Code Sample 1 - function and return I

1) function a(){

2) console.log('hello');

3) }

4) console.log('Dojo');

RESULT:

Dojo

// function not called

=====================================

Code Sample 2 - function and return I

1) function a(){

2) console.log('hello');

3) return 15;

4) }

5) x = a();

6) console.log('x is', x);

RESULT:

hello

x is 15

=====================================

Code Sample 3 - function and return I

1) function a(n){

2) console.log('n is', n);

3) return n+15;

4) }

5) x = a(3);

6) console.log('x is', x);

RESULT:

n is 3

x is 18

// return value n(3) + 15 = 18

=====================================

Code Sample 4 - function and return I

1) function a(n){

2) console.log('n is', n);

3) y = n\*2;

4) return y;

5) }

6) x = a(3) + a(5);

7) console.log('x is', x);

RESULT:

n is 3

n is 5

x is 16

=====================================

Code Sample 5 - order of operations I

1) function op(a,b){

2) c = a+b;

3) console.log('c is', c);

4) return c;

5) }

6) x = op(2,3) + op(3,5);

7) console.log('x is', x);

RESULT:

c is 5

c is 8

x is 13

// x = 5 + 8

// x = 13

=====================================

Code Sample 6 - order of operations II

1) function op(a,b){

2) c = a+b;

3) console.log('c is', c);

4) return c;

5) }

6) x = op(2,3) + op(3,op(2,1)) + op(op(2,1),op(2,3));

7) console.log('x is', x);

RESULT:

c is 5

c is 3

c is 6

c is 3

c is 5

c is 8

x is 19

// x = op(5) + op(3, op(3)) + op(op(3), op(5))

// x = 5 + op(6) + op(8)

// x = 5 + 6 + 8

// x = 19

=====================================

Code Sample 7 - scoping

1) var x = 15;

2) function a(){

3) var x = 10;

4) }

5) console.log(x);

6) a();

7) console.log(x);

RESULT:

15

15

// function called but there's no passing/taking value happenDiagram

Description automatically generated