Java module 2

Exercises Day 1

1.1 - Functions	Create and use a function
Instructions	Step 1: Create a function that returns the area of a rectangle given the rectangle's width and height. Step 2: Create a program that asks the user for the width and height of a rectangle and outputs the rectangle's area. It should use the function created in step 1.

Expected output	Rectangle width:
	>>> 10
	Rectangle height:
	>>> 5
	The area of the rectangle is 50.

1.2 - Procedures	Create and use a procedure
Instructions	Step 1: Create a procedure that prints a number's multiplication table. The procedure should receive the number in a parameter. Step 2: Create a program that asks the user for a number and then prints the number's multiplication table. It should use the procedure created in step 1.
Expected output	Enter a number: >>> 3 3 x 1 = 3 3 x 2 = 6 3 x 3 = 9 3 x 4 = 12 3 x 5 = 15 3 x 6 = 18 3 x 7 = 21 3 x 8 = 24 3 x 9 = 27 3 x 10 = 30

2.1 - JavaDoc	Implement the function

Instructions	Based on this javaDoc, implement the code
	
	/**
	* Checks if a number is prime.
	*
	* @param num the number to check
	* @return true if the number is prime, otherwise false
	*/
	public static boolean isPrime(int num) {
	// Implementation code here
	}

2.2 - JavaDoc	Write the JavaDoc
Instructions	Write the JavaDoc for the methods implemented in exercises 1.1 and 1.2.

3.1 - Arrays	Find the maximum
Instructions	Write a function that finds the maximum number in an array of integers. Then use this function in a program that requests 5 integers to the user, stores them in an array and outputs the maximum.

Expected output	Number 1:
	>>> 9
	Number 2:
	>>> 167
	Number 3:
	>>> -43
	Number 4:
	>>> 33
	Number 5:
	>>>63
	The maximum is 167

3.2 - Lists	Store names
Instructions	Write two functions: - The first one should request names from the user, store them in a list and return this list. - The second one should receive a list as a parameter and print the names stored within the list. Then write a program that will call both functions.

Expected output	How many names would you like to store?
	>>> 3
	Name 1:
	>>> Ana

	Name 2:
	>>> Carol
	Name 3:
	>>> John
	The names are Ana, Carol, John