

```

package model;

import static org.junit.jupiter.api.Assertions.*;

import java.time.LocalDate;

import org.junit.jupiter.api.Test;

import exceptions.DayAndIDException;
import exceptions.Under18YearsOldException;

class TiendaTest {

    private int today = LocalDate.now().getDayOfMonth();
    private Tienda tienda = new Tienda();

    public void setupEscenary1() {

    }

    @Test
    void test1() {
        setupEscenary1();
        if(today%2 == 0) {
            try {
                tienda.addCustomer("1006584131", "CC");
            } catch (Under18YearsOldException e) {
                System.out.println("No se permite el ingreso a menores
de edad\n");
                fail();
            } catch (DayAndIDException e) {
                System.out.println("No cumple con el Pico y Cedula,
por lo que no puede entrar a la tienda\n");
                fail();
            }
        } else {
            try {
                tienda.addCustomer("1006584121", "CC");
            } catch (Under18YearsOldException e) {
                System.out.println("No se permite el ingreso a menores
de edad\n");
                fail();
            } catch (DayAndIDException e) {
                System.out.println("No cumple con el Pico y Cedula,
por lo que no puede entrar a la tienda\n");
                fail();
            }
        }
        assertEquals("CC", tienda.getCustomers().get(0).getIdType());
        if(today%2 == 0) {
            assertEquals("1006584131",
tienda.getCustomers().get(0).getId());
        } else {
            assertEquals("1006584121",
tienda.getCustomers().get(0).getId());
        }
    }
}

```

```

    }

}

@Test
void test2() {
    setupEscenary1();
    if(today%2 == 0) {
        try {
            tienda.addCustomer("1006584131", "CC");
        } catch (Under18YearsOldException e) {
            System.out.println("No se permite el ingreso a menores
de edad\n");
            fail();
        } catch (DayAndIDException e) {
            System.out.println("No cumple con el Pico y Cedula,
por lo que no puede entrar a la tienda\n");
            fail();
        }
    } else {
        try {
            tienda.addCustomer("1006584121", "CC");
        } catch (Under18YearsOldException e) {
            System.out.println("No se permite el ingreso a menores
de edad\n");
            fail();
        } catch (DayAndIDException e) {
            System.out.println("No cumple con el Pico y Cedula,
por lo que no puede entrar a la tienda\n");
            fail();
        }
    }
    assertEquals(1, tienda.getCount());
    assertEquals(false, tienda.getCustomers().isEmpty());
}

@Test
void test3() {
    setupEscenary1();
    try {
        tienda.addCustomer("11121313", "TI");
        fail();
    } catch (Under18YearsOldException e) {
        System.out.println("No se permite el ingreso a menores de
edad\n");
    } catch (DayAndIDException e) {
        System.out.println("No cumple con el Pico y Cedula, por lo
que no puede entrar a la tienda\n");
        fail();
    }
}

@Test
void test4() {
    setupEscenary1();

```

```

        if(today%2 == 0) {
            try {
                tienda.addCustomer("1006584121", "CC");
                fail();
            } catch (Under18YearsOldException e) {
                System.out.println("No se permite el ingreso a menores
de edad\n");
            } catch (DayAndIDException e) {
                System.out.println("No cumple con el Pico y Cedula,
por lo que no puede entrar a la tienda\n");
                fail();
            }
        } else {
            try {
                tienda.addCustomer("1006584131", "CC");
                fail();
            } catch (Under18YearsOldException e) {
                System.out.println("No se permite el ingreso a menores
de edad\n");
                fail();
            } catch (DayAndIDException e) {
                System.out.println("No cumple con el Pico y Cedula,
por lo que no puede entrar a la tienda\n");
            }
        }
    }
}
}

```