

Практика к Лекции 3

Упражнение 1-1 Использование конструкторов.

Цель упражнения: Изучить предназначение конструкторов.

Описание упражнения: В этом упражнении вы добавите в классы проекта `StockListProject` необходимые конструкторы.

1) Добавьте в класс `GenericItem` три конструктора:

- i) `public GenericItem(String name, float price, Category category)`
- ii) `public GenericItem(String name, float price, GenericItem analog)`
- iii) `public GenericItem()`

2) Добавьте в класс `FoodItem` три конструктора:

- i) `public FoodItem(String name, float price, FoodItem analog, Date date, short expires)`
- ii) `public FoodItem(String name, float price, short expires)`
- iii) `public FoodItem(String name)`

3) Частные конструкторы класса `FoodItem` (№ 2 и 3) должны обращаться к общему конструктору (№ 1), передавая ему часть параметров в виде значений по умолчанию.

4) Добавьте в класс `GenericItem` статическое поле `static int currentID`, хранящее максимальный назначенный ID товара в текущей сессии.

5) Добавьте в конструкторы класса `GenericItem` строку, автоматически инициализирующую поле ID товара очередным свободным номером: `this.ID = GenericItem.currentID++;`

Упражнение 2-1. Обобщенное программирование

Цель упражнения: Изучить возможности, предоставляемые настраиваемыми типами.

- **Описание упражнения:** В этом упражнении вы воспользуетесь возможностями `generics` для создания настраиваемых типов.

1) Требуется создать класс, в котором есть метод `sum` для получения суммы числового массива.

Точный тип элементов массива не определен, известно только, что они являются наследниками числового типа `Number`.

- a) Создать класс `U0901WorkArray`. В заголовке класса должно быть указано ограничение по типу `Number` - `<T extends Number>`
- b) На уровне экземпляра класса должна быть объявлена переменная `arrNums` – массив с типом, указанным в ограничении:
`T[] arrNums;`
- c) В классе требуется объявить конструктор с параметром `numP` – массивом, с типом соответствующим ограничению. В этом конструкторе переменной `arrNums` присваивается

ссылка входного параметра:

```
arrNums=numP;
```

- d) Создаем метод `sum` без входных параметров, но с возвращаемым типом – `double`. (Чтобы хватило на любого наследника типа `Number`).
- e) В методе `sum` объявляется переменная `doubleWork` типа `double`, в которую в цикле `for` будут инкрементироваться значения массива `arrNums`.
- f) Значение переменной `doubleWork` возвращается из метода `sum`.
- g) Создать класс `U0901Main` с методом `main` для проверки работоспособности класса `U0901WorkArray`.
- h) В методе `main` этого класса объявить 4 переменные:
 - i) Массив `Integer`-значений `intArr` и заполнить его несколькими значениями:

```
Integer intArr[]={10,20,15}
```
 - ii) Массив `Float`-значений `floatArr` и заполнить его произвольными значениями в цикле `for` или `while`.
 - iii) Экземпляр класса `U0901WorkArray` с именем `insWorkArrayInt` и инициировать ее конструктором, в котором в качестве параметра передается массив `intArr`.
 - iv) Экземпляр класса `U0901WorkArray` с именем `insWorkArrayFloat` и инициировать ее конструктором, в котором в качестве параметра передается массив `floatArr`.
- i) Для переменных-экземпляров класса `U0901WorkArray` вызвать метод `sum` и вывести на экран полученное значение.
- j) При желании, можно проверить работоспособность ограничения – создать массив `String`-значений и передать его в качестве параметра в конструктор экземпляра класса `U0901WorkArray`. Должна быть ошибка времени компиляции.

Упражнение 3-1. Работа со строками

Цель упражнения: Изучить методы класса `String`.

Описание упражнения: В этом упражнении вы воспользуетесь возможностями класса `String` для разбора строки текста. Строка будет представлять собой структурированное описание товара, на основе которого требуется создать экземпляр класса `FoodItem`. Полученный алгоритм будет в дальнейшем использован для массовой загрузки описаний товаров из текстового файла.

- 1) Пусть имеется текстовая строка `"Конфеты 'Маска' ;45;120"`, хранящая информацию о пищевом товаре в формате `<name>;<price>;<expires>`. Объявите в методе `main` класса `Main` переменную `line` типа `String`. Присвойте ей значение `"Конфеты 'Маска' ;45;120"`.
- 2) Объявите массив текстовых строк `item_fld`.
- 3) Разбейте строку `line` на поля (без разделителей) и заполните значениями полей массив `item_fld` (используйте метод `split()` класса `String`).

- 4) Создайте новый объект класса `FoodItem` на основании элементов массива `item_fld[]`.
Вызовите у этого объекта метод `printAll`, чтобы убедиться в его корректном создании.