



فاز دوم – تغییر نام

با پشت سر گذاشتن فاز اول ، حال ما دارای تمام اطلاعات مورد نیاز برای آغاز مبهم سازی خود هستیم که اولین قدم آن فرایند تغییر نام است. این پروسه در ظاهر ساده ترین قدم در مبهم سازیست زیرا صرفاً باید نام تمام متغیر های تعریف شده را تغییر داد. اما باید در نظر داشت که برای جلوگیری از سینتکس ارور و ارور های فانکشنال باید موارد استفاده از هر یک از این متغیر ها نیز عین به عین و طبق نام جدید آن تغییر کند.

به طور کلی شما مجدداً تکه کد ورودی را پارس کرده و با بررسی شناسه ها و تطبیق آن ها با دیتابیس موجود باید تصمیم بگیرید که در درجه ی اول آیا نام این شناسه باید تغییر کند و اگر این چنین است، این تغییر نام باید طی اعمال شروطی بر اساس اطلاعات دیتابیس باشد که از معادل بودن این دو و صحت عمل تغییر نام مطمئن شویم.

این شروط میتوانند شامل مواردی مانند تطبیق نام شناسه، نام فایل، کلاس استفاده شده و موارد دیگر باشد که پیدا کردن و اعمال آن ها بر عهده ی شماست.

```
class TestClass {
    public static void main(String[] args) {
        int a = 3;
        System.out.println(a + 2);
    }
}
```



```
class New_TestClass {
    public static void New_main(String[] args) {
        int New_a = 3;
        System.out.println(New_a + 2);
    }
}
```

همانطور که میدانید در فاز اول تمرکز ما تماماً بر تعریف شناسه ها بود اما در فاز دوم و سوم تمرکز اصلی ما به موارد استفاده و کاربرد های این شناسه ها معطوف میشود که به طور کلی دارای قوانین گرامری متفاوتی هستند و برای هر نوع تعریف تنها یک سری قانون مشخص قابل پذیرش هستند.

حل این صورت مسئله با روش های متنوعی قابل انجام است اما پیشنهاد میشود که برای هر یک از تایپ های قابل قبول که در فاز یک مورد استخراج قرار دادید، با دیباگ و تست های ساده، تایپ های متناظر که به عنوان موارد استفاده ی این تعاریف مطرح میشوند را پیدا کنید و در الگوریتم خود اعمال نمایید.

دلیل کلی این بررسی ها و اعمال شروط مختلف به این دلیل است که باز بودن دست برنامه نویس در نوشتن کد خود، حالات خاص بی شماری را بوجود می آورد که در این پروژه از بسیاری از آنها صرف نظر شده است اما مقابله ی حداقلی با برخی حالات خاص برای شاکله ی کلی یک مبهم سازی سطحی الزامیست.

این حالات خاص عموماً خود را در فرایند زمان گیر دیباگ نشان میدهند که ما برای سادگی کار از پروژه های شی گزایی ساده استفاده میکنیم تا هم با برخی از چالش های این فاز روبرو شوید هم زمان صرف شده برای دیباگ آن سرسام آور و بیش از حد درگیر کننده نباشد.

یکی از این حالات خاص که توجه خاصی به آن داریم، وجود شناسه های هم نام در فایل های مختلف هستند (میتوانند از تایپ های مختلف یا یکسان باشند) که پس از تغییر نام باید دارای نام های منحصر به فرد باشند، توجه به این نکته موجب میشود تا فیلترینگ اعمال شده در این پروسه دقیق تر شود و صرفاً توجه به نام شناسه کافی نباشد.

- فایل اول: MyClass.java

```
public class MyClass {  
    int data = 10;  
  
    public void displayData() {  
        System.out.println("Data value is: " + data);  
    }  
}
```

- فایل دوم: DataProcessor.java

```
public class DataProcessor {  
    public void data() {  
        System.out.println("Processing data...");  
    }  
  
    public void processData() {  
        data();  
    }  
}
```

در این مثال، هر دو کلاس از `data` استفاده می‌کنند، اما با معانی مختلف. در `MyClass`، `data` یک متغیر عددی است که مقداری دارد اما در کلاس `DataProcessor` یک تابع است که کار خاصی انجام می‌دهد.

پس از فرآیند تغییر نام در یک سیستم مبهم‌سازی کد، باید اطمینان حاصل شود که این دو استفاده متفاوت از `data` نام‌های جدید و منحصر به فردی دریافت می‌کنند، تا هنگام اجرای کد هیچ تداخلی بین آنها رخ ندهد.

فاز دوم پروژه

لطفا قبل از ارسال پروژه به نکات زیر توجه کنید:

۱. فایل ارسالی شما تنها یک فایل فشرده شده شامل کد و نتایج پروژه باشد. (.zip/.rar)

۲. نام فایل شما باید به صورت روبرو باشد:

YourFullName_YourStudentID

۳. توجه شود، ارائه به صورت مجازی برگزار خواهد شد، تمامی افراد گروه در روز ارائه باید حضور داشته باشند، در صورت عدم حضور هر یک از اعضا، نمره به آن شخص تعلق نخواهد گرفت.

🔔 مهلت ارسال در سامانه VC:

جمعه - ۱۸ خرداد - ۱۴۰۳

😊 موفق باشید