Testing in DevOps

DOu – Certified Tester in DevOps (CTD)
Exercise Solutions

HO-3.2.2(HO-0)
Step 1 - Getting Sample Code(Dark Launch)

- Launch URL https://github.com/umangsaltuniv/EM_DarkLaunch
- Click Fork at right top section
- "EM_Dark Launch" repository will be added on your GitHub account

HO-3.2.2(HO-0)
Step 2 - Cloning Sample Code(Dark Launch)

- After forking "EM_Dark Launch" repository on GitHub Web, Click "Code" button
- Click "Open with GitHub Desktop"
- Click "Open GitHubDesktop"
- GitHub Desktop UI will be launched
- Browser **C:\DO-United\GitHubRepo** path under "Local path" section
- Click Clone
- GitHub Desktop UI will open the cloned repository and code will be downloaded on local machine under **C:\DO-United\GitHubRepo**

HO-3.2.2(HO-0)
Exercise - Demonstrate how dark launch can be implemented for a given system

- Implement dark launch on system

HO-3.2.2(HO-0)
Exercise Solution - Demonstrate how dark launch can be implemented for a given system

- Launch below URL in browser:

    https://app.launchdarkly.com/signup

- Do Sign Up

- Do Login if not already done

- Go to "Create a feature flag" section

- Enter feature flag name **login-button-color**

- Click CREATE FEATURE FLAG button

HO-3.2.2(HO-0)
Exercise Solution - Demonstrate how dark launch can be implemented for a given system

- Make sure Java & Production will be auto-filled under "Set up the application" section
- Scroll down the page and note down your SDK key

  E.g. sdk-14f672d4-aadd-40cf-975a-9da50ed3b146

- Go to Feature flags section on "Darkly" website
- Verify feature flag is created and turned-off
- Click login-button-color link

Set up the application

Select your language: Select your environment: ℹ

    🍸 Java   ▾   ■ Production   ▾
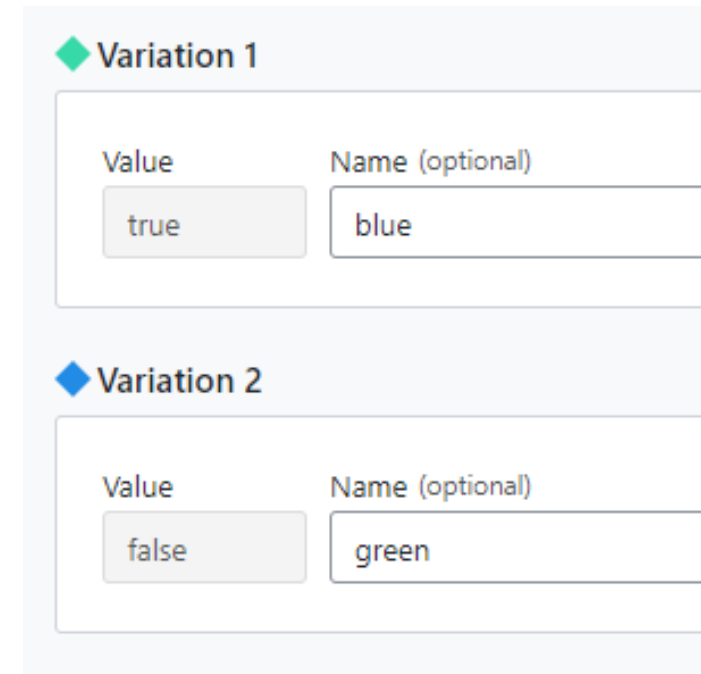
e. Then, in `main` create a new `LDClient` with your *environment-specific* SDK key:

```
public class App {
    public static void main(String... args) throws IOException {
        LDClient ldClient = new LDClient("sdk-14f672d4-aadd-40cf-975a-9da50ed3b146");
    }
}
```

HO-3.2.2(HO-0)
Exercise Solution - Demonstrate how dark launch can be implemented for a given system

- Go to Variations section

- Enter "blue" for true variation1

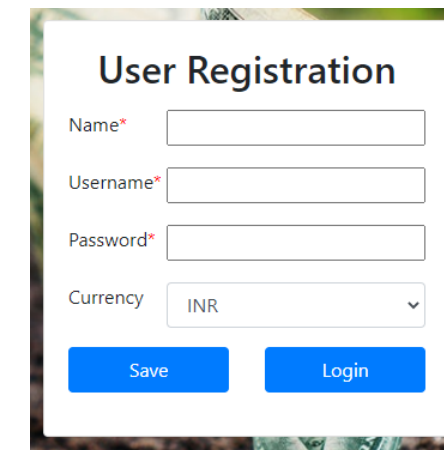- Enter "green" for false variation2

- Click SAVE WITH COMMENT button

HO-3.2.2(HO-0)
Exercise Solution - Demonstrate how dark launch can be implemented for a given system

- Open **EM_DarkLaunch** project code in eclipse(that you cloned earlier in pre-requisite section)

  Note: Steps for Opening the code in eclipse are given in Comment section

- Go to **UserController.java** file under src/main/java/com.expense.controller

- Update the sdk key in code that you noted down from Darkly website

- Launch the Expense Manager app by running **ExpenseApplication.java** file under s**rc/main/java/com.expense**

- Launch Expense Manager app on browser using

  http://localhost:8088

- Go to Expense Manager's Registration page

- Login button will be displayed in green color(as feature is turned-off on Darkly website)

- Now turn on the "login-button-color" feature from "Darkly" website by giving comment & environment(production)

- Now refresh the registration page, now Login button will be displayed in blue color(as feature is turned-on on Darkly website)