

## Практическое занятие № 11

**Тема:** Коллекции, списки, одномерные и многомерные массивы, работа с датой и временем в ЯП Java, Kotlin и JavaScript. Использование подпрограмм в ЯП Java, Kotlin и JavaScript. Сравнение возможностей языков, их достоинства и недостатки.

### Литература:

1. Монахов В.В. Язык программирования Java. СПб.: БХВ-Петербург, 2011. 704 с.
2. Шилдт Г. Java: руководство для начинающих. М.: Вильямс, 2012. 624 с.
3. Гонсалвес Э. Изучаем Java EE 7. СПб.: Питер, 2014. 640 с.
4. Скин Дж., Гринхол Д. Kotlin. Программирование для профессионалов. СПб.: Питер, 2020. – 464 с.: (Серия «Для профессионалов»).
5. Хорстман К.С. Современный JavaScript для нетерпеливых. Addition Wesley. Перевод на русский. ДМК Пресс. Москва, 2021. 288 с.

Самостоятельный расширенный поиск по теме практической работы.

### Вопросы

1. Обработка *одномерных, двумерных и многомерных массивов* в ЯП Java, Kotlin и JavaScript. Правила их объявления, обращение к ним. Примеры и скриншоты.
2. *Коллекции*, как *умные массивы*. *Итераторы* и их методы. *Списки* как варианты коллекций. Преобразование коллекций в массивы и обратно. Примеры и скриншоты.
3. *Строки* как объекты, в языках ООП Java, Kotlin и JavaScript. Преобразование строк. Длинные строки. Класс `StringBuffer`. Примеры и скриншоты.
4. Типы–*перечисления* в ЯП Java, Kotlin и JavaScript. Правила их использования. Применение зарезервированного слова *enum*. Примеры и скриншоты.
5. Работа с датой и временем в ЯП Java, Kotlin и JavaScript. Классы `java.util.Time` и `java.util.Date`. Функция `System.currentTimeMillis()`. Конструкторы для создания объектов с заданной датой и временем. Примеры и скриншоты.
6. Исключительные ситуации (*exceptions*) и методы их обработки. Защищённый блок программного кода *try catch*. Перехват *exceptions* в Java, Kotlin и JavaScript. Примеры.
7. Достоинства и недостатки использования *коллекций, списков и массивов*, обработки даты и времени в ЯП Java, Kotlin и JavaScript. Сравнение скриншотов.
8. Функции в Java, Kotlin и JavaScript. Передача примитивных типов данных в функции. Методы как функции. Передача примитивных аргументов в функцию. Формальные и фактические параметры в ЯП Java, Kotlin и JavaScript. Примеры и скриншоты.
9. Передача данных в *подпрограмму*. Локальные и глобальные переменные, правила их использования. в языках ООП Java, Kotlin и JavaScript. Примеры и скриншоты.
10. Модификаторы доступа *private, protected* и *public*. Правила видимости в языках ООП Java, Kotlin и JavaScript. Примеры и скриншоты.
11. Ссылка на объект *this* Java, Kotlin и JavaScript. Применение ссылки *this*, скриншоты. *Классы данных* (data classes) для автоматизации генерации кода классов, для хранения данных. Особенности их применения в ЯП Java, Kotlin и JavaScript. Примеры.
12. Сравнительный анализ ЯП при использовании *подпрограмм*. Методы, процедуры и функции. Особенности их применения в Java, Kotlin и JavaScript. Примеры.
13. Самостоятельное изучение самой свежей научной литературы (научных статей, монографий, учебников 2021 – 2025 годов издания) по теме занятия.