Назначение лекции

Знакомство с особенностями ввода текущей даты в требуемом формате и специальной структурой данных «перечисление».

Новые слова

**Перечисление** – это специальная структура для хранения данных, схожего значения..

Что было сделано

В предыдущей работе наше приложение MyEvent научилось выдавать список доступных режимов работы и выполнять контроль правильности номера режима. Если номер был введен не верно, то ввод повторялся.

Продолжение

В этой лекции начинаем рассматривать реализацию режима «Ввод». В этом режиме предполагается вводить текущую дату, название события, количество времени, состояние и дату выполнения события.

В лекции пока объясняется реализация ввода текущей даты и названия события.

Для реализации потребуются дополнительные классы:

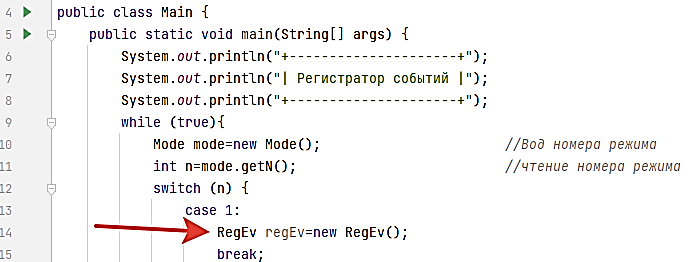
RegEv – регистратор событий

DateReg – регистратор текущей даты

DirEnum – главное перечисление

ReadNameEv – читатель названия события

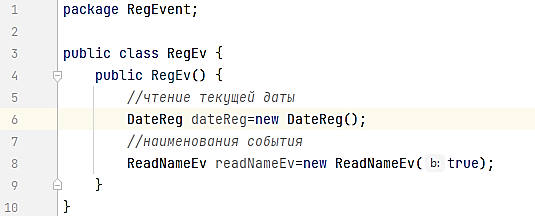
Для создания объекта regEv определим место в конструкторе класса Main, для режима 1:



RegEv regEv=new RegEv();

Создание объекта regEv принадлежащий классу RegEv

# Класс RegEv



DateReg dateReg=new DateReg();

Создание нового объекта dateReg принадлежащий классу DateReg

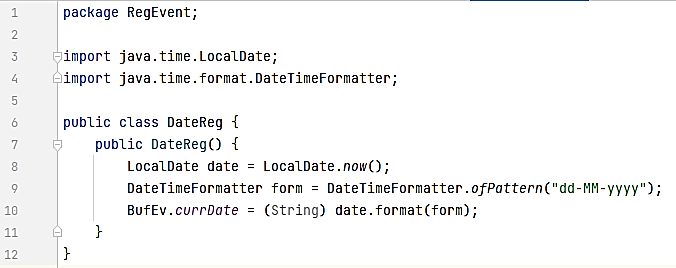
Класс предназначен для ввода текущей даты. При этом автоматически вызывается конструктор класса и выполняется код, который размещен в конструкторе.

ReadNameEv readNameEv=new ReadNameEv(true);

Создание нового объекта readNameEv принадлежащий классу ReadNameEv

Класс предназначен для ввода названия события. При этом автоматически вызывается конструктор класса и выполняется код, который размещен в конструкторе. В конструктор передается параметр (здесь, true). Назначение параметра: запрашивать имя события или нет. Из дальнейшего будет понятно назначение.

# Класс DateReg



Для работы с датой в Java предлагается несколько стандартных библиотек.

**import java.time.LocalDate;**

Одна из библиотек для работы с датой.

**import java.time.format.DateTimeFormatter;**

Библиотека для работы с различными форматами даты

**LocalDate date = LocalDate.*now*();**

Создание нового объекта **date** который содержит текущую дату для данного региона. Дата представлена в своем особом формате.

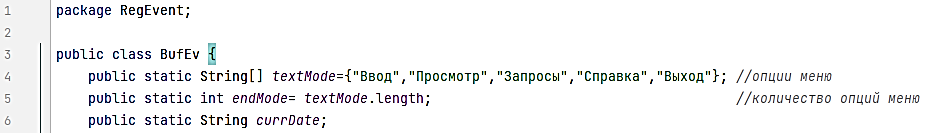
**DateTimeFormatter form = DateTimeFormatter.*ofPattern*("dd-MM-yyyy");**

Определяет формат **form** в виде **dd-MM-yyyy (**день, месяц, год)

**BufEv.*currDate* = (String) date.format(form);**

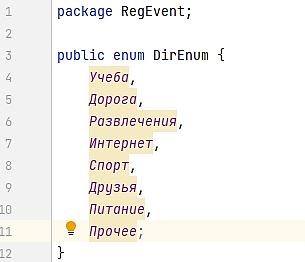
В строковую переменную ***currDate*** которая находится в классе **BufEv** записывается дата из **date** в формате **form**

Переменную ***currDate*** необходимо объявить в классе **BufEv**



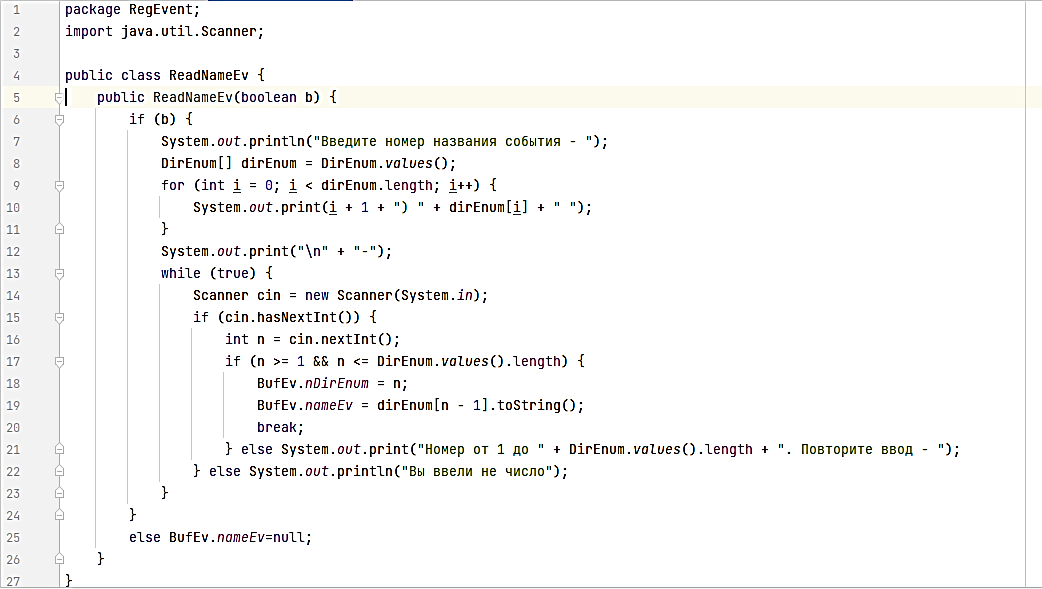
# Перечисление DirEnum

Перечисление – это специальная структура для данных, схожего значения.



# Класс ReadNameEv

В классе реализован алгоритм ввода наименования события, на основе перечисления *DirEnum*



**import java.util.Scanner;**

Билиотека для потокового ввода с клавиатуры

**if (b) {**

Подтверждение на ввод названия события. Если не подтверждается, то ввод наименования не будет выполнятся и название события будет пустым. Такой режим понадобится в дальнейщем.

**System.*out*.println("Введите номер названия события:");**

Выдача на экран приглашения

**DirEnum[] dirEnum = DirEnum.*values*();**

Создание массива **dirEnum** элементами которого будут элементы перечисления **DirEnum .** Массив более удобен для работы.

**for (int i = 0; i < dirEnum.length; i++) {**

Цикл для вывода массива названий событий.

**System.*out*.print(i + 1 + ") " + dirEnum[i] + " ");**

Выдача названий событий в виде:



**System.*out*.print("\n" + "-");**

Приглашение для ввода номера события на следующей строке.

while (true) {

Бесконечный цикл для ввода названий событий, пока не будет введено верно.

**Scanner cin = new Scanner(System.*in*);**

Ввод в поток **cin** данных с клавиатуры

**if (cin.hasNextInt()) {**

Проверка потока на целое число.

**int n = cin.nextInt();**

Запись в переменную **n** номера события из потока **cin**

**if (n >= 1 && n <= DirEnum.*values*().length) {**

Проверка номера на допустимость, где **length** количество элементов в перечислении.

**BufEv.*nDirEnum* = n;**

Запись в переменную ***nDirEnum*** порядкового номера события из перечисления

**BufEv.*nameEv* = dirEnum[n - 1].toString();**

break;

Запись в переменную ***nameEv*** названия события в виде строки. При этом происходит выход из цикла.

**} else System.*out*.print("Номер от 1 до " + DirEnum.*values*().length + ". Повторите ввод - ");**

Условие перехода к повтору цикла при ошибочном номере названия события.

**} else System.*out*.println("Вы ввели не число");**

Условие перехода к повтору цикла, если ввели не число.

}

Конец цикла

}

Конец условия if (b) {

else BufEv.*nameEv*=null;

Условие для отсутствия ввода номера названия события.

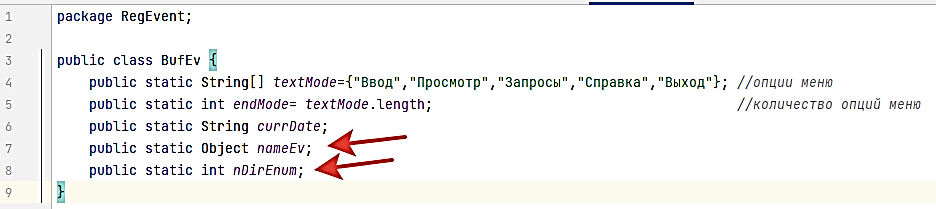
}

Конец конструктора

}

Конец класса

# Изменения в классе BufEv

Конец лекции