SUBJECT : PROGRAMMING IN JAVA

SUBJECT CODE : CSI26W1

LEARNING UNIT: 3 **PRACTICAL EXERCICE 1**

Instructions:

- This task may me completed in groups (maximum of 3 students per group)

Task 1:

a) Create a class named `Lease` with fields that hold an apartment tenant's name, apartment number, monthly rent amount, and term of the lease in months. Include a default constructor that initializes the name to XXX, the apartment number to 0, the rent to 1000, and the term to 12. Also include methods to get and set each of the fields. Include a non-static method named `addPetFee()` that adds R250 to the monthly rent value and calls a static method named `explainPetPolicy()` that explains the pet fee.

b) Create a class named `TestLease` whose `main()` method declares four Lease objects named lease1, lease2, lease3, and lease4. Create a `getData()` method that prompts a user for values for each field for a Lease, and return a newly constructed Lease object to the main() method, where it is assigned to one of main()'s first three Lease objects. Do not prompt the user for values for the fourth Lease object, but let it hold the default values. After the four Lease objects have been assigned values, pass the lease1 object to a `showValues()` method that displays the data. Then call the `addPetFee()` method with the lease1 object, and confirm that the fee explanation statement is displayed. Next, call the `showValues()` method for the lease1 object again and confirm that the pet fee has been added to the rent. Finally, call the `showValues()` method with each of the other three objects. Confirm that three hold the values you supplied as input and one holds the constructor default values.

Task 2:

a) Create a class named `BloodData` that includes String fields that hold a blood type (the four blood types are O, A, B, and AB) and an Rh factor (the factors are + and –). Create a default constructor that sets the fields to O and + and an overloaded constructor that requires values for both fields. Include get and set methods for each field.

b) Create an application named `TestBloodData` that declares two objects. Prompt the user for values for one, and use the default constructor values for the other. Display the details of both objects. Then change the values in the default object so it uses the user's values, and display the details for the object again to confirm the changes are made correctly.

c) Create a class named `Patient` that includes an ID number, age, and `BloodData`. Provide a default constructor that sets the ID number to 0, the age to 0, and the `BloodData` values to the default `BloodData` values (O and +). Create an overloaded constructor that provides values for each field. Also provide get methods for each field.

d) Create an application that declares three `Patient` objects. Use all default values for one `Patient` object, prompt the user for values for the second `Patient` object, and for the third object, prompt the user for `Patient` data but use default values for that Patient's `BloodData`.

~~~~~~~~~~~~~~~~~~
~~~~~~~~~~~~~~~~~~