

Tecnologías de la Información y Comunicación

Área: Desarrollo de Software

Programación orientada a objetos

Milton Batres

Reporte de práctica:

3.5.2 SH Programar las excepciones generadas a través de las cláusulas apropiadas en el manejo de errores en tiempo de ejecución.

TID22M

Alumno(s):

Carlos Eduardo Reyes Quintero

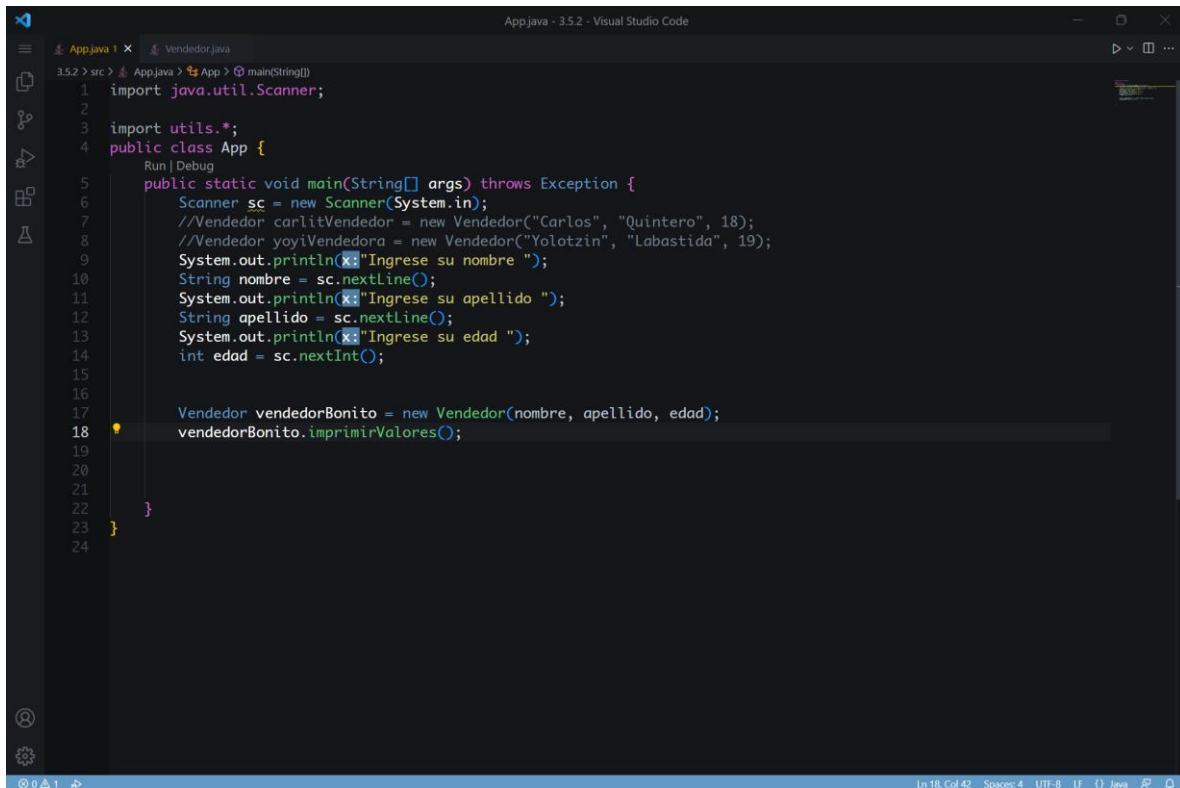
Rene Alejandro Ramos Castro

Chihuahua, Chih., Mex.

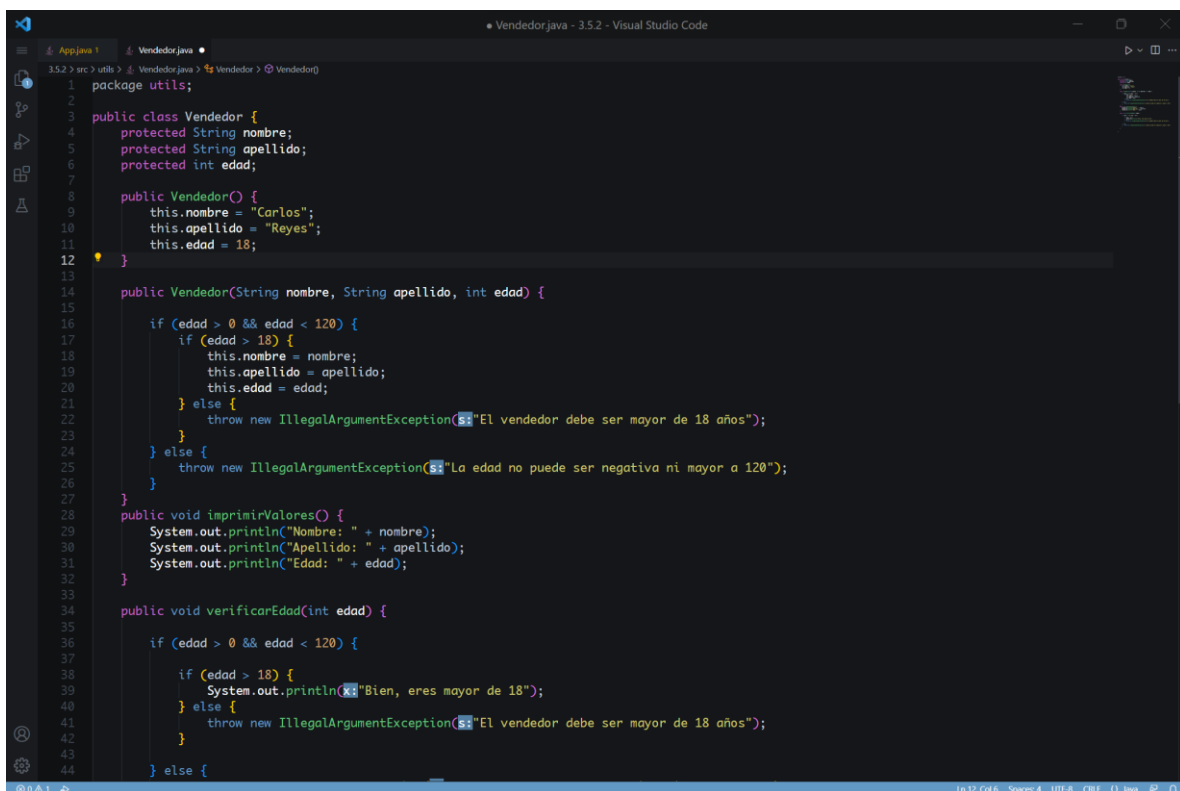
Fecha de realización de la práctica: 19/03/2023

Fecha de entrega el reporte: 20/03/2024

REPORTE DE PRÁCTICA

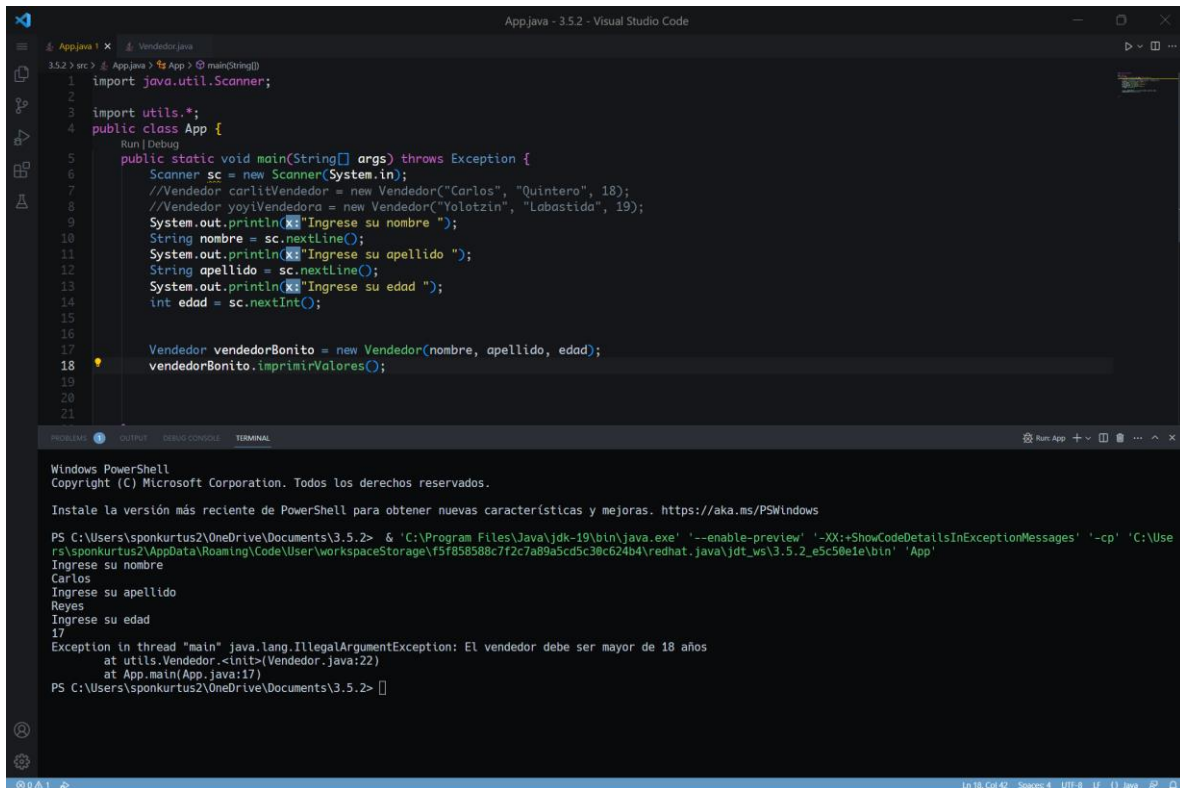


```
App.java 1 x Vendedor.java
3.5.2 > src > App.java > App > main(String[])
1 import java.util.Scanner;
2
3 import utils.*;
4 public class App {
5     Run | Debug
6     public static void main(String[] args) throws Exception {
7         Scanner sc = new Scanner(System.in);
8         //Vendedor carlitVendedor = new Vendedor("Carlos", "Quintero", 18);
9         //Vendedor yoyiVendedora = new Vendedor("Yolotzin", "Labastida", 19);
10        System.out.println(x;"Ingrese su nombre ");
11        String nombre = sc.nextLine();
12        System.out.println(x;"Ingrese su apellido ");
13        String apellido = sc.nextLine();
14        System.out.println(x;"Ingrese su edad ");
15        int edad = sc.nextInt();
16
17        Vendedor vendedorBonito = new Vendedor(nombre, apellido, edad);
18        vendedorBonito.imprimirValores();
19
20
21    }
22
23 }
24
```



```
Vendedor.java • 3.5.2 - Visual Studio Code
3.5.2 > src > utils > Vendedor.java > Vendedor > Vendedor()
1 package utils;
2
3 public class Vendedor {
4     protected String nombre;
5     protected String apellido;
6     protected int edad;
7
8     public Vendedor() {
9         this.nombre = "Carlos";
10        this.apellido = "Reyes";
11        this.edad = 18;
12    }
13
14    public Vendedor(String nombre, String apellido, int edad) {
15
16        if (edad > 0 && edad < 120) {
17            if (edad > 18) {
18                this.nombre = nombre;
19                this.apellido = apellido;
20                this.edad = edad;
21            } else {
22                throw new IllegalArgumentException(x;"El vendedor debe ser mayor de 18 años");
23            }
24        } else {
25            throw new IllegalArgumentException(x;"La edad no puede ser negativa ni mayor a 120");
26        }
27    }
28
29    public void imprimirValores() {
30        System.out.println("Nombre: " + nombre);
31        System.out.println("Apellido: " + apellido);
32        System.out.println("Edad: " + edad);
33    }
34
35    public void verificarEdad(int edad) {
36
37        if (edad > 0 && edad < 120) {
38
39            if (edad > 18) {
40                System.out.println(x;"Bien, eres mayor de 18");
41            } else {
42                throw new IllegalArgumentException(x;"El vendedor debe ser mayor de 18 años");
43            }
44        } else {
45
46        }
47    }
48 }
```

REPORTE DE PRÁCTICA



The screenshot shows a Visual Studio Code editor with a Java file named `Vendedor.java`. The code defines a `Vendedor` class with a constructor that takes `nombre`, `apellido`, and `edad` as parameters. It also has a `main` method that uses a `Scanner` to input these values. The `main` method creates a `Vendedor` object and calls `imprimirValores()`. The terminal window shows the execution of the program, where the user inputs `Carlos`, `Reyes`, and `17`. An exception is thrown because the age is less than 18.

```
App.java - 3.5.2 - Visual Studio Code
1  import java.util.Scanner;
2
3  import utils.*;
4  public class App {
5      public static void main(String[] args) throws Exception {
6          Scanner sc = new Scanner(System.in);
7          //Vendedor carlitVendedor = new Vendedor("Carlos", "Quintero", 18);
8          //Vendedor yoyiVendedora = new Vendedor("Yoletzin", "Labastida", 19);
9          System.out.println("Ingrese su nombre ");
10         String nombre = sc.nextLine();
11         System.out.println("Ingrese su apellido ");
12         String apellido = sc.nextLine();
13         System.out.println("Ingrese su edad ");
14         int edad = sc.nextInt();
15
16         Vendedor vendedorBonito = new Vendedor(nombre, apellido, edad);
17         vendedorBonito.imprimirValores();
18     }
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
```

```
Windows PowerShell
Copyright (C) Microsoft Corporation. Todos los derechos reservados.

Instale la versión más reciente de PowerShell para obtener nuevas características y mejoras. https://aka.ms/PSWindows

PS C:\Users\sponkurtus2\OneDrive\Documents\3.5.2> & 'C:\Program Files\Java\jdk-19\bin\java.exe' '--enable-preview' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\sponkurtus2\AppData\Roaming\Code\User\workspaceStorage\15f858588c7f2c7a89a5cd5c30c624b4\redhat.java\jdk_ws\3.5.2_esc50e1e\bin' 'App'
Ingrese su nombre
Carlos
Ingrese su apellido
Reyes
Ingrese su edad
17
Exception in thread "main" java.lang.IllegalArgumentException: El vendedor debe ser mayor de 18 años
    at utils.Vendedor.<init>(Vendedor.java:22)
    at App.main(App.java:17)
PS C:\Users\sponkurtus2\OneDrive\Documents\3.5.2>
```

La función de este código es crear un objeto `Vendedor` solo si ciertas condiciones se cumplen, y estas condiciones son la edad, dentro de la clase `vendedor` tenemos en el constructor una condicional, que checa la edad, si la edad es correcta y mayor a 18, podemos crear un vendedor, de no ser así, el programa terminara automáticamente dándonos la excepción de que el vendedor debe ser si o si mayor a 18 años o que la edad debe estar correcta.

Contenido

Objetivo:	5
Material:	5
Desarrollo:	5
Conclusiones:	5
Bibliografía:	5

Objetivo:

El objetivo de esta practica es conocer mas acerca de como funcionan las excepciones en Java.

Material:

Vscode.

Desarrollo:

Primero leí la practica para ver que tenia que hacer, luego leí un poco acerca de como funcionan las excepciones para poder hacer el código, después lo programe, hice el documento y listo.

Conclusiones:

Las excepciones son algo que no solemos tomar tanto en cuenta, pero que sin duda pueden ser útiles y deberíamos usarlas.

Bibliografía: