

```
ED EstructuraDatosAplicadas master Current File
Array.kt For.kt ExamenDos.kt Empleados.kt
1 data class Empleados(val noEmp: Int, var nombre: String, var puesto: String, var sueldo: Double) { 2 Usages new *
2 fun muestra() { 2 Usages new *
3     println("Numero: $noEmp, Nombre: $nombre, Puesto: $puesto, Sueldo: $sueldo")
4 }
5 }
6
7 val lista = arrayListOf<Empleados>() 7 Usages
8
9 fun main() { new *
10     var opcion: Int
11     do {
12         println("/// MENU DE EMPLEADOS ///")
13         println("1) agregar empleado")
14         println("2) buscar empleado")
15         println("3) modificar empleado")
16         println("4) eliminar empleado")
17         println("5) consultar todos los empleados")
18         println("6) salir")
19         print("elige una opcion: ")
20         opcion = readln().toIntOrNull() ?: -1
21         when (opcion) {
22             1 -> {
23                 try {
24                     print("ingresa el numero de empleado: ")
25                     val noEmp = readln().toInt()
26                     print("dame el nombre: ")
27                     val nombre = readln()
```

```
ED EstructuraDatosAplicadas master Current File
Array.kt For.kt ExamenDos.kt Empleados.kt
9 fun main() { new *
28         print("dame el puesto: ")
29         val puesto = readln()
30         print("dame el sueldo: ")
31         val sueldo = readln().toDouble()
32         agregarEmpleado(noEmp, nombre, puesto, sueldo)
33     } catch (e: Exception) {
34         println("datos no validos intenta de nuevo")
35     }
36 }
37 2 -> {
38     try {
39         print("ingresa el numero de empleado: ")
40         val noEmp = readln().toInt()
41         buscarEmpleado(noEmp)
42     } catch (e: Exception) {
43         println("datos invalidos.")
44     }
45 }
46 3 -> {
47     try {
48         print("ingresa el numero de empleado a modificar: ")
49         val noEmp = readln().toInt()
50         modificarEmpleado(noEmp)
51     } catch (e: Exception) {
52         println("entrada invalida")
53     }
}
```

```
ED EstructuraDatosAplicadas master
Current File

Array.kt For.kt ExamenDos.kt Empleados.kt x
9 fun main() { new *
54 }
55 4 -> {
56     try {
57         print("ingresa el numero de empleado a eliminar: ")
58         val noEmp = readln().toInt()
59         eliminarEmpleado(noEmp)
60     } catch (e: Exception) {
61         println("entrada invalida")
62     }
63 }
64 5 -> consultarEmpleados()
65 6 -> println("adios")
66 else -> println("opcion invalida")
67 }
68 } while (opcion != 6)
69 }
70
71 fun agregarEmpleado(noEmp: Int, nombre: String, puesto: String, sueldo: Double) { 1Usage new *
72     val existe = lista.any { it.noEmp == noEmp }
73     if (!existe) {
74         lista.add(Empleados(noEmp, nombre, puesto, sueldo))
75         println("empleado agregado")
76     } else {
77         println("el empleado ya existe")
78     }
79 }
```

EstructuraDatosAplicadas > src > Empleados.kt 6:1 CRLF UTF-8 4 spaces

```
ED EstructuraDatosAplicadas master
Current File

Array.kt For.kt ExamenDos.kt Empleados.kt x
80
81 fun buscarEmpleado(noEmp: Int) { 1Usage new *
82     val empleado = lista.find { it.noEmp == noEmp }
83     if (empleado != null) {
84         empleado.muestra()
85     } else {
86         println("el empleado con ese numero no existe")
87     }
88 }
89
90 fun modificarEmpleado(noEmp: Int) { 1Usage new *
91     val empleado = lista.find { it.noEmp == noEmp }
92     if (empleado != null) {
93         println("empleado encontrado ingresa los nuevos datos: ")
94         print("nuevo nombre: ")
95         empleado.nombre = readln()
96         print("nuevo puesto: ")
97         empleado.puesto = readln()
98         print("nuevo sueldo: ")
99         empleado.sueldo = readln().toDoubleOrNull() ?: empleado.sueldo
100         println("empleado modificado")
101     } else {
102         println("el empleado con ese numero no existe")
103     }
104 }
105
106 fun eliminarEmpleado(noEmp: Int) { 1Usage new *
```

EstructuraDatosAplicadas > src > Empleados.kt 6:1 CRLF UTF-8 4 spaces

ED EstructuraDatosAplicadas master

Current File

Array.kt For.kt ExamenDos.kt Empleados.kt

90 fun modificarEmpleado(noEmp: Int) { 1 Usage new \*  
101 } else {  
102 println("el empleado con ese numero no existe")  
103 }  
104 }  
105  
106 fun eliminarEmpleado(noEmp: Int) { 1 Usage new \*  
107 val eliminado = lista.removeIf { it.noEmp == noEmp }  
108 if (eliminado) {  
109 println("empleado eliminado")  
110 } else {  
111 println("no se encontro el empleado")  
112 }  
113 }  
114  
115 fun consultarEmpleados() { 1 Usage new \*  
116 if (lista.isEmpty()) {  
117 println("lista vacia")  
118 } else {  
119 lista.forEach { it.muestra() }  
120 }  
121 }

4 84

EstructuraDatosAplicadas > src > Empleados.kt 6:1 CRLF UTF-8 4 spaces

ED EstructuraDatosAplicadas master

Current File

Array.kt For.kt ExamenDos.kt Empleados.kt

Run EmpleadosKt

C:\Users\luffy\.jdk\openjdk-24.0.1\bin\java.exe "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2025.1.1.1\lib\idea\_rt.jar" -Dfile.encoding=UTF-8  
/// MENU DE EMPLEADOS ///  
1) agregar empleado  
2) buscar empleado  
3) modificar empleado  
4) eliminar empleado  
5) consultar todos los empleados  
6) salir  
elige una opcion: 1  
ingresa el numero de empleado: 1  
dame el nombre: Erick  
dame el puesto: SubGerente  
dame el sueldo: 8000  
empleado agregado  
/// MENU DE EMPLEADOS ///  
1) agregar empleado  
2) buscar empleado  
3) modificar empleado  
4) eliminar empleado  
5) consultar todos los empleados  
6) salir  
elige una opcion: 1  
ingresa el numero de empleado: 2  
dame el nombre: Marcos

EstructuraDatosAplicadas > src > Empleados.kt > main 18:15 CRLF UTF-8 4 spaces

ED EstructuraDatosAplicadas master

Current File

Array.kt For.kt ExamenDos.kt Empleados.kt

Run EmpleadosKt

```
dame el puesto: Gerente
dame el sueldo: 15000
empleado agregado
/// MENU DE EMPLEADOS ///
1) agregar empleado
2) buscar empleado
3) modificar empleado
4) eliminar empleado
5) consultar todos los empleados
6) salir
elige una opcion: 1
ingresa el numero de empleado: 3
dame el nombre: silvia
dame el puesto: Auxiliar
dame el sueldo: 3000
empleado agregado
/// MENU DE EMPLEADOS ///
1) agregar empleado
2) buscar empleado
3) modificar empleado
4) eliminar empleado
5) consultar todos los empleados
6) salir
elige una opcion: 2
```

EstructuraDatosAplicadas > src > Empleados.kt > main 18:15 CRLF UTF-8 4 spaces

ED EstructuraDatosAplicadas master

Current File

Array.kt For.kt ExamenDos.kt Empleados.kt

Run EmpleadosKt

```
ingresa el numero de empleado: 2
Numero: 2, Nombre: Marcos, Puesto: Gerente, Sueldo: 15000.0
/// MENU DE EMPLEADOS ///
1) agregar empleado
2) buscar empleado
3) modificar empleado
4) eliminar empleado
5) consultar todos los empleados
6) salir
elige una opcion: 3
ingresa el numero de empleado a modificar: 3
empleado encontrado ingresa los nuevos datos:
ruevo nombre: Silvia
ruevo puesto: Ayudante General
ruevo sueldo: 4000
empleado modificado
/// MENU DE EMPLEADOS ///
1) agregar empleado
2) buscar empleado
3) modificar empleado
4) eliminar empleado
5) consultar todos los empleados
6) salir
elige una opcion: 5
```

EstructuraDatosAplicadas > src > Empleados.kt > main 18:15 CRLF UTF-8 4 spaces

```
ED EstructuraDatosAplicadas master
Current File
Array.kt For.kt ExamenDos.kt Empleados.kt
Run EmpleadosKt
elige una opcion: 5
Numero: 1, Nombre: Erick, Puesto: SubGerente, Sueldo: 8000.0
Numero: 2, Nombre: Marcos, Puesto: Gerente, Sueldo: 15000.0
Numero: 3, Nombre: Silvia, Puesto: Ayudante General, Sueldo: 4000.0
/// MENU DE EMPLEADOS ///
1) agregar empleado
2) buscar empleado
3) modificar empleado
4) eliminar empleado
5) consultar todos los empleados
6) salir
elige una opcion: 4
ingresa el numero de empleado a eliminar: 2
empleado eliminado
/// MENU DE EMPLEADOS ///
1) agregar empleado
2) buscar empleado
3) modificar empleado
4) eliminar empleado
5) consultar todos los empleados
6) salir
elige una opcion: 5
Numero: 1, Nombre: Erick, Puesto: SubGerente, Sueldo: 8000.0
Numero: 3, Nombre: Silvia, Puesto: Ayudante General, Sueldo: 4000.0
EstructuraDatosAplicadas > src > Empleados.kt > main 18:15 CRLF UTF-8 4 spaces
```

```
ED EstructuraDatosAplicadas master
Current File
Array.kt For.kt ExamenDos.kt Empleados.kt
Run EmpleadosKt
empleado eliminado
/// MENU DE EMPLEADOS ///
1) agregar empleado
2) buscar empleado
3) modificar empleado
4) eliminar empleado
5) consultar todos los empleados
6) salir
elige una opcion: 5
Numero: 1, Nombre: Erick, Puesto: SubGerente, Sueldo: 8000.0
Numero: 3, Nombre: Silvia, Puesto: Ayudante General, Sueldo: 4000.0
/// MENU DE EMPLEADOS ///
1) agregar empleado
2) buscar empleado
3) modificar empleado
4) eliminar empleado
5) consultar todos los empleados
6) salir
elige una opcion: 6
adios
Process finished with exit code 0
EstructuraDatosAplicadas > src > Empleados.kt > main 18:15 CRLF UTF-8 4 spaces
```