NAME: HUZAIFA SUFYAN

ANSWER #1

OPERATORS IN DART

**1. Arithmetic Operators:**

This class of operators contains those operators used to perform arithmetic operations on the operands. They are binary operators i.e they act on two operands.

ADDITION(+): Use to add two operands.

SUBTRACTION(-): Use to subtract two operands

UNARY MINUS (-expr): It is used to reverse the sign of the expression.

MULTIPLY(\*): Use to multiply two operands.

DIVISION(/): Use to divide two operands.

DIVISION(~/): Use to divide two operands but give output in integer.

MODULUS(%): Use to give the remainder of two operands.

Example: Using Arithmetic Operators in the program :

void main()

{

int a = 2;

int b = 3;

// Adding a and b

var c = a + b;

print("Sum of a and b is $c");

// Subtracting a and b

var d = a - b;

print("The difference between a and b is $d");

// Using unary minus

var e = -d;

print("The negation of difference between a and b is $e");

// Multiplication of a and b

var f = a \* b;

print("The product of a and b is $f");

// Division of a and b

var g = b / a;

print("The quotient of a and b is $g");

// Using ~/ to divide a and b

var h = b ~ / a;

print("The quotient of a and b is $h");

// Remainder of a and b

var i = b % a;

print("The remainder of a and b is $i");

}

**2. RELATIONAL OPERATORS**

This class of operators contains those operators which are used to perform relational operations on the operands.

**Greater than(>)**: check which operand is bigger and give the result as a boolean expression.

**LESS THAN(<):** check which operand is smaller and give the result as a boolean expression.

**GREATER THAN AND EQUAL TO( >=)**: Check which operand is greater or equal to each other and give the result as a boolean expression

**LESS THAN EQUAL TO(<=):** Check which operand is less than or equal to each other and give the result as a boolean expression.

**EQUAL TO( = ):** Check whether the operand is equal to each other or not and give the result as a boolean expression.

**NOT EQUAL TO ( !=):** Check whether the operand is not equal to each other or not and give the result as a boolean expression.

**EXAMPLE USING RELATIONAL OPERATORS**

void main()

{

int a = 2;

int b = 3;

// Greater between a and b

var c = a > b;

print("a is greater than b is $c");

// Smaller between a and b

var d = a < b;

print("a is smaller than b is $d");

// Greater than or equal to between a and b

var e = a >= b;

print("a is greater than b is $e");

// Less than or equal to between a and b

var f = a <= b;

print("a is smaller than b is $f");

// Equality between a and b

var g = b == a;

print("a and b are equal is $g");

// Unequality between a and b

var h = b != a;

print("a and b are not equal is $h");

**}**

**3. Assignment Operators:**

This class of operators contains those operators used to assign value to the operands.

**EQUAL TO (=):** Use to assign values to the expression or variable

**ASSIGNMENT OPERATOR (??):** Assign the value only if it is null.

**Example: Using Assignment Operators in the program**

void main()

{

int a = 5;

int b = 7;

// Assigning value to variable c

var c = a \* b;

print(c);

// Assigning value to variable d

var d;

d ? ? = a + b; // Value is assign as it is null

print(d);

// Again trying to assign value to d

d ? ? = a - b; // Value is not assign as it is not null

print(d);

}

**Logical Operators:**

This class of operators contains those operators used to logically combine two or more conditions of the operands.

**AND OPERATOR (&&):** Use to add two conditions and if both are true than it will return true.

**OR OPERATOR (||):** Use to add two conditions and if even one of them is true than it will return true.

**NOT OPERATOR (!):** It is use to reverse the result.

**EXAMPLE OF LOGICAL OPERATORS.**

void main()

{

int a = 5;

int b = 7;

// Using And Operator

bool c = a > 10 && b < 10;

print(c);

// Using Or Operator

bool d = a > 10 || b < 10;

print(d);

// Using Not Operator

bool e = !(a > 10);

print(e);

}

**Conditional Operators:**

This class of operators contain those operators which are used to perform comparison on the operands.

**Conditional Operator(?):** It is a simple version of the if-else statement. If the condition is true than expersion1 is executed else expersion2 is executed

**Conditional Operator(??):**If expersion1 is non-null returns its value else returns expression2 value.

**EXAMPLE OF CONDITIONAL OPERATORS.**

void main()

{

int a = 5;

int b = 7;

// Conditional Statement

var c = (a < 10) ? "Statement is Correct, Geek" : "Statement is Wrong, Geek";

print(c);

// Conditional statement

int n;

var d = n ? ? "n has Null value";

print(d);

// After assigning value to n

n = 10;

d = n ? ? "n has Null value";

print(d);

}

**ANSWER #2:**

// Ans #2

void main() {

int ticket = 600;

int num\_of\_tickets = 5;

int a = ticket\* num\_of\_tickets ;

print("5 tickets price is $a");

}

**ANSWER #3:**

void main() {

var array1 = <int> [1,2,3,4,5,6,7];

var array2 = <int> [3,5,6,7,9,10];

print(array1.difference(array2));

}

**ANSWER #4 :**

? and ?? both are conditional operaters but we use ? operator when the variable is initialized with the value and ?? when the variable is not initialized with a value.

**ANSWER #5:**

1: **NUMBER:** Number is the data type that is used to hold the numeric data.

Keyword: int, double, num

Example:

Int num1 = 5;

Int num2= 6;

Print(num1);

Print(num2);

2: **STRING:** It is used to represent a sequence of characters.

Keyword: string.

Example:

String string1 = 'Babar';

String string2 = 'Azam ';

Print(string1 + string2);

3: **BOOLEANS:** It is represent Boolean values true and false

Keyword : bool.

Example:

String string1 = 'Babar';

String string2 = 'Azam ';

bool a = (string1 == string2);

print(a);

4: **LIST**: It is used to represent a collection of objects.

Keyword: List.

Example:

List a = new List(3);

a[0] = 'Babar';

a[1] = 'Rizwan';

a[2] = 'Afridi';

print(a);

5:**MAP:** It represents a set of values as key-value pairs.

Keyword: Map.

Example:

List a = new Map();

a[‘first’] = 'Babar';

a[‘second’] = 'Rizwan';

a[‘third’] = 'Afridi';

print(a);

**ANSWER #6 :**

void main() {

// a)

var array=[7,14,21,28,35,42,49,56,63,70];

// b)

var array\_1=[1,2,3,4,5,6,7,8,9,10];

// c)

for(int i=1;i>=10;i++){

print('7 \* $i = ${7\*i}');

}

}

**ANSWER #7:**

import 'dart:io';

void main() {

String pass = "zefi123";

String pass1 = "zefi321";

print("Enter your pass");

String? p = stdin.readLineSync();

if(p==null){

print("Please enter your pass!!!");

}

else if(p==pass || p==pass1){

print("Please enter your correct pass");

}

else{

print("Please enter your incorrect pass");

}

}

**ANSWER #8:**

void main() {

var st\_name = ["shahan","huzefa", "muneef"];

var st\_marks = [410, 416, 428];

int tot\_marks = 500;

print("Student name: ${st\_name[0]}\nStudent score is: ${st\_marks[0]}\nPercentage is: ${st\_marks[0]\*100/tot\_marks}\n");

print("Student name: ${st\_name[1]}\nStudent score is: ${st\_marks[1]}\nPercentage is: ${st\_marks[1]\*100/tot\_marks}\n");

print("Student name: ${st\_name[2]}\nStudent score is: ${st\_marks[2]}\nPercentage is: ${st\_marks[2]\*100/tot\_marks}");

}

**ANSWER #9 :**

**FIVE LEGAL VARIABLE NAMES.**

1: customer\_name

2: std\_name

3: current\_month

4: ticket

5: bill\_amount

**FIVE ILLEGAL VARIABLE NAMES.**

1: #customer\_name

2: 4std\_name

3: current month

4: 0\_ticket

5: @bill\_amount

**ANSWER #10 :**

void main() {

String a = 'Hyderabad';

String b = a.replaceAll('Hyder', 'Islam');

print(b);

}

**ANSWER #11 :**

void main() {

String customer\_name = 'Sir Ali Mughal';

String current\_month = '10/03/2022';

int no\_of\_units = 1800;

double charges\_per\_unit = 50.50;

double net\_amount = no\_of\_units \* charges\_per\_unit;

int late\_pay\_surcharge = 5000;

double gross\_amount = net\_amount + late\_pay\_surcharge ;

// Due date bill

print("Due date 15/3/22\nCustomer name: $customer\_name\nCurrentmonth:$current\_month\nNumber of Units: $no\_of\_units\nCharges per Unit: $charges\_per\_unit\nLate fee Charges: $late\_pay\_surcharge\nTotal amount: $net\_amount\n\n");

// After date bill

print("After date 16/3/22\nCustomer name: $customer\_name\nCurrentmonth:$current\_month\nNumber of Units: $no\_of\_units\nCharges per Unit: $charges\_per\_unit\nLate fee Charges: $late\_pay\_surcharge\nTotal amount(After date): $gross\_amount");

}