



# INHERITANCE

## Checklist

Olga Smolyakova

**Ответьте на вопросы, приведенные ниже. Нарисуйте рисунки, позволяющие вам объяснить ответы.**

<p>Напишите (нарисуйте) алгоритм, описывающий порядок вызова конструкторов при создании объектов производного класса. (Постарайтесь предусмотреть все случаи)</p>	<p>1) Инициализация статических переменных базового класса(БК). 2)Иниц-ция статических переменных производного класса (ПК). 3) Иниц-ия и присваивание значений нестатических переменных БК. 4) Вызов конструктора БК. 5) Иниц-ия и присваивание значений нестатических переменных ПК. 6) Вызов конструктора ПК.</p>
<p>Как вы думаете, зачем необходимо переопределение методов? Приведите примеры переопределенных методов. Может ли при переопределении метода меняться тип возвращаемого значения?</p>	<p>Переопределение методов - это как часть полиморфизма. Один метод и множество реализаций. Переопределёнными могут быть toString(), equals(), hashCode(); При переопределении тип возвращаемого значения меняться не может.</p>
<p>Объясните, как вы понимаете смысл фразы “ссылка базового типа может ссылаться на объекты производных типов” и “объект подкласса может быть использован везде, где используется объект его суперкласса.”</p>	<p>Ссылка базового типа может ссылаться на любой объект производного типа и соответственно по этой ссылке может быть вызван любой метод, который есть в базовом типе, но реализация будет производного типа. Ну и соответственно ссылка на объект подкласса может быть присвоена ссылке суперкласса. Восходящее преобразование. Вызываться могут методы, которые есть в суперклассе.</p>
<p>Попробуйте порассуждать на тему, наследуются или нет статические методы.</p>	<p>Думаю статические методы наследуются и они копируются из базового также как и переменные и нестатические методы.</p>

<p>Как вы понимаете, то такое перегруженные методы? Приведите примеры перегруженных методов.</p>	<p>Методы перегружаются, чтобы не писать методы, которые делают одно и то же с разными названиями, т.е. по сути при небольшом изменении в типе и(или) количестве параметров не надо придумывать название метода, а просто перегружается старый. Также это упрощает использование перегруженных методов.</p>
<p>Как вы думаете, зачем в языке Java надо применять ключевое слово <code>final</code>. Что может быть <code>final</code> в Java.</p>	<p><code>final</code> применяется, 1) чтобы сделать переменную неизменяемой; 2) чтобы запретить переопределение метода 3) чтобы запретить наследование класса</p>
<p>Перечислите методы класса <code>Object</code>, которые вы знаете. Расскажите для чего они предназначены и как работают.</p>	<p><code>equals</code> - сравнивает равенство объектов; <code>hashCode</code> - вычисляет уникальный хэш код объекта; <code>toString</code> - выводит значения всех полей объекта в консоль/лог</p>
<p>Для чего следует применять метод <code>equals()</code>. Знаете ли вы “где” в Java метод <code>equals()</code> применяется неявно? Перечислите правила переопределения метода <code>equals()</code></p>	<p><code>equals</code> применяется для сравнения объектов Может <code>equals</code> неявно используется в строках, коллекциях. Правила: 1) проверить, что объект один и тот же; 2) проверить, что объект не <code>null</code> и что классы объектов совпадают 3) сравнить все поля объектов начиная с числовых, так как они сравниваются быстрее</p>

<p>Что такое хэш-код? Что такое хэш-код объекта? Объясните, почему хэш-коды двух различных объектов могут совпасть. Перечислите известные вам правила переопределения метода hashCode()</p>	<p>Хэш код - сгенерированное число с помощью хэш функции. Хэш код объекта - число, которое генерируется в методе hashCode() на основании значений полей объекта и используется для сравнения объектов.</p> <p>Как таковых правил перегрузки не встречал и минимально могу вычлениить следующее:</p> <ol style="list-style-type: none"><li>1) использовать переменную prime со значением 31</li><li>2) проверять строки на null и после этого вызывать у них hashCode;</li></ol>
<p>Расскажите, когда применяется метод toString() и как его необходимо переопределять</p>	<p>toString() применяется для вывода значений строк или других объектов на консоль или в лог. Для корректного использования требуется переопределение toString()</p>