Java code refactoring with or without ChatGPT

Survey response 1

Response ID
2

Date submitted
2024-01-09 12:47:05

Last page
13

Start language
en

Seed
330681714

Date started
2024-01-09 12:20:39

Date last action
2024-01-09 12:47:05

Total time
1592.04

Survey questionnaires (Part 1)

How old are you?
30

How many years of experience do you have with Java programming?
7

For how many years have you been programming for larger software projects e.g. in a company? Please enter a number between 0 and 30.
7

How many years of experience do you have with code refactoring?
7

Did you study programming or computer science at a university?
Yes

During your education, how many courses did you take where Java was the primary language?
4

On a scale from 1 to 5, how would you rate your Java programming expertise (e.g 1-very inexperienced, 5-very experienced)?
3

How would you compare your Java expertise to those with over 20 years of practical experience (e.g 1-very inexperienced, 5-very experienced)?
2

How would you rate your Java expertise in comparison to your peers or colleagues (e.g 1-very inexperienced, 5-very experienced)?
3

How often have you used Chat GPT (e.g 1-low, 5-high)?
3

Have you used ChatGPT for code refactoring tasks (e.g 1-low, 5-high)?
3

What is the average size of Java professional projects you typically work on, categorized as small-scale (up to 900 lines of code), medium-scale (900 to 40,000 lines of code), or large-scale (exceeding 40,000 lines of code)?
medium-scale

Group time: Survey questionnaires (Part 1)
10.38

Task Explanation

Tasks Overview: In each of the two sections, you will encounter five Java code snippets that require refactoring. The first five snippets must be refactored without assistance from ChatGPT, while the last five snippets can be refactored with the aid of ChatGPT. Primarily, you have two alternatives: Without Assistance: Refactor the code on your own, relying on your existing knowledge and skills. With ChatGPT Assistance: Utilize the assistance of ChatGPT to receive suggestions and guidance for refactoring the code. Timing: Each assignment must be completed within a strict time constraint of 3 minutes. You must complete the work within a 3-minute timeframe, otherwise, timeouts will occur. Efficiently allocate your time to ensure timely completion of all jobs. Instructions: Read the code: Begin by thoroughly understanding the provided Java code snippet. Refactor: Apply your refactoring skills to improve the code based on the given criteria (readability, efficiency, maintainability, etc.).

Group time: Task Explanation
5.49

Question 1 for Pretest (Part 2)

Refactor the below code snippet without ChatGPT within 3 minutes. public double getPayAmount() { double result; if (isDead) { result = deadAmount(); } else { if (isSeparated) { result = separatedAmount(); } else { if (isRetired) { result = retiredAmount(); } else { result = normalPayAmount(); } } } return result;}

public double getPayAmount() {
  double result;
  if (isDead) {
  result = deadAmount();
  } else {
  if (isSeparated) {
  result = separatedAmount();
  } else {
  if (isRetired) {
  result = retiredAmount();
  } else {
  result = normalPayAmount();
  }
  }
  }
  return result;
}

Group time: Question 1 for Pretest (Part 2)
152.42

Question 2

Refactor the below code snippet without ChatGPT within 3 minutes. public class Customer { private String name; private String address; private double balance; public Customer(String name, String address) { this.name = name; this.address = address; this.balance = 0; } public void deposit(double amount) { this.balance += amount; } public void withdraw(double amount) { this.balance -= amount; } public double getBalance() { return balance; } }

@Getter
@Setter
public class Customer {
  private String name;
  private String address;
  private double balance;
 	
  public Customer(String name, String address) {
  this.name = name;
  this.address = address;
  this.balance = 0;
  }
 	
  public void deposit(double amount) {
  if (amount >= 0) {
  this.balance += amount;
  } else {
  throw new IllegalArgumentException("Deposit amount cannot be negative.");
  }
  }
 	
  public void withdraw(double amount) {
  if (amount >= 0) {
  if (balance >= amount) {
  this.balance -= amount;
  } else {
  throw new IllegalArgumentException("Insufficient balance for withdrawal.");
  }
  } else {
  throw new IllegalArgumentException("Withdrawal amount cannot be negative.");
  }
  }
}

Group time: Question 2
206.66

Question 3

Refactor the below code snippet without ChatGPT within 3 minutes. public class Customer { private String name; private String address; private double balance; public Customer(String name, String address, double initialBalance) { this.name = name; this.address = address; this.balance = initialBalance; } public void processPayment(double amount) { if (amount > balance) { throw new InsufficientFundsException(); } balance -= amount; } public void printStatement() { System.out.println("Customer name: " + name); System.out.println("Customer address: " + address); System.out.println("Customer balance: " + balance); }}

```
@Getter
@Setter
@Log4j2
public class Customer {
    private String name;
    private String address;
    private double balance;

    public Customer(String name, String address, double initialBalance) {
        this.name = name;
        this.address = address;
        this.balance = initialBalance;
    }

    public void processPayment(double amount) {
        if (amount < 0) {
            throw new IllegalArgumentException("Payment amount cannot be negative.");
        }

        if (amount > balance) {
            throw new IllegalArgumentException("Insufficient funds for the payment.");
        }

        balance -= amount;
    }
public void printStatement() {
        LOGGER.info("Customer name: " + getName());
        LOGGER.info("Customer address: " + getAddress());
        LOGGER.info("Customer balance: " + getBalance());
    }
}
```

Group time: Question 3

226.77

Question 4

Refactor the below code snippet without ChatGPT within 3 minutes. public class ShippingService { public double calculateShippingCost(Order order) { double totalPrice = order.getTotalPrice(); double weight = order.getWeight(); if (totalPrice > 100) { if (weight > 10) { return totalPrice * 0.2; } else { return totalPrice * 0.05; } } else { return 0; } }}

```
public class ShippingService {
    public double calculateShippingCost(Order order) {
        double totalPrice = order.getTotalPrice();
        double weight = order.getWeight();

        if (totalPrice > 100 && weight > 10) {
            return totalPrice * 0.2;
        }

        if (totalPrice > 100) {
            return totalPrice * 0.05;
        }

        return 0;
    }
}
```

Group time: Question 4
70.06

Question 5

Refactor the below code snippet without ChatGPT within 3 minutes. import java.util.concurrent.atomic.AtomicInteger; public class Order { private String customerName; private String productName; private double price; private int orderId; private static final AtomicInteger orderIdGenerator = new AtomicInteger(1000); public Order(String customerName, String productName, double price) { this.customerName = customerName; this.productName = productName; this.price = price; this.orderId = orderIdGenerator.incrementAndGet(); } public String toString() { String nameAndPrice = customerName + "," + String.valueOf(price); return nameAndPrice + "," + orderId; }}

import java.util.concurrent.atomic.AtomicInteger;
 	
public class Order {
  private final String customerName;
  private final String productName;
  private final double price;
  private final int orderId;
 	
  private static final AtomicInteger orderIdGenerator = new AtomicInteger(1000);
 	
  public Order(String customerName, String productName, double price) {
  this.customerName = customerName;
  this.productName = productName;
  this.price = price;
  this.orderId = orderIdGenerator.incrementAndGet();
  }
 	
  public String toString() {
  return String.format("%s,%.2f,%d", customerName, price, orderId);
  }
 	
  public int getOrderId() {
  return orderId;
  }
}

Group time: Question 5
68.65

Question 1 for Posttest (Part 3)

Refactor the below code snippet with ChatGPT within 3 minutes. public double getPayAmount() { double result; if (isDead) { result = deadAmount(); } else { if (isSeparated) { result = separatedAmount(); } else { if (isRetired) { result = retiredAmount(); } else { result = normalPayAmount(); } } } return result;}

public double getPayAmount() {
  double result;
  switch(true) {
  case isDead:
  result = deadAmount();
  break;
  case isSeparated:
  result = separatedAmount();
  break;
  case isRetired:
  result = retiredAmount();
  break;
  default:
  result = normalPayAmount();
  }
  return result;
}

Group time: Question 1 for Posttest (Part 3)
47.6

Question 2

Refactor the below code snippet with ChatGPT within 3 minutes. public class Customer { private String name; private String address; private double balance; public Customer(String name, String address) { this.name = name; this.address = address; this.balance = 0; } public void deposit(double amount) { this.balance += amount; } public void withdraw(double amount) { this.balance -= amount; } public double getBalance() { return balance; } }

```
public class Customer {
    private String name;
    private String address;
    private double balance;

    public Customer(String name, String address) {
        this.name = name;
        this.address = address;
        this.balance = 0;
    }

    public void deposit(double amount) {
        if (amount >= 0) {
            this.balance += amount;
        } else {
            throw new IllegalArgumentException("Deposit amount cannot be negative.");
        }
    }

    public void withdraw(double amount) {
        if (amount >= 0) {
            if (balance >= amount) {
                this.balance -= amount;
            } else {
                throw new IllegalArgumentException("Insufficient balance for withdrawal.");
            }
        } else {
            throw new IllegalArgumentException("Withdrawal amount cannot be negative.");
        }
    }

    public double getBalance() {
        return balance;
    }

    // Getter and setter methods for name and address
    public String getName() {
        return name;
    }

    public void setName(String name) {
        this.name = name;
    }

    public String getAddress() {
        return address;
    }

    public void setAddress(String address) {
        this.address = address;
    }
}
```

Group time: Question 2

42.36

Question 3

Refactor the below code snippet with ChatGPT within 3 minutes. public class Customer { private String name; private String address; private double balance; public Customer(String name, String address, double initialBalance) { this.name = name; this.address = address; this.balance = initialBalance; } public void processPayment(double amount) { if (amount > balance) { throw new InsufficientFundsException(); } balance -= amount; } public void printStatement() { System.out.println("Customer name: " + name); System.out.println("Customer address: " + address); System.out.println("Customer balance: " + balance); }}

```
public class Customer {
    private String name;
    private String address;
    private double balance;

    public Customer(String name, String address, double initialBalance) {
        this.name = name;
        this.address = address;
        this.balance = initialBalance;
    }

    public void processPayment(double amount) {
        if (amount > balance) {
            throw new InsufficientFundsException("Insufficient funds to process payment for customer: " + name);
        }
        balance -= amount;
    }

    public void printStatement() {
        System.out.println("Customer name: " + getName());
        System.out.println("Customer address: " + getAddress());
        System.out.println("Customer balance: " + getBalance());
    }

    public String getName() {
        return name;
    }

    public String getAddress() {
        return address;
    }

    public double getBalance() {
        return balance;
    }
}
```

Group time: Question 3

47.23

Question 4

Refactor the below code snippet with ChatGPT within 3 minutes. public class ShippingService { public double calculateShippingCost(Order order) { double totalPrice = order.getTotalPrice(); double weight = order.getWeight(); if (totalPrice > 100) { if (weight > 10) { return totalPrice * 0.2; } else { return totalPrice * 0.05; } } else { return 0; } }}

```
public class ShippingService {
    public double calculateShippingCost(Order order) {
        double totalPrice = order.getTotalPrice();
        double weight = order.getWeight();

        if (totalPrice > 100) {
            return (weight > 10) ? totalPrice * 0.2 : totalPrice * 0.05;
        } else {
            return 0;
        }
    }
}
```

Group time: Question 4
48.95

Question 5

Refactor the below code snippet with ChatGPT within 3 minutes. import java.util.concurrent.atomic.AtomicInteger; public class Order { private String customerName; private String productName; private double price; private int orderId; private static final AtomicInteger orderIdGenerator = new AtomicInteger(1000); public Order(String customerName, String productName, double price) { this.customerName = customerName; this.productName = productName; this.price = price; this.orderId = orderIdGenerator.incrementAndGet(); } public String toString() { String nameAndPrice = customerName + "," + String.valueOf(price); return nameAndPrice + "," + orderId; }}

import java.util.concurrent.atomic.AtomicInteger;
 	
public class Order {
  private String customerName;
  private String productName;
  private double price;
  private int orderId;
 	
  private static final AtomicInteger orderIdGenerator = new AtomicInteger(1000);
 	
  public Order(String customerName, String productName, double price) {
  this.customerName = customerName;
  this.productName = productName;
  this.price = price;
  this.orderId = orderIdGenerator.incrementAndGet();
  }
 	
  @Override
  public String toString() {
  return String.format("%s,%.2f,%d", customerName, price, orderId);
  }
}

Group time: Question 5
51.32

Interview Question (Part 4)

Can you share your experiences using ChatGPT for code refactoring? What were the specific benefits or advantages you observed during the process?
faster to process something

In what ways did ChatGPT enhance your productivity and efficiency in completing code refactoring tasks? Please provide specific examples.
While doing code review it can give different aspects and simplifies code review.

Did ChatGPT help you discover new refactoring techniques or approaches that you were previously unaware of? If yes, please elaborate on these insights.
Yes. you will not think of different solution every time. you can get different solutions from ChatGPT so that you can find that fits for your application or context.

How did ChatGPT contribute to the maintainability and readability of the code you produced during refactoring? Were there any notable improvements or challenges in this aspect?
challenges is that you can get wrong output from ChatGPT so you will need to verify whether the provided solution is correct or not. sometimes it doesn't give appropriate professional solutions. so you cannot absolutely rely on ChatGPT's answer all the time.

Were there any specific challenges or limitations you encountered while using ChatGPT for code refactoring? How did you overcome them, if at all?
It sometimes gives you non-optimal solution but you need to ask it to give more smarter answer or you need to specify what adjustment you need in that piece of code then it revises it's solution and after couple of attempts later you might get your desired solution.

In what scenarios do you believe AI assistance, like ChatGPT, is most beneficial for code refactoring? Conversely, are there situations where you think it might be less effective or not suitable at all?
you can get AI assistance for code refactoring but you cannot absolutely rely on it's answers. you have to verify it cause sometimes it will exaggerate the refactoring which is not professional way to do it.

How does ChatGPT's performance vary depending on the complexity of the code?
it takes a bit more time to answer and might not be able to give you proper answer directly.

Would you recommend ChatGPT to other Java programmers?
yes. only for code reviewing and still you will need to check things on your own.

Group time: Interview Question (Part 4)
614.15