1. void main() {

Print(“Hello, Dart! Welcome to Programming.”);

}

1. void main() {

Int a = 10, b = 5;

Print(“Sum: ${a + b}”);

Print(“Difference: ${a – b}”);

Print(“Product: ${a \* b}”);

Print(“Quotient: ${a / b}”);

}

1. void main() {

Int num = 11;

If (num % 2 == 0) {

Print(“The number is Even.”);

} else {

Print(“The number is Odd.”);

}

}

1. int add(int a, int b) => a + b;

Int subtract(int a, int b) => a – b;

Int multiply(int a, int b) => a \* b;

Double divide(int a, int b) => a / b;

Void main() {

Int a = 8, b = 4;

Print(“Addition: ${add(a, b)}”);

Print(“Subtraction: ${subtract(a, b)}”);

Print(“Multiplication: ${multiply(a, b)}”);

Print(“Division: ${divide(a, b)}”); }

1. void main() {

Int n = 10;

Int a = 0, b = 1;

Print(“Fibonacci Series:”);

For (int I = 0; I < n; i++) {

Print(a);

Int temp = a + b;

A = b;

B = temp;

}

6 ) void main() {

List<int> numbers = [5, 3, 8, 1, 2];

Numbers.add(7);

Numbers.sort();

Print(“Sorted List: $numbers”);

}

7) void main() {

Set<int> numbers = {1, 2, 3, 4};

Numbers.add(5);

Numbers.remove(2);

Print(“Final Set: $numbers”);

}

8) void main() {

Map<String, dynamic> person = {

‘name’: ‘Alice’,

‘age’: 25,

‘city’: ‘New York’

};

Person[‘country’] = ‘USA’;

Person[‘age’] = 26;

Print(“Updated Map: $person”);

}

9) void main() {

List<int> numbers = [5, 2, 9, 1, 5, 6];

For (int I = 0; I < numbers.length – 1; i++) {

For (int j = 0; j < numbers.length – I – 1; j++) {

If (numbers[j] > numbers[j + 1]) {

Int temp = numbers[j];

Numbers[j] = numbers[j + 1];

Numbers[j + 1] = temp;

}

}

}

Print(“Sorted List: $numbers”);

}

10) int binarySearch(List<int> list, int target) {

Int left = 0, right = list.length – 1;

While (left <= right) {

Int mid = (left + right) ~/ 2;

If (list[mid] == target) return mid;

If (list[mid] < target) {

Left = mid + 1;

} else {

Right = mid – 1;

}

}

Return -1;

}

Void main() {

List<int> sortedList = [1, 3, 5, 7, 9, 11];

Int target = 7;

Int result = binarySearch(sortedList, target);

If (result != -1) {

Print(“Number found at index: $result”);

} else {

Print(“Number not found.”);

}

}

11) class Car {

String brand, model;

Int year;

Car(this.brand, this.model, this.year);

Void displayInfo() {

Print(“Car Details:”);

Print(“Brand: $brand”);

Print(“Model: $model”);

Print(“Year: $year”);

}

}

Void main() {

Car car = Car(“Toyota”, “Corolla”, 2020);

Car.displayInfo();

12) class Animal {

Void makeSound() {

Print(“Animal makes a sound”);

}

}

Class Dog extends Animal {

@override

Void makeSound() {

Print(“Dog barks”);

}

}

Void main() {

Animal animal = Animal();

Animal.makeSound();

Dog dog = Dog();

Dog.makeSound();

}

13) void greet(String name, [String message = “Welcome”]) {

Print(“Hello $name, $message”);

}

Void main() {

Greet(“Alice”);

Greet(“Bob”, “Good Morning”);

}

14) void main() {

List<int> numbers = [1, 2, 3, 4, 5];

Numbers.forEach((number) {

Print(“Square of $number: ${number \* number}”);

});

}

15) void main() {

String input = “madam”;

String reversed = input.split(‘’).reversed.join(‘’);

If (input == reversed) {

Print(“The string ‘$input’ is a palindrome.”);

} else {

Print(“The string ‘$input’ is not a palindrome.”);

}

}