Task 5: Functional Interfaces

Create a method that accepts functions as parameters using Predicate, Function, Consumer, and Supplier interfaces to operate on a Person object.

```
package test;

import java.util.function.Consumer;

import java.util.function.Function;

import java.util.function.Predicate;

import java.util.function.Supplier;


class Person {
    private String name;
    private int age;
    public Person(String name, int age) {
        this.name = name;
        this.age = age;
    }
    public String getName() {
        return name;
    }
    public int getAge() {
        return age;
    }
    public void setName(String name) {
        this.name = name;
    }
    public void setAge(int age) {
        this.age = age;
    }
    @Override
    public String toString() {
        return "Person{" +
                "name='" + name + '\'' +
```

```
            ", age=" + age +
            '}';
    }
}
public class FunctionInterfacesExample {
    public static void processPerson(Person person,
                                     Predicate<Person> predicate,
                                     Function<Person, String> function,
                                     Consumer<Person> consumer,
                                     Supplier<Person> supplier) {
        if (predicate.test(person)) {
            String info=function.apply(person);
            System.out.println("Person info: " + info);
            consumer.accept(person);
            Person newPerson=supplier.get();
            System.out.println("New Person created: "+newPerson);
        } else {
            System.out.println("Predicate condition not satisfied for person: "+person);
        }
    }

    public static void main(String[] args) {
        Person person = new Person("Alice", 30);
        processPerson(
                person,
                p -> p.getAge() >= 18,
                p -> "Name: " + p.getName() + ", Age: " + p.getAge(),
                p -> p.setAge(p.getAge() + 1),
                () -> new Person("Bob", 25)
        );
        System.out.println("Person after processing: " + person);
```

```
    }
}
```