Inheritance: Super keyword

1. Super

Super

1. Super

Use SUPER to access fields and methods of the super class

1. Super

```
public class Jeans extends Clothing{


    String type; //slim, fit, ..


    public String display() {
        return "Jeans{" +
                "cod='" + cod + '\'' +
                ", price=" + price +
                ", size='" + size + '\'' +
                ", color='" + color + '\'' +
                ", genre=" + genre +
                ", type='" + type + '\'' +
                '}';
    }
}
```

Jeans.java

1. Super

```
public class Clothing {
   String cod;
   double price;
   String size;
   String color;
   char genre; //W==Woman, M==Man


   public String display() {
      return "cod='" + cod + '\'' +
            ", price=" + price +
            ", size='" + size + '\'' +
            ", color='" + color + '\'' +
            ", genre=" + genre;
   }
}
```

We want to continue to use the original method, and ADD extras for each subclass individually.

super

Clothing.java

1. Super

```
public class Jeans extends Clothing{


    String type; //slim, fit, ..


    public String display() {
      return "Jeans{" +
          "cod='" + cod + '\'' +
          ", price=" + price +
          ", size='" + size + '\'' +
          ", color='" + color + '\'' +
          ", genre=" + genre +
          ", type='" + type + '\'' +
          '}';
    }
}
```

Jeans.java

1. Super

```
public class Jeans extends Clothing{

    String type; //slim, fit, ..

    public String display() {
      return "Jeans{" + super.display() +
            ", type='" + type + '\'' +
            '}';
    }
}
```

Access to super class display method

Jeans.java

1. Super

```
public class Main {

    public static void main(String[] args) {

        Jeans jeans = new Jeans();
        jeans.cod = "9663/310";
        jeans.price = 39.95;
        jeans.size = "40";
        jeans.color = "blue";
        jeans.genre = 'M';
        jeans.type = "slim";
        System.out.println(jeans.display());
            //Prints: Jeans{cod='9663/310', price=39.95, size='40', color='blue', genre=M,        //type='slim'}
    }
}
```

Main.java

1. Super

¿Cuál es la salida por consola al ejecutar el siguiente código?

```
Truck myCar = new Truck();
myCar.m1();
myCar.m2();
```

```
public class Car{
  public void m1(){ System.out.println("car 1"); }
  public void m2(){ System.out.println("car 2"); }

}
```

```
public class Truck extends Car{
  public void m1(){ System.out.println("truck 1"); }
  public void m2(){ super.m1(); }
}
```

1. Super

¿Cuál es la salida por consola al ejecutar el siguiente código?

```
Truck myCar = new Truck();
myCar.m1(); //Prints truck 1
myCar.m2();
```

```
public class Car{
  public void m1(){ System.out.println("car 1"); }
  public void m2(){ System.out.println("car 2"); }

}
```

```
public class Truck extends Car{
  public void m1(){ System.out.println("truck 1"); }
  public void m2(){ super.m1(); }
}
```

1. Super

¿Cuál es la salida por consola al ejecutar el siguiente código?

```
Truck myCar = new Truck();
myCar.m1(); //Prints truck 1
myCar.m2(); //Prints car 1
```

```
public class Car{
  public void m1(){ System.out.println("car 1"); }
  public void m2(){ System.out.println("car 2"); }

}
```

```
public class Truck extends Car{
  public void m1(){ System.out.println("truck 1"); }
  public void m2(){ super.m1(); }
}
```

1. Super

super()

1. Super

```
public class Clothing {
    String cod;
    double price;
    String size;
    String color;
    char genre; //W==Woman, M==Man


    public String display() {
        return "cod='" + cod + '\'' +
                ", price=" + price +
                ", size='" + size + '\'' +
                ", color='" + color + '\'' +
                ", genre=" + genre;
    }
}
```

Clothing.java

1. Super

```
public class Clothing {
   String cod;
   double price;
   String size;
   String color;
   char genre; //W==Woman, M==Man

   public Clothing(String cod, double price, String size, String color, char genre) {
    this.cod = cod;
    this.price = price;
    this.size = size;
    this.color = color;
    this.genre = genre;
   }
   public String display() {
      return "cod='" + cod + '\'' +
             ", price=" + price +
             ", size='" + size + '\'' +
             ", color='" + color + '\'' +
             ", genre=" + genre;
   }
}
```

Clothing.java

1. Super

```
public class Jeans extends Clothing{

   String type; //slim, fit, ..

   public Jeans(String cod, double price, String size,
                String color, char genre, String type) {
     super(cod, price, size, color, genre);
     this.type = type;
   }

   public String display() {
    return "Jeans{" + super.display() +
          ", type='" + type + '\'' +
          '}';
   }
}
```

Jeans.java

1. Super

```
public class Jeans extends Clothing{

   String type; //slim, fit, ..

   public Jeans(String cod, double price, String size,
                String color, char genre, String type) {
     super(cod, price, size, color, genre);
     this.type = type;
   }

   public String display() {
     return "Jeans{" + super.display() +
          ", type='" + type + '\'' +
          '}';
   }
}
```

1st call the superclass constructor

Jeans.java

1. Super

```
public class Jeans extends Clothing{

   String type; //slim, fit, ..

   public Jeans(String cod, double price, String size,
                String color, char genre, String type) {
     super(cod, price, size, color, genre);
     this.type = type;
   }

   public String display() {
     return "Jeans{" + super.display() +
          ", type='" + type + '\'' +
          '}';
   }
}
```

2nd add more code (if necessary)

Jeans.java

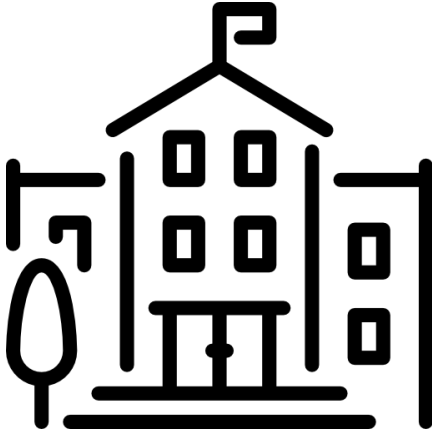
1. Super

```
public class Jeans extends Clothing{

   String type; //slim, fit, ..

   public Jeans(String cod, double price, String size,
                String color, char genre, String type) {
     this.type = type;
     super(cod, price, size, color, genre);
   }

   public String display() {
     return "Jeans{" + super.display() +
          ", type='" + type + '\'' +
          '}';
   }
}
```

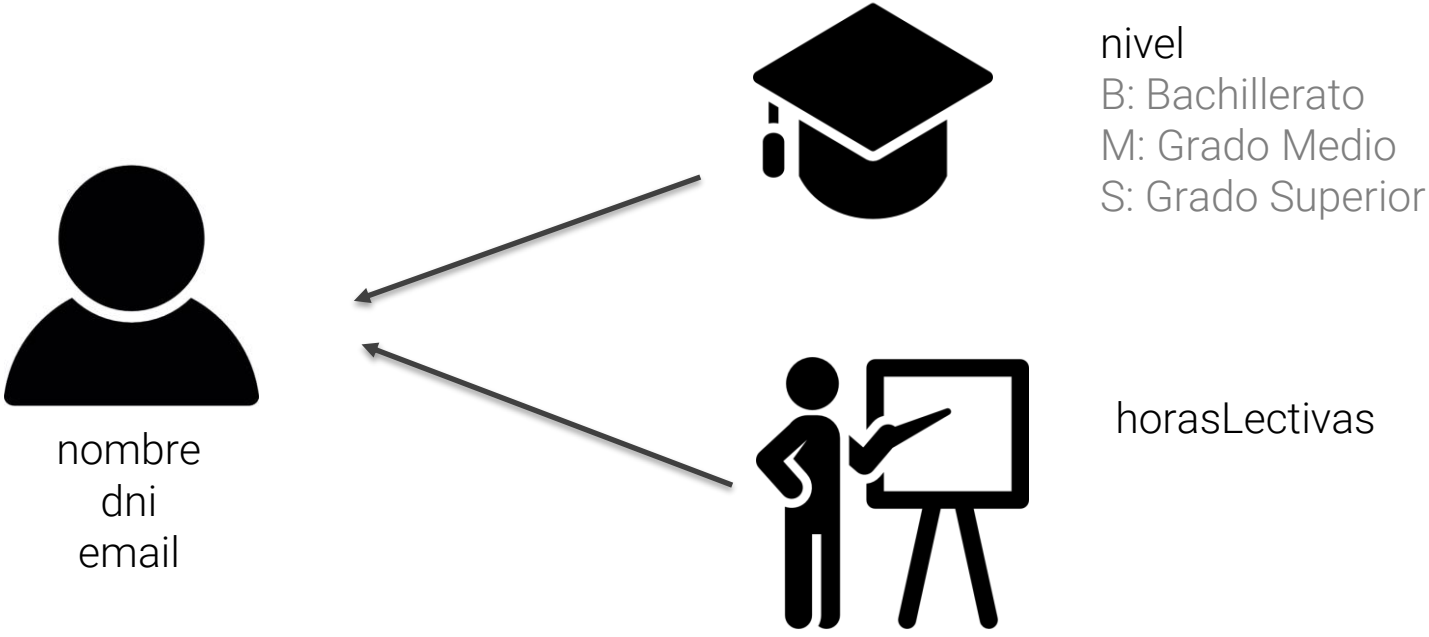
Error compilation!!
A call to super class constructor must be the first statement

Jeans.java

CENTRO DE ESTUDIOS FP



CENTRO DE ESTUDIOS FP

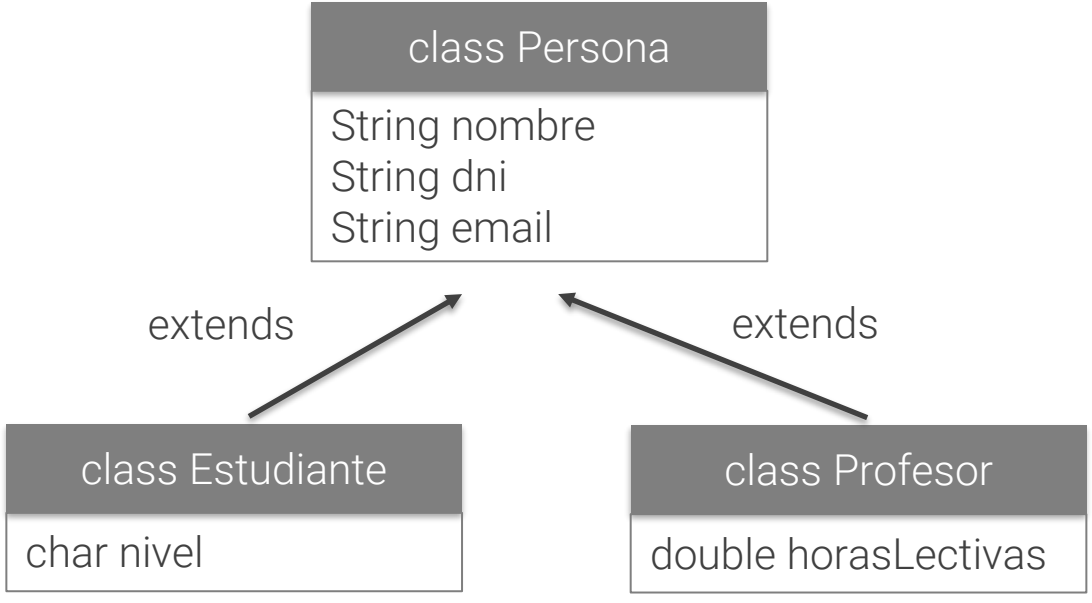


nombre
dni
email

nivel
B: Bachillerato
M: Grado Medio
S: Grado Superior

horasLectivas

CENTRO DE ESTUDIOS FP



CENTRO DE ESTUDIOS FP

nombre: Maria Sanz
dni: 1234567X
email: example@example.com
nivel: B

CENTRO DE ESTUDIOS FP

```
public class Persona {
   String nombre;
   String dni;
   String email;

   public Persona(String nombre, String dni, String email) {
      this.nombre = nombre;
      this.dni = dni;
      this.email = email;
   }

   public Persona(String nombre, String dni) {
      this.nombre = nombre;
      this.dni = dni;
      this.email = "example@example.com";
   }

}
```

CENTRO DE ESTUDIOS FP

```
public class Estudiante extends Persona {
   char nivel;

   public Estudiante(String nombre, String dni, String email, char nivel){
      super(nombre, dni, email);
      this.nivel = nivel;
   }

   public Estudiante(String nombre, String dni, char nivel){
      super(nombre, dni);
      this.nivel = nivel;
   }
}
```

CENTRO DE ESTUDIOS FP

```
public class Estudiante extends Persona {
   char nivel;

   public Estudiante(String nombre, String dni, String email, char nivel){
      super(nombre, dni, email);
      this.nivel = nivel;
   }

   public Estudiante(String nombre, String dni, char nivel){
      super(nombre, dni);
      this.nivel = nivel;
   }
}
```

“Do what you think is interesting, do something that you think is fun and worthwhile, because otherwise you won't do it well anyway “

Brian Kernigan, científico de la computación que trabajó el laboratorios Bell y ayudó en la creación del sistema operativo Unix.