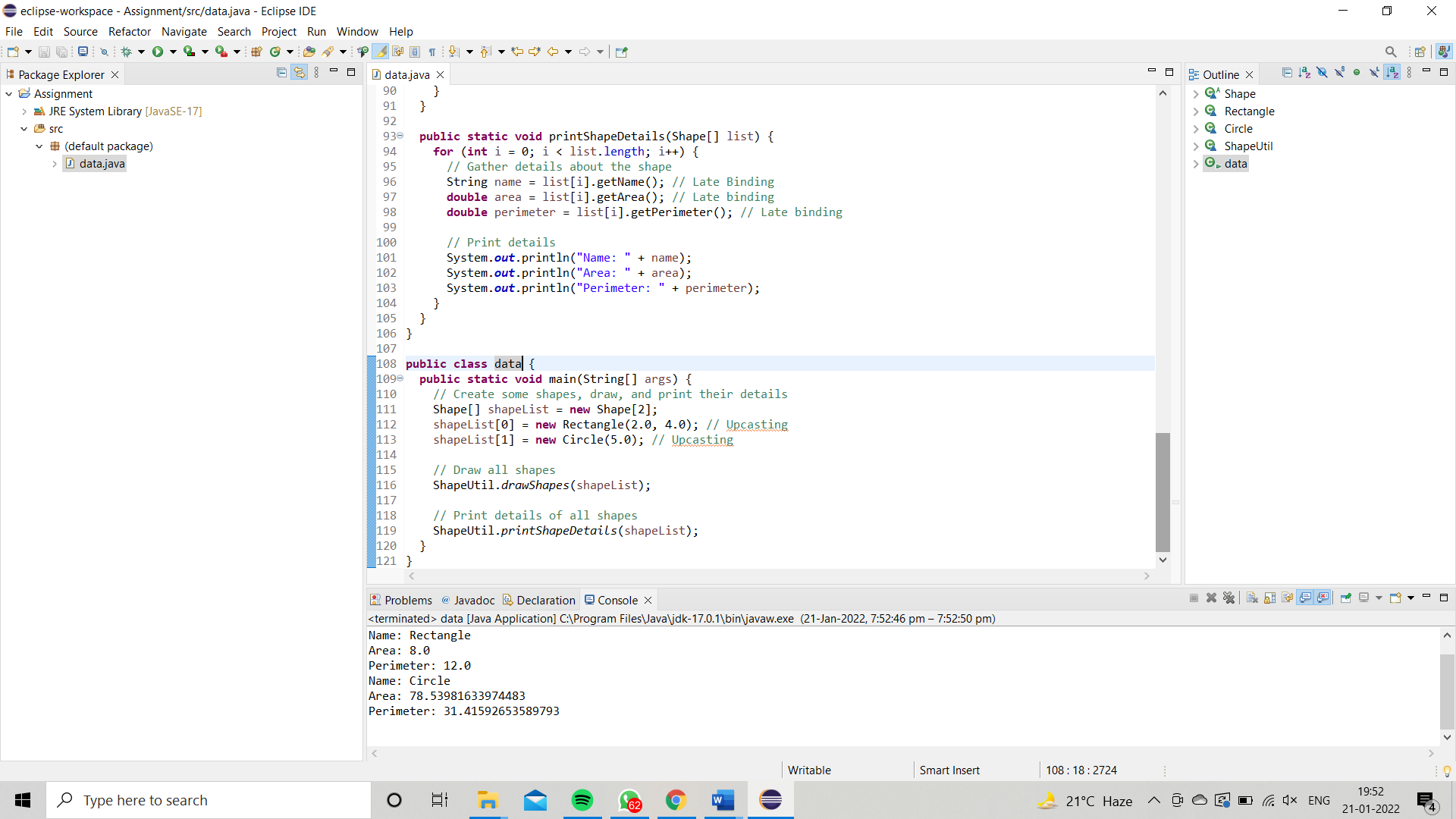
Hence, if you want to prevent instantiation of a class directly you can declare it abstract.



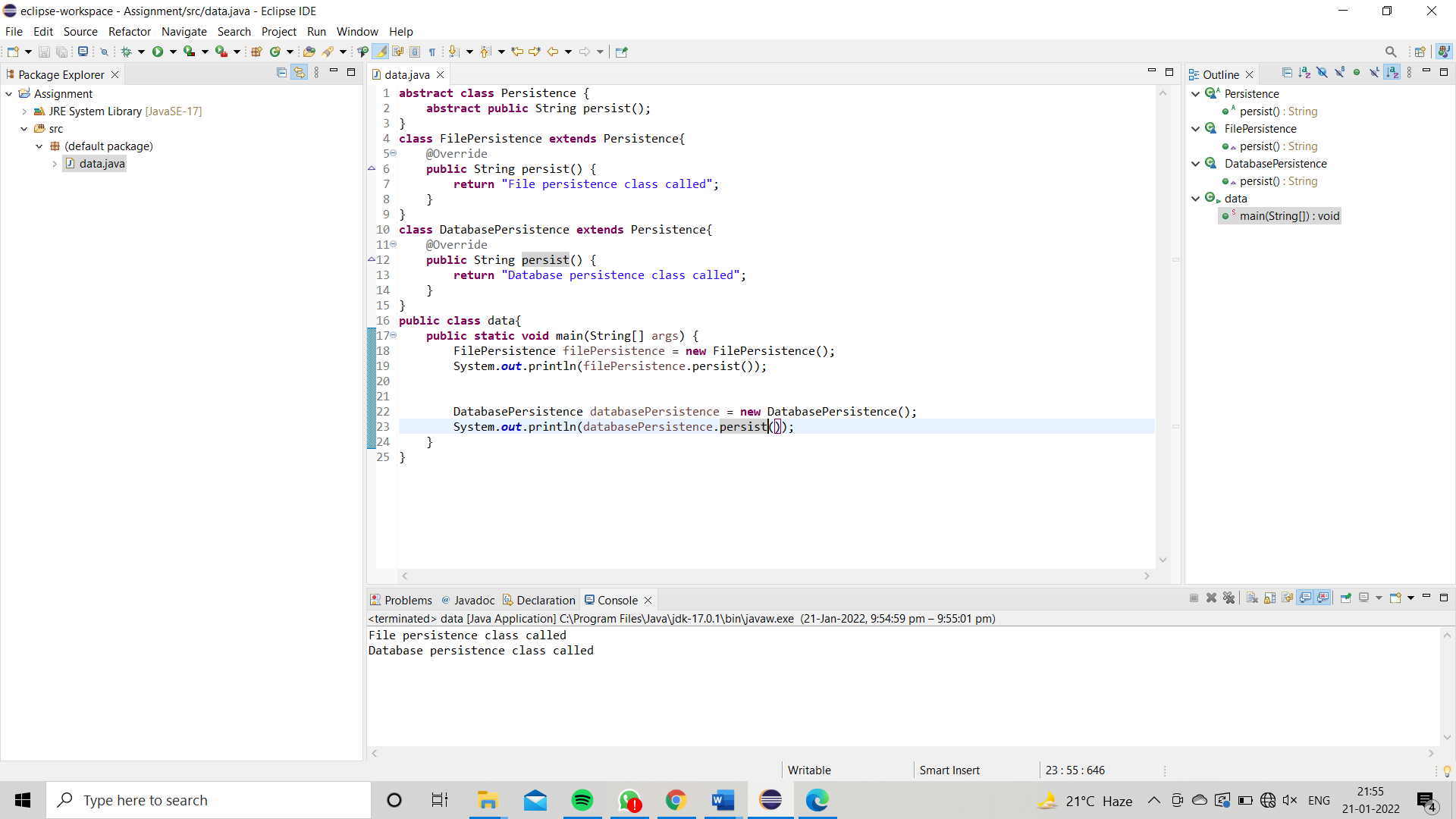




**Q5.Write the classes Line, Rectangle, Cube etc. & make the Shape as their base class. Add an abstract draw() method in the class Shape & draw all shapes**



[**Q6**](https://adapt.in.capgemini.com/mod/vpl/view.php?id=2145)**. Write an abstract class ‘Persistence’ along with two sub classes ‘FilePersistence’ & ‘DatabasePersistence’. The base class with have an abstract method persist() which will be overridden by its sub classes. Write a client who gets the Persistence object at runtime & invokes persist() method on it without knowing whether data is being saved in File or in Database.**



**Q7 Develop an application for Dessert shop. The application should allow owner to add items like Candy, Cookie or Ice Cream in the shop storage. Also customers should be able to place an order.**

