

NAME : MAKVANA NEEL

ROLL NO : CE064

ID : 20CEUOS086

BATCH : A4

Lab 02 code and comment :

Github link:

<https://github.com/NK16082002/SDP-LABS/tree/main/LAB2>

```
import 'package:lab2_tutorial2/lab2_tutorial2.dart' as lab2_tutorial2;

void main(List<String> arguments) {
  /*
   // Topic : Comments

   // Below Statement Is Single Line Comment
   // This Is A Comment. It Is Not Executed

   // below is a example of how we can use single line comment for multi-line
comment
   // This is also a comment
   // over multiple lines

   // Below is an example of comment block
```

```

/* This is multiline comment */

// Below is A Example of nest comment
/* this is a comment
/* and inside there is another
comment */
back */

// below is a example of documentation comment
/// I am a documentation comment
/// hi
/**
 * hi
 */

// Below statement print Hello,Dart Apprentice reader!"
// print("Hello,Dart Apprentice reader!");
*/

/*

// Topic : Statement and expressions

// Below statement print Hello,Dart Apprentice reader!"
// print("Hello,Dart Apprentice reader!");

// Example Of If Statement
int a=5;
if(a>0){
    print("yes!!");
}

// Example of Expression:
// print 42
// print(42);
// print 5
// print(3 + 2);
// print the statement
// print("Hello,Dart Apprentice reader!");
// below statement give us error of undefined name
// print(x);

*/

// Topic : Arithmetic Expressions:

// simple operations:

```

```

// print 2+6=8
// print(2 + 6);
// print 10-2=8
// print(10 - 2);
// print 2*4=8
// print(2 * 4);
// print 24/3=8.0
// print(24 / 3);

// Decimal Numbers:

// print 3.142857142857143
// print(22 / 7);

// if we want integer division
// print 3
// print(22 ~/ 7);

// Euclidean modulo operation
// it print 8 because remainder is 8
// print(28 % 10);

// order of operations:

// it print 32
// because it will first calculate the expression which are in () (parenthesis)
the it evalute the value from left to right so firt 8000/50 is evaluate and its
subtract 32 then he compute 29%5 and then perform ~/ between them And Give Result
32
// print(((8000 / (5 * 10)) - 32) ~/ (29 % 5));

// first it will calculate 350/5 because / has higher precedence then + so it
will give us 72
// print(350 / 5 + 2);

// parenthesis has higher precedence compare to / so it will first perform 5+2
and then divide from 350
// print(350 / (5 + 2));

// Topic : Math Functions:

// We Have To Import dart:math library
// These convert an angle from degrees to radians, and then compute the sine and
cosine respectively.
// it will take pi value as 3.14 and perform the expression and give it's sin
value
// print(sin(45 * pi / 180));
// it will take pi value as 3.14 and perform the expression and give it's cos
value
// print(cos(135 * pi / 180));
// it will give us a value of square root of 2

```

```

// print(sqrt(2));

// These compute the maximum of two numbers respectively.
// print(max(5, 10));
// These compute the minimum of two numbers respectively.
// print(max(-5, -10));
// it will first compute sqrt(2) and pi/2 and then give us max of this two
numbers
// print(max(sqrt(2), pi / 2));

// print the value of 1 over the square root of 2 in Dart.
// It Will Give Us 1 As Answer.
// print(sin(pi / 2));

// Topic : Naming data

// Identifiers can include both, characters and digits. However, the identifier
cannot begin with a digit.
// Identifiers cannot include special symbols except for underscore (_) or a dollar
sign ($).
// They must be unique.
// Identifiers cannot be keywords.
// Identifiers are case-sensitive.
// Identifiers cannot contain spaces.

// Variables :

// This Below statement declares a variable called number of type int and set the
value is 10.
// The int part of the statement is known as a type annotation.
// int number = 10;

// If you want to change the value of a variable, then you can just give it a
different value of the same type:
// int number = 10;
// number = 15;

// This Below statement declares a variable called apple of type double and set the
value is 3.14159.
// double apple = 3.14159;

// This Below statement print true
// print(10.isEven);

// This Below statement print 3.
// print(3.14159.round());

// Topic : Type safety

// int myInteger = 10;
// myInteger = 3.14159; // No, no, no. That's not allowed.

```

// The num type can be either an int or a double, However, the string value 'ten' is of a different type, so the compiler complains.

```
// num myNumber;  
// myNumber = 10;      //ok  
// myNumber = 3.14159; //ok  
// myNumber = "t";     //No, no , no
```

// This lets you assign any data type you like to your variable, and the compiler won't warn you about anything.

```
// dynamic myNumber = 10;      //ok  
// myNumber = 3.14159;        //ok  
// myNumber = "t";            //ok
```

// Type inference :

// There's no need to tell Dart that 10 is an integer. Dart the type and makes someNumber an int.

```
// var someNumber = 10;  
// someNumber = 15; // OK  
// someNumber = 3.14159; // No, no, no.
```

// Constants :

// constant variable is immutable , we can't change it's value after assigning it.

```
// const myConstant = 10;  
// myConstant = 0; // Not allowed.
```

// final constants :

```
// Here is another example of a runtime value:  
// final hoursSinceMidnight = DateTime.now().hour;
```

// If you try to change the final constant after it's already been set, This will produce error.

```
// final hoursSinceMidnight = DateTime.now().hour;  
// hoursSinceMidnight = 0;
```

// Mini-exercises :

// 1. Declare a constant of type int called myAge and set it to your age.

```
// Ans1 :  
// const myAge = 19;
```

// 2. Declare a variable of type double called averageAge.

```
// Initially, set the variable to your own age. Then, set it to  
// the average of your age and your best friend's age.
```

```
// Ans2 :
```

```

// double averageAge = 19;
// averageAge = 20;

// 3. Create a constant called testNumber and initialize it
// with whatever integer you'd like. Next, create another
// constant called evenOdd and set it equal to testNumber
// modulo 2. Now change testNumber to various numbers.
// What do you notice about evenOdd?
// Ans3 :
// const testNumber = 1025;
// const evenOdd = testNumber % 2;

// Increment and decrement :

// var counter = 0;
// counter += 1;
// counter = 1;
// counter -= 1;
// counter = 0;

// In other words, the code above is shorthand for the following:

// var counter = 0;
// counter = counter + 1;
// counter = counter - 1;

// If you only need to increment or decrement by 1, then you can use the ++ or --
operators:

// var counter = 0;
// counter++; // 1
// counter--; // 0

// This given Below given operation is perform similar operations for
multiplication and division, respectively:
// double myValue = 10;

// same as myValue = myValue * 3; // myValue = 30.0;
// myValue *= 3;
// same as myValue = myValue / 2; // myValue = 15.0;
// myValue /= 2;

// Chanllanges :

// Challenge 1: Variables
// Declare a constant int called myAge and set it equal to your
// age. Also declare an int variable called dogs and set that
// equal to the number of dogs you own. Then imagine you
// bought a new puppy and increment the dogs variable by one.
// Ans1 :
// const myAge = 19;

```

```
// int dogs = 0;
// dogs++;
// print(dogs);

// Challenge 2: Make it compile
// Modify the first line so that the code compiles. Did you use var?
// Ans2 :
// var age = 16;
// print(age); //print : 16
// age = 30;
// print(age); //print : 30

// Challenge 3: Compute the answer

// Consider the following code:
// const x = 46;
// const y = 10;

// Work out what each answer equals when you add the
// following lines of code to the code above:

// const answer1 = (x * 100) + y;
// const answer2 = (x * 100) + (y * 100);
// const answer3 = (x * 100) + (y / 10);

// Ans3 :
// print(answer1); // 4610
// print(answer2); // 5600
// print(answer3); // 4601.0

// Challenge 4: Average rating
// Ans4 :
// const rating1 = 3.4;
// const rating2 = 4.2;
// const rating3 = 4.5;
// const averageRating = (rating1 + rating2 + rating3) / 3;
// print(averageRating); // 4.033333333333333
}
```