۱۰ سؤال مفهومی از جلسه ۴۴

سؤال ۱

چرا وقتی توی کنترلر بعد از ارسال فرم `return view()`می‌کنیم، با مشکل مواجه می‌شیم؟

جواب:
چون ما در واقع مستقیم از **POST**، یک تکه HTML برمی‌گردونیم. این باعث میشه:

- اگر کاربر صفحه رو رفرش کنه، دوباره همون **POST**تکرار بشه (Duplicate Submission).
- با زدن دکمه Back یا Refresh مرورگر خطای *Confirm Form Resubmission*یا *Document Expired*نمایش داده بشه.

سؤال ۲

الگوی PRG (Post Redirect Get)دقیقا چه کاری انجام می‌ده؟

جواب:

- P = Post → کاربر فرم رو سابمیت می‌کنه.
- R = Redirect → اگر مشکلی بود (مثلاً ولیدیشن)، سرور به صفحه جدید) مثل login)با کد وضعیت ۳۰۲ ریدایرکت می‌کنه.
- G = Get → مرورگر دوباره یک درخواست GET برای صفحه ارسال می‌کنه و خطاها نمایش داده می‌شن.

اینطوری مشکل Refresh و Back حل میشه.

سؤال ۳

کد وضعیت 302در ریدایرکت چه معنایی داره؟

جواب:
یعنی "Found" یا "Moved Temporarily". مرورگر باید به یک URL جدید بره. این همون رفتاریه که توی PRG استفاده میشه.

سؤال ۴

تفاوت `$_SESSION['errors']` با فلش‌کردن خطاها (`$_SESSION['_flash']`)چیه؟

جواب:

- `$_SESSION['errors']`: **خطاها توی** session **می‌مونن تا وقتی کاربر مرورگر رو ببنده یا** logout **کنه.**
- `$_SESSION['_flash']`: **خطاها فقط برای** **یک درخواست بعدی** **نگهداری می‌شن و بعد خودکار حذف می‌شن.**

سؤال ۵

چرا بهتره کارهای مربوط به session **رو توی یک کلاس** Session **مدیریت کنیم؟**

جواب:

- برای اینکه کد تمیزتر بشه.
- از تکرار دستورات جلوگیری میشه.
- اگر بعداً خواستیم روش مدیریت session رو تغییر بدیم، فقط توی همون کلاس تغییر می‌دیم.

سؤال ۶

تفاوت `flush()` و `destroy()` در کلاس Session چیه؟

جواب:

- `flush()`: همه‌ی داده‌های session رو خالی می‌کنه ولی session هنوز فعاله.
- `destroy()`: علاوه بر خالی کردن session ، خود session رو نابود می‌کنه و کوکی مرتبط با اون هم حذف میشه.

سؤال ۷

چرا جفری گفت نمی‌خواد `$_SESSION['_flash']` مستقیم توی کنترلرها استفاده بشه؟

جواب:
چون این جزئیات داخلی (implementation detail) هستن. باید فقط از متدهای `Session::flash()` یا `Session::get()` استفاده کنیم تا اگر روزی ساختار تغییر کرد، کدهای دیگه دستکاری نشن.

سؤال ۸

روش کار `Session::get($key, $default)` چیه؟

جواب:
اول دنبال داده توی `$_SESSION['_flash']` می‌گرده.
اگر نبود، دنبال داده‌ی معمولی توی `$_SESSION` می‌گرده.
اگر هیچ‌کدوم نبود، مقدار `default` رو برمی‌گردونه.

سؤال ۹

چرا از `Session::has($key)` استفاده می‌کنیم؟

جواب:
برای اینکه راحت بتونیم بفهمیم یک کلید (مثل `errors`توی session وجود داره یا نه، بدون نیاز به نوشتن `isset($_SESSION['key'])`.

سؤال ۱۰

در نهایت، وقتی کاربر logout می‌کنه، چرا باید هم `flush()` داشته باشیم هم `destroy()`؟

جواب:
چون logout یعنی کاربر کامل خارج بشه. پس:

- flush **کافی نیست (session هنوز فعاله).**
- destroy **باعث میشه session از بین بره و کوکی هم حذف بشه → امنیت بالاتر.**

💻 ۱۰ تمرین کدنویسی

تمرین ۱

یک متد ساده `put` برای Session بنویسید.
جواب:

```
class Session {
    public static function put($key, $value) {
        $_SESSION[$key] = $value;
    }
}
```

تمرین ۲

متد `get` با مقدار پیش‌فرض پیاده‌سازی کنید.

جواب:

```
public static function get($key, $default = null) {
    return $_SESSION[$key] ?? $default;
}
```

تمرین ۳

متد `has` بسازید که وجود کلید رو بررسی کنه.

جواب:

```
public static function has($key) {
    return isset($_SESSION[$key]);
}
```

تمرین ۴

یک متد `flash` بسازید که داده رو فقط برای یک درخواست نگه داره.

جواب:

```
public static function flash($key, $value) {
    $_SESSION['_flash'][$key] = $value;
}
```

تمرین ۵

یک متد `unflash` بسازید که داده‌های فلش شده رو حذف کنه.

جواب:

```
public static function unflash() {
    unset($_SESSION['_flash']);
}
```

تمرین ۶

یک متد `flush` **بسازید که کل** session **رو خالی کنه.**

جواب:

```
public static function flush() {
    $_SESSION = [];
}
```

تمرین ۷

متد `destroy` **بنویسید تا کل** session **نابود بشه.**

جواب:

```
public static function destroy() {
    $_SESSION = [];
    if (ini_get("session.use_cookies")) {
        $params = session_get_cookie_params();
        setcookie(session_name(), '', time() - 42000,
            $params["path"], $params["domain"],
            $params["secure"], $params["httponly"]
        );
    }
    session_destroy();
}
```

تمرین ۸

یک فرم Login **ساده بسازید که اگر ولیدیشن خطا داشت، از** `Session::flash('errors', 'Invalid credentials')` **استفاده کنه و بعد** redirect **بشه.**

جواب (خلاصه):

```
if ($_POST['password'] !== 'secret') {
    Session::flash('errors', 'رمز عبور اشتباه است');
    header('Location: /login');
    exit;
}
```

تمرین ۹

در صفحه login اگر خطا وجود داشت، آن را نمایش دهید.

جواب:

```
if (Session::has('errors')) {
    echo "<p style='color:red'>" .
Session::get('errors') . "</p>";
}
```

تمرین ۱۰

در logout از `Session::destroy()` استفاده کنید.

جواب:

```
if (isset($_POST['logout'])) {
    Session::destroy();
    header("Location: /");
    exit;
}
```