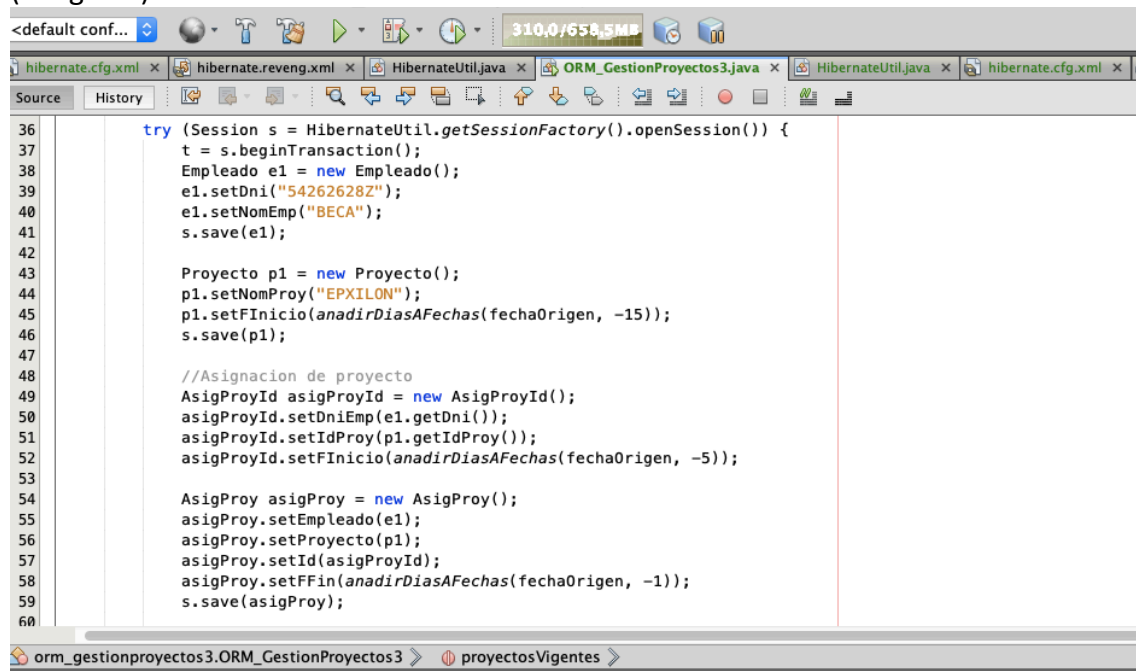


En este ejercicio 3, hemos realizado tanto el ejercicio 3 como el 5,6 y 7.

Insertando datos de empleados, proyectos y sus correspondiente fechas en la asignación de proyectos usando la clase Calendar y Date de Java para restar o aumentar días (Imagen 1).

También se muestra un listado detallado de todas las tablas (Imagen 2), así como dos listados más mostrando información sobre la asignación de proyectos con fechas de FIN pasadas (No vigentes) (Imagen 3) y otro listado con asignación de proyectos que tienen fecha de FIN futuras o NULAS (Vigentes) (Imagen 4) haciendo uso de cláusulas y funciones de SQL dentro de la consulta FROM de HQL.

(Imagen 1):



```
36 try (Session s = HibernateUtil.getSessionFactory().openSession()) {
37     t = s.beginTransaction();
38     Empleado e1 = new Empleado();
39     e1.setDni("54262628Z");
40     e1.setNomEmp("BECA");
41     s.save(e1);
42
43     Proyecto p1 = new Proyecto();
44     p1.setNomProy("EPXILON");
45     p1.setFinicio(anadirDiasAFechas(fechaOrigen, -15));
46     s.save(p1);
47
48     //Asignacion de proyecto
49     AsigProyId asigProyId = new AsigProyId();
50     asigProyId.setDniEmp(e1.getDni());
51     asigProyId.setIdProy(p1.getIdProy());
52     asigProyId.setFinicio(anadirDiasAFechas(fechaOrigen, -5));
53
54     AsigProy asigProy = new AsigProy();
55     asigProy.setEmpleado(e1);
56     asigProy.setProyecto(p1);
57     asigProy.setId(asigProyId);
58     asigProy.setFFin(anadirDiasAFechas(fechaOrigen, -1));
59     s.save(asigProy);
60 }
```

(Imagen 2)

The screenshot shows a Java IDE with a project named `orm_gestionproyectos3`. The code in `ORM_GestionProyectos3` defines two queries:

```

Query query2 = s.createQuery("FROM Empleado").setReadOnly(true);
List<Empleado> lEmpleados = query2.getResultList();
System.out.println(" TABLA EMPLEADOS:");
System.out.println("-----");
System.out.println(" DNI    NOMBRE ");
System.out.println();
for (Empleado item : lEmpleados) {
    System.out.println(" " + item.getDni() + "    " + item.getNomEmp());
    System.out.println("-----");
}
System.out.println("-----Fin de tabla-----");

Query query3 = s.createQuery("FROM AsigProy").setReadOnly(true);
List<AsigProy> lAsigProy = query3.getResultList();
System.out.println(" TABLA ASIG_PROY:");
System.out.println("-----");
System.out.println(" DNI    ID_PROY    FECHAIni    FECHAFin ");
System.out.println();
for (AsigProy item : lAsigProy) {
    System.out.println(" " + item.getId().getDniEmp() + "    " + item.getId().getIdProy() + "    " + item.getFechaIni() + "    " + item.getFechaFin());
    System.out.println("-----");
}

```

The output window shows the results of the second query:

DNI	ID_PROY	FECHAIni	FECHAFin
54262628Z	10	2020-02-08	2020-02-12
74262678M	9	2020-02-08	2020-02-12
74345678B	1	2020-02-18	null
74345678D	2	2020-02-18	null

(Imagen 3)

The screenshot shows the same IDE with additional code for querying non-vigilant and vigilent projects:

```

public static void selectTablas() {...40 lines}

public static void proyectosNoVigilantes(){
    try (Session s = HibernateUtil.getSessionFactory().openSession()) {
        Query query3 = s.createQuery("FROM AsigProy WHERE FFIn> date(now()) or FFIn is null").setReadOnly(true);
        List<AsigProy> lAsigProy = query3.getResultList();
        System.out.println(" TABLA ASIG_PROY NO VIGENTES:");
        System.out.println("-----");
        System.out.println(" DNI    ID_PROY    FECHAIni    FECHAFin ");
        System.out.println();
        for (AsigProy item : lAsigProy) {
            System.out.println(" " + item.getId().getDniEmp() + "    " + item.getId().getIdProy() + "    " + item.getFechaIni() + "    " + item.getFechaFin());
            System.out.println("-----");
        }
        System.out.println("-----Fin de tabla-----");
    }
}

public static void proyectosVigilantes(){...15 lines}

```

The output window shows the results of the `proyectosNoVigilantes` query:

DNI	ID_PROY	FECHAIni	FECHAFin
74345678B	1	2020-02-18	null
74345678D	2	2020-02-18	null
74365678S	3	2020-02-18	null
74365678Z	4	2020-02-18	null

(Imagen 4)

The screenshot shows an IDE with a Java file named `ORM_GestionProyectos3`. The code defines three static methods: `selectTablas()`, `proyectosNoVigentes()`, and `proyectosVigentes()`. The `proyectosVigentes()` method uses Hibernate to query the database for active projects and prints them in a table format.

```
128 public static void selectTablas() { ...40 lines }
129
144 public static void proyectosNoVigentes() { ...15 lines }
145
146 public static void proyectosVigentes(){
147     try (Session s = HibernateUtil.getSessionFactory().openSession()) {
148         Query query3 = s.createQuery("FROM AsigProy WHERE FFin<= date(now())").setReadOnly(true);
149         List<AsigProy> lAsigProy = query3.getResultList();
150         System.out.println(" TABLA ASIG_PROY VIGENTES:");
151         System.out.println("-----");
152         System.out.println(" DNI      ID_PROY    FECHAIni    FECHAFin ");
153         System.out.println();
154         for (AsigProy item : lAsigProy) {
155             System.out.println(" " + item.getId().getDniEmp()+ "      " + item.getId().getIdProy()+ "      "+ite
156             System.out.println("-----");
157         }
158         System.out.println("-----Fin de tabla-----");
159     }
160 }
161
162
```

The output window shows the result of the `proyectosVigentes()` method, displaying a table of active projects.

```
-----Fin de tabla-----
TABLA ASIG_PROY VIGENTES:
DNI      ID_PROY    FECHAIni    FECHAFin
54262628Z  10    2020-02-08    2020-02-12
74262678M  9    2020-02-08    2020-02-12
74362678R  8    2020-02-08    2020-02-12
```