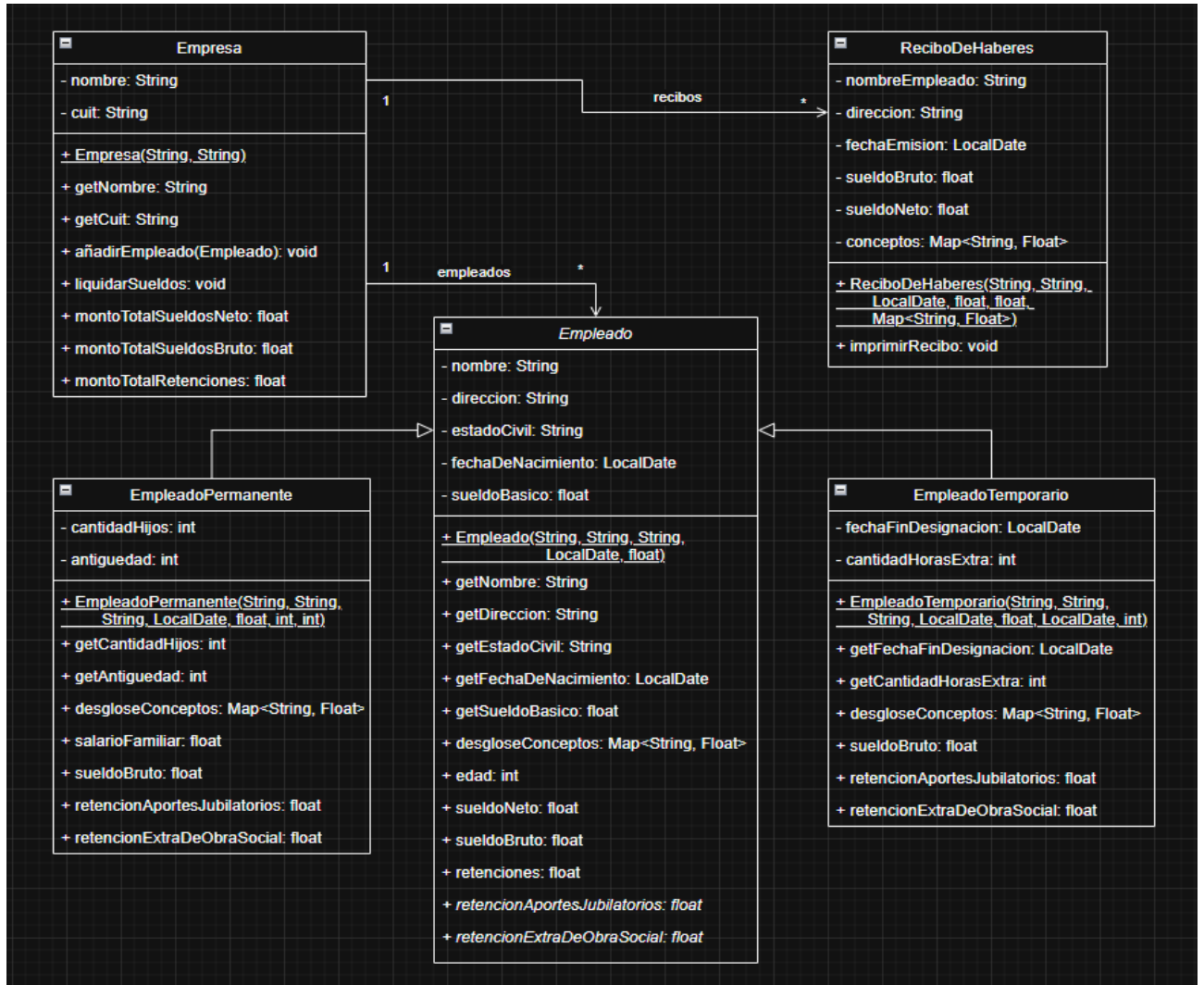


TRABAJO PRÁCTICO N°2: UML

● EJERCICIO 1:

- 1.A: Realizar un diagrama de clases UML del modelo descrito anteriormente.



- 1.B: Implementar en un lenguaje OO que usted conozca el modelo del punto anterior.

- 1.C: Muestre en dos scripts de código cómo instancia un escenario concreto en el cual se dispare I) el cálculo del total de sueldos neto y II) la liquidación de sueldo.

```
public class EmpresaTest {
    private Empresa volkswagen;
    private EmpleadoPermanente jorge;
    private EmpleadoTemporario ricardo;
    private EmpleadoContratado juana;

    @BeforeEach
    public void setUp() {
        volkswagen = new Empresa("Volkswagen", "20-99104531-7");

        jorge = new EmpleadoPermanente("Jorge", "Av. La Plata 3981", "Casado", LocalDate.of(1979, 9, 22), 435000f, 2, 10);
        ricardo = new EmpleadoTemporario("Ricardo", "Edmundo Fierro 847", "Soltero", LocalDate.of(2002, 1, 22), 948000f, LocalDate.of(2025, 8, 31), 4);
        juana = new EmpleadoContratado("Juana", "Garibaldi 910", "Soltero", LocalDate.of(1993, 4, 18), 999000f, 10, "Tarjeta");

        volkswagen.añadirEmpleado(jorge);
        volkswagen.añadirEmpleado(ricardo);
        volkswagen.añadirEmpleado(juana);
        volkswagen.liquidarSueldos();
    }

    @Test
    public void funcionamientoDeLosGetter() {
        assertEquals("Volkswagen", volkswagen.getNombre());
        assertEquals("20-99104531-7", volkswagen.getCuit());

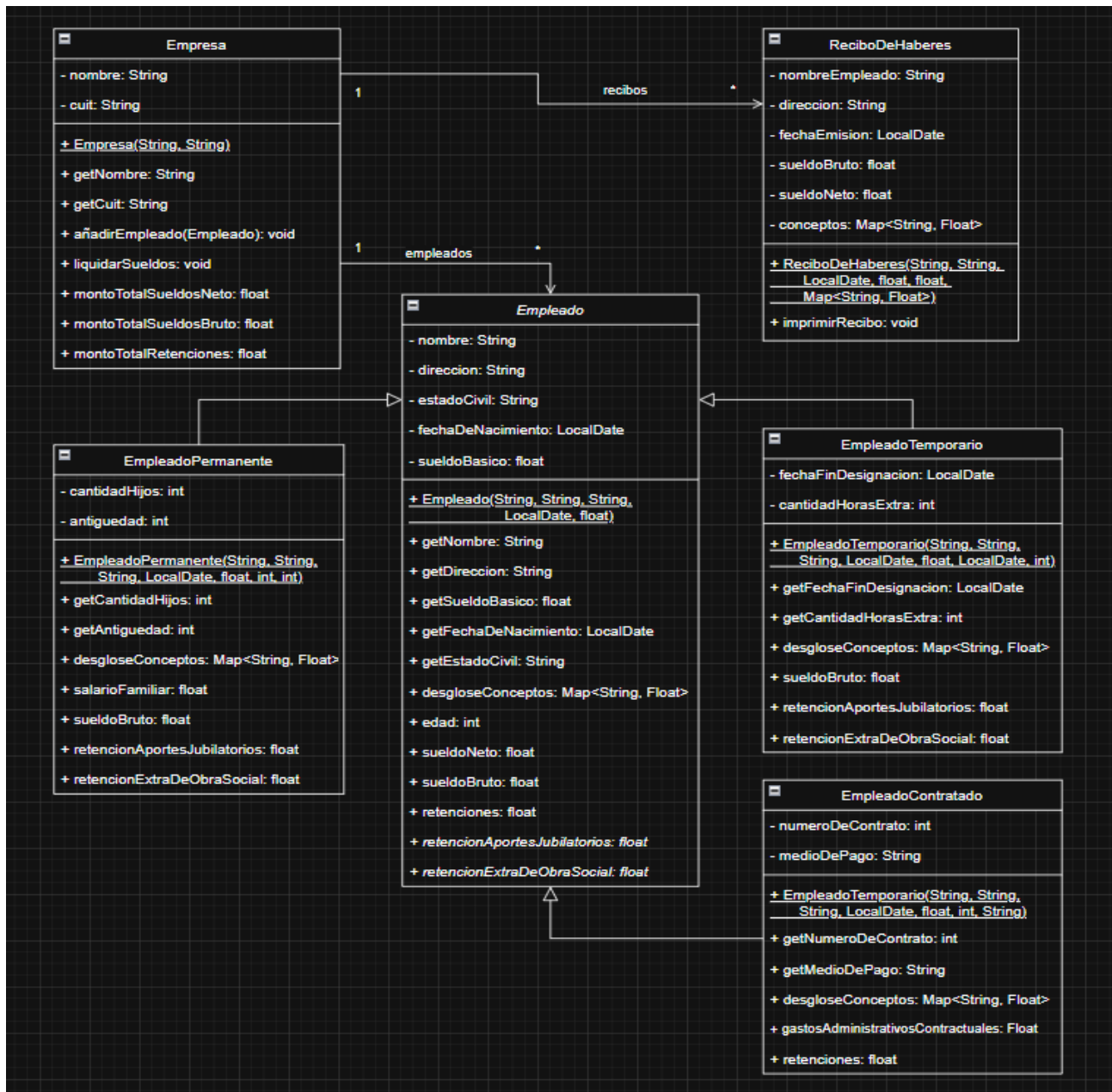
        assertEquals(2084343f, volkswagen.montoTotalSueldosNeto());
        assertEquals(2383060f, volkswagen.montoTotalSueldosBruto());
        assertEquals(298717f, volkswagen.montoTotalRetenciones());
    }

    @Test
    public void imprimirRecibos() {
        List<ReciboDeHaber> recibos = volkswagen.getRecibos();

        for (ReciboDeHaber recibo : recibos) {
            recibo.imprimirRecibo();
        }
    }
}
```

● EJERCICIO 2:

● 2.A: Extender el diagrama UML del Ejercicio 1 para incluir esta extensión.



● 2.B: Extender el código del Ejercicio 1 para incluir esta extensión

● 2.C: Para realizar esta extensión del modelo, ¿necesitó modificar la clase Empresa? ¿Por qué?

- No se necesitó modificar la clase empresa porque no forma parte de las responsabilidades de la nueva extensión, porque a la hora de liquidar los sueldos trata a cada uno de los empleados de forma homogénea sin diferenciación, por ende, si es subclase de Empleado no debería de necesitarse ninguna modificación.

- 2.D: Dada esta extensión del modelo, actualice los scripts de código de instanciación de escenarios para incluir también ahora empleados contratados.

```
public class EmpresaTest {
    private Empresa volkswagen;
    private EmpleadoPermanente jorge;
    private EmpleadoTemporario ricardo;
    private EmpleadoContratado juana;

    @BeforeEach
    public void setUp() {
        volkswagen = new Empresa("Volkswagen", "20-99104531-7");

        jorge = new EmpleadoPermanente("Jorge", "Av. La Plata 3981", "Casado", LocalDate.of(1979, 9, 22), 435000f, 2, 10);
        ricardo = new EmpleadoTemporario("Ricardo", "Edmundo Fierro 847", "Soltero", LocalDate.of(2002, 1, 22), 948000f, LocalDate.of(2025, 8, 31), 4);
        juana = new EmpleadoContratado("Juana", "Garibaldi 910", "Soltero", LocalDate.of(1993, 4, 18), 999000f, 10, "Tarjeta");

        volkswagen.añadirEmpleado(jorge);
        volkswagen.añadirEmpleado(ricardo);
        volkswagen.añadirEmpleado(juana);
        volkswagen.liquidarSueldos();
    }

    @Test
    public void funcionamientoDeLosGetter() {
        assertEquals("Volkswagen", volkswagen.getNombre());
        assertEquals("20-99104531-7", volkswagen.getCuit());

        assertEquals(2084343f, volkswagen.montoTotalSueldosNeto());
        assertEquals(2383060f, volkswagen.montoTotalSueldosBruto());
        assertEquals(298717f, volkswagen.montoTotalRetenciones());
    }

    @Test
    public void imprimirRecibos() {
        List<ReciboDeHaber> recibos = volkswagen.getRecibos();

        for (ReciboDeHaber recibo : recibos) {
            recibo.imprimirRecibo();
        }
    }
}
```