# Lab: Inheritance

In case **Java Zip File** format is required you need to make a .zip file of the package in which the main class is present, together with all other classes if there are any, and submit it in [Judge](https://alpha.judge.softuni.org/contests/1574).

# Part I: Inheritance

## Single Inheritance

Create two classes named Animal and Dog.

Animal with a single public method eat() that prints: **"eating…"**

Dog with a single public method bark() that prints: **"barking…"**

TheDog should inherit from the Animal.





### Hints

Use the **extends** keyword to build a hierarchy.

## Hierarchical Inheritance

Create three classes named Animal, Dog, and Cat.

Animal with a single public method eat() that prints: **"eating…"**

Dog with a single public method bark() that prints: **"barking…"**

Cat with a single public method meow() that prints: **"meowing…"**

Dog and Cat should inherit from Animal.





# Part II: Reusing Classes

## Random Array List

Create a RandomArrayList class that has all the functionality of an ArrayList.

Add an additional function that **returns** and **removes** a random element from the list.

* Public method: getRandomElement(): Object

## Stack of Strings

Create a class StackOfStrings that can store only strings and has the following functionality:

* Private field: **data: ArrayList<String>**
* Public method: push(String item): void
* Public method: pop(): String
* Public method: peek(): String
* Public method: isEmpty(): boolean



### Hints

Use composition/delegation to have a field in which to store the stack's data.