

Proyecto 1º Trimestre

Jairo Rastrojo Rodríguez

Hospital:

En este proyecto vamos a simular la gestión de datos de la zona de urgencias de un hospital privado.

Para ello, emplearemos diversos elementos de la programación orientada a objetos.

Clases:

Las clases son descripciones de un conjunto de objetos similares, consta de métodos y de datos que resumen las características de los conjuntos de objetos, es decir es la declaración de un tipo de objeto usado para abstraer datos.

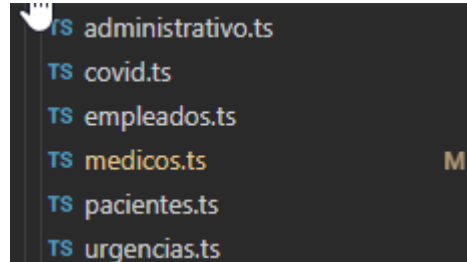
Este proyecto dispone de dos clases principales, la clase pacientes y la clase empleados, los cuales tienen cada una dos subclases heredadas:

-Empleados: dispone de las subclases médicos y administrativos, los cuales no solo heredan de empleados si no que disponen de sus propios campos y sobrescriben métodos de la empleados.

-Pacientes: tiene dos clases que heredan de ella, la clase urgencias y la clase covid, que heredan campos de pacientes y sobrescriben métodos de pacientes.

Usando export class “Nombre de clase” creamos una clase

```
import { Paciente } from "../pacientes";  
  
export class Empleado {
```



TS administrativo.ts
TS covid.ts
TS empleados.ts
TS medicos.ts
TS pacientes.ts
TS urgencias.ts

Herencias:

Las herencias permite definir nuevas clases basadas de una que ya existe para reutilizar código generando así una jerarquía de clases.

Heredamos escribiendo “export class “Nombre subclase” extend “clase de la que hereda”

```
import {Empleado} from "../empleados";
import {Paciente} from "../pacientes";

export class Medicos extends Empleado{
```

Polimorfismo

El polimorfismo se da cuando creamos un método en la superclase, y en la clase heredada sobrescribimos el método haciendo cambios en el mismo, pudiendo usar el método específico dependiendo de lo que queramos hacer.

Un ejemplo de polimorfismo sería lo siguiente, en la clase paciente se ha creado el método pago, y en la subclase covid, sobrescribimos el método para adecuarlo a dicha subclase

```
pago(): number{
  let pago: number;
  pago = this._preciobase;
  if (this._seguro == false){
    pago = 80;
  } else {
    pago = 0
  }
  return pago
}
```

```
pago(): number{
  let pago: number;
  pago = super.pago();
  if (this._seguro == false){
    if (this._test == 'antigeno'){
      pago += 50 + pago;
    } else if (this._test == 'PCR'){
      pago += 90 + pago
    } else if (this._test == 'serologico'){
      pago += 45 + pago
    }
  } else {
    pago = 0
  }
  return pago
}
```

Menús

Disponemos de tres menús:

Menú principal: El cual nos da acceso a 2 submenús

Menú atender: Es un submenú que permite gestionar los pacientes

Menú emplear: Menú que permite gestionar los trabajadores

Mongo Atlas

Hacer persistente nuestro proyecto en una base de datos es en sí el echo de subir los datos de los objetos, transformados con el uso de los schemas a un lenguaje mongo y subirlo (en este caso) a mongo atlas.