

Java module 3

Exercises Day 2 (A)

1 - Enum	Toy shop
Instructions	<p>We will create a program to manage a toy shop inventory.</p> <p>Step 1: Create an Enum called ToyType to allow classifying toys into categories. Some of the categories could be: STUFFED_ANIMAL, VEHICLE or PUZZLE.</p> <p>Step 2: Create a class to represent a toy that exists within the toy shop. It should be possible to store the name of the toy and its type.</p> <p>Step 3: Create a main program that will allow the store manager to add a toy to the shop's inventory.</p> <p>Step 4: Run the program and take a look at the object that was created using the debugger.</p>
Expected output	<p>Toy name: >>> HotWheels Winter car Toy type (STUFFED_ANIMAL, VEHICLE, PUZZLE): >>> VEHICLE Toy added to the inventory.</p>

2 - Inheritance	Adding books to the shop
Instructions	<p>We will continue developing the program for the toy shop. The owner now wants to sell books.</p> <p>Step 1: Create a parent class called Article. The Article class should allow storing the price of the article..</p> <p>Step 2: Create a class called Book that is a child class of Article. The Book class should allow storing the book's title and author.</p> <p>Step 3: Update the Toy class created in the previous exercise to be a child class of Article.</p> <p>Step 4: Modify the main program to allow adding Books and Toys to the inventory.</p> <p>Step 5: Add a method to the Article class that will allow printing the article information to the output. Override this method for each of the child classes such that it prints all relevant information of each article.</p>
Expected output	<p>Would you like to (t) add a toy, (b) add a book or (p) print the list of all articles? >>> t</p>

	<p>Toy name: >>> HotWheels Winter car Toy type (STUFFED_ANIMAL, VEHICLE, PUZZLE): >>> VEHICLE Toy price: >>> 3,5 Toy added to the inventory. Would you like to (t) add a toy, (b) add a book or (p) print the list of all articles? >>> b Book title: >>> Harry Potter Book author: >>> J. K. Rowling Book price: >>> 27 Book added to the inventory. Would you like to (t) add a toy, (b) add a book or (p) print the list of all articles? >>> p Toy: HotWheels Winter car - 3,5 euros. Book: Harry Potter by J. K. Rowling - 27 euros.</p>
--	--

3 - Polymorphism	Promotional price
Instructions	<p>We will continue developing the program for the toy shop.</p> <p>Step 1: Create a polymorphic method to print the information of each article. This new method should have the same name as the one created in the previous exercise but it should receive the amount of discount that should be applied to the price of the articles.</p> <p>Step 2: Update the main program to allow the manager to print all articles with a 20% discount.</p>
Expected output	<p>Would you like to (t) add a toy, (b) add a book or (p) print the list of all articles? >>> t Toy name: >>> HotWheels Winter car Toy type (STUFFED_ANIMAL, VEHICLE, PUZZLE): >>> VEHICLE Toy price: >>> 3,5 Toy added to the inventory. Would you like to (t) add a toy, (b) add a book or (p) print the list of all articles? >>> b Book title: >>> Harry Potter Book author: >>> J. K. Rowling Book price:</p>

	<pre>>>> 27 Book added to the inventory. Would you like to (t) add a toy, (b) add a book or (p) print the list of all articles? >>> p Would you like to apply a 20% discount to the prices (y/n)? >>> n Toy: HotWheels Winter car - 3,5 euros. Book: Harry Potter by J. K. Rowling - 27 euros. Would you like to (t) add a toy, (b) add a book or (p) print the list of all articles? >>> p Would you like to apply a 20% discount to the prices (y/n)? >>> y Toy: HotWheels Winter car - 2,8 euros. Book: Harry Potter by J. K. Rowling - 21,6 euros.</pre>
--	---