

Класе, Наслеђивање и Git

- Више не качимо у формату Zadatak1, Zadatak2, ...
- Качимо имена класа директно, нпр Sastojak.java, Rezept.java, ...

Конкретно за овај домаћи:

```
У фолдеру NedeljaX/sreda
Налазе се класе:
- Sastojak.java
- SastojakRecepta.java
- Rezept.java
- Main.java
```

- За домаћи:
 - Задаци 0,1,2,3
 - Не обазирати се на toString у трећем задатку ако вам прави проблем
 - Задатак 0 радите у току рада, тестирајте сваку класу редом
 - Покушајте да урадите први део 4. задатка, ако сте урадили све остало
 - Прављење GitHub налога
 - Качење прва три задатка на GitHub (Мора и даље на GDrive такође)
 - Име репозиторијума: **IT-Bootcamp-QA-Domaci** - Или ћирилични/енглески еквивалент
 - Име фолдера у ком се налазе класе: **Nasledjivanje** - Или ћирилични/енглески еквивалент

- Додати све get и set методе које су вам потребне
- Додати све конструкторе из примера, и додатне ако су вам потребни

0. Main класу која садржи main методу

- У њој инстанцирати креиране класе (Направити објекте)
- Такође позвати све направљене методе (Тестирати методе и објекте)

1. Направити класу Састојак која садржи следеће:

- Назив састојка
- Цену састојка (По јединици мере, због једноставности)
 - Јаје - Цена по Комаду
 - Млеко - Цена по Милилитру
 - Брашно - Цена по Граму

```
// Пример прављења:
new Sastojak("Јаје", 12.5);
```

2. Направити класу СастојакРецепта која наслеђује Састојак и има додатно:

- Количина састојка (У јединици мере, као изнад)
 - Описује колико нам састојка треба за неки конкретан рецепт
- Метода: **ukupnaCena()**
 - Враћа укупну цену састојка

```
// Пример прављења и коришћења:
x = new SastojakRecepta("Brasno", 0.075, 300.0); // Ово описује састојак који кошта 0.075 динара по граму, а
за рецепт је потребно 300 грама
x.ukupnaCena(); // Враћа 22.5 | Објашњење: 300 * 0.075
```

3. Направити класу Рецепт која садржи следеће:

- Назив рецепта
- Тежину рецепта (Број од 1 до 5)
 - Редом означавају следеће нивое: Почетни, Лак, Средњи, Тежак, Мајсторски
- Колекцију састојака (типа СастојакРецепта)
- Подразумевани конструктор, get и set методе за све
- `dodajSastojak(SastojakRecepta sastojak)`
 - Додаје састојак у колекцију
- `ukloniSastojak(SastojakRecepta sastojak)`
 - Брише састојак из колекције (Водити рачуна о референцама/вредностима)
- `toString` враћа String у следећем формату:

Шпагете у белом сосу:
Тежина: Средњи ниво
Састојци: Шпагете, Павлака за кување, Бели лук, Ћуреће месо, Маслац, Јаја, Бибер, Со

- Напомена: Океј је и неки други формат ако је све читко и садржи МАКАР исто информација као пример изнад!
- Објашњење: Ниво рецепта је био 3, и зато исписујемо `Средњи ниво`

4. На класу Рецепт додати методе које враћају:

- Укупну цену рецепта
- Рецепт скалиран за X процената
 - Враћа нови Рецепт који садржи све исте састојке и атрибуте
 - Али сваки састојак је скалиран (нпр ако је X = 50, количина сваког састојка се полови)

5. Направити статичке методе (Могу да буду у класи Рецепт):

- `filtrirajRecepte(ArrayList<Recept> recepti, int tezina)`
 - Враћа све рецепте одређене тежине
- `filtrirajRecepte(ArrayList<Recept> recepti, int[] tezina)`
 - Враћа све рецепте одређених тежина