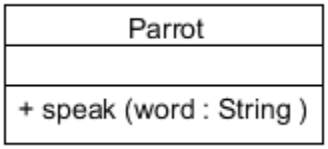
Lab Parrot

UML Class Diagram:



Learning Outcomes:

- Declare a class that is intended to be instantiated
- Declare a method to provide a functionality
- Create an instance of the class you declared
- Access the method using the dot operator
- Increase your familiarity with UML class diagrams
- Review the use of Scanner

- Create a project called **LabParrot**
- Add 2 files to the project: Parrot.java and ParrotTest.java
- In Parrot.java do the following:
  - Create a public class called Parrot
  - Inside the class create a public method called speak.

    The method speak has one String parameter named word and no return value (i.e. return type **void**)

    The method header looks like this:

    `public void speak(String word)`
  - The parrot repeats anything he is told.

    We implement this behavior by printing the word passed as an argument.
- In ParrotTest.java create the main method

  Inside the main method do the following:
  - Create an instance of Scanner named input
  - Create an instance of Parrot named myParrot

    You create a new instance by calling the default constructor like this:

    `Parrot myParrot = new Parrot();`
  - Read in a text (i.e. use Scanner to let the user choose what s/he would like to say to the parrot);

    make sure to prompt the user before you read in the text

    Create a String variable called text to temporarily store the input read
  - Call the method speak of the instance myParrot and pass the variable text as argument

    NOTE: the variable name passed does not have to match the parameter name

    The method call looks like this:

    `myParrot.speak(text);`

Sample Output1:

```
What would you like to say to the parrot? hi
hi
```

Sample Output2:

```
What would you like to say to the parrot? how are you?
how are you?
```