

Blockchain Ethereum Solidity

Módulo 06 – Desafío

¿Qué son los desafíos?

- Son **ejercicios complementarios** a los realizados en clase.
- Permiten **practicar y consolidar los conocimientos** adquiridos.
- Es recomendable que los realices antes de rendir el examen de la clase.



Ejercicio 1

Completar el siguiente caso de test en Solidity:

```
pragma solidity >=0.4.25 <0.6.0;
import "truffle/Assert.sol";
import "truffle/DeployedAddresses.sol";
import "../contracts/Token.sol";

... TestToken {
  ... testInitialBalanceUsingDeployedContract() {
    Token token = ...(DeployedAddresses.Token());

    uint expected = 10000;

    Assert.equal(token.getBalance(...), expected, "Owner should have 10000 tokens initially");
  }
}
```

...

...

```
function testInitialBalanceWithNewMetaCoin() {  
    MetaCoin meta = new MetaCoin();  
  
    uint expected = 10000;  
  
    Assert.equal(meta.getBalance(tx.origin), expected, "Owner should have 10000 MetaCoin  
initially");  
}  
}
```

Ejercicio 2

Completar el siguiente caso de test en Javascript:

```
const Token = artifacts.require("Token");

...("Token", accounts => {
  it("should put 10000 tokens in the first account", () =>
    ...deployed()
      .then(instance => instance....call(accounts[0]))
      .then(balance => {
        assert....(
          balance.valueOf(),
          10000,
          "10000 wasn't in the first account"
        );
      }));
});
```

¡Terminaste el módulo!
Estás listo para rendir el examen