**Practica 4: Pruebas de Software**

**2. Actividades**

**2.1.-Pruebas unitarias de la clase Empleado**

**2.1.1.-Diseñar los casos de prueba para la clase Empleado utilizando las técnicas de partición equivalente y AVL vistas en clase**

Usando AVL y partición equivalente obtenemos las siguientes clases de equivalencia válidas y no válidas:

|  |  |  |
| --- | --- | --- |
| **Parámetro** | **Clases Equivalencia Válidas** | **Clases Equivalencia No Válidas** |
| **Fecha** | **1.[Hoy-5 años, Hoy]**  **2.[Hoy-10 años, Hoy-5 años)**  **3.[Hoy-20 años, Hoy-10 años)**  **4.[Hoy-∞, Hoy-20 años)** | **7.{Hoy+1día, null}** |
| **Categoría** | **5.{DIRECTIVO/EJECUTIVO/OBRERO}** | **8.null,**  **!{DIRECTIVO/EJECUTIVO/OBRERO}** |
| **Baja** | **6.{TRUE/FALSE}** | **9.null** |

A partir de estas clases de equivalencia, obtenemos los siguientes casos de prueba (1-wise) formato: ([Fecha, categoría, Baja]):

**CASOS DE PRUEBA VÁLIDOS**

1.[DIRECTIVO, Hoy, true]=1125€

2. [GESTOR, Hoy-3 años, false]=1200€

3. [OBRERO, Hoy-5 años+1día, true]=750€

4. [DIRECTIVO, Hoy-5 años, false]=1550€

5. [GESTOR, Hoy-7 años, true]=937.5€

6. [OBRERO, Hoy-10 años+1dia, false]=1050€

7. [GESTOR, Hoy-10años, true]=975€

8. [DIRECTIVO, Hoy-15 años, false]=1600€

9. [GESTOR, Hoy-20 años+1día, true]=975€

10. [GESTOR, Hoy-20 años, false]=1400€

11. [GESTOR, Hoy-50 años, true]=1050€

**CASOS DE PRUEBA NO VÁLIDOS**

12.[GESTOR, Hoy+1día, true]

13. [GESTOR, null, true]

14. [null, Hoy, true]

15. [PEPE, Hoy, true]

16. [GESTOR, Hoy, null]

(15 y 16 no probables en código, comentadas dentro de la clase EmpleadoTest)

**2.1.5.-Cuando todas las pruebas de caja negra sean correctas, ejecutar pruebas de caja blanca y comprobar la cobertura**

Al necesitar aplicar las pruebas de caja blanca en base a una cobertura de condiciones y decisiones, todas las decisiones de dentro de una condición y todas las condiciones han de ser evaluadas en verdadero y falso, esto ya esta hecho en las pruebas de caja negra

1.-fechaContratacion==null🡪True: **Caso de prueba 13**

2.-fechaContratacion==null🡪False: **Casos de prueba 1-12 y 14-16**

3.-fechaContratación.isAfter(LocalDate.now())🡪True: **Caso de prueba 12**

4.-fechaContratación.isAfter(LocalDate.now())🡪False: **Casos de prueba 1-11 y 13-16**

5.-Decisión 1🡪True: **Casos de prueba 12,13,14,15,16**

6.-Decisión 1🡪False: **Casos de prueba 1-11**

7.-Directivo🡪True: **Casos de prueba 1,4,8**

8.-Directivo🡪False: **Casos de prueba 1,3,5,6,7,9,10,11,12,13,14,15,16**

9.-Gestor🡪True: **Casos de prueba 2,5,7,9,10,11**

10.-Gestor🡪False: **Casos de prueba 1,3,4,6,8**

11.-Obrero🡪True: **Casos de prueba 3,6**

12.-Obrero🡪False: **Casos de prueba 1,2,4,5,7,8,9,10,11,12,13,14,15,16**

13.-fechaContratación.isBefore(LocalDate.now().minusYears(5).plusDays(1))🡪True:

**Casos de prueba 1,2,3**

14.-fechaContratación.isBefore(LocalDate.now().minusYears(5).plusDays(1))🡪False:

**Casos de prueba 4,5,6,7,8,9,10,11,13,14,15,16**

15.-fechaContratación.isAfter(LocalDate.now().minusYears(10))🡪True:

**Casos de prueba 1,2,3**

16.-fechaContratación.isAfter(LocalDate.now().minusYears(10))🡪False:

**Casos de prueba 4,5,6,7,8,9,10,11**

17.- fechaContratación.isBefore(LocalDate.now().minusYears(10).plusDays(1))🡪True:

**Casos de prueba 4,5,6**

18.- fechaContratación.isBefore(LocalDate.now().minusYears(10).plusDays(1))🡪False:

**Casos de prueba 1,2,3,7,8,9,10,11**

19.-fechaContratación.isAfter(LocalDate.now().minusYears(20))🡪True:

**Casos de prueba 4,5,6**

20.-fechaContratación.isAfter(LocalDate.now().minusYears(20))🡪False

**Casos de prueba 1,2,3,7,8,9,10,11**

21.-fechaContratación.isBefore(LocalDate.now().minusYears(20).plusDays(1))🡪True:

**Casos de prueba 10,11**

22.-fechaContratación.isBefore(LocalDate.now().minusYears(20).plusDays(1))🡪False:

**Casos de prueba 1,2,3,4,5,6,7,8,9**

23.-baja🡪True: **Casos de prueba 1,3,5,7,9,11,12,13,14,15**

24.-baja🡪False: **Casos de prueba 2,4,6,8,10**

**2.2.1.-Indicar qué casos de prueba se consideraría necesario aplicar en dicha prueba de integración**