

# 1. Domácí úloha 04

## Základní informace:

- **Účel:** příprava a využití `Přeppravka`, statická tovární metoda, výčtový typ
- **Kostrá:** `04_PresouvaniOsob.zip`
- **Odevzdávaný soubor:** `04_PresouvaniOsob.jar`

## Zadání:

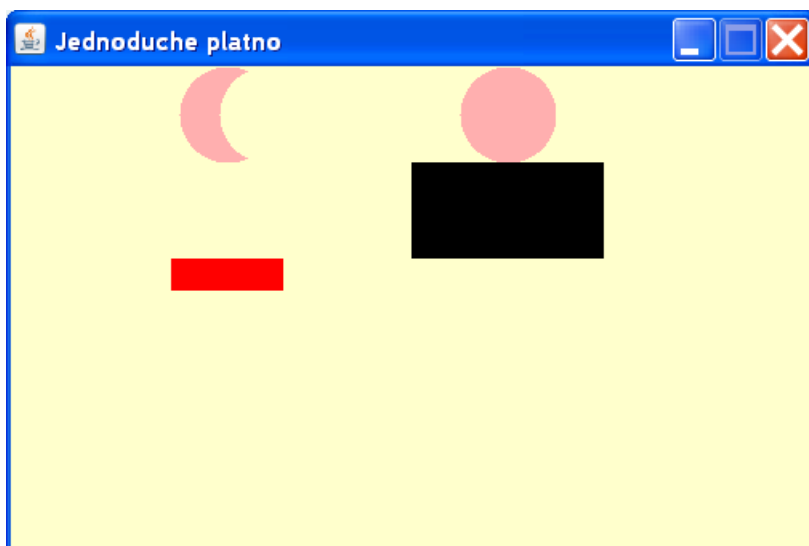
- připravte třídu `Rozmer`, která je podle návrhového vzoru `Přeppravka`
- připravte výčtový typ `Pohlavi`
- upravte třídu `Osoba`, kterou jste vytvářeli v minulém DU
- nijak nedoplňujte předpřipravenou třídu `TestOsoby` (budete v ní pouze postupně odkomentovávat těla testovacích metod)
- do Portálu odevzdáte JAR soubor celého projektu

## Postup řešení:

- stáhněte si soubor `04_PresouvaniOsob.zip`, rozbalte jej - NEotvírejte projekt v BlueJ
- do rozbaleného adresáře nakopírujte soubor `Osoba.java`, který jste odevzdávali v minulém DU
- v BlueJ otevřete projekt `04_PresouvaniOsob`
- připravte třídu `Rozmer`, která je podle návrhového vzoru `Přeppravka`
  - třída má atributy
    - ♦ `int sirka`
    - ♦ `int vyska`
  - napište metody
    - ♦ `public Rozmer(int sirka, int vyska)`
    - ♦ `public int getSirka()`
    - ♦ `public int getVyska()`
    - ♦ `public String toString()`, která vrátí např.:

```
Rozmer[sirka=200, vyska=100]
```
  - při vytváření třídy `Rozmer` pište průběžně Javadoc komentáře
  - třídu `Rozmer` otestujte pomocí příkazu `testRozmer()` třídy `TestOsoby`

- pro účely duck-testů doplňte bezparametrický konstruktor `public Rozmer()`, který nastaví šířku i výšku na nulové hodnoty
  - ♦ využijte možnost přetížení konstruktorů známou již ze třídy `Osoba`
  - ♦ bezparametrický konstruktor otestujte pomocí příkazu `testBezparRozmer()` třídy `TestOsoby`
- výslednou třídu `Rozmer` otestujte pomocí PMD a odstraňte případné problémy
- upravte třídu `Osoba` tak, že bude umět využívat `Přepřavky` `Pozice` a `Rozmer`
  - přidejte konstruktory (pouze kopírujte a vhodně doplňujte volání již existujících konstruktorů):
    - ♦ `public Osoba(Pozice pozice, Barva barvaTela)`
    - ♦ `public Osoba(Pozice pozice, int velikostHlavy, Barva barvaTela)`
    - ♦ `public Osoba(Pozice pozice, int velikostHlavy, double pomerHlavaTelo, double pomerTelo, Barva barvaTela)`
    - ♦ správnost funkce konstruktorů ověřte pomocí `testKonstruktorůPozicRůznýchOsob()`, kterou před prvním použitím odkomentujte
  - přidejte metody:
    - ♦ `public void nakresli()` - vykreslí hlavu i tělo instance voláním jejich metod `nakresli()`
    - ♦ `public Pozice getPozice()` - vrátí pozici instance
    - ♦ `public void setPozice(int x, int y)` - přemístí instanci na dané souřadnice
      - pozor na skutečnost, že hlava může být širší než tělo a naopak
        - je vhodné určit si vektor posuvu osoby (rozdíl původních a nových souřadnic), který se pak aplikuje na hlavu i na tělo jednotně, tzn. přičítá se k jejich dosavadním souřadnicím
    - ♦ `public void setPozice(Pozice p)` - přemístí instanci na danou pozici
    - ♦ `public Rozmer getRozmer()` - vrátí rozměr instance
  - správnost funkce metod ověřte pomocí `testProhodPozice()`, kterou před prvním použitím odkomentujte (včetně pomocné metody `pomProhodPozice()`)



- ♦ chybné zobrazení ženy po provedení přesunu je záměrná nedokonalost v testu, která nemá nic společného v kódu ve třídě `Osoba`
- ♦ tuto nedokonalost se nesnažte nijak odstraňovat, pouze si uvědomte, že existuje a zdůvodněte si, proč vznikla

■ upravte třídu `Osoba` tak, že bude dávat k dispozici statické tovární metody

- přidejte statické tovární metody `getBeznyMuz(Pozice pozice)` a `getBeznaZena(Pozice pozice)`, které vrátí instance běžného muže (modrá barva těla) a ženy (červená barva těla)
- správnost funkce metod ověřte pomocí `testStatickaTovarniMetoda()`, kterou před prvním použitím odkomentujte

■ připravte třídu výčtového typu `Pohlavi`

- třída bude velice jednoduchá, bude mít pouze instance `MUZ` a `ZENA`
- nesnažte se vkládat do tohoto typu další informace, například o barvě náležející pro `MUZ` či `ZENA` - podobné informace sem nepatří

■ upravte třídu `Osoba` tak, že bude umět využívat výčtový typ `Pohlavi`

- přidejte konstruktor (pouze kopírujte a vhodně doplňujte):

- ♦ kontrakt a signatura jsou

```

/*****
 * Vytvoří na definované pozici
 * instanci o implicitních rozměrech hlavy a těla
 * Pohlavi.MUZ bude mít modrou barvu těla
 * Pohlavi.ZENA bude mít červenou barvu těla
 *
 * @param pozice pozice instance
 * @param pohlavi jedna z hodnot výčtového typu Pohlavi
 */
public Osoba(Pozice pozice, Pohlavi pohlavi)

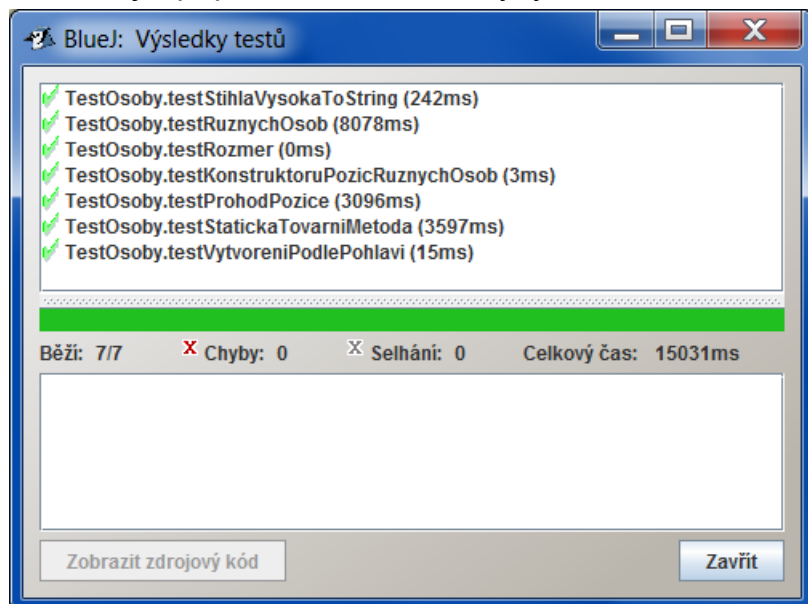
```

- ♦ implementační poznámka - nejprve vytvořte standardní osobu s implicitní barvou těla

- ♦ dalšími příkazy tělo přebarvěte na požadovanou barvu, použijte např. konstrukci:

```
if (pohlavi == Pohlavi.MUZ) {
```

- správnost funkce konstrukturu ověřte pomocí *testVytvoreniPodlePohlavi()*
  - výslednou třídu `Osoba` otestujte pomocí PMD a odstraňte případné problémy
- spusťte příkazem *Testovat* vše z místní nabídky třídy `TestOsoby`
- odstraňujte případné zavlečené chyby, dokud nedostanete konečný výsledek



- na závěr otestujte pomocí Duck-testů celou svoji práci a odstraňte případné problémy
- celý projekt již známým způsobem zabalte do JAR souboru `04_PresouvaniOsob.jar`, který budete odevzdávat