

Here are simplified Java object-oriented exercises focusing only on **HAS** (attributes/data) and **DOES** (methods/behaviors) without constructors:

Exercise 1: Library Management System

```
public class Book {
    // HAS (Data/Attributes)
    String title;
    String author;
    String isbn;
    boolean isAvailable;

    // DOES (Methods/Behaviors)
    void checkout() {
        // Mark book as checked out
    }

    void returnBook() {
        // Mark book as returned
    }

    void displayInfo() {
        // Show book details
    }
}

public class Member {
    // HAS
    String name;
    int memberId;
    int booksCount;

    // DOES
    void borrowBook() {
        // Borrow a book
    }

    void returnBook() {
        // Return a book
    }

    void viewProfile() {
        // Display member information
    }
}
```

```
public class Library {  
    // HAS  
    String libraryName;  
    int totalBooks;  
    int totalMembers;  
  
    // DOES  
    void addBook() {  
        // Add new book to library  
    }  
  
    void registerMember() {  
        // Register new member  
    }  
  
    void searchBook() {  
        // Find books by title or author  
    }  
}
```

Exercise 2: Banking System

```
public class BankAccount {
    // HAS
    String accountNumber;
    String ownerName;
    double balance;
    boolean isActive;

    // DOES
    void deposit() {
        // Add money to account
    }

    void withdraw() {
        // Remove money from account
    }

    void checkBalance() {
        // Display current balance
    }

    void closeAccount() {
        // Deactivate account
    }
}

public class ATM {
    // HAS
    String location;
    double cashAvailable;
    boolean isOperational;

    // DOES
    void dispenseCash() {
        // Give cash to customer
    }

    void acceptDeposit() {
        // Accept cash/check deposits
    }
}
```

```
void printReceipt() {  
    // Print transaction receipt  
}  
}
```

Exercise 3: E-commerce System

```
public class Product {
    // HAS
    String productName;
    double price;
    int stockQuantity;
    String category;

    // DOES
    void updatePrice() {
        // Change product price
    }

    void updateStock() {
        // Change stock quantity
    }

    void displayProduct() {
        // Show product details
    }
}

public class ShoppingCart {
    // HAS
    int totalItems;
    double totalAmount;
    boolean isEmpty;

    // DOES
    void addItem() {
        // Add product to cart
    }

    void removeItem() {
        // Remove product from cart
    }

    void calculateTotal() {
        // Calculate total price
    }
}
```

```
    void clearCart() {  
        // Empty the cart  
    }  
}  
  
public class Customer {  
    // HAS  
    String customerName;  
    String email;  
    String address;  
    int orderCount;  
  
    // DOES  
    void placeOrder() {  
        // Create new order  
    }  
  
    void updateProfile() {  
        // Change customer details  
    }  
  
    void viewOrderHistory() {  
        // Show past orders  
    }  
}
```

Exercise 4: School Management System

```
public class Student {
    // HAS
    String studentName;
    int studentId;
    int grade;
    double gpa;

    // DOES
    void attendClass() {
        // Mark attendance
    }

    void takeExam() {
        // Take a test
    }

    void submitAssignment() {
        // Submit homework
    }

    void calculateGPA() {
        // Compute grade point average
    }
}

public class Teacher {
    // HAS
    String teacherName;
    String subject;
    int yearsExperience;
    int studentsCount;

    // DOES
    void teachLesson() {
        // Conduct class
    }

    void gradeExam() {
        // Grade student tests
    }
}
```



```
void createAssignment() {  
    // Make homework  
}  
  
void takeAttendance() {  
    // Record student attendance  
}  
}  
  
public class Course {  
    // HAS  
    String courseName;  
    String courseCode;  
    int credits;  
    int maxStudents;  
  
    // DOES  
    void enrollStudent() {  
        // Add student to course  
    }  
  
    void dropStudent() {  
        // Remove student from course  
    }  
  
    void scheduleClass() {  
        // Set class time  
    }  
}
```

Exercise 5: Vehicle Management System

```
public class Car {  
    // HAS  
    String make;  
    String model;  
    int year;  
    String color;  
    boolean engineRunning;  
  
    // DOES  
    void startEngine() {  
        // Turn on the car  
    }  
  
    void stopEngine() {  
        // Turn off the car  
    }  
  
    void accelerate() {  
        // Speed up  
    }  
  
    void brake() {  
        // Slow down  
    }  
  
    void honk() {  
        // Sound the horn  
    }  
}  
  
public class Motorcycle {  
    // HAS  
    String brand;  
    int engineSize;  
    boolean hasWindshield;  
    int speed;  
  
    // DOES  
    void startEngine() {  
        // Turn on motorcycle
```

```
}

void wheelie() {
    // Lift front wheel
}

void lean() {
    // Tilt for turning
}

void revEngine() {
    // Make loud noise
}
}

public class Truck {
    // HAS
    double cargoCapacity;
    double currentLoad;
    int numberOfAxles;
    boolean trailerAttached;

    // DOES
    void loadCargo() {
        // Put items in truck
    }

    void unloadCargo() {
        // Remove items from truck
    }

    void attachTrailer() {
        // Connect trailer
    }

    void checkWeight() {
        // Verify load limits
    }
}
```

Exercise 6: Restaurant System

```
public class Restaurant {  
    // HAS  
    String restaurantName;  
    int tableCount;  
    String cuisineType;  
    boolean isOpen;  
  
    // DOES  
    void openRestaurant() {  
        // Start business for the day  
    }  
  
    void closeRestaurant() {  
        // End business for the day  
    }  
  
    void reserveTable() {  
        // Book table for customer  
    }  
  
    void serveCustomer() {  
        // Provide service  
    }  
}
```

```
public class Chef {  
    // HAS  
    String chefName;  
    String specialty;  
    int yearsExperience;  
    boolean isWorking;  
  
    // DOES  
    void cookMeal() {  
        // Prepare food  
    }  
  
    void createMenu() {  
        // Design new dishes  
    }  
}
```

```
void checkIngredients() {
    // Verify supplies
}

void trainStaff() {
    // Teach cooking techniques
}
}

public class Order {
    // HAS
    int orderNumber;
    String customerName;
    double totalAmount;
    boolean isCompleted;

    // DOES
    void addItem() {
        // Add dish to order
    }

    void removeItem() {
        // Remove dish from order
    }

    void calculateTotal() {
        // Compute final price
    }

    void markComplete() {
        // Finish the order
    }
}
```

Exercise 7: Hospital System

```
public class Patient {
    // HAS
    String patientName;
    int age;
    String illness;
    boolean isAdmitted;

    // DOES
    void checkIn() {
        // Register at hospital
    }

    void takeTest() {
        // Undergo medical examination
    }

    void takeMedicine() {
        // Consume prescribed drugs
    }

    void discharge() {
        // Leave hospital
    }
}

public class Doctor {
    // HAS
    String doctorName;
    String specialization;
    int patientCount;
    boolean isAvailable;

    // DOES
    void examinePatient() {
        // Check patient condition
    }

    void prescribeMedicine() {
        // Give medication
    }
}
```

```
void performSurgery() {  
    // Conduct operation  
}  
  
void writeReport() {  
    // Document findings  
}  
}
```

Practice Instructions:

1. **Pick one scenario** and implement all classes
2. **Focus on realistic data:** What would each object actually have?
3. **Think about behaviors:** What would each object actually do?
4. **Keep it simple:** Don't worry about implementation details yet
5. **Use meaningful names:** Make your code self-explanatory

Next Steps: Once comfortable with HAS/DOES, you can add:

- Method parameters
- Return types
- Access modifiers (public, private)
- Constructors

Start with the scenario that interests you most!