**Number 1**

The following will be printed out

First of all, hoisting takes place for all variable declarations.

**Undefined** – Function c gets invoked when variable x has been hoisted but not initialized hence its value is undefined

**8** – The argument value passed to the function for parameter a is 8. Since the printing is in the local scope of the function, 8 gets printed out.

**8** – In the inner function, the value for the local variable a in the inner function is assigned to local variable b and hence the value of b in the inner function is 8.

**9** – Once we are out of the inner function scope (i.e. function f), we go back to the outer function scope (i.e. function c) where the value of b is 9. This is the value that was passed as an argument to the function c.

**10** – The value of b which is 10 is determined basing on the global scope and not in the function since we are out of scope on this function.

**1** – Since x is declared in the global scope, once hoisting is complete, at the time of execution, it will have a value of 1 in global scope.

**Number 2**

Global scope defines variables, objects and functions that can be accessed from anywhere in an application. In a browser, global variables are only deleted when the browser window is closed.

Local scope defines variables, objects and functions that can only be accessed within the scope where they are defined. Local variables defined in a function are deleted immediately when the function completes.

**Number 3**

A - NO (B and C are defined in the local scope);

B - YES (A is defined in global scope);

C - NO (C is defined in a local scope);

D - YES (A is a global variable);

E - YES (Possible due to closure);

**Number 4**

81 – In the first call of myFunction(), the value of x will be determined via lexical scoping and thus its value will be set to 9 which is defined in the global scope. This will give a result of 81.

25 – Before the second call of myFunction(), the value for the global variable x is set to 5. Thus the function will give a result of 25.

**Number 5**

10 – When function bar() is invoked, the variable foo is undefined in this functions scope since its only been hoisted and not initialized. When execution reaches the if statement i.e. (!foo) evaluates to true and it is inside the if statement that the value of foo is set to 10. The alert will display a value of 10.