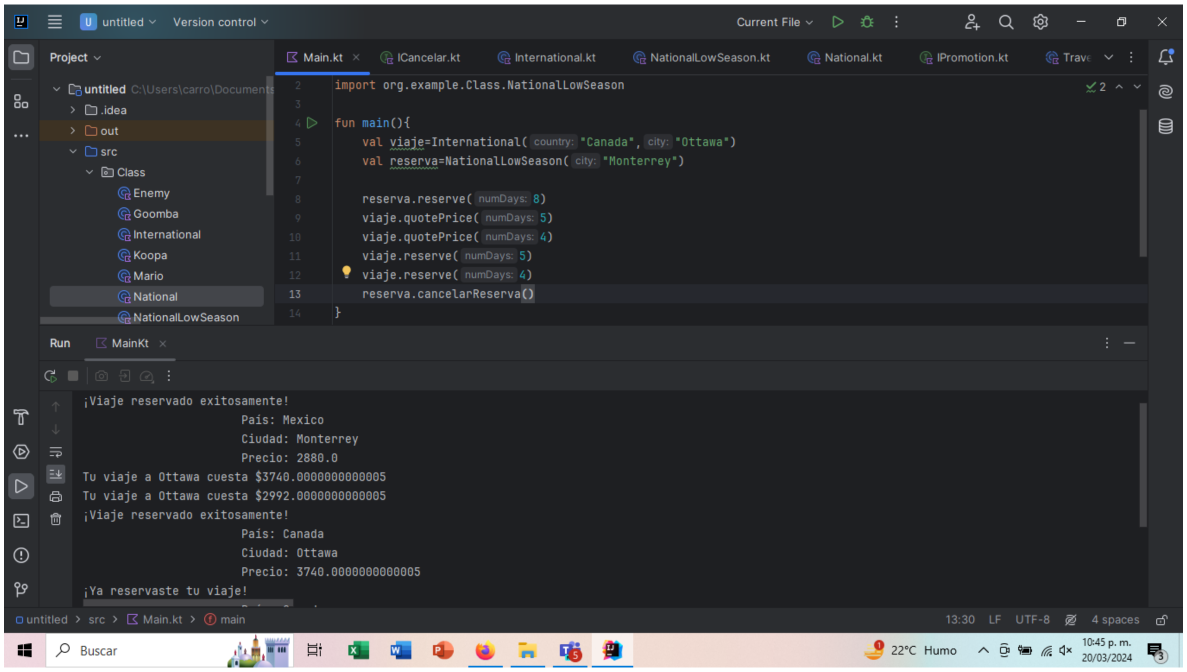
MORALES LUCIO ERICK ALEJANDRO BOLETA:2022601788 SECUENCIA:6NM61

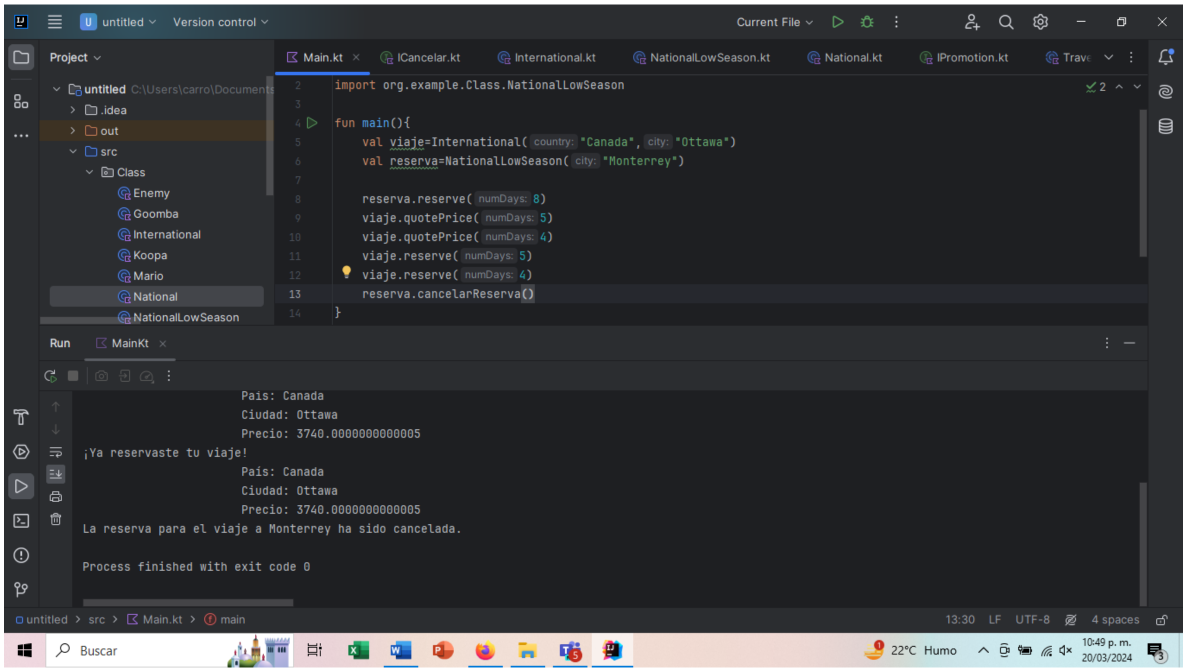


```
import org.example.Class.NationalLowSeason

fun main(){
    val viaje=International( country: "Canada", city: "Ottawa")
    val reserva=NationalLowSeason( city: "Monterrey")

    reserva.reserve( numDays: 8)
    viaje.quotePrice( numDays: 5)
    viaje.quotePrice( numDays: 4)
    viaje.reserve( numDays: 5)
    viaje.reserve( numDays: 4)
    reserva.cancelarReserva()
}
```

```
¡Viaje reservado exitosamente!
                       Pais: Mexico
                       Ciudad: Monterrey
                       Precio: 2880.0
Tu viaje a Ottawa cuesta $3740.0000000000005
Tu viaje a Ottawa cuesta $2992.0000000000005
¡Viaje reservado exitosamente!
                       Pais: Canada
                       Ciudad: Ottawa
                       Precio: 3740.0000000000005
¡Ya reservaste tu viaje!
```



```
import org.example.Class.NationalLowSeason

fun main(){
    val viaje=International( country: "Canada", city: "Ottawa")
    val reserva=NationalLowSeason( city: "Monterrey")

    reserva.reserve( numDays: 8)
    viaje.quotePrice( numDays: 5)
    viaje.quotePrice( numDays: 4)
    viaje.reserve( numDays: 5)
    viaje.reserve( numDays: 4)
    reserva.cancelarReserva()
}
```

```
                       Pais: Canada
                       Ciudad: Ottawa
                       Precio: 3740.0000000000005
¡Ya reservaste tu viaje!
                       Pais: Canada
                       Ciudad: Ottawa
                       Precio: 3740.0000000000005
La reserva para el viaje a Monterrey ha sido cancelada.

Process finished with exit code 0
```

```
CLASE INTERNATIONAL
package org.example.Class

import org.example.Interfaces.ICancelar

class International (override val country: String, override val city:
String): Travel(),ICancelar {
    protected val destination = mapOf(
        "Alemania" to mapOf(
            "Munich" to 980,
            "Berlin" to 820,
            "Francort" to 850
        ),
        "Chile" to mapOf(
            "Santiago" to 643,
            "Valparaiso" to 721
        ),
        "Canada" to mapOf(
            "Vancouver" to 620,
            "Ottawa" to 680,
            "Montreal" to 600
        )
    )




    override fun getPrice(numDays: Int):Double {
        val fee = destination[country]?.get(city)?: 0
        val impuesto:Double = when (country) {
            "Alemania" -> 1.16
            "Chile" -> 1.05
            "Canada" -> 1.10
            else -> 1.0 // Valor predeterminado si el país no está
especificado
        }

        return fee.toDouble() * numDays.toDouble() * impuesto

    }

    override fun cancelarReserva() {
        if (!reserved) {
            println("No hay ninguna reserva para cancelar.")
        } else {
            reserved = false
            paidAmount = 0.0
            println("La reserva para el viaje a $city ha sido
cancelada.")
        }
    }



}
```

INTERFAS

```
package org.example.Interfaces
interface ICancelar{

        fun cancelarReserva()
}
```

MORALES LUCIO ERICK ALEJANDRO   BOLETA:2022601788   SECUENCIA:6NM61

CLASSE NATIONALLOWSEASON

```
package org.example.Class

import org.example.Interfaces.IPromotion

class NationalLowSeason(city: String) : National(city), IPromotion {
    override  val discount = 10.0 //es porcentaje, o sea 10%
    override val typeDiscount = 0.0 //0 para porcentaje, 1 para cantidad
    override fun getPrice(numDays: Int): Double {
        val amount:Double = super.getPrice(numDays)
        return if (amount == 0.0) 0.0 else getDiscountPrice(amount)
    }
}
```

CLASSE NATIONAL

```
package org.example.Class

import org.example.Interfaces.ICancelar

open class National(override val city:String):Travel(),ICancelar {
    override val country = "Mexico"

    protected val fees = mapOf(
        "Monterrey" to 400,
        "Guadalajara" to 350,
        "CDMX" to 360,
        "San Cristóbal de las casas" to 240,
        "San Miguel de Allende" to 320
    )

    override fun getPrice(numDays: Int): Double {
        val fee = fees[city]
        return if (fee==null) 0.0 else fee.toDouble()*numDays
    }
    override fun cancelarReserva() {
        if (!reserved) {
            println("No hay ninguna reserva para cancelar.")
        } else {
            reserved = false
            paidAmount = 0.0
            println("La reserva para el viaje a $city ha sido 
cancelada.")
        }
    }

}
```