1)
1,8,8,9,10,1

2)
Every line of code is defined inside the global scope unless its inside a function, as every function has its own local scope.

3)
no, yes, no, yes,yes

4)
81, 25

5)
10

6)

```
var counter = (function () {
var privateCounter = 0;
return {
add: function () {
privateCounter++;
},
reset: function () {
privateCounter = 0;
},
value: function () {
return privateCounter;
},
};
})();
```

7)
privateCounter is the free variable.
For a closure function, its the variable that’s not one of its parameters or local variables.

8)

```
var counter = (function () {
var privateCounter = 0;
function make_addr(inc) {
return function () {
privateCounter = privateCounter + inc;
};
}
return {
add: function () {
privateCounter++;
},
reset: function () {
privateCounter = 0;
},
add5: make_addr(5),
add7: make_addr(7),
value: function () {
return privateCounter;
},
};
})();
```

9)
we can put all of these functions and variables in a module or an IIFE.

10)

```
var Employee = (function () {
var name = "";
var age = 0;
var salary = 0;
function getAge() {
return age;
}
function getSalary() {
return salary;
}
function getName() {
return name;
}
function setAge(newAge) {
age = newAge;
}
function setSalary(newSalary) {
salary = newSalary;
}
function setName(newName) {
name = newName;
}
function increaseSalary(percentage) {
salary = getSalary + getSalary * (percentage / 100);
}
function incrementAge() {
age = getAge++;
}

return {
setAge,
setSalary,
setName,
increaseSalary,
incrementAge,
};
})();
```

11)

```
var Employee = (function () {
var name = "";
var age = 0;
var salary = 0;
function getAge() {
return age;
}
function getSalary() {
return salary;
}
function getName() {
return name;
}
return {
setAge: function (newAge) {
age = newAge;
},
setSalary: function (newSalary) {
salary = newSalary;
},
setName: function (newName) {
name = newName;
},
increaseSalary: function (percentage) {
salary = getSalary + getSalary * (percentage / 100);
},
incrementAge: function () {
age = getAge++;
},
};
})();
```

12)

```
var Employee = (function () {
var name = "";
var age = 0;
var salary = 0;
let myObject = {};
myObject.getAge = function () {
return age;
};
myObject.getSalary = function () {
return salary;
};
myObject.getName = function () {
return name;
};
myObject.setAge = function (newAge) {
age = newAge;
};
myObject.setSalary = function (newSalary) {
salary = newSalary;
};
myObject.setName = function (newName) {
name = newName;
};
myObject.increaseSalary = function (percentage) {
salary = getSalary + getSalary * (percentage / 100);
};
myObject.incrementAge = function () {
age = getAge++;
};

return myObject;
})();
```

13)

```
Employee.extention = {
address: "",
setAddress: function (newAddress) {
this.address = newAddress;
},
getAddress: function () {
return this.address;
},
};
```

14)
“Error: Hattori”

15)
“Success: Hattori”

16)
“success”
“error”