



Date/time



Определение

Дата и время — 2 варианта

представления:

- Человеческое время — часы, минуты, дни, недели, месяцы
- Машинное время — миллисекунды от нулевой точки отсчета (1 января 1970 года, 00:00:00)

В Java для работы с датами и временем существует несколько ключевых классов в пакете `java.time`, представленных в Java 8 и позже.

Эти классы предназначены для упрощения работы с датами, временем и их форматированием.



class LocalDate

```
package 145.slides;

import java.time.LocalDate;

public class LocalDateExample {
    public static void main(String[] args) {
        LocalDate today = LocalDate.now();
        System.out.println("Сегодня. дата: " + today);

        LocalDate specificDate = LocalDate.of(
            2024, 3, 16
        );
        System.out.println("Дата: " + specificDate);

        LocalDate tomorrow = today.plusDays(1);
        System.out.println("Завт. дата: " + tomorrow);

        LocalDate yesterday = today.minusDays(1);
        System.out.println("Вчер. дата: " + yesterday);
    }
}
```

LocalDate представляет дату без времени и часового пояса (используется время вашего компьютера). Этот класс часто используется, когда вам нужно работать только с датой.



class LocalDateTime

```
package l45.slides;

import java.time.LocalDateTime;

public class LocalDateTimeExample {
    public static void main(String[] args) {
        LocalDateTime now = LocalDateTime.now();
        System.out.println(
            "Текущие дата и время: " + now
        );

        LocalDateTime specificDateTime = LocalDateTime.of(
            2024, 3, 18, 13, 45
        );
        System.out.println(
            "Конкретные дата и время: " + specificDateTime
        );

        LocalDateTime later = now.plusHours(2)
            .plusMinutes(30);
        System.out.println(
            "Позже на 2 часа 30 минут: " + later
        );
    }
}
```

LocalDateTime комбинирует дату и время, но, как и LocalDate, не содержит информации о часовом поясе.



class DateTimeFormatter

```
package l45.slides;

import java.time.LocalDate;
import java.time.LocalDateTime;
import java.time.format.DateTimeFormatter;

public class DateTimeFormatterExample {
    public static void main(String[] args) {
        LocalDate today = LocalDate.now();

        DateTimeFormatter formatter = DateTimeFormatter
            .ofPattern("dd.MM.yyyy");
        String formattedDate = today.format(formatter);
        System.out.println(
            "Отформатированная дата: " + formattedDate
        );

        LocalDateTime now = LocalDateTime.now();
        DateTimeFormatter formatter2 = DateTimeFormatter
            .ofPattern("dd.MM.yyyy HH:mm:ss");
        String formattedDate2 = now.format(formatter2);
        System.out.println(
            "Отформатированная дата-время: " + formattedDate2
        );
    }
}
```

DateTimeFormatter используется для форматирования и разбора дат (парсинг, преобразование строки в объект) в соответствии с заданным шаблоном.



class DateTimeFormatter

```
package l45.slides;

import java.time.LocalDate;
import java.time.LocalDateTime;
import java.time.format.DateTimeFormatter;

public class DateTimeFormatterParseExample {
    public static void main(String[] args) {
        String dateString = "04.07.1776";
        DateTimeFormatter formatter = DateTimeFormatter
            .ofPattern("dd.MM.yyyy");
        LocalDate parsedDate = LocalDate
            .parse(dateString, formatter);
        System.out.println(
            "Разобранная дата: " + parsedDate
        );

        String dateString2 = "04.07.1776 12:12:43";
        DateTimeFormatter formatter2 = DateTimeFormatter
            .ofPattern("dd.MM.yyyy HH:mm:ss");
        LocalDateTime parsedDate2 = LocalDateTime
            .parse(dateString2, formatter2);
        System.out.println(
            "Разобранная дата-время: " + parsedDate2
        );
    }
}
```

DateTimeFormatter используется для форматирования и разбора дат (парсинг, преобразование строки в объект) в соответствии с заданным шаблоном.



Контрольная точка

- Понятно ли?

Если все ясно, ставим плюсы, иначе - задаем вопросы.



The end