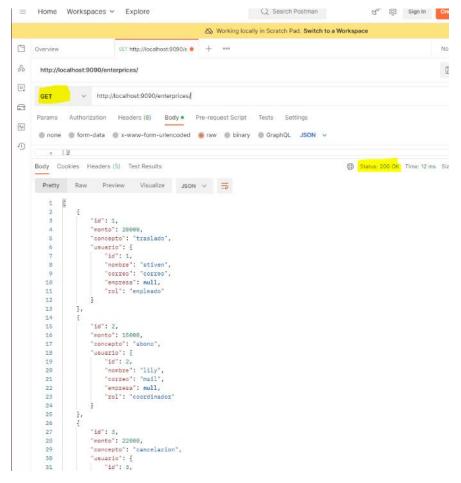
Servicios Movimientos Dinero

```
2 usages 🚨 Reyezonia *
@Service
public class MovimientoDineroService {
   9 usages
    @Autowired //se conecta con el repo
    MovimientoDineroRepository movimientoDineroRepo;
    1 usage 🚨 Reyezonia *
    public List<MovimientoDinero> getAllMovimientoDinero(){//listar
       List<MovimientoDinero> listaMovimientoDinero = new ArrayList<>();
       movimientoDineroRepo.findAll().forEach(movimientoDinero -> ListaMovimientoDinero.add(movimientoDinero));
       return listaMovimientoDinero;
    //metodo para buscar
    3 usages 🐣 Reyezonia *
    public MovimientoDinero getMovimientoDinero(Integer id){ //buscar con id
       return movimientoDineroRepo.findById(id).get();
    //guardar o actualizar
    2 usages new *
    public MovimientoDinero actualizarMovimientoDinero(MovimientoDinero movDinero){
       MovimientoDinero movimientoDinero = movimientoDineroRepo.save(movDinero);
       return movimientoDinero;
    //metodo para eliminar
    1 usage 🚨 Reyezonia
    public boolean eliminarMovimientoDinero(Integer id){
        movimientoDineroRepo.deleteById(id);
        if (getMovimientoDinero(id)!= null){
           return false;
       }
       return true;
```

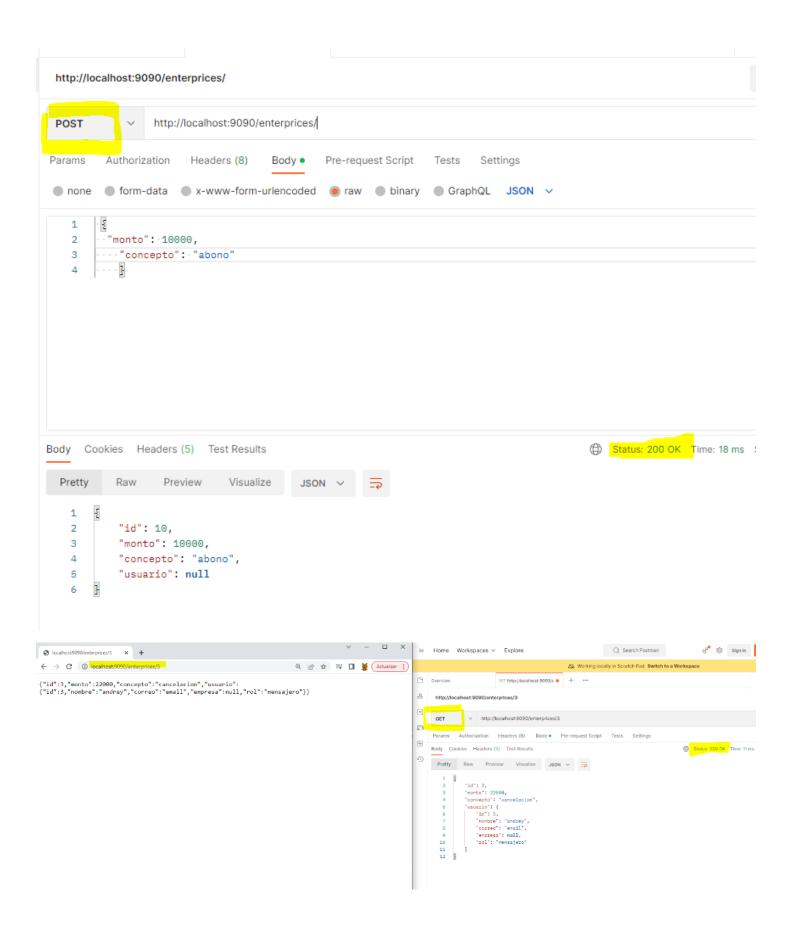
Controlador Movimiento Dinero

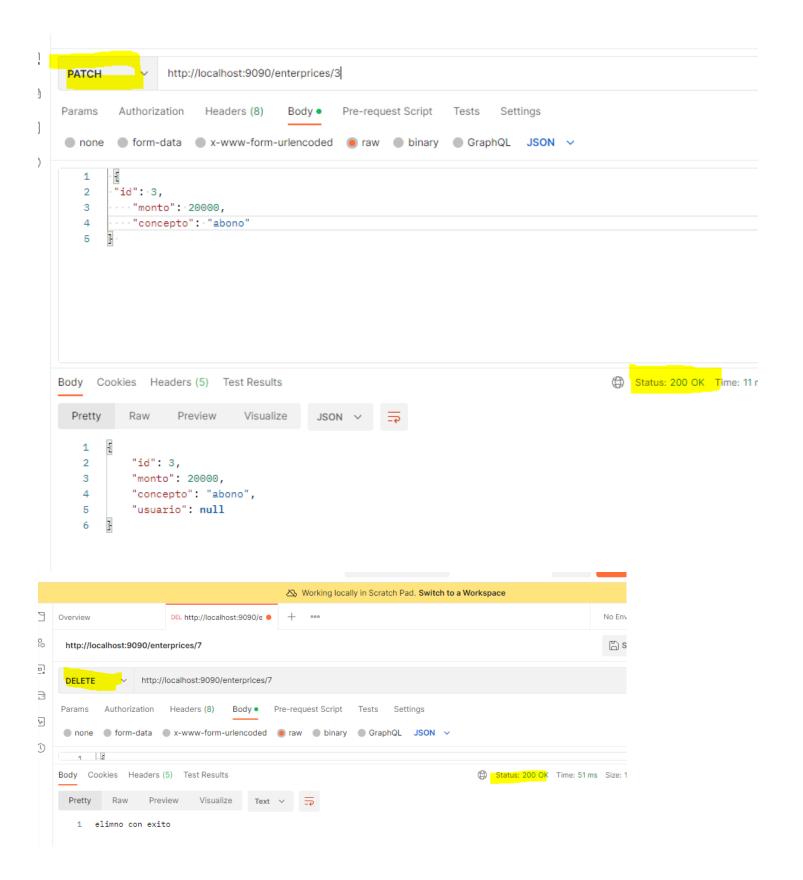
```
6 usages
@Autowired //se comunica con el service
MovimientoDineroService movDineroServ;
@GetMapping({"/enterprices"})//json de todas las empresa se ve el listado
public List<MovimientoDinero> verMovimiento(){
    return movDineroServ.getAllMovimientoDinero();
@PostMapping({"/enterprices"})//se guarda el json del body como un nuevo registro
public MovimientoDinero guardarMovimiento(@RequestBody MovimientoDinero movDinero){
    return this.movDineroServ.actualizarMovimientoDinero(movDinero);
}
@GetMapping(path="/enterprices/{id}")//buscar por id
public MovimientoDinero buscarPorId(@PathVariable("id") Integer id){
    return this.movDineroServ.getMovimientoDinero(id);
♣ Reyezonia *
@PatchMapping(path="/enterprices/{id}")//actualizar por id
public MovimientoDinero actualizarMovimiento(@PathVariable("id") Integer id, @RequestBody MovimientoDinero movDinero){
    MovimientoDinero movimientoDinero = movDineroServ.getMovimientoDinero(id);
    movimientoDinero.setMonto(movDinero.getMonto());
    movimientoDinero.setConcepto(movDinero.getConcepto());
    movimientoDinero.setUsuario(movDinero.getUsuario());
    return this.movDineroServ.actualizarMovimientoDinero(movimientoDinero);
}
4 Reyezonia *
@DeleteMapping(path="/enterprices/{id}")
public String eliminarmovimiento(@PathVariable("id") Integer id){
    boolean respuesta = this.movDineroServ.eliminarMovimientoDinero(id);
   if (respuesta == true){
        return "elimino con exito";
    }else{
        return "no se ha eliminado";
```

Resultados 200









← → C ① localhost:9090/enterprices

@ ₺ ☆

```
[{"id":2,"monto":15000,"concepto":"abono","usuario":
{"id":2, "nombre": "lily", "correo": "mail", "empresa": null, "rol": "coordinador"}},
{"id":9, "monto":0, "concepto":null, "usuario":null},
{"id":10, "monto":10000, "concepto": "abono", "usuario":null},
{"id":3, "monto":20000, "concepto": "abono", "usuario": null}]
```