

Flutter & Dart Complete Guide

Prepared by Eng. Yousef Salah Kassem

DAFL 1
IDE & Problem Solving

Lecture 3



Chapter

03

Table Of Content

- Introduction
- Installation
- User Inputs
- Problem Solving Questions



Introduction

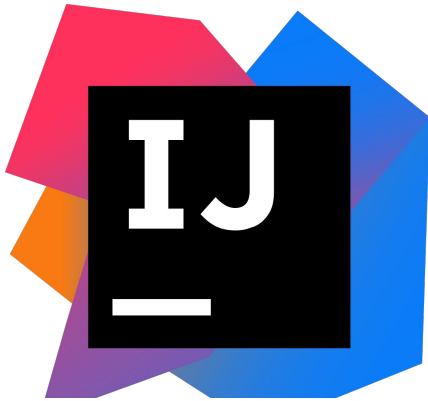
IDE

- ◆ **An IDE, or Integrated Development Environment, enables programmers to consolidate the different aspects of writing a computer program.**
- ◆ IDEs increase programmer productivity by combining common activities of writing software into a single application: editing source code, building executables, and debugging.

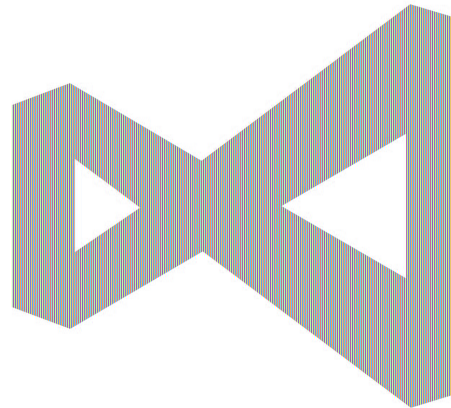


Chapter
03

Types Of IDE



intellij



VsCode



**Android
Studio**



Chapter
03

Visual Studio Code

Visual Studio Code, also commonly referred to as VS Code,^[9] is an integrated development environment developed by [Microsoft](#) for [Windows](#), [Linux](#), macOS and [web browsers](#).^{[10][11]} Features include support for [debugging](#), [syntax highlighting](#), [intelligent code completion](#), [snippets](#), [code refactoring](#), and embedded version control with Git. Users can change the theme, keyboard shortcuts, preferences, and install extensions that add functionality.

Installation

Installation Link

- Dart SDK: <https://dart.dev/get-dart>
- VS Code: <https://code.visualstudio.com/download>
- Android Studio:
https://developer.android.com/studio?gad_source=1&gbraid=0AAAAAC-I OZnzhEPXDunSnP1yL5hJs_wIW&gclid=CjwKCAjwpbi4BhByEiwAMC8JnbLt0qTQzFaZZaICxlg9CY4y5Gc_3gyGe_p_oCkb5d6CuyVGShlq0RoCNxsQAvD_BwE&gclsrc=aw.ds

Videos to Install

<https://www.youtube.com/watch?v=IxU3MjRunWI&t=201s>

User Inputs

Standard Input in Dart

In Dart programming language, you can take standard input from the user through the console via the use of **.readLineSync()** function. To take input from the console you need to import a library, named **dart:io** from libraries of Dart.

```
// importing dart:io file
import 'dart:io';

void main()
{
    print("Enter your name?");
    // Reading name of the Geek
    String? name = stdin.readLineSync(); // null safety in name string

    // Printing the name
    print("Hello, $name! \nWelcome to GeeksforGeeks!!");
}
```

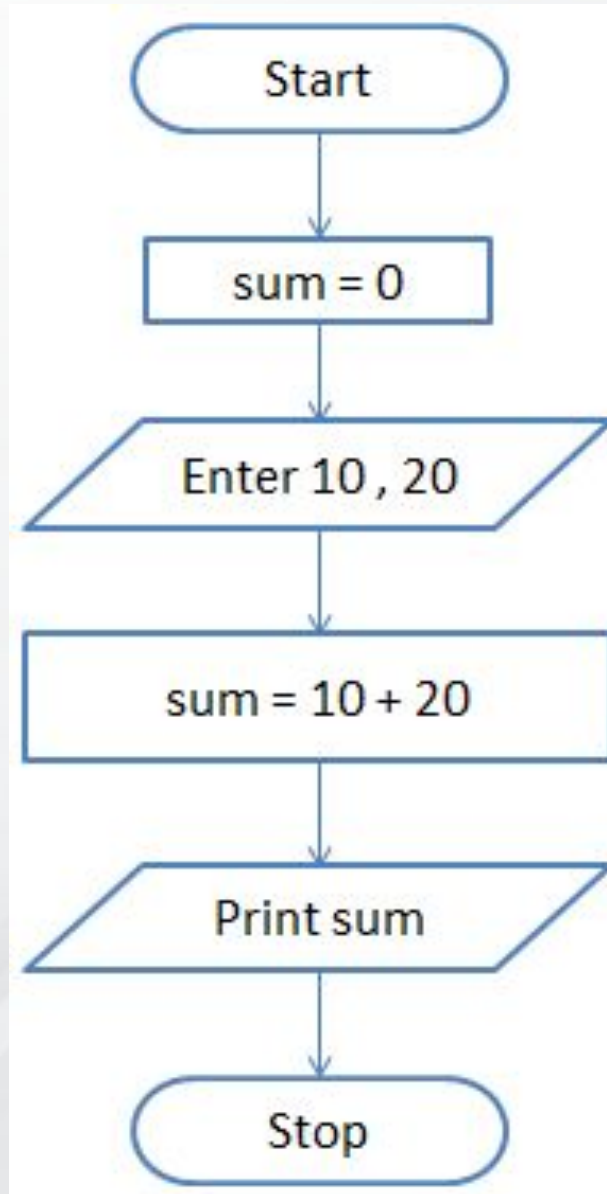
```
import 'dart:io';

void main()
{
    print("-----GeeksForGeeks-----");
    print("Enter first number");
    int? n1 = int.parse(stdin.readLineSync()!);

    print("Enter second number");
    int? n2 = int.parse(stdin.readLineSync()!);

    // Adding them and printing them
    int sum = n1 + n2;
    print("Sum is $sum");
}
```

Problem Solving



Chapter
03

Example 1

Alex, Sam, and Jamie are organizing a lemonade sale for charity. They want to track their sales and revenue with a Dart program. Here's how they solve the problem using Dart:

1. **Greeting Message:** Start by greeting themselves with "Welcome to the Lemonade Sale!".
2. **Track Sales:** Set the price of a lemonade glass to \$2.0 and track how many glasses are sold.
3. **String Interpolation:** Display how many glasses were sold and the total revenue.
4. **Temperature Conversion:** Convert the temperature of the day (25°C) from Celsius to Fahrenheit to predict good selling weather.
5. **Discount Rule:** Apply a discount of 10% if they sell more than 50 glasses.
6. **Sales Loop:** Simulate lemonade sales by updating the count every hour until they sell 100 glasses.
7. **Customer Freebie:** Every 5th customer gets a free glass.

Chapter
03**Example 1****Breakdown:**

1. **Greeting Message:** The program starts by printing a welcome message.
2. **Track Sales:** Variables like `pricePerGlass` and `glassesSold` are initialized and updated as sales occur.
3. **String Interpolation:** Sales and revenue are printed dynamically using `$` for interpolation.
4. **Temperature Conversion:** A simple formula converts 25°C to Fahrenheit and prints it.
5. **Discount Application:** The program checks if more than 50 glasses are sold, and applies a 10% discount if true.
6. **Sales Loop:** A `while` loop simulates selling 10 glasses every hour until they reach 100 glasses.
7. **Free Lemonade for Every 5th Customer:** A `for` loop checks every 5th customer and prints a message.

Chapter
03

Problem 2: Concert Ticket Sales Tracker

Alex, Sam, and Jamie are organizing a concert and need a Dart program to help them track ticket sales. They want to:

1. **Set Ticket Prices:** Declare the price per ticket and set an initial number of sold tickets.
2. **Calculate Revenue:** Calculate the total revenue and display it using string interpolation.
3. **Track Age Group:** Depending on the customer's age, they want to print a message offering discounts to seniors (60+) and children (under 12).
4. **Loop through Ticket Sales:** Use a loop to simulate ticket sales until they reach the target of 200 tickets.
5. **Ternary Operator for Late Fee:** Add a late fee if the total tickets are sold after a certain hour, using a ternary operator.

Chapter
03

Problem 3: Smart Home Water Usage Monitor

Alex, Sam, and Jamie are creating a smart home system to monitor water usage and alert them when their water consumption exceeds a certain limit. Here's how they use Dart to solve the problem:

1. **Set Water Usage Limit:** Declare a water usage limit in liters and initialize the current usage.
2. **Monitor Usage:** For each hour, the system checks if their water usage exceeds the limit.
3. **String Concatenation for Alerts:** Use string concatenation to alert the user when water usage exceeds 100 liters.
4. **Upper/Lowercase Alert:** Convert alert messages to uppercase or lowercase, depending on severity.
5. **Track Water Usage:** Use a `for` loop to simulate water usage being tracked for 24 hours.
6. **Discount for Conserving Water:** If they used less than 200 liters in a day, offer a 5% discount on the water bill using an `if/else` statement.

Problem 4: Fitness Tracker Monitoring

1. Tracks the number of steps taken each day for a week.
2. Calculates total steps, average steps, and whether the user met their goal.
3. Converts steps into kilometers and uses loops to find the day with the most steps.

What is Next ?

<https://zapp.run/>



Thank You