CAT-1 LAB (13 April 2023) L51 + L52

ITA5004 Lab

1. **Create a Java program that validates passwords based on certain rules.**

The program should have an abstract class "PasswordValidator" that defines a method "validatePassword" that accepts a password string and throws a custom exception "PasswordException" if the password is not valid.

The program should have a concrete class "BasicPasswordValidator" that extends the PasswordValidator class and implements an interface "PasswordLengthValidator" that defines a method "validatePasswordLength" that throws a custom exception "PasswordLengthException" if the password is less than 8 characters.

The program should have a concrete class "AdvancedPasswordValidator" that extends the PasswordValidator class and implements an interface "PasswordCharacterValidator" that defines a method "validatePasswordCharacters" that throws a custom exception "PasswordCharacterException" if the password does not contain at least one uppercase letter, one lowercase letter, and one digit.

The program should have a class "PasswordException" that extends the Exception class and is thrown by the "validatePassword" method when the password is invalid. The program should handle these exceptions and display meaningful error messages to the user. The program should have a method "isPasswordValid" in both the BasicPasswordValidator and AdvancedPasswordValidator classes that return true if the password is valid according to the rules and false otherwise.

……
……

2. **Design a billing system in Java that allows users to create and process invoices.**

   The program should include an interface "Taxable" that defines a method "calculateTax" that calculates the tax on a given amount based on a tax rate.

   The program should include an abstract class, "Invoice", that has properties such as invoice number, customer name, and amount.

   The program should have a class "ServiceInvoice" that extends the Invoice class and implements the Taxable interface. The ServiceInvoice class should have additional properties such as service description and hourly rate.

The program should have a class "ProductInvoice" that extends the Invoice class and implements the Taxable interface. The ProductInvoice class should have additional properties such as product description and unit price.

The program should allow users to create ServiceInvoices and ProductInvoices, calculate tax on the invoice amounts, and display the invoice details, including the total amount and tax amount. The program should also allow users to search for invoices by invoice number.

If an invoice is not found, the program should throw a custom exception, "InvoiceNotFoundException".