Массивы

- Наиболее эффективный способ организации размещения объектов и произвольного доступа к ним (линейность) но постоянен в размере. ArrayList - позволяет добавлять элементы, создает новый массив и копирует элементы туда, но менее эффективен
- Хранение конкретного типа, можно хранить примитивы
- Основное преимущество- эффективность
- Массивы как полноценные объекты
- Индефикатор массива - ссылка на объект, созданный в куче, хранящий ссылки на другие объекты. Может быть создан неявно или с new. Обязательная часть объекта - поле длинна. Длина не обязательно показывает, сколько объектов находится в массиве, но показывает, сколько может находиться
- При массив = массив присваевается ссылка на один массив в общей куче
- Многомерные массивы
- - параметры в фигурных скобках - каждая новая - переход на новый уровень
- Для создания можно воспользоваться ключевым словом new
- Векторы массивов, образующих матрицу, могут иметь произвольную длину

- Массив и обобщение
- Плохо сочетаются лруг с другом. Нельзя создать массив параметр.типов (чтоб он хранил какой-то тип по каждому значению, потому что тип должен быть конкретным)
- Но тип массива может быть параметризован
- Можно создать массив обджект и помещать в него все, кроме примитивных типов
- Создание тестовых данных
- Arrays.fill()- заполняет указанный диапазон значений одними тестовыми данными
- Генератор данных
- Можно использовать в качестве инициализации массива (например генератор счетчика) с динамическими полями count
- Класс Arrays – содержит набор стат методов для выполнения нескольких операций над массивами - equalse, fill, sort, binarySearch, toString, hashCode, asList
- Копирование массивов, перегружен для всех типов. Передача исходного массива, применого, смещение блоков, количество копируемых элементов
- Копироваться могут массивы примитивов и массивы объектов. Такое копирование - поверхностное. Метод не выполняет автоматической упаковки и распоковки

- Сравнение массивов
- Имеется перегруженный метод equals() - для сравнения целых массивов, для прим и обдж
- В массивах должно быть равное кол-во элементов, каждый элемент должен быть эквивалентен соответствующему.
- В случае помещения строк - "myString" new String("vyString") - равны
- Сравнения элементов массива
- Используется стратегия (изменяющаяся часть кода в отдельном классе, объект передается неизменному коду, использует стратегию для реализации алгоритма)
- Два способа предоставить описание процесса сравнения - Comparable - можно реализовать для каждого класса
- Для написания собственного метода для
- сравнения - отдельный класс с интерфейсом Comparator (или можно переопределить equalse)
- Сортировка массива - есть встроенные методы
- Quik sort - для простейших типов
- Сортировка слиянием - для объектов
- Поиск в отсортированном массиве - бинарный, при дупликатах - первый дубликат

Контейнеры лучше массивов во всем, кроме производительности

С добавлением автоматической упаковки хранение примитивов упростилось

Контейнеры предпочтительнее