**BROWSER FACTORY** public class BrowserFactory { public enum Browser { CHROME, FIREFOX, IEXPLORER; } final WebDriver webDriver; DesiredCapabilities capabilities; switch (browserType) { case FIREFOX: FirefoxProfile profile = new FirefoxProfile(); profile.setAssumeUntrustedCertificateIssuer(false); webDriver = new FirefoxDriver(profile); break; case IEXPLORER: webDriver = new InternetExplorerDriver(); break; case OPERA: webDriver = new OperaDriver(); break; case SAFARI: webDriver = new SafariDriver(); break; case CHROME: webDriver = new ChromeDriver(); break; case PHANTOMJS: capabilities = DesiredCapabilities.phantomjs(); webDriver = new PhantomJSDriver(capabilities); break; case RWD\_CHROME: capabilities = DesiredCapabilities.chrome(); capabilities.setCapability("webdriver.chrome.driver", "~/Documents/chromedriver"); webDriver = new RemoteWebDriver(new URL("http://127.0.0.1:5551/wd/hub"), capabilities); break; case RWD\_PHANTOMJS: capabilities = DesiredCapabilities.phantomjs(); capabilities.setCapability("phantomjs.binary.path", "/usr/bin/phantomjs"); webDriver = new RemoteWebDriver(new URL("http://127.0.0.1:5551/wd/hub"), capabilities); break; case RWD\_FIREFOX: capabilities = DesiredCapabilities.firefox(); webDriver = new RemoteWebDriver(new URL("http://127.0.0.1:5551/wd/hub"), capabilities); break; case RWD\_IE8\_VM: capabilities = DesiredCapabilities.firefox(); //capabilities.setCapability("webdriver.ie.driver", "C:\\selenium\\IEDriverServer.exe"); webDriver = new RemoteWebDriver(new URL("http://10.10.22.113:5551/wd/hub"), capabilities); break; case SAUCELABS: capabilities = DesiredCapabilities.firefox(); capabilities.setCapability("name", "Lexus Selenium"); webDriver = new RemoteWebDriver( new URL("http://" + saucelabsUsername + ":" + saucelabsKey + "@ondemand.saucelabs.com:80/wd/hub"), capabilities); break; default: throw new IllegalArgumentException("Invalid browser type set in class injection " + browserType.getBrowserTypeString()); } initWebDriver(webDriver); return webDriver; }