```dart
// Dart OOP Example: Book, Employee, Appliance
void main() {
  print("=== BOOK EXAMPLE ===");
  Book book1 = Book("Dart Essentials", "Firoj", 500);
  Book book2 = Book("Flutter in Action", "Belal", 750);

  book1.display();
  book2.display();

  print("\n=== EMPLOYEE EXAMPLE ===");
  Manager manager = Manager("Firoj", 120000, "HR");
  Developer developer = Developer("Belal", 100000, "Dart");

  manager.displayInfo();
  developer.displayInfo();

  print("\n=== APPLIANCE EXAMPLE ===");
  Fan fan = Fan();
  Light light = Light();

  fan.turnOn();
  fan.turnOff();

  light.turnOn();
  light.turnOff();
}

// ------------------- BOOK CLASS -------------------
class Book {
  String title;
  String author;
  double price;

  // Constructor
  Book(this.title, this.author, this.price);

  // Method to calculate discounted price
  double discountedPrice(double discountPercent) {
    return price - (price * discountPercent / 100);
  }

  // Display book details
  void display() {
    print("Title: $title");
    print("Author: $author");
    print("Original Price: \$$${price.toStringAsFixed(2)}");
```

```dart
    print("Discounted Price (10%): \$$${discountedPrice(10).toStringAsFixed(2)}");
    print("------------------------");
  }
}

// ------------------- EMPLOYEE CLASSES -------------------
class Employee {
  String name;
  double salary;

  Employee(this.name, this.salary);
}

class Manager extends Employee {
  String department;

  Manager(String name, double salary, this.department) : super(name, salary);

  void displayInfo() {
    print("Manager Name: $name");
    print("Salary: \$$${salary.toStringAsFixed(2)}");
    print("Department: $department");
    print("------------------------");
  }
}

class Developer extends Employee {
  String programmingLanguage;

  Developer(String name, double salary, this.programmingLanguage)
      : super(name, salary);

  void displayInfo() {
    print("Developer Name: $name");
    print("Salary: \$$${salary.toStringAsFixed(2)}");
    print("Programming Language: $programmingLanguage");
    print("------------------------");
  }
}

// ------------------- APPLIANCE CLASSES -------------------
abstract class Appliance {
  void turnOn();
  void turnOff();
}
```

```
class Fan extends Appliance {
  @override
  void turnOn() {
    print("Fan is now running");
  }

  @override
  void turnOff() {
    print("Fan has been turned off");
  }
}

class Light extends Appliance {
  @override
  void turnOn() {
    print("Light is switched on");
  }

  @override
  void turnOff() {
    print("Light is switched off");
  }
}
```