



Creación, Compilación y Ejecución de Código Java



MÓDULO: TALLER DE APLICACIONES MÓVILES

SEMANA: 03

Docente: Iván Ayala Ayala

Estudiante: Alvaro Moyano – Danay Parra

Índice

<i>Introducción</i>	<i>3</i>
<i>Código.....</i>	<i>4</i>
<i>Directorio</i>	<i>7</i>
<i>Compilación</i>	<i>8</i>
<i>Ejecución del Código</i>	<i>9</i>
<i>Historias de Usuario.....</i>	<i>10</i>
<i>Bibliografía</i>	<i>12</i>

Introducción

El desarrollo de este proyecto surge como una forma práctica de aplicar los conocimientos de programación en Java a un caso cotidiano, como lo es un servicio de delivery. La idea principal es crear un programa que permita registrar información básica de un vehículo, un cliente y su compra, para luego calcular el costo del envío en función del monto y la distancia recorrida. Este ejercicio no solo busca comprobar el correcto uso de estructuras como variables, entradas por teclado y condicionales, sino también afianzar la lógica necesaria para resolver problemas reales a través de la programación. A lo largo del proceso se incluye, además, la documentación correspondiente, con el fin de organizar de mejor manera el proyecto y simular un entorno de trabajo más profesional.

Código

Se realiza código.

```
Delivery: Bloc de notas
Archivo Edición Formato Ver Ayuda
// Importamos la clase Scanner desde la librería java.util
import java.util.Scanner;

// Declaración de la clase principal llamada "Delivery"
public class Delivery {
    // Método principal donde inicia la ejecución del programa
    public static void main(String[] args) {
        // Se crea un objeto Scanner para leer datos desde el teclado
        Scanner scanner = new Scanner(System.in);

        // =====
        // Registro del Vehículo Repartidor
        // =====
        System.out.println("=== Registro del Vehículo Repartidor ===");

        // Se solicita al usuario ingresar la marca del vehículo
        System.out.print("Ingrese Marca: ");
        String marca = scanner.nextLine(); // Se guarda el dato en la variable "marca"

        // Se solicita al usuario ingresar el modelo del vehículo
        System.out.print("Ingrese Modelo: ");
        String modelo = scanner.nextLine(); // Se guarda en la variable "modelo"

        // Se solicita la cilindrada
        System.out.print("Ingrese Cilindrada: ");
        String cilindrada = scanner.nextLine(); // Se guarda en la variable "cilindrada"

        // Se solicita el tipo de combustible
        System.out.print("Ingrese Tipo de combustible: ");
        String combustible = scanner.nextLine(); // Se guarda en "combustible"

        // Se solicita capacidad en pasajeros
        System.out.print("Ingrese Capacidad en pasajeros: ");
        int pasajeros = scanner.nextInt(); // Se guarda en "pasajeros"
        scanner.nextLine(); // Limpia el buffer para evitar errores al leer cadenas después de presionar Enter
    }
}
```

- Se crea la Clase y se solicitan al Usuario datos de entrada del Vehículo.

```
// =====
// Registro del Cliente
// =====
System.out.println("\n=== Registro del Cliente ===");

// Se pide nombre del cliente
System.out.print("Ingrese Nombre: ");
String nombre = scanner.nextLine(); // Se guarda en "nombre"

// Se pide edad del cliente
System.out.print("Ingrese Edad: ");
int edad = scanner.nextInt(); // Se guarda en "edad"
```

- Se le solicita información del Cliente al Usuario.

```

// =====
// Datos de la Compra
// =====
System.out.println("\n=== Datos de la Compra ===");

// Se pide el monto de la compra
System.out.print("Ingrese Monto de la compra: ");
int montoCompra = scanner.nextInt(); // Se guarda en "montoCompra"

// Se pide los kilómetros de distancia
System.out.print("Ingrese Kilómetros de distancia: ");
int km = scanner.nextInt(); // Se guarda en "km"

// =====
// Cálculo del costo del delivery
// =====
int costoDelivery = 0; // Se inicializa en 0

// Si la compra es menor a $25.000 el costo es km * 300
if (montoCompra < 25000) {
    costoDelivery = km * 300;
}
// Si la compra está entre $25.000 y $49.999 el costo es km * 150
else if (montoCompra >= 25000 && montoCompra <= 49999) {
    costoDelivery = km * 150;
}
// Si la compra es mayor o igual a $50.000 el delivery es gratis
else if (montoCompra >= 50000) {
    costoDelivery = 0;
}

// Se calcula el total a pagar (compra + delivery)
int totalPagar = montoCompra + costoDelivery;

```

- Se le solicita información de la Compra al Usuario.
- Se realiza el cálculo del cobro total del servicio

```

// =====
// Mostrar resumen del pedido
// =====
System.out.println("\n=== RESUMEN DEL PEDIDO ===");
System.out.println("Cliente: " + nombre + " | Edad: " + edad);
System.out.println("Vehículo: " + marca + " " + modelo +
    " | Cilindrada: " + cilindrada +
    " | Combustible: " + combustible +
    " | Capacidad: " + pasajeros + " pasajeros");
System.out.println("Monto de la compra: $" + montoCompra);
System.out.println("Kilómetros recorridos: " + km + " km");
System.out.println("Costo por delivery: $" + costoDelivery);
System.out.println("TOTAL A PAGAR: $" + totalPagar);
    }
}
, gra

```

- Se muestran datos de salida ingresados por el Usuario utilizando System.out

Directorio

Se muestra directorio donde podemos visualizar el Delivery.java

```
C:\> Símbolo del sistema
Microsoft Windows [Versión 10.0.19045.6216]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\alvar>d:

D:\>dir
El volumen de la unidad D es Respaldo
El número de serie del volumen es: 7092-414C

Directorio de D:\

08-07-2025  17:57    <DIR>          2024
08-07-2025  17:17    156.591.982  2024.rar
08-07-2025  19:23    <DIR>          2025
31-08-2025  14:42    2.500 Delivery.java
07-07-2025  20:50    <DIR>          Descargas
20-08-2025  20:30    <DIR>          Programas
25-08-2025  20:18    <DIR>          Programas Instalados
07-07-2025  23:34    2.537.611.776 Ubuntu_CyberEss.ova
                3 archivos  2.694.206.258 bytes
                5 dirs   306.562.207.744 bytes libres

D:\>
```

Compilación

Mediante el código: `javac Delivery.java` se compila el archivo `Delivery.class`

```
C:\> Símbolo del sistema

08-07-2025  19:23    <DIR>          2025
01-09-2025  21:17             2.497 Delivery.java
07-07-2025  20:50    <DIR>          Descargas
20-08-2025  20:30    <DIR>          Programas
25-08-2025  20:18    <DIR>          Programas Instalados
07-07-2025  23:34    2.537.611.776 Ubuntu_CyberEss.ova
                3 archivos  2.694.206.255 bytes
                5 dirs  306.562.207.744 bytes libres

D:\>javac Delivery.java

D:\>dir
El volumen de la unidad D es Respaldo
El número de serie del volumen es: 7092-414C

Directorio de D:\

08-07-2025  17:57    <DIR>          2024
08-07-2025  17:17    156.591.982 2024.rar
08-07-2025  19:23    <DIR>          2025
01-09-2025  21:18    2.419 Delivery.class
01-09-2025  21:17    2.497 Delivery.java
07-07-2025  20:50    <DIR>          Descargas
20-08-2025  20:30    <DIR>          Programas
25-08-2025  20:18    <DIR>          Programas Instalados
07-07-2025  23:34    2.537.611.776 Ubuntu_CyberEss.ova
                4 archivos  2.694.208.674 bytes
                5 dirs  306.562.203.648 bytes libres

D:\>
```


Ejecución del Código

```
ca\ Símbolo del sistema
25-08-2025 20:18 <DIR> Programas Instalados
07-07-2025 23:34 2.537.611.776 Ubuntu_CyberEss.ova
                4 archivos 2.694.208.674 bytes
                5 dirs 306.562.203.648 bytes libres

D:\>java Delivery
=== Registro del Vehículo Repartidor ===
Ingrese Marca: Mazda
Ingrese Modelo: 3
Ingrese Cilindrada: 2.0
Ingrese Tipo de combustible: Gasolina
Ingrese Capacidad en pasajeros: 5

=== Registro del Cliente ===
Ingrese Nombre: Ivan
Ingrese Edad: 30

=== Datos de la Compra ===
Ingrese Monto de la compra: 9000
Ingrese Kilómetros de distancia: 10

=== RESUMEN DEL PEDIDO ===
Cliente: Ivan | Edad: 30
Vehículo: Mazda 3 | Cilindrada: 2.0 | Combustible: Gasolina | Capacidad: 5 pasajeros
Monto de la compra: $9000
Kilómetros recorridos: 10 km
Costo por delivery: $3000
TOTAL A PAGAR: $12000

D:\>
```

Mediante el código: java Delivery se ejecuta el código creado.

Historias de Usuario

Historia 1

¿Quién?: Cliente,

¿Qué quiere?: ingresar su nombre y edad,

¿Para qué?: que el sistema me identifique correctamente en el pedido.

- Criterio de aceptación: El sistema debe solicitar y mostrar nombre y edad en el resumen del pedido.

Historia 2

¿Quién?: cliente,

¿Qué quiere?: que el costo del delivery dependa del monto de su compra,

¿Para qué?: pagar un precio justo por el servicio.

- Criterio de aceptación:
 - Si la compra es $< \$25.000 \rightarrow$ el delivery se calcula como $\text{km} * 300$.
 - Si la compra está entre $\$25.000$ y $\$49.999 \rightarrow$ el delivery se calcula como $\text{km} * 150$.
 - Si la compra es $\geq \$50.000 \rightarrow$ el delivery es gratis.

Historia 3

¿Quién?: empresa,

¿Qué quiere?: registrar el vehículo repartidor,

¿Para qué?: llevar control del transporte utilizado en las entregas.

- Criterio de aceptación: El sistema debe solicitar y mostrar la información del vehículo en el resumen.

Conclusión

A partir del trabajo realizado, recordamos el funcionamiento de java mediante la obtención de datos de entrada y posterior salida para la correcta ejecución de un programa. también, consideramos que documentar el proceso y detallarlo proporciona claridad a la hora de que otro programador pueda utilizar nuestro código de manera más fácil de entender.

Utilizar declaraciones de variables como, por ejemplo, String, fueron de mucha ayuda para poder almacenar la información que se pidió con los comandos Scanner(System.in) y System.out,

Bibliografía

AIEP. (2023). Taller de Aplicaciones Mviles. *Semana 3. Material de apoyo.*