La Salle

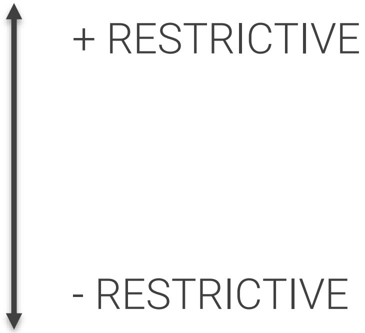
1. Attributes and methods access modifiers
2. Getter and setter methods
3. Summary

Encapsulation: Controlling access to members of a class

Attributes and methods access modifiers

1. Attributes and methods access modifiers

- private
- no modifier (package private)
- protected
- public



+ RESTRICTIVE

- RESTRICTIVE

1. Attributes and methods access modifiers

- **private**: attribute or method can only be accessed withing the class
- **no modifier (package private):** attribute or method can accessed withing the class and from classes withing the same package
- **protected:** attribute or method can accessed withing the classes withing the same package and by all subclasses
- **public:** atrribute or method can accessed withing the class and by all the other classes

INHERITANCE

1. Attributes and methods access modifiers

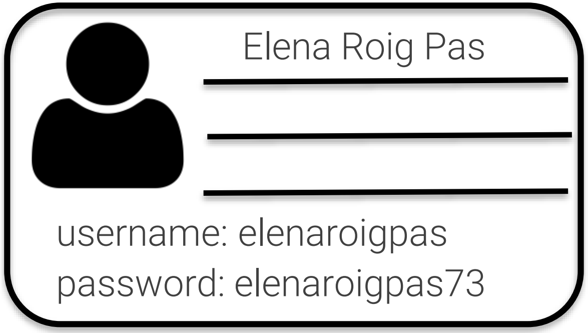
SCHOOL LEARNING PLATFORM

User.java

```
public class User{
   String name;
   String userName;
   String userPassword;

   User(String name) {
      this.name = name;
      this.userName = createUserName();
      this.userPassword = createUserPassword();
   }

   String createUserName(){
      return name.replaceAll("\\s","").toLowerCase();
   }
   String createUserPassword(){
      int num1 = (int) (Math.random() * 100) + 1;
      int num2 = (int) (Math.random() * 100) + 1;
      return this.userName+num1+num2;
   }
   String display(){ //block of code }
}
```



Elena Roig Pas

username: elenaroigpas

password: elenaroigpas73

1. Attributes and methods access modifiers

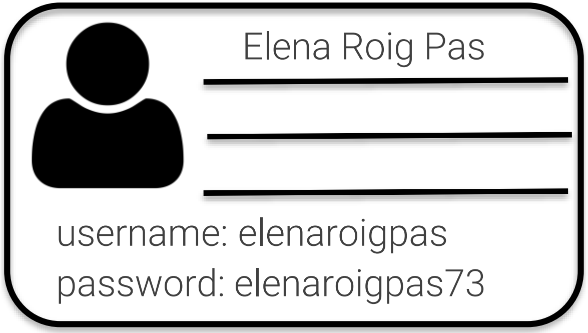
SCHOOL LEARNING PLATFORM

User.java

```
public class User{
 private String name;
 private String userName;
 private String userPassword;

   User(String name) {
      this.name = name;
      this.userName = createUserName();
      this.userPassword = createUserPassword();
   }

   String createUserName(){
      return name.replaceAll("\\s","").toLowerCase();
   }
   String createUserPassword(){
      int num1 = (int) (Math.random() * 100) + 1;
      int num2 = (int) (Math.random() * 100) + 1;
      return this.userName+num1+num2;
   }
   String display(){ //block of code }
}
```



Elena Roig Pas

username: elenaroigpas

password: elenaroigpas73

1. Attributes and methods access modifiers

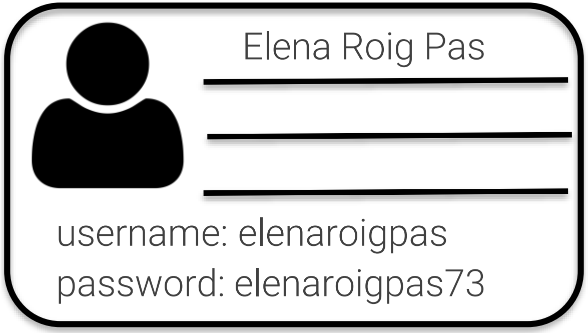
SCHOOL LEARNING PLATFORM

User.java

```
public class User{
 private String name;
 private String userName;
 private String userPassword;

   User(String name) {
      this.name = name;
      this.userName = createUserName();
      this.userPassword = createUserPassword();
   }

 private String createUserName(){
      return name.replaceAll("\\s","").toLowerCase();
   }
 private String createUserPassword(){
      int num1 = (int) (Math.random() * 100) + 1;
      int num2 = (int) (Math.random() * 100) + 1;
      return this.userName+num1+num2;
   }
  String display(){ //block of code }
}
```



Elena Roig Pas

username: elenaroigpas

password: elenaroigpas73

1. Attributes and methods access modifiers

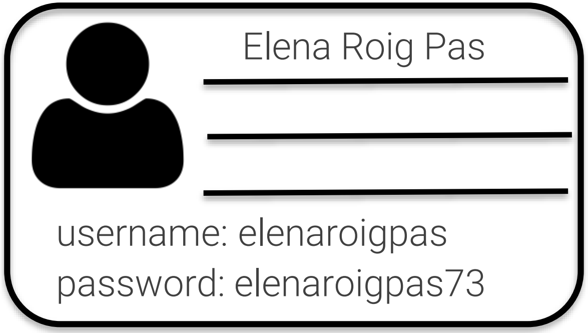
SCHOOL LEARNING PLATFORM

User.java

```
public class User{
 private String name;
 private String userName;
 private String userPassword;

 public User(String name) {
      this.name = name;
      this.userName = createUserName();
      this.userPassword = createUserPassword();
   }

 private String createUserName(){
      return name.replaceAll("\\s","").toLowerCase();
   }
 private String createUserPassword(){
      int num1 = (int) (Math.random() * 100) + 1;
      int num2 = (int) (Math.random() * 100) + 1;
      return this.userName+num1+num2;
   }
 public String display(){ //block of code }
}
```


Elena Roig Pas

username: elenaroigpas

password: elenaroigpas73

1. Attributes and methods access modifiers

SCHOOL LEARNING PLATFORM

User.java

```
public class User{
 private String name;
 private String userName;
 private String userPassword;

 public User(String name) {
      this.name = name;
      this.userName = createUserName();
      this.userPassword = createUserPassword();
   }

 private String createUserName(){
      return name.replaceAll("\\s","").toLowerCase();
   }
 private String createUserPassword(){
      int num1 = (int) (Math.random() * 100) + 1;
      int num2 = (int) (Math.random() * 100) + 1;
      return this.userName+num1+num2;
   }
  public String display(){ //block of code }
}
```

Main.java

User newUser = new User(“Elena Roig Pas”);

~~newUser.name = “Maria Elena Roig Pas”~~;

String userPassword= ~~newUser.createUserPassword~~();

Error compilation

Getter and setter methods

2. Getter and setter methods

Attributes that shall be readable and/or updatable by other classes

A GENERIC EXAMPLE

```java
public class MyClass {

    private String myAttribute;

    public MyClass(String myAttribute) {
        this.myAttribute = myAttribute;
    }

}
```

MyClass.java

2. Getter and setter methods

A GENERIC EXAMPLE

```java
public class MyClass {

    private String myAttribute;

    public MyClass(String myAttribute) {
        this.myAttribute = myAttribute;
    }

    public void setMyAttribute(String myAttribute) {
        this.myAttribute = myAttribute;
    }

}
```

Adding a setter method

MyClass.java

A GENERIC EXAMPLE

```
public class MyClass {

    private String myAttribute;

    public MyClass(String myAttribute) {
        this.myAttribute = myAttribute;
    }

    public void setMyAttribute(String myAttribute) {
        this.myAttribute = myAttribute;
    }

    public String getMyAttribute() {
        return myAttribute;
    }
}
```

Adding a getter method

MyClass.java

2. Getter and setter methods

A GENERIC EXAMPLE

```java
public class MyClass {

    private String myAttribute;

    public MyClass(String myAttribute) {
        this.myAttribute = myAttribute;
    }

    public void setMyAttribute(String myAttribute) {
        this.myAttribute = myAttribute;
    }

    public String getMyAttribute() {
        return myAttribute;
    }
}
```

myAttribute has a setter and getter method

MyClass.java

2. Getter and setter methods

A GENERIC EXAMPLE

```java
public class MyClass {

    private String myAttribute;

    public MyClass(String myAttribute) {
        this.myAttribute = myAttribute;
    }

    public void setMyAttribute(String myAttribute) {
        this.myAttribute = myAttribute;
    }

    public String getMyAttribute() {
        return myAttribute;
    }
}
```

MyClass.java

```java
MyClass object = new MyClass(“Name”);
object.setMyAttribute(“AnotherName”);
System.out.println(object.getMyAttribute());
```

Main.java

2. Getter and setter methods

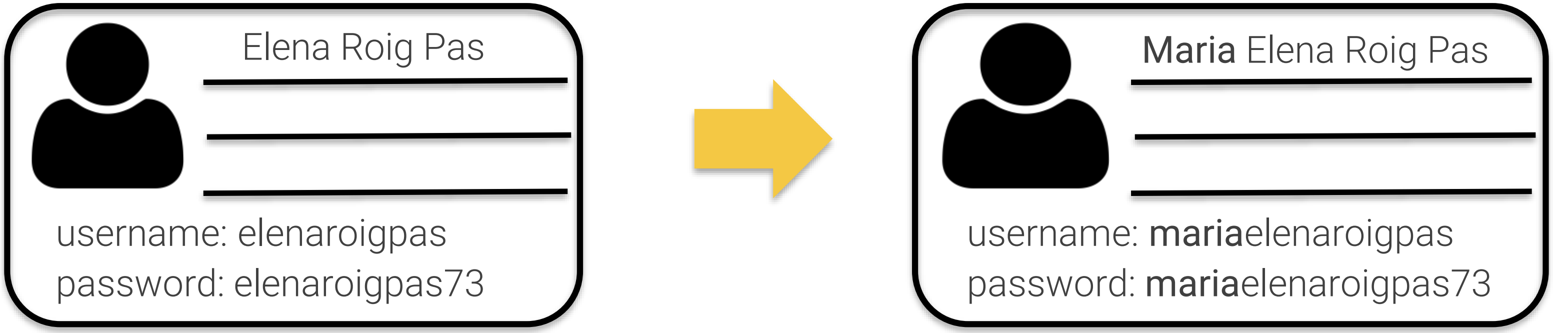
Attributes that shall be readable and/or updatable by other classes

IMPORTANTE

The presence of numerous getter and setter methods is a red flag that the program isn't necessarily well designed == isn't well encapsulated information

2. Getter and setter methods

SCHOOL LEARNING PLATFORM



Elena Roig Pas

username: elenaroigpas
password: elenaroigpas73

Maria Elena Roig Pas

username: **maria**elenaroigpas
password: **maria**elenaroigpas73

2. Getter and setter methods

SCHOOL LEARNING PLATFORM

User.java

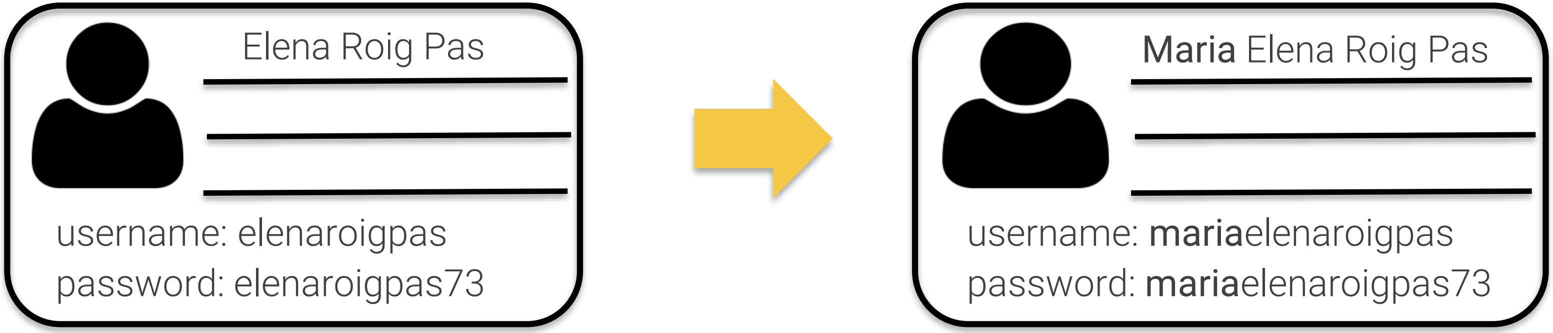
```
public class User{
 private String name;
 private String userName;
 private String userPassword;

 public User(String name) {
      this.name = name;
      this.userName = createUserName();
      this.userPassword = createUserPassword();
   }

  …
 public void setName(String name){
    this.name = name;
 }
}
```

Adding a setter method

SCHOOL LEARNING PLATFORM



Elena Roig Pas

username: elenaroigpas
password: elenaroigpas73

Maria Elena Roig Pas

username: **maria**elenaroigpas
password: **maria**elenaroigpas73

2. Getter and setter methods

SCHOOL LEARNING PLATFORM

User.java

```java
public class User{
 private String name;
 private String userName;
 private String userPassword;

 public User(String name) {
      this.name = name;
      this.userName = createUserName();
      this.userPassword = createUserPassword();
   }

  …
 public void setName(String name){
    this.name = name;
    this.userName = createUserName();
    this.userPassword = createUserPassword();
 }
}
```

Adding a setter method

2. Getter and setter methods

SCHOOL LEARNING PLATFORM

User.java

```
public class User{
 private String name;
 private String userName;
 private String userPassword;

 public User(String name) {
      this.name = name;
      this.userName = createUserName();
      this.userPassword = createUserPassword();
   }

  …
 public void setName(String name){
    this.name = name;
    this.userName = createUserName();
    this.userPassword = createUserPassword();
 }
}
```

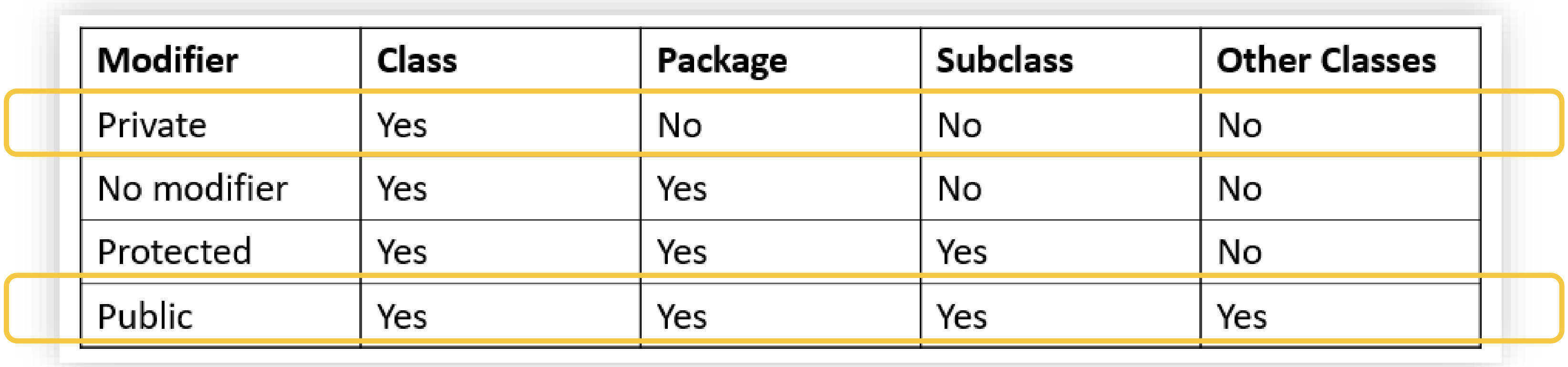
Main.java

```
User newUser = new User(“Elena Roig Pas”);
……
newUser.setName("Maria Elena Perez Roig");
```

Summary

3. Summary

Accesibility matrix



Modifier	Class	Package	Subclass	Other Classes
Private	Yes	No	No	No
No modifier	Yes	Yes	No	No
Protected	Yes	Yes	Yes	No
Public	Yes	Yes	Yes	Yes

3. Summary

To define a well-encapsulated class:

- ✓ Define its **attributes** as **private** variables. Allow access or manipulation to these variables using public methods (getters & setters).
- ✓ Define **public constructors**.
- ✓ Define **public methods** to implement operations (accesible from other objects).
- ✓ **Private methods** are **helper methods.**

“La carrera se hace en público, el talento en privado“

Marilyn Monroe, actriz de cine estadounidense

La Salle