A desktop Java application:

(1) In order to learn and remember a little about object-oriented programming (OOP) in Java, let's develop a desktop Java application;

(2) Create a desktop Java application: in Eclipse, option **File → New → Java Project**. Give to the project the name of our department of undergraduate: **Prograd** and then click in **Finish**.

(3) Create a package called **br.edu.ufabc.prograd.model**: right click over the folder **src**, option **New → Package.**

(4) Inside the package, create a class called **Student**: right click over the package, option **New → Class.** The class should have the following code:

```
package br.edu.ufabc.prograd.model;
public class Student {
	private Long id;
	private String name;
	private String email;
	private String address;
}
```

(5) Notice that we created the attribute **id** with type **Long,** not **long**, because Long initialize the attribute with default value **null**. We make this way to avoid errors in the database;

(6) Now we convert the class Student in a *javabean*. *Javabeans* are classes which have the constructor without arguments and access methods of the type *get* and *set*. To make that automatically in Eclipse, use the shortcut **ctrl + 3** and type **generate**. Among the options, select **Generate Getters and Setters**. In the next window, select all attributes.

(7) Create a package named **br.edu.ufabc.prograd.tests**: right click over the folder **src**, option **New → Package.**

(8) Inside this new package, create a class named **CreateStudent:** right click over the package, option **New → Class.** The class should have the following code:

```
package br.edu.ufabc.prograd.tests;

import br.edu.ufabc.prograd.model.Student;

public class CreateStudent {
	public static void main(String[] args) throws SQLException {
		Student student = new Student();
		student.setName("Jack");
		student.setEmail("jack@gmail.com");
		student.setAddress("Av. Sherbrooke, 5001");
		student.setId((long) 3);


		System.out.println(student .getName());
		System.out.println(student .getEmail());
		System.out.println(student .getAddress());
		System.out.println(student .getId());
	}
}
```

(9) Now create an array of type Student and store data of some students. Make a loop to list all of them.

TYPE 1:

```
//**********************************
package br.edu.ufabc.prograd.tests;

import br.edu.ufabc.prograd.model.Student;

public class CreateStudent {
	public static void main(String[] args) {
		Student student1 = new Student();
		student1.setName("Jack");
		student1.setEmail("jack@gmail.com");
		student1.setAddress("Sherbrooke Street, 5001");
		student1.setId((long) 1);

		Student student2 = new Student();
		student2.setName ("John");
		student2.setEmail("john@gmail.com");
		student2.setAddress ("Papineau Street, 5001");
		student2.setId((long) 2);

		Student student3 = new Student();
		student3.setName ("Mary");
		student3.setEmail("maria@gmail.com");
		student3.setAddress ("Joffre Avenue, 5001");
		student3.setId((long) 3);

		Student students[] = new Student[3];
		students[0] = student1;
		students[1] = student2;
		students[2] = student3;

		for (int i = 0; i < students.length; i++) {
			System.out.println(students[i].getName());
			System.out.println(students[i].getEmail());
			System.out.println(students[i].getAddress());
			System.out.println(students[i].getId());
		}
	}
}
```

TYPE 2:

```
//**********************************
package br.edu.ufabc.prograd.tests;

import br.edu.ufabc.prograd.model.Student;

public class CreateStudent {
	public static void main(String[] args) {
		Student students[] = new Student[3];

		students[0] = new Student();
		students[0].setName("Jack");
		students[0].setEmail("jack@gmail.com");
		students[0].setAddress("Sherbrooke Street, 5001");
		students[0].setId((long) 1);

		students[1] = new Student();
		students[1].setName ("John");
		students[1].setEmail("john@gmail.com");
		students[1].setAddress ("Papineau Street, 5001");
		students[1].setId((long) 2);

		students[2] = new Student();
		students[2].setName ("Mary");
		students[2].setEmail("maria@gmail.com");
		students[2].setAddress ("Joffre Avenue, 5001");
		students[2].setId((long) 3);

		for (int i = 0; i < students.length; i++) {
			System.out.println(students[i].getName());
			System.out.println(students[i].getEmail());
			System.out.println(students[i].getAddress());
			System.out.println(students[i].getId());
		}
	}
}
```

(10) What is the difference between the two types in the above exercise?