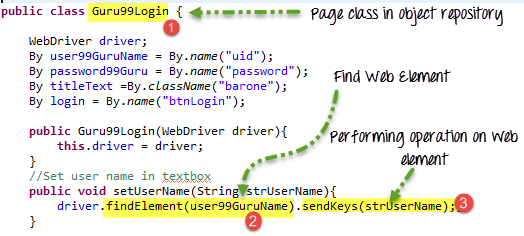
**Page Factory vs POM**

1. A Page Object Model is a test design pattern which says organize page objects as per pages in such a way that scripts and page objects can be differentiated easily. A Page Factory is one way of implementing PageObject Model which is inbuilt in selenium.
2. In POM, you define locators using ‘***By’***while in Page Factory, you use ***FindBy***annotation to define page objects.
3. **Page Object Model**is a **design approach** while **PageFactory**is a **class**which **provides implementation of Page Object Model**design approach.
4. **POM**is **not optimal**as it does not provide **lazy initialization**while Page Factory provides lazy initialization.
5. Plain POM will not help in **StaleElementReferecneException**while Page Factory takes care of this exception by relocating web element every time whenever it is used.
6. In plain page object model, you need to initialize every page object individually otherwise you will encounter NullPointerException while In PageFactory all page objects are initialized (Lazily) by using initElements() method.

**Page Object Model:**

**Page Object Model (POM)** is a design pattern, popularly used in test automation that creates Object Repository for web UI elements. The advantage of the model is that it reduces code duplication and improves test maintenance.



**What is Page Factory in Selenium?**

**Page Factory in Selenium** is an inbuilt Page Object Model framework concept for Selenium WebDriver but it is very optimized. It is used for initialization of Page objects or to instantiate the Page object itself. It is also used to initialize Page class elements without using “FindElement/s.”

Here as well, we follow the concept of separation of Page Object Repository and Test Methods. Additionally, with the help of class PageFactory in Selenium, we use annotations **@FindBy** to find WebElement. We use initElements method to initialize web elements

