Syntax Errors:

a) What will be the output of the following function? If there is a problem in it, explain the solution.

```
function add() public pure returns (uint) {
    uint a = 7;
    uint b = 9;
    return a - b;
}
```

Answer:

we are returning the difference not the sum.

If we don't want to change the body of the function we will put int int as the return type so we can return negative values

function add() public pure returns (uint) {

uint a = 7;

uint b = 9;

return a + b;

}

b) What will be the output of the following function? If there is a problem in it, explain the solution.

```
function divide() public pure returns (uint) {
    uint a = 19;
    uint b = 9;
    return a / b;
}
```

Answer:

Output will be 2

There is no problem with it.

c) Debug the following function and explain how it can be fixed.

```
function transfer(address receiver, uint numberOfTokens) public returns (bool) {
    require(numberOfTokens <= balances[receiver]);
    balances[msg.sender] = balances[msg.sender] - numberOfTokens;
    balances[receiver] = balances[receiver] + numberOfTokens;
    return true;
}
```

Answer:

- Original function checks if the number of tokens are less than or equal to the receiver's current tokens/balance, if sender send tokens that are greater than its existing tokens the function will go through the require statement which should not happen.

Fix:

- We check if sender's tokens are less than their current tokens

```
function transfer(address receiver, uint numberofTokens) public returns (bool) {
    require(numberofTokens <= balances[msg.sender]);
    balances[msg.sender] = balances[msg.sender] – numberOfTokens;
    balances[receiver] = balances[receiver] + numberofTokens;
    return true;
}
```

d) How many times can this function be called?

```
uint private papersChecked = 1;
function check() private {
    require(papersChecked < 10);
    papersChecked++;
}
```

Internally, it can be called 9 times, externally, it cannot be called

e) Explain the error in the following function. How can it be fixed?

```
string _totalSupply = 0;
function mint(address account, uint256 amount) onlyOwner public {
    require(account != address(0));
    _totalSupply += amount;
}
```

Answer:

- _totalSupply should be a uint not a string
- require(account == msg.sender(0)) to access account from 0^{th} index

Fix:

```
uint _totalSupply = 0;
  function mint(address account, uint256 amount) public {
    require(account == msg.sender(0));
    _totalSupply += amount;
  }
```