

1. Projekt oks-08

Základní informace:

- **Účel:** prakticky se seznámit s použitím Selenium WebDriver a pokročilými možnostmi organizace rozsáhlejší skupiny testů
- **Kostrá:** oks-data-08.zip
- **Odevzdávaný soubor:** oks-08.jar

Zadání:

- napište automatizované testy pomocí Selenium WebDriver na všechny testovací případy, které selhaly v projektu oks-07
- testy již jsou začleněny do hierarchické struktury, takže je možné spouštění jednotlivých testů i jejich skupin

Popis vstupních dat:

- stáhněte si soubor oks-data-08.zip a rozbalte jej
 - všechny soubory jsou v kódování UTF-8 bez BOM
- v adresáři src/oks08 se nachází zdrojové .java soubory
 - tento adresář budete celý importovat do IntelliJ a budete doplňovat jen některé -- dále výslovně uvedené -- soubory
 - podrobný popis struktury celého adresáře je uveden v přednáškách
 - v adresáři src/oks08/tests je předpřipravená kompletní kostra testů ve struktuře odpovídající projektům oks-06 a oks-07 pro:
 - ♦ TC.01.01 -- Happy day scenario a Mazání
 - ♦ všech 11 selhaných testů z projektu oks-07
- v adresáři kontrola se nachází pro kontrolu před odevzdáním
 - adresář obsahuje následující podadresáře a soubor:
 - ♦ oks08 - adresář s testovacími .class soubory
 - ♦ vysledky - adresář se vzorovým souborem výsledku vzor-vysledky-testu-08.txt
 - ♦ kontrola.bat - překlad a spuštění kontrolního programu
- v adresáři lib se nachází potřebné knihovny pro JUnit 5 a Selenium WebDriver
 - adresář obsahuje jedenáct .jar souborů
 - je zde navíc i chromedriver-100.exe
 - ♦ jedná se o nouzové řešení pro první pokusné spuštění, které ale nemusí všem fungovat

- ♦ stáhněte si verzi driveru pro svoji platformu a pro aktuální verzi prohlížeče

Postup řešení:

- v IntelliJ založte nový projekt `oks-prj-08`

- nastavte celému projektu kódování UTF-8

File / Settings / Editor / File encodings / Project encoding: UTF-8

- do adresáře `src` projektu přetáhněte celý adresář `src/oks08`

- již známým způsobem připojte všechny JAR knihovny z adresáře `lib`

- **nebudete přidávat žádnou novou třídu !**

- budete jen doplňovat některé stávající

- otevřete soubor `src/oks08/support/Const.java` a upravte v něm cestu alespoň k jednomu driveru vašeho existujícího webového prohlížeče

- před tím zkontrolujte, že máte verzi driveru pro aktuální verzi prohlížeče
- nemusíte aktualizovat drivery pro všechny tři uvedené prohlížeče, stačí minimálně jeden

```
public static final String CHROME_DRIVER_URI =  
    "C:/Program Files/Java/selenium/chromedriver-100.exe";
```

- dále zde zkontrolujte volbu používaného prohlížeče

```
public static final Settings.Browsers WEB_BROWSER_TYPE =  
    Drivers.Browsers.CHROME;
```

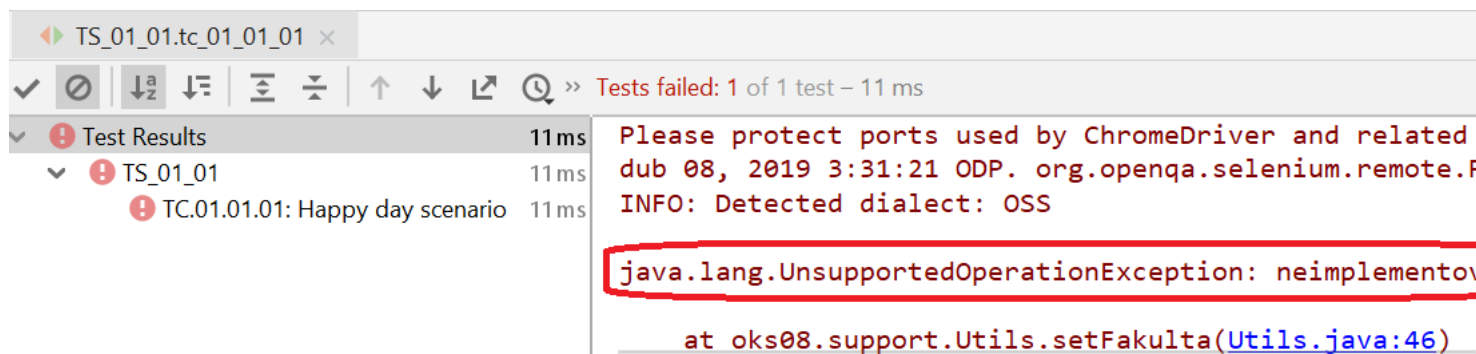
- otevřete soubor `src/oks08/tests/ts_01/TS_01_01.java`

- je v něm kompletní kód obou testovacích případů zde označených jako

- ♦ `tc_01_01_01()`

- ♦ `tc_01_01_02()`

- v tomto souboru nebudete nic měnit
- spustíte kontrolně `tc_01_01_01()`
- pokud je správně nastaven driver webového prohlížeče, spustí se prohlížeč a v zápětí skončí
- test selže, protože nejsou implementovány metody `Utils.setFakulta()` atd.



- na všech místech ve všech třídách, kde budete doplňovat kód, je příkaz:

```
throw new UnsupportedOperationException("neimplementovano");
```

■ ve třídě `Id` je předvyplněna většina ID

- doplňte zbývající

■ u všech testů neměňte nic z hlavičky testu, např. toto ponechte všude bez jakékoliv změny:

```
@Test
@DisplayName("TC.01.01.01: Happy day scenario")
@Tag(Tagy.AKTIVNI)
void tc_01_01_01() {
```

■ vraťte se ke třídě `TS_01_01`

- v ní nebudete nic měnit -- je kompletní, ale budete oba testy spouštět tak dlouho, dokud neprojdou
- pro úspěšné spuštění musíte implementovat metody ze třídy `Utills`
 - ♦ doplníte zde jednoduchý kód využívající Seleniových příkazů
 - žádná z metod by neměla mít tělo delší než čtyři řádky

Note

Zde si uvědomte výhodu oddělení části Selenia od části testů.

- pokud projdou úspěšně oba testy, pak by doplňování následujících testů měla být jen rutinní práce
 - ♦ má však u některých testů svá specifika

■ `TS_04_02_01`

- zde je většina nastavení pro oba testy stejné -- umístěte jej do `@BeforeEach`

■ `TS_04_02_02`

- běžný test, který by již neměl dělat žádné problémy

■ `TS_04_02_05`

- běžný test, který by již neměl dělat žádné problémy

■ TS_04_02_06

- běžný test, který by již neměl dělat žádné problémy

■ TS_04_03_01

- použijte `Utils.isZobrazenaChyba()`

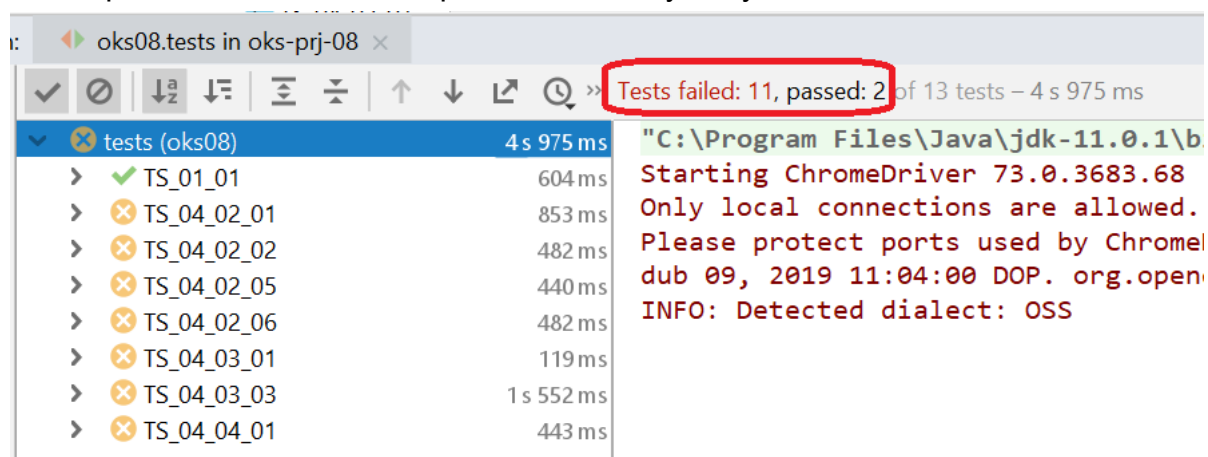
■ TS_04_03_03

- zde se téměř doslovně opakují čtyři testy
 - ♦ společné nastavení provedte v `@BeforeEach`
 - ♦ vyhodnocení testu bude opět pro všechny stejné, proto bude provedeno v `@AfterEach`, které nechte beze změny
 - zde je s výhodou použít `assertAll()`, který vyhodnotí tři věci najednou

■ TS_04_04_01

- běžný test, který by již neměl dělat žádné problémy

■ po dokončení úprav ve všech testech spusťte z všechny testy z adresáře `tests`



■ následně spusťte `Kontrola_Prj_08`

- výsledek vypsaný do nově vzniklého souboru `vysledky-testu-08.txt` by měl být totožný s obsahem souboru `vzor-vysledky-testu-08.txt`

■ můžete provést experimenty s jinými driversy apod.

■ po skončení práce nastavte v `Const` driver `HtmlUnit`

```
public static final Settings.Browsers WEB_BROWSER_TYPE =  
    Drivers.Browsers.HTML_UNIT;
```

- a ještě jednou kontrolně spusťte `Kontrola_Prj_08`
 - ♦ občas testy proběhnou rychle, ale pak se dlouho (desítky sekund) čeká na ukončení práce `HtmlUnit` driveru

Warning

Bez tohoto nastavení nebudou moci být testy spuštěny na validátoru a bude vyhozena výjimka:
`JupiterEngineExecutionContext close SEVERE`

Kontrola úplnosti řešení:

- adresář `src` z IntelliJ překopírujte někam do svého pomocného adresáře, např. `D:\zzz`
 - **vymažte** z něj `zzz/src/oks-08/Kontrola_Prj_08.java`
 - dále sem překopírujte obsah adresáře `kontrola/oks08` a soubor `kontrola.bat`
 - dále sem překopírujte obsah adresáře `lib`
- spusťte soubor `kontrola.bat` nebo z něj použijte příslušné příkazy
 - Pozor - pokud pracujete v Linuxu, bude zřejmě nutné v nastavení *classpath* změnit oddělovač souborů `:`:

```
-cp apiguardian-api-1.0.0.jar;junit-jupiter-api-5.4.1.jar ...
```

na:

```
-cp apiguardian-api-1.0.0.jar:junit-jupiter-api-5.4.1.jar ...
```

a generování `sources.txt` je:

```
find -name "*.java" > sources.txt
```

- porovnejte obsahy souborů `vysledky-testu-08.txt` a `vysledky/vzor-vysledky-testu-08.txt`
 - postupujte stejně jako v projektu `oks-01` včetně řešení případného problému s kódováním diakritiky
 - pokud nebudou identické, tak ani neodevzdávejte a hledejte příčinu rozdílu

Příprava souborů k odevzdání:

- v adresáři, ve kterém byla prováděna kontrola, použijte příkaz:

```
jar cMf oks-08.jar src
```

- výsledkem bude soubor `oks-08.jar`, který budete odevzdávat
 - opravdu není spustitelný, neobsahuje `.class` soubory ani Javadoc
 - jeho velikost by měla být do 20 KB - pokud je větší, je v něm něco navíc
- po úspěšném odevzdání na Portále, proklikněte výsledné OK a, prosím, vyplňte v tabulce časovou náročnost této úlohy