

Pro Java №2

```
public class SmallDog extends Dog
```

Ключевое слово `extends` в Java используется для указания наследования, при котором класс `SmallDog` наследует свойства и методы класса `Dog`.

В Java один класс может наследоваться только от одного другого класса. Это называется одиночным наследованием (*single inheritance*).

В наследовании подкласс (дочерний класс) может иметь свои собственные дополнительные поля, методы и конструкторы, наряду с теми, которые он наследует от родительского класса.

В Java ты можешь переопределить (*override*) метод родительского класса в дочернем классе, чтобы изменить его поведение. Для этого используется аннотация `@Override`, которая помогает гарантировать, что метод действительно переопределяет метод из суперкласса.

В Java, когда ты вызываешь родительский метод с помощью ключевого слова `super`, ты можешь переопределить этот метод в дочернем классе. Если ты используешь `super.methodName()` в дочернем классе, то вызывается метод родительского класса. Если хочешь, чтобы дочерний класс не использовал метод родительского класса, то нужно удалить или закомментировать вызов `super.methodName()`.

Классы, участвующие в наследовании, должны находиться в логической иерархии, где один класс (родительский `Dog`) передает свои свойства и методы другому классу (дочернему `Puppy`).

Любой класс, даже если он явно не наследуется от другого класса, фактически является потомком класса `Object`.

Метод `toString()` возвращает строковое представление объекта. Переопределение этого метода позволяет выводить значения полей объекта в удобочитаемом формате.

Если в классе-наследнике ты хочешь вызвать метод `toString()` родительского класса и добавить к нему

дополнительную информацию, ты можешь использовать ключевое слово `super` для вызова родительского метода `toString()`.

Если ты хочешь переопределить метод `hashCode()` в подклассе и использовать хэш-код родительского класса, ты можешь использовать `super.hashCode()` для обращения к методу родительского класса, а `this.hashCode()` для обращения к текущему объекту.

Когда создаётся объект класса-наследника, конструктор родительского класса вызывается автоматически перед выполнением конструктора класса-наследника.

Многоуровневое наследование в Java позволяет создавать иерархию классов, где один класс наследуется от другого, а затем этот класс может наследоваться следующим классом и так далее.

В Java объекты классов, связанных через механизм наследования, можно объединить в одну структуру, такую как массив, список или любую другую коллекцию.