Note:

This is a case where the videos are easier to understand than the pdfs alone.

The videos comment the slides step by step, this helps build your understanding.

Don’t read and rush. Watch the video instead!

So we used so far boxes which contain objects like String (text) or Integer (round numbers)

```
String nameOfTheUser;
Integer ageOfTheUser;
```

The good news is, boxes can contain anything, not just text or numbers. That’s actually 2 good news:

1. **We can create new kinds of objects, with their own actions, and put them inside boxes, and this will enable us to invent the functions of our app**

2. **People all around the world have already created many different objects, with many actions, and they share them freely and we can use them in our apps**

Actually Codename One, the tool we use in this MOOC, is made of objects and we can use them and their actions in our apps to connect to the Internet, access the camera of the phone, etc.

How to put these objects we created (or somebody else) in a box?

Let’s take the example of a Label (you know, these pieces of text we put on the screen of the mobile phone in the designer)

First, we create a box which I call “myText”:

```
myText;
```

And we remember to put “Label” in front of it to say that it is a box specialized in containing Labels:

```
Label myText;
```

And finally we create a new label and put it in the box. Contrary to String and Integers, which are very simple objects, almost all other objects need to be created like that when we put them in the box:

new
Label()
myText

```
myText = new Label();
```

Don’t forget the upper case initial in the name of the object!

The same method works if we want to create a new screen for our mobile app:

First, we create a box called "myScreen" which will contain screens:

The box "myScreen", specialized in containing Forms. Yes, screens are called « forms » by codename one…

```
Form myScreen;
```

Then we create a screen and put it in the box:

```
myScreen = new Form();
```

Now, we can explore what kind of actions we can do with a screen. Maybe show it in the app? So easy, write a dot after myScreen and explore all possible actions. The one we want is:

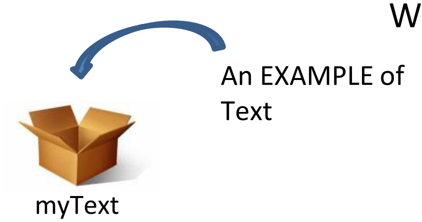
```
myScreen.show();
```

If you write that in the code of your app (we'll show you where exactly), you will be able to change which screen is shown to the user. Yes, you know how to code the navigation from one screen to another!!

Special actions

This page is a bit complicated, you can skip it and come back later if you want

We have seen that objects have values and actions. For example:
We create a box specialized in containing text, and we add some text to it:

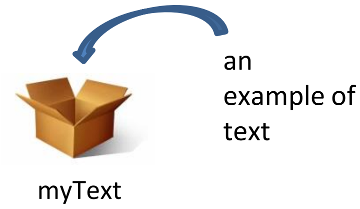


An EXAMPLE of Text

myText

```
String myText;
myText = “An EXAMPLE of Text”;
```

And we can apply some actions to the content of the box (this one puts every letter in lower case):

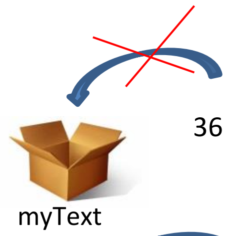


an example of text

myText

```
myText = myText.toLowerCase();
```

Well, there are other special actions we can use. These special actions can be found by writing the type of object (here, String) and adding a dot to it.
For example, this would not work because a number cannot go into a box made for text:

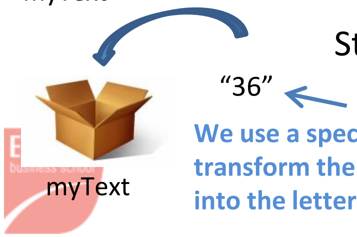


36

myText

~~myText = 36;~~

What we can do is using a **special action** by adding a dot to the type of object of the box (here, String – don't forget the upper case “S!”). This action “valueOf” takes a number (36), transform it into text, and adds it to the box:



“36”

myText

We use a special action to transform the number 36 into the letters « 36 »

```
myText = String.valueOf(36);
```