Name of Student : Pushkar Sane		
Roll Number : 45		**LAB Assignment Number:** 8
Title of LAB Assignment : Execution of smart contract using truffle framework.		
DOP : 15-08-2024		**DOS :** 16-09-2024
CO Mapped : CO5	**PO Mapped:** PO1,PO2, PO3, PO4, PO7, PO9, PSO1, PSO2	**Signature:**

PRACTICAL 8

Aim: Execution of smart contract using truffle framework.

Theory:

Truffle framework:

- Truffle is a world-class development environment, testing framework and asset pipeline for blockchains using the Ethereum Virtual Machine (EVM), aiming to make life as a developer easier.
- Truffle is widely considered the most popular tool for blockchain application development with over 1.5 million lifetime downloads. Truffle supports developers across the full lifecycle of their projects, whether they are looking to build on Ethereum, Hyperledger, Quorum, or one of an ever-growing list of other supported platforms.
- Paired with Ganache, a personal blockchain, and Drizzle, a front-end dApp development kit, the full Truffle suite of tools promises to be an end-to-end dApp development platform.

  1. Built-in smart contract compilation, linking, deployment and binary management.

  Automated contract testing for rapid development.

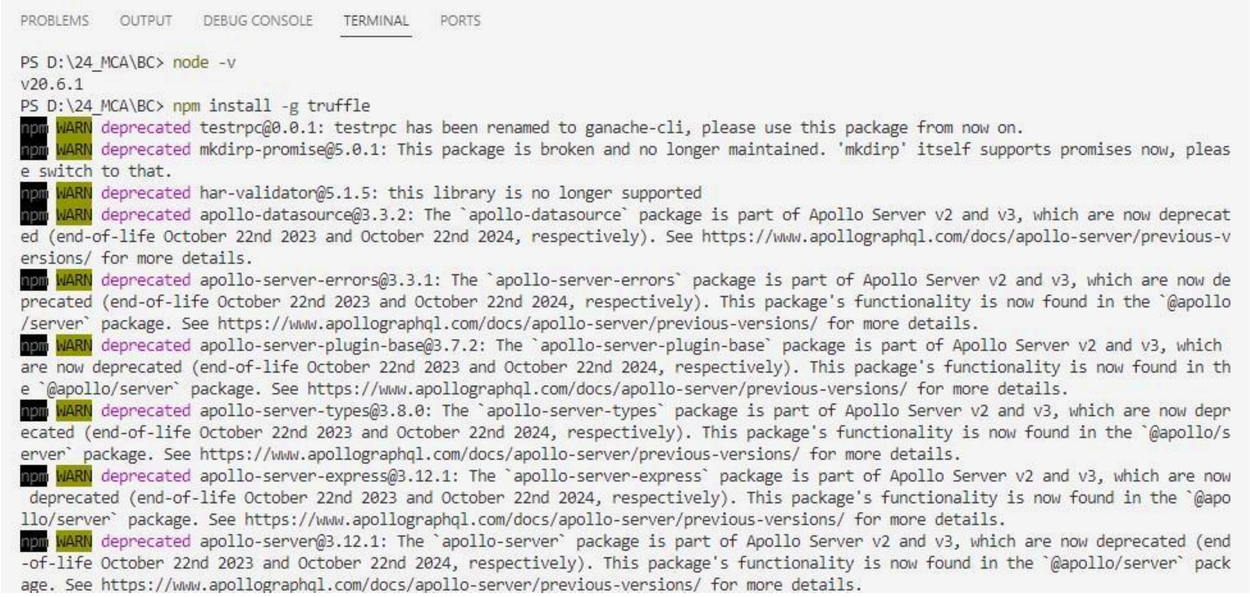
  2. Scriptable, extensible deployment & migrations framework.
  3. Network management for deploying to any number of public & private networks.
  4. Package management with EthPM & NPM, using the ERC190 standard.
  5. Interactive console for direct contract communication.
  6. Configurable build pipeline with support for tight integration.
  7. External script runner that executes scripts within a Truffle environment

Smart Contract:

A smart contract is a stand-alone script usually written in Solidity and compiled into binary or JSON and deployed to a specific address on the blockchain. In the same way that we can call a specific URL endpoint of a RESTful API to execute some logic through an HttpRequest, we can similarly execute the deployed smart contract at a specific address by submitting the correct data along with the necessary Ethereum to call the deployed and compiled Solidity function.

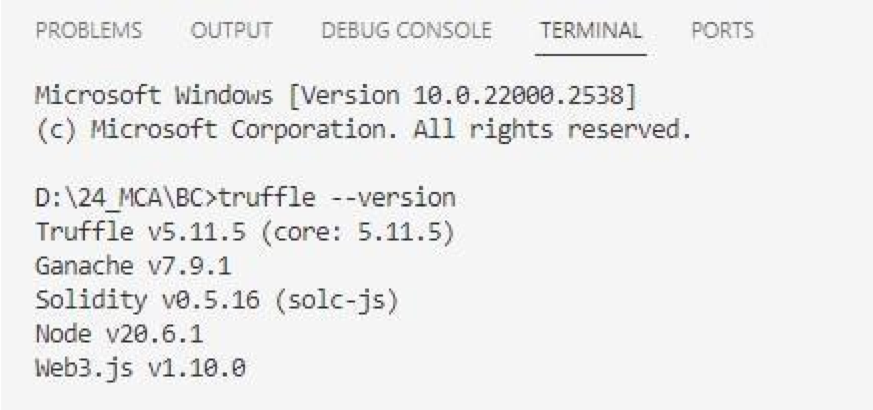
Installation Steps:

- **npm install -g truffle**



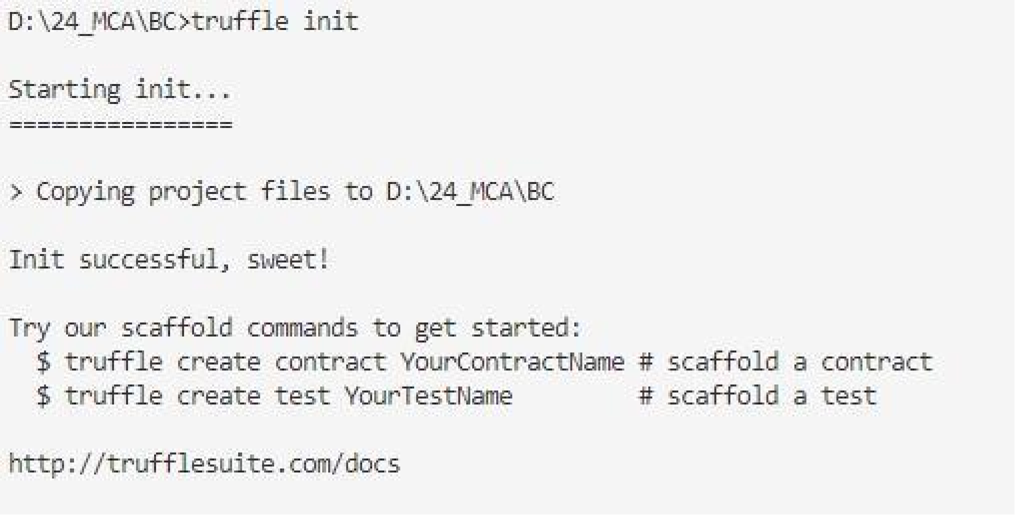
```
PS D:\24_MCA\BC> node -v
v20.6.1
PS D:\24_MCA\BC> npm install -g truffle
npm WARN deprecated testrpc@0.0.1: testrpc has been renamed to ganache-cli, please use this package from now on.
npm WARN deprecated mkdirp-promise@5.0.1: This package is broken and no longer maintained. 'mkdirp' itself supports promises now, pleas
e switch to that.
npm WARN deprecated har-validator@5.1.5: this library is no longer supported
npm WARN deprecated apollo-datasource@3.3.2: The `apollo-datasource` package is part of Apollo Server v2 and v3, which are now deprecat
ed (end-of-life October 22nd 2023 and October 22nd 2024, respectively). See https://www.apollographql.com/docs/apollo-server/previous-v
ersions/ for more details.
npm WARN deprecated apollo-server-errors@3.3.1: The `apollo-server-errors` package is part of Apollo Server v2 and v3, which are now de
precated (end-of-life October 22nd 2023 and October 22nd 2024, respectively). This package's functionality is now found in the `@apollo
/server` package. See https://www.apollographql.com/docs/apollo-server/previous-versions/ for more details.
npm WARN deprecated apollo-server-plugin-base@3.7.2: The `apollo-server-plugin-base` package is part of Apollo Server v2 and v3, which
are now deprecated (end-of-life October 22nd 2023 and October 22nd 2024, respectively). This package's functionality is now found in th
e `@apollo/server` package. See https://www.apollographql.com/docs/apollo-server/previous-versions/ for more details.
npm WARN deprecated apollo-server-types@3.8.0: The `apollo-server-types` package is part of Apollo Server v2 and v3, which are now depr
ecated (end-of-life October 22nd 2023 and October 22nd 2024, respectively). This package's functionality is now found in the `@apollo/s
erver` package. See https://www.apollographql.com/docs/apollo-server/previous-versions/ for more details.
npm WARN deprecated apollo-server-express@3.12.1: The `apollo-server-express` package is part of Apollo Server v2 and v3, which are now
 deprecated (end-of-life October 22nd 2023 and October 22nd 2024, respectively). This package's functionality is now found in the `@apo
llo/server` package. See https://www.apollographql.com/docs/apollo-server/previous-versions/ for more details.
npm WARN deprecated apollo-server@3.12.1: The `apollo-server` package is part of Apollo Server v2 and v3, which are now deprecated (end
-of-life October 22nd 2023 and October 22nd 2024, respectively). This package's functionality is now found in the `@apollo/server` pack
age. See https://www.apollographql.com/docs/apollo-server/previous-versions/ for more details.
```

- **truffle --version**



```
Microsoft Windows [Version 10.0.22000.2538]
(c) Microsoft Corporation. All rights reserved.

D:\24_MCA\BC>truffle --version
Truffle v5.11.5 (core: 5.11.5)
Ganache v7.9.1
Solidity v0.5.16 (solc-js)
Node v20.6.1
Web3.js v1.10.0
```

- **truffle init**

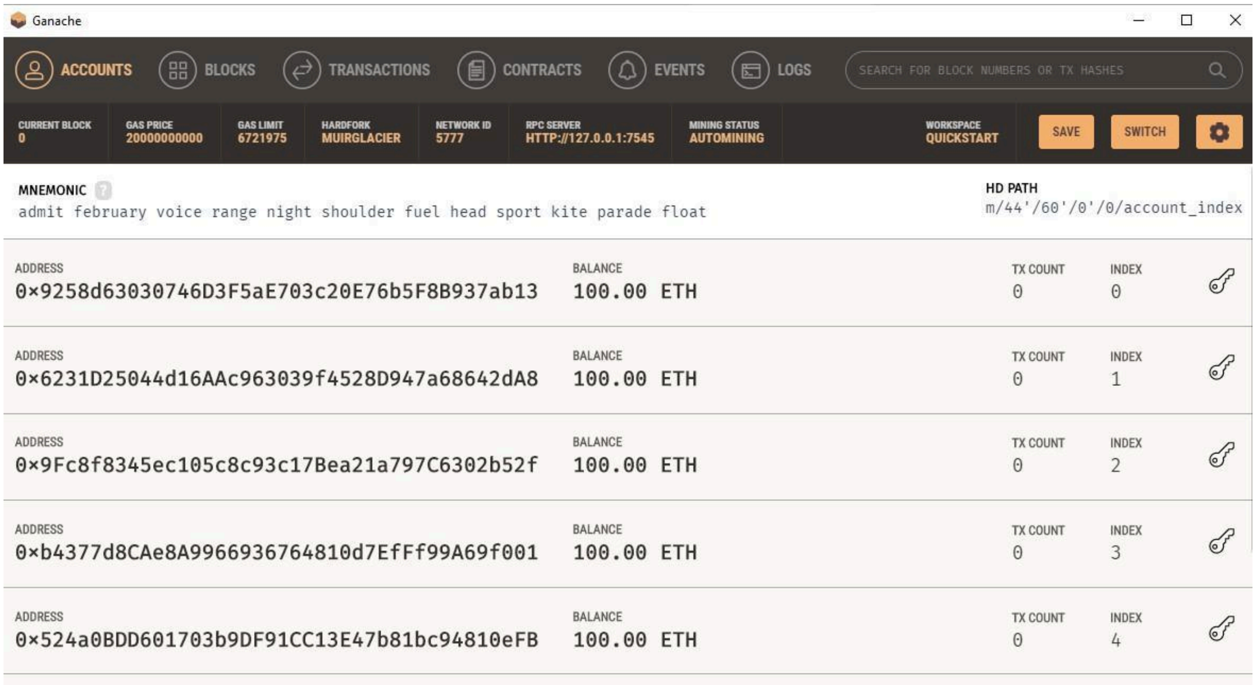


```
D:\24_MCA\BC>truffle init

Starting init...
================

> Copying project files to D:\24_MCA\BC

Init successful, sweet!

Try our scaffold commands to get started:
  $ truffle create contract YourContractName # scaffold a contract
  $ truffle create test YourTestName         # scaffold a test

http://trufflesuite.com/docs
```

MyContract.sol

```
pragma solidity >=0.5.0 <0.9.0; contract
MyContract { string value; constructor() public {
value = "myValue"; } function get() public view
returns(string memory) { return value; } function
set(string memory _value) public { value = _value;
}
}
```

- **Start Ganache:**



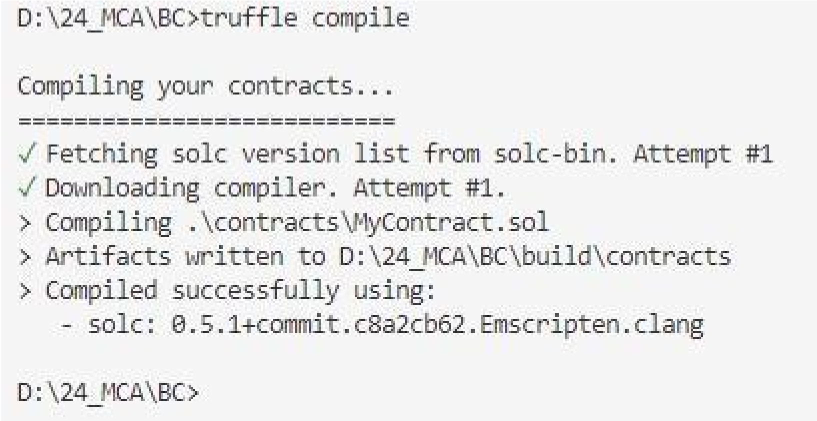
CURRENT BLOCK	GAS PRICE	GAS LIMIT	HARDFORK	NETWORK ID	RPC SERVER	MINING STATUS	WORKSPACE
0	20000000000	6721975	MUIRGLACIER	5777	HTTP://127.0.0.1:7545	AUTOMINING	QUICKSTART

MNEMONIC
admit february voice range night shoulder fuel head sport kite parade float

HD PATH
m/44'/60'/0'/0/account_index

ADDRESS	BALANCE	TX COUNT	INDEX
0×9258d63030746D3F5aE703c20E76b5F8B937ab13	100.00 ETH	0	0
0×6231D25044d16AAc963039f4528D947a68642dA8	100.00 ETH	0	1
0×9Fc8f8345ec105c8c93c17Bea21a797C6302b52f	100.00 ETH	0	2
0×b4377d8CAe8A9966936764810d7EfFf99A69f001	100.00 ETH	0	3
0×524a0BDD601703b9DF91CC13E47b81bc94810eFB	100.00 ETH	0	4

- **truffle compile**



```
D:\24_MCA\BC>truffle compile

Compiling your contracts...
===========================
✓ Fetching solc version list from solc-bin. Attempt #1
✓ Downloading compiler. Attempt #1.
> Compiling .\contracts\MyContract.sol
> Artifacts written to D:\24_MCA\BC\build\contracts
> Compiled successfully using:
   - solc: 0.5.1+commit.c8a2cb62.Emscripten.clang

D:\24_MCA\BC>
```

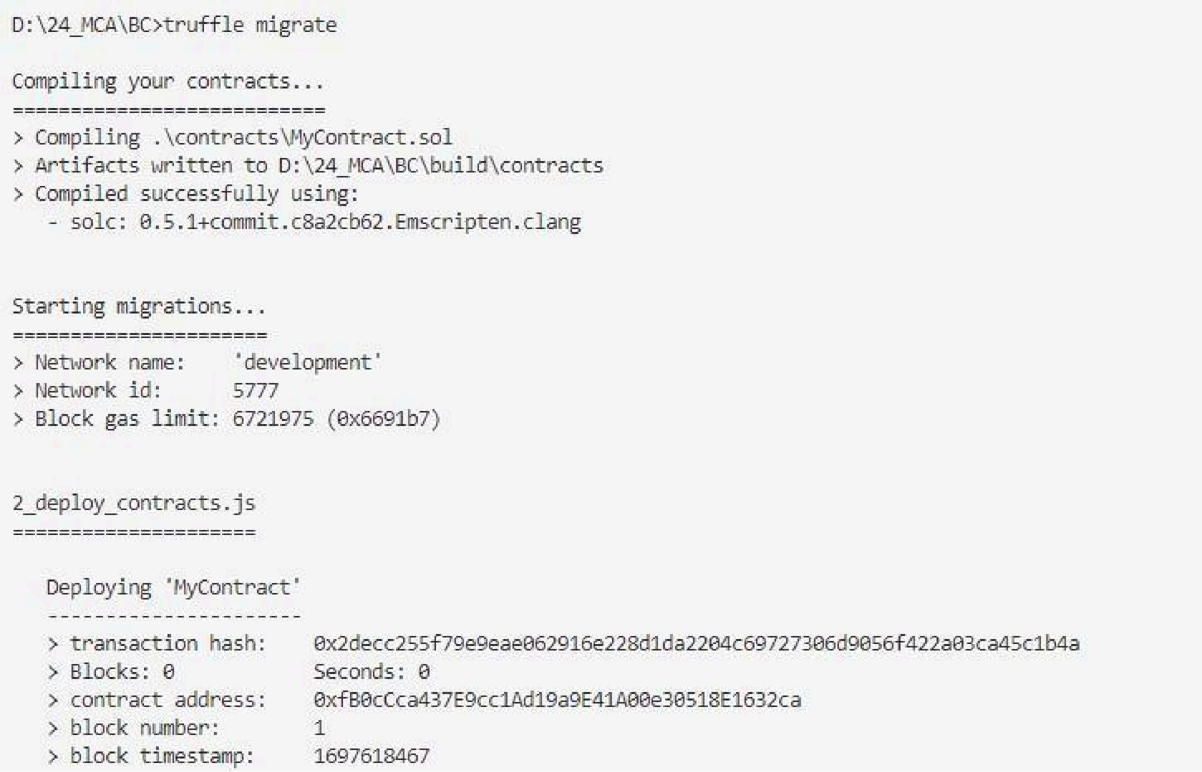
Migration > 2_deploy_contract.js var MyContract = artifacts.require("./MyContract.sol"); module.exports = function(deployer)

{

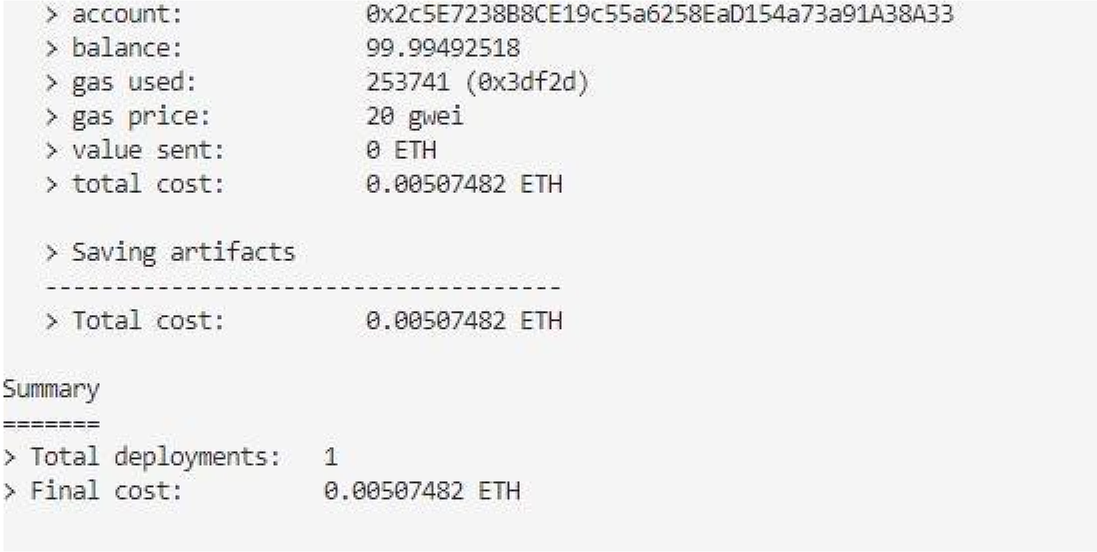
deployer.deploy(MyContract);

};

- **truffle migrate**

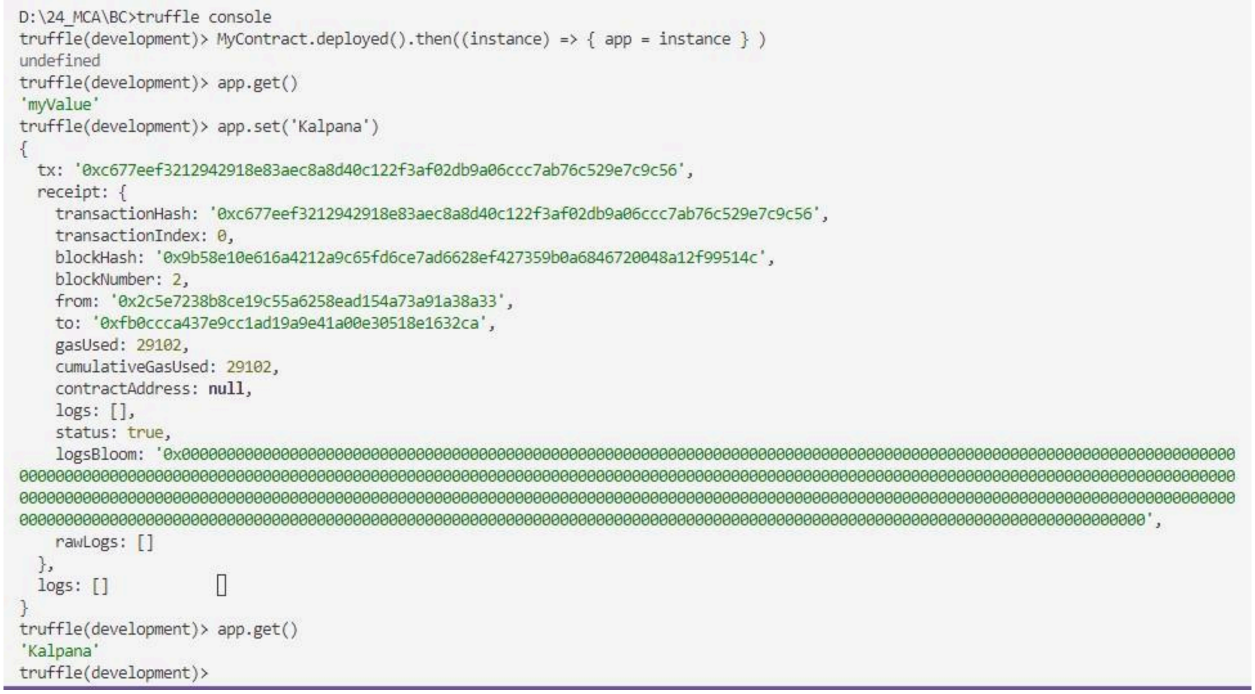


```
D:\24_MCA\BC>truffle migrate

Compiling your contracts...
===========================
> Compiling .\contracts\MyContract.sol
> Artifacts written to D:\24_MCA\BC\build\contracts
> Compiled successfully using:
   - solc: 0.5.1+commit.c8a2cb62.Emscripten.clang


Starting migrations...
======================
> Network name:    'development'
> Network id:      5777
> Block gas limit: 6721975 (0x6691b7)


2_deploy_contracts.js
=====================

   Deploying 'MyContract'
   ----------------------
   > transaction hash:    0x2decc255f79e9eae062916e228d1da2204c69727306d9056f422a03ca45c1b4a
   > Blocks: 0            Seconds: 0
   > contract address:    0xfB0cCca437E9cc1Ad19a9E41A00e30518E1632ca
   > block number:        1
   > block timestamp:     1697618467
```



```
   > account:             0x2c5E7238B8CE19c55a6258EaD154a73a91A38A33
   > balance:             99.99492518
   > gas used:            253741 (0x3df2d)
   > gas price:           20 gwei
   > value sent:          0 ETH
   > total cost:          0.00507482 ETH

   > Saving artifacts
   -------------------------------------
   > Total cost:          0.00507482 ETH

Summary
=======
> Total deployments:   1
> Final cost:          0.00507482 ETH
```

- **truffle console**



```
D:\24_MCA\BC>truffle console
truffle(development)> MyContract.deployed().then((instance) => { app = instance } )
undefined
truffle(development)> app.get()
'myValue'
truffle(development)> app.set('Kalpana')
{
  tx: '0xc677eef3212942918e83aec8a8d40c122f3af02db9a06ccc7ab76c529e7c9c56',
  receipt: {
    transactionHash: '0xc677eef3212942918e83aec8a8d40c122f3af02db9a06ccc7ab76c529e7c9c56',
    transactionIndex: 0,
    blockHash: '0x9b58e10e616a4212a9c65fd6ce7ad6628ef427359b0a6846720048a12f99514c',
    blockNumber: 2,
    from: '0x2c5e7238b8ce19c55a6258ead154a73a91a38a33',
    to: '0xfb0ccca437e9cc1ad19a9e41a00e30518e1632ca',
    gasUsed: 29102,
    cumulativeGasUsed: 29102,
    contractAddress: null,
    logs: [],
    status: true,
    logsBloom: '0x000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000',
    rawLogs: []
  },
  logs: []
}
truffle(development)> app.get()
'Kalpana'
truffle(development)>
```

1) Create a Bank Account contract and implement the following services:

Deposit

Withdraw (keep a condition that only the owner of the contract can withdraw)

Receive Ether

Transfer Ether

Check Balance

Code:

Mycontract.sol:

```
pragma  solidity  ^0.5.0;
contract MyContract
{ uint balance=31300; function getBalance() public
   view returns (uint)
   {  return  balance;  }  function  deposit(uint
   newDeposit) public
   { balance=balance+newDeposit;
   } }
```

```
pragma solidity ^0.5.0; contract
FC1
{ function receiveDeposit() payable public
   { } function getbalance() public view returns
   (uint)
   { return address(this).balance;
   } }

pragma solidity ^0.5.0; contract
FC2
{    address    owner;
   constructor() public
   {    owner=msg.sender;    }    function
   receiveDeposit() payable public
   { } function getbalance() public view returns
   (uint) { return address(this).balance;
   } function withdraw(uint funds) public
   { msg.sender.transfer(funds);
   } }

pragma solidity ^0.5.0; contract
FC3
{    address    owner;
   constructor() public
   { owner=msg.sender;
   } modifier ifOwner()
   { if(owner != msg.sender)
      {
         revert(); }
      else
      {
         _;
```

```
    } } function receiveDeposit() payable
  public
  { } function getbalance() public view returns
  (uint)
  { return address(this).balance;
  } function withdraw(uint funds) public ifOwner
  { msg.sender.transfer(funds);
  }
}
```

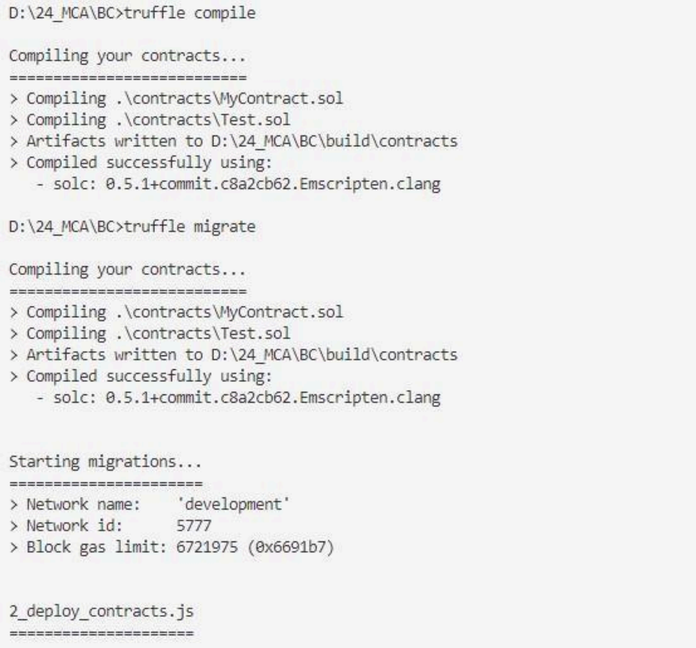
2_deploy_contracts.sol

```
var MyContract = artifacts.require("./MyContract.sol");
var FC1=artifacts.require("./FC1.sol")
var FC2=artifacts.require("./FC2.sol")
var FC3=artifacts.require("./FC3.sol")
var    Test    =    artifacts.require("./Test.sol");    module.exports    =
function(deployer)
{
deployer.deploy(MyContract);
deployer.deploy(FC1);
deployer.deploy(FC2);
deployer.deploy(FC3);
deployer.deploy(Test);
};
```

Output:

Compile:

Migrate:

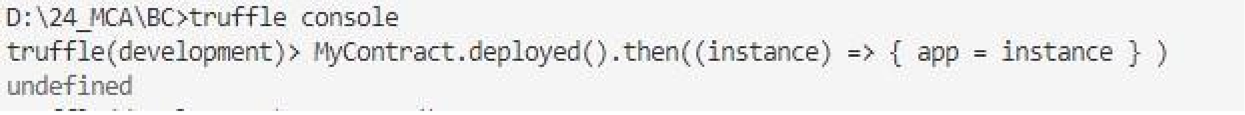


```
D:\24_MCA\BC>truffle compile

Compiling your contracts...
===========================
> Compiling .\contracts\MyContract.sol
> Compiling .\contracts\Test.sol
> Artifacts written to D:\24_MCA\BC\build\contracts
> Compiled successfully using:
   - solc: 0.5.1+commit.c8a2cb62.Emscripten.clang

D:\24_MCA\BC>truffle migrate

Compiling your contracts...
===========================
> Compiling .\contracts\MyContract.sol
> Compiling .\contracts\Test.sol
> Artifacts written to D:\24_MCA\BC\build\contracts
> Compiled successfully using:
   - solc: 0.5.1+commit.c8a2cb62.Emscripten.clang


Starting migrations...
======================
> Network name:    'development'
> Network id:      5777
> Block gas limit: 6721975 (0x6691b7)


2_deploy_contracts.js
=====================
```

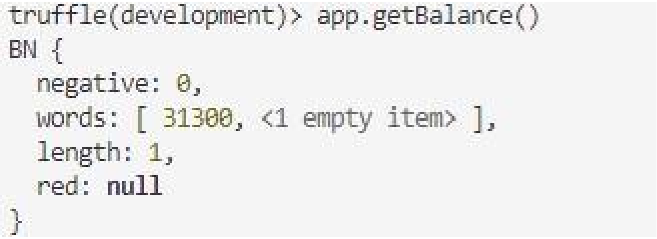
Check Balance: truffle

console

Mycontract.deployed().then((instance)=>{app = instance}) app.getBalance()

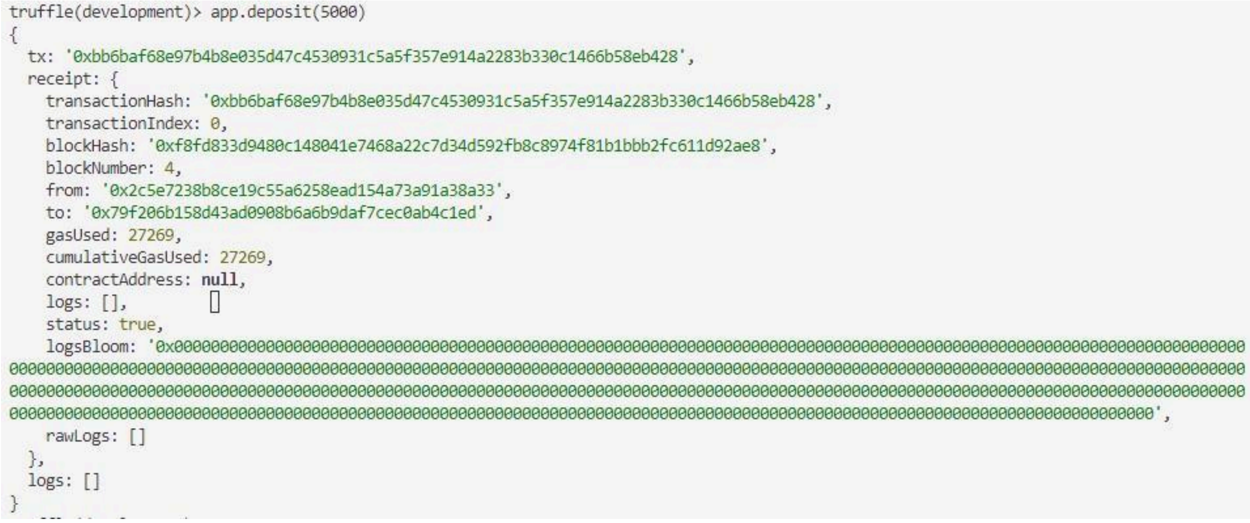


```
D:\24_MCA\BC>truffle console
truffle(development)> MyContract.deployed().then((instance) => { app = instance } )
undefined
```

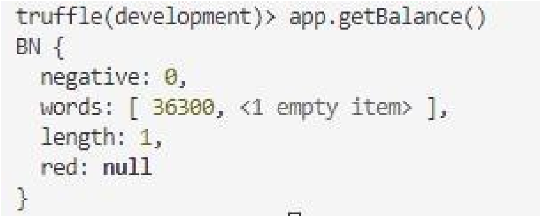


```
truffle(development)> app.getBalance()
BN {
  negative: 0,
  words: [ 31300, <1 empty item> ],
  length: 1,
  red: null
}
```

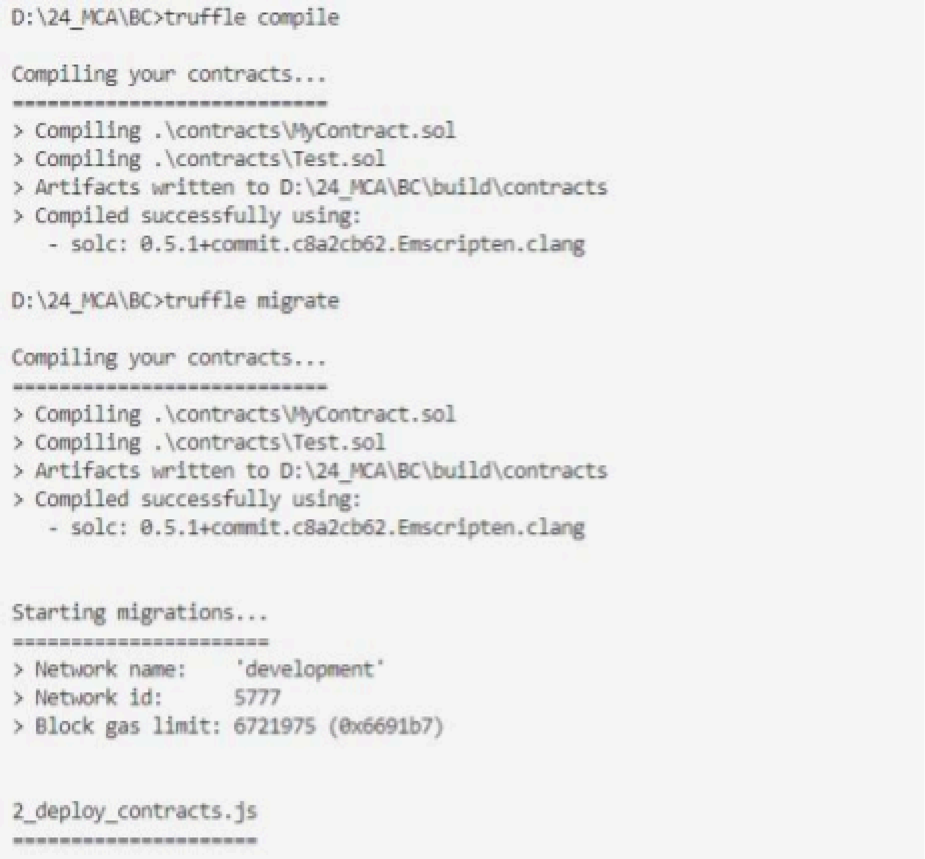
Deposit 5000:



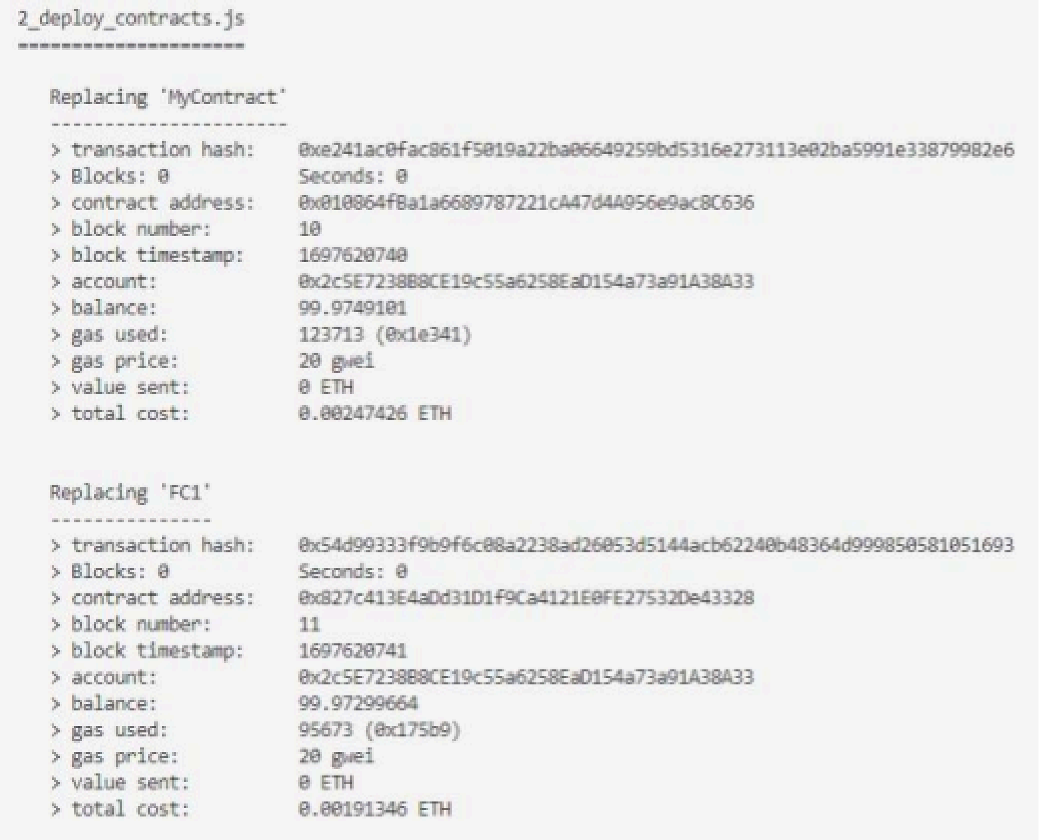
```
truffle(development)> app.deposit(5000)
{
  tx: '0xbb6baf68e97b4b8e035d47c4530931c5a5f357e914a2283b330c1466b58eb428',
  receipt: {
    transactionHash: '0xbb6baf68e97b4b8e035d47c4530931c5a5f357e914a2283b330c1466b58eb428',
    transactionIndex: 0,
    blockHash: '0xf8fd833d9480c148041e7468a22c7d34d592fb8c8974f81b1bbb2fc611d92ae8',
    blockNumber: 4,
    from: '0x2c5e7238b8ce19c55a6258ead154a73a91a38a33',
    to: '0x79f206b158d43ad0908b6a6b9daf7cec0ab4c1ed',
    gasUsed: 27269,
    cumulativeGasUsed: 27269,
    contractAddress: null,
    logs: [],
    status: true,
    logsBloom: '0x0000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000',
    rawLogs: []
  },
  logs: []
}
```



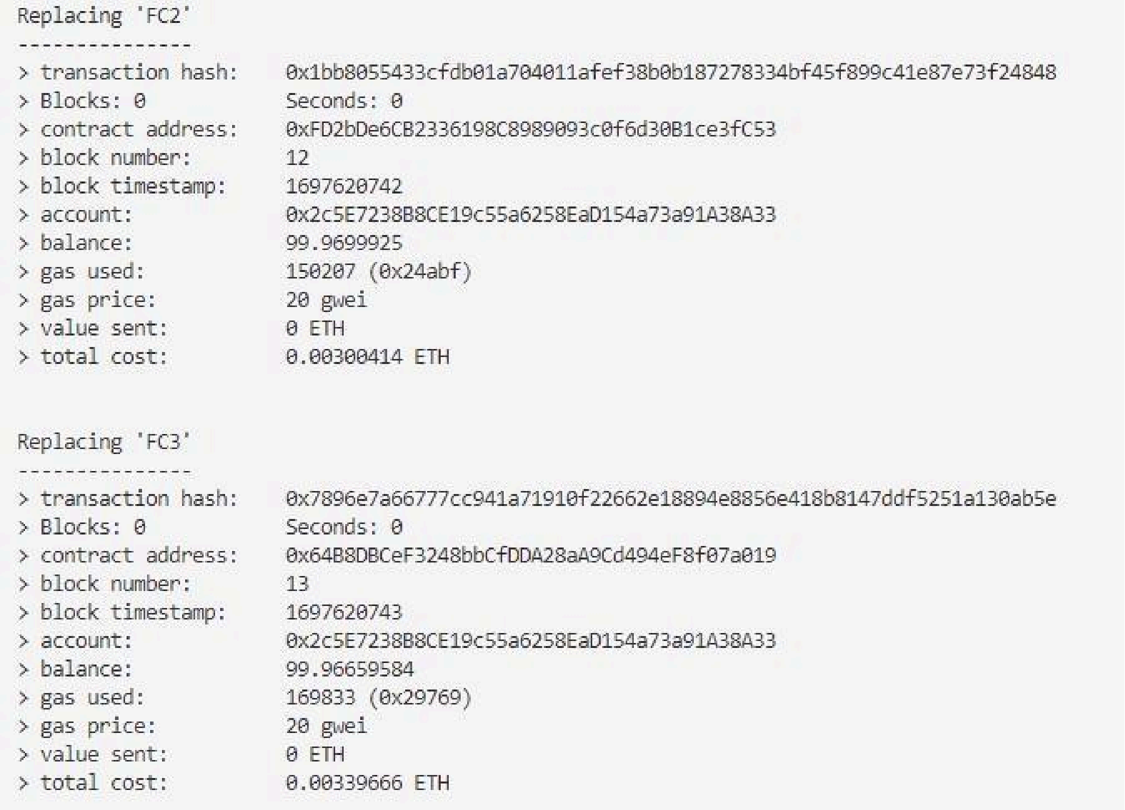
```
truffle(development)> app.getBalance()
BN {
  negative: 0,
  words: [ 36300, <1 empty item> ],
  length: 1,
  red: null
}
```



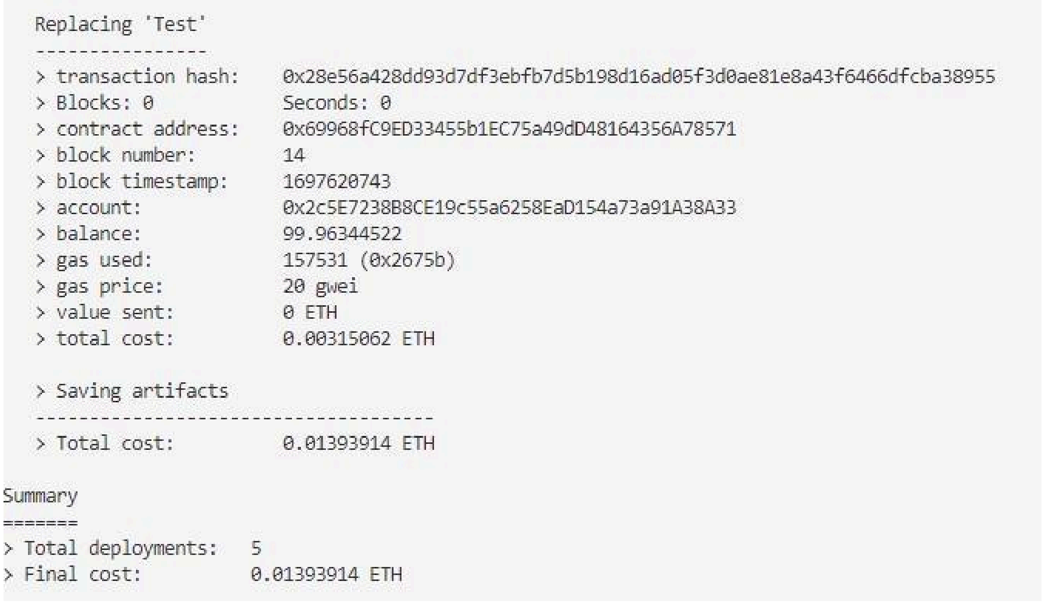
```
D:\24_MCA\BC>truffle compile

Compiling your contracts...
===========================
> Compiling .\contracts\MyContract.sol
> Compiling .\contracts\Test.sol
> Artifacts written to D:\24_MCA\BC\build\contracts
> Compiled successfully using:
   - solc: 0.5.1+commit.c8a2cb62.Emscripten.clang

D:\24_MCA\BC>truffle migrate

Compiling your contracts...
===========================
> Compiling .\contracts\MyContract.sol
> Compiling .\contracts\Test.sol
> Artifacts written to D:\24_MCA\BC\build\contracts
> Compiled successfully using:
   - solc: 0.5.1+commit.c8a2cb62.Emscripten.clang


Starting migrations...
======================
> Network name:    'development'
> Network id:      5777
> Block gas limit: 6721975 (0x6691b7)


2_deploy_contracts.js
=====================
```



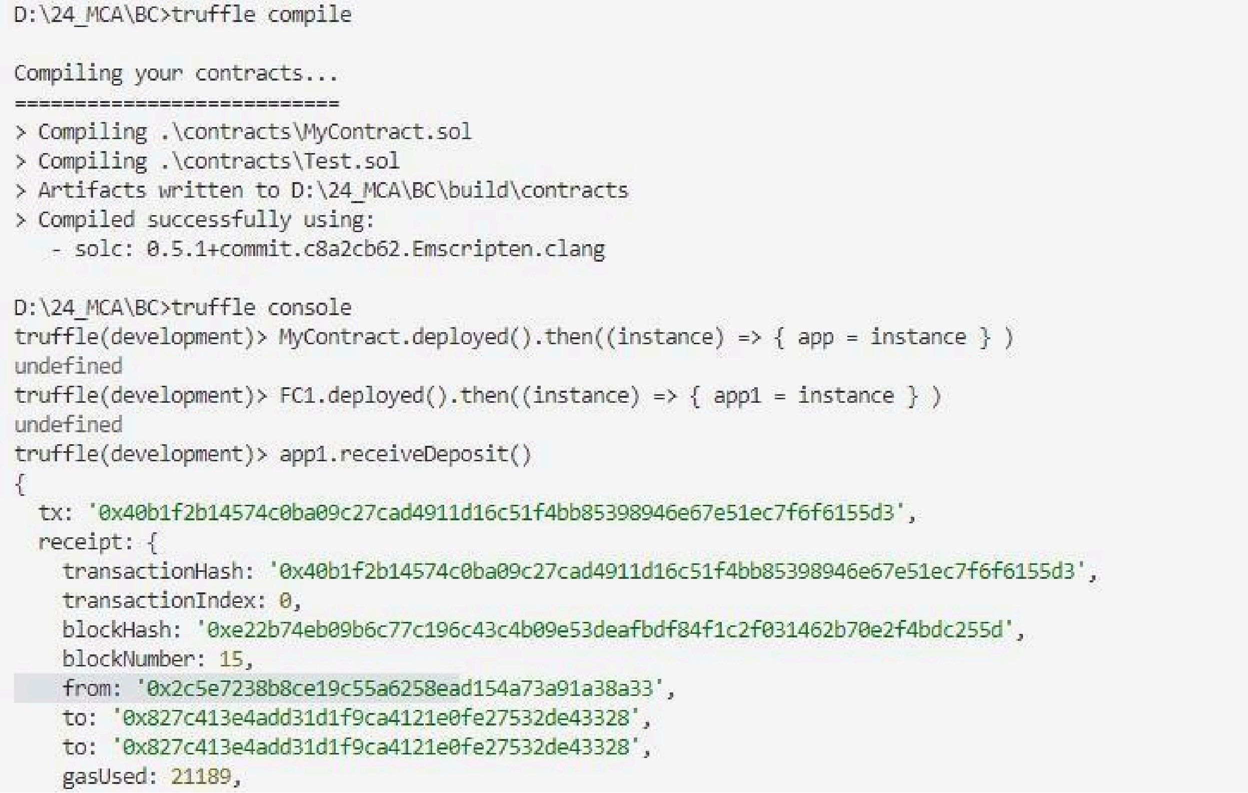
```
2_deploy_contracts.js
=====================

   Replacing 'MyContract'
   ----------------------
   > transaction hash:    0xe241ac0fac861f5019a22ba06649259bd5316e273113e02ba5991e33879982e6
   > Blocks: 0            Seconds: 0
   > contract address:    0x010864fBa1a6689787221cA47d4A956e9ac8C636
   > block number:        10
   > block timestamp:     1697620740
   > account:             0x2c5E7238B8CE19c55a6258EaD154a73a91A38A33
   > balance:             99.9749101
   > gas used:            123713 (0x1e341)
   > gas price:           20 gwei
   > value sent:          0 ETH
   > total cost:          0.00247426 ETH


   Replacing 'FC1'
   ---------------
   > transaction hash:    0x54d99333f9b9f6c08a2238ad26053d5144acb62240b48364d999850581051693
   > Blocks: 0            Seconds: 0
   > contract address:    0x827c413E4aDd31D1f9Ca4121E0FE27532De43328
   > block number:        11
   > block timestamp:     1697620741
   > account:             0x2c5E7238B8CE19c55a6258EaD154a73a91A38A33
   > balance:             99.97299664
   > gas used:            95673 (0x175b9)
   > gas price:           20 gwei
   > value sent:          0 ETH
   > total cost:          0.00191346 ETH
```



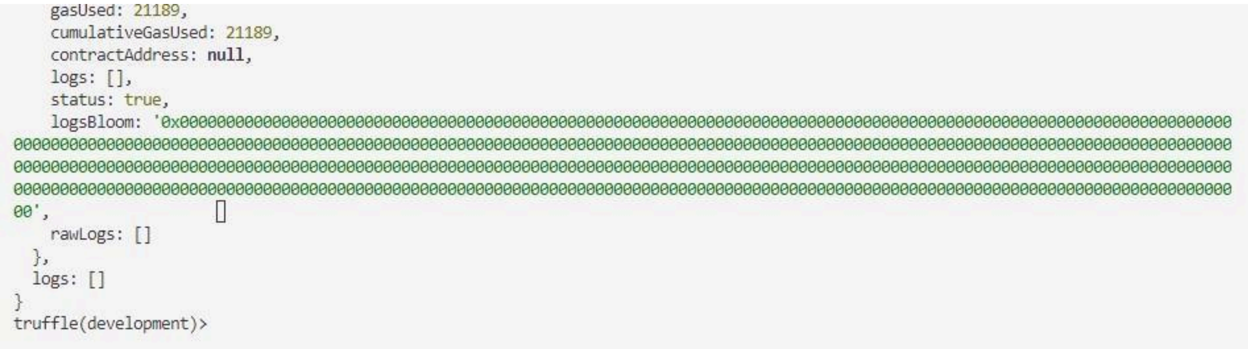
```
Replacing 'FC2'
---------------
> transaction hash:    0x1bb8055433cfdb01a704011afef38b0b187278334bf45f899c41e87e73f24848
> Blocks: 0            Seconds: 0
> contract address:    0xFD2bDe6CB2336198C8989093c0f6d30B1ce3fC53
> block number:        12
> block timestamp:     1697620742
> account:             0x2c5E7238B8CE19c55a6258EaD154a73a91A38A33
> balance:             99.9699925
> gas used:            150207 (0x24abf)
> gas price:           20 gwei
> value sent:          0 ETH
> total cost:          0.00300414 ETH


Replacing 'FC3'
---------------
> transaction hash:    0x7896e7a66777cc941a71910f22662e18894e8856e418b8147ddf5251a130ab5e
> Blocks: 0            Seconds: 0
> contract address:    0x64B8DBCeF3248bbCfDDA28aA9Cd494eF8f07a019
> block number:        13
> block timestamp:     1697620743
> account:             0x2c5E7238B8CE19c55a6258EaD154a73a91A38A33
> balance:             99.96659584
> gas used:            169833 (0x29769)
> gas price:           20 gwei
> value sent:          0 ETH
> total cost:          0.00339666 ETH
```



```
   Replacing 'Test'
   ----------------
   > transaction hash:    0x28e56a428dd93d7df3ebfb7d5b198d16ad05f3d0ae81e8a43f6466dfcba38955
   > Blocks: 0            Seconds: 0
   > contract address:    0x69968fC9ED33455b1EC75a49dD48164356A78571
   > block number:        14
   > block timestamp:     1697620743
   > account:             0x2c5E7238B8CE19c55a6258EaD154a73a91A38A33
   > balance:             99.96344522
   > gas used:            157531 (0x2675b)
   > gas price:           20 gwei
   > value sent:          0 ETH
   > total cost:          0.00315062 ETH

   > Saving artifacts
   -------------------------------------
   > Total cost:          0.01393914 ETH

Summary
=======
> Total deployments:   5
> Final cost:          0.01393914 ETH
```

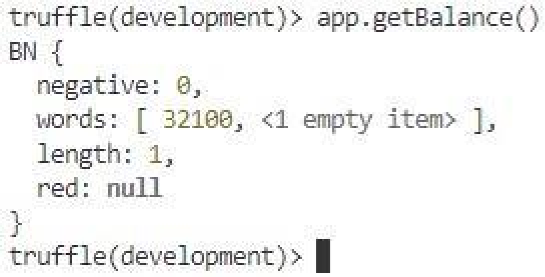


```
D:\24_MCA\BC>truffle compile

Compiling your contracts...
===========================
> Compiling .\contracts\MyContract.sol
> Compiling .\contracts\Test.sol
> Artifacts written to D:\24_MCA\BC\build\contracts
> Compiled successfully using:
   - solc: 0.5.1+commit.c8a2cb62.Emscripten.clang

D:\24_MCA\BC>truffle console
truffle(development)> MyContract.deployed().then((instance) => { app = instance } )
undefined
truffle(development)> FC1.deployed().then((instance) => { app1 = instance } )
undefined
truffle(development)> app1.receiveDeposit()
{
  tx: '0x40b1f2b14574c0ba09c27cad4911d16c51f4bb85398946e67e51ec7f6f6155d3',
  receipt: {
    transactionHash: '0x40b1f2b14574c0ba09c27cad4911d16c51f4bb85398946e67e51ec7f6f6155d3',
    transactionIndex: 0,
    blockHash: '0xe22b74eb09b6c77c196c43c4b09e53deafbdf84f1c2f031462b70e2f4bdc255d',
    blockNumber: 15,
    from: '0x2c5e7238b8ce19c55a6258ead154a73a91a38a33',
    to: '0x827c413e4add31d1f9ca4121e0fe27532de43328',
    to: '0x827c413e4add31d1f9ca4121e0fe27532de43328',
    gasUsed: 21189,
```



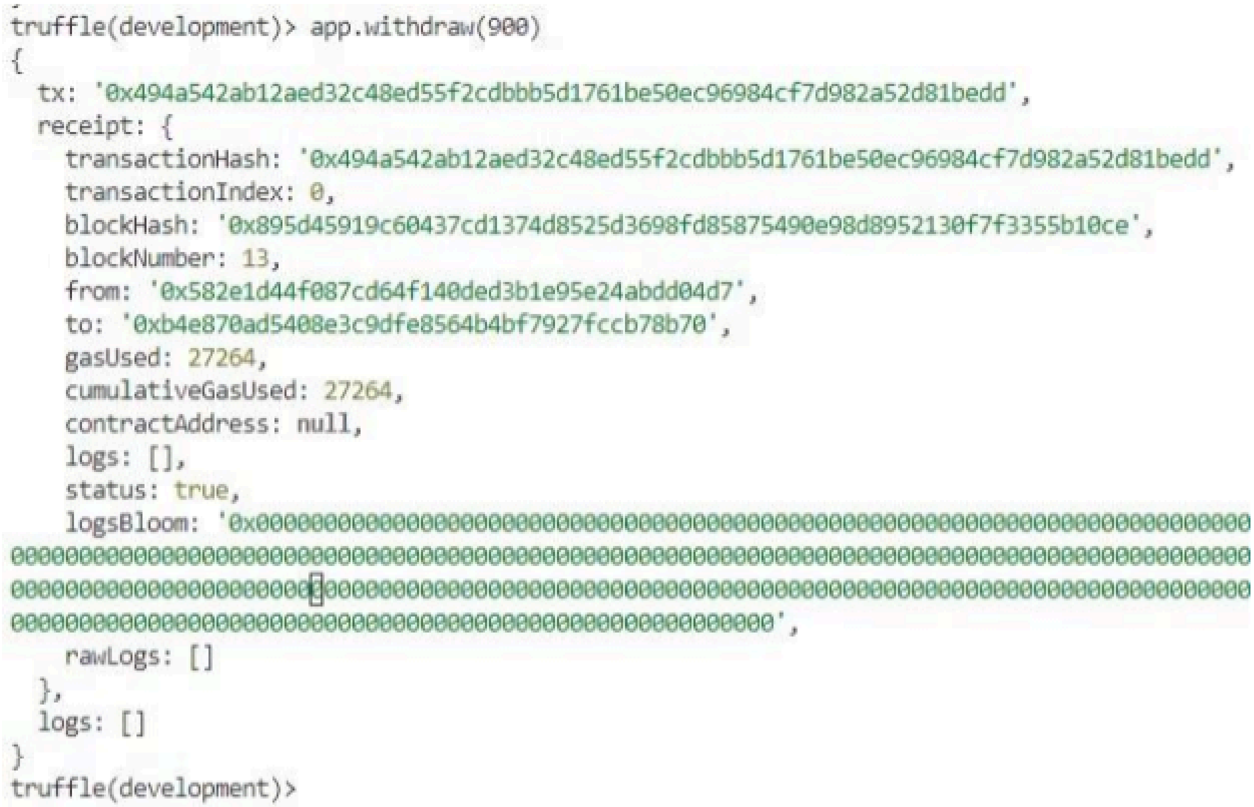
```
    gasUsed: 21189,
    cumulativeGasUsed: 21189,
    contractAddress: null,
    logs: [],
    status: true,
    logsBloom: '0x000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
00',
    rawLogs: []
  },
  logs: []
}
truffle(development)>
```

Receive Ether:

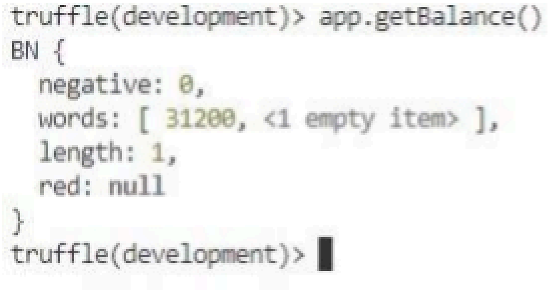


```
truffle(development)> app.getBalance()
BN {
  negative: 0,
  words: [ 32100, <1 empty item> ],
  length: 1,
  red: null
}
truffle(development)>
```

Withdraw 900:



```
truffle(development)> app.withdraw(900)
{
  tx: '0x494a542ab12aed32c48ed55f2cdbbb5d1761be50ec96984cf7d982a52d81bedd',
  receipt: {
    transactionHash: '0x494a542ab12aed32c48ed55f2cdbbb5d1761be50ec96984cf7d982a52d81bedd',
    transactionIndex: 0,
    blockHash: '0x895d45919c60437cd1374d8525d3698fd85875490e98d8952130f7f3355b10ce',
    blockNumber: 13,
    from: '0x582e1d44f087cd64f140ded3b1e95e24abdd04d7',
    to: '0xb4e870ad5408e3c9dfe8564b4bf7927fccb78b70',
    gasUsed: 27264,
    cumulativeGasUsed: 27264,
    contractAddress: null,
    logs: [],
    status: true,
    logsBloom: '0x0000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000',
    rawLogs: []
  },
  logs: []
}
truffle(development)>
```



```
truffle(development)> app.getBalance()
BN {
  negative: 0,
  words: [ 31200, <1 empty item> ],
  length: 1,
  red: null
}
truffle(development)>
```

2) Create a Smart contract to simulate function overloading . Execute the contract using the truffle framework.

Code:

Test.sol:

pragma solidity ^0.5.0; contract Test { function getSum(uint
a, uint b) public pure returns(uint){

    return a + b;

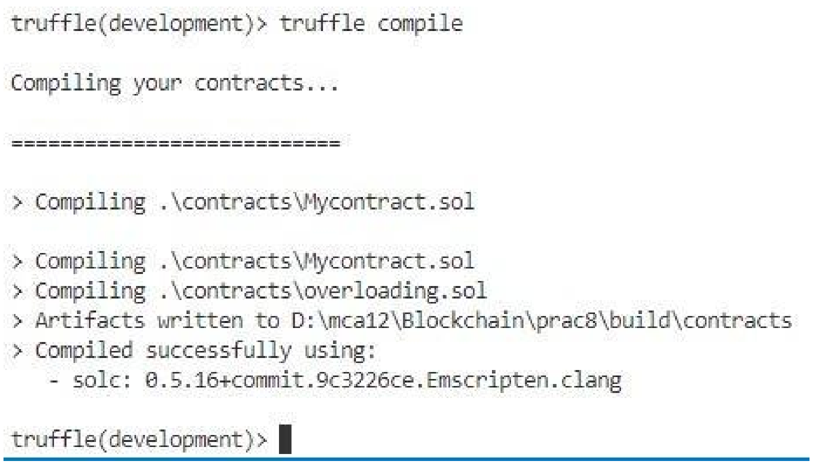
  } function getSum(uint a, uint b, uint c) public pure returns(uint){

```
    return a + b + c;
  }
  function callSumWithTwoArguments(uint a, uint b) public pure returns(uint){
    return getSum(a,b);
  } function callSumWithThreeArguments(uint a, uint b, uint c) public pure returns(uint){
    return getSum(a,b,c);
  }
}
```

2_deploy_contract.js:

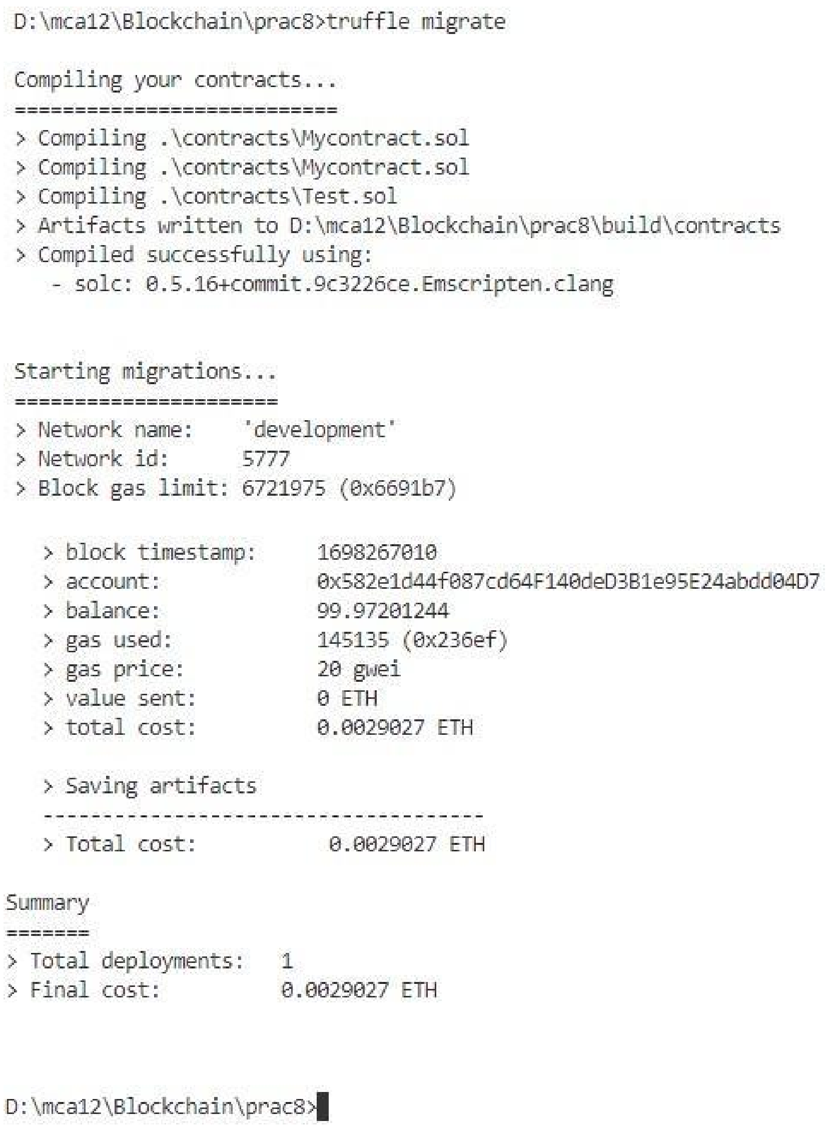
```
var Test = artifacts.require("./Test.sol"); module.exports =
function(deployer)
{ deployer.deploy(Test);
};
```

Output:



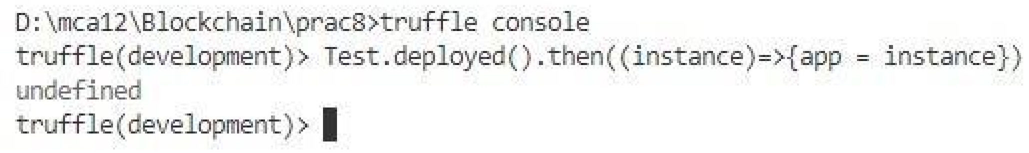
```
truffle(development)> truffle compile

Compiling your contracts...

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

> Compiling .\contracts\Mycontract.sol

> Compiling .\contracts\Mycontract.sol
> Compiling .\contracts\overloading.sol
> Artifacts written to D:\mca12\Blockchain\prac8\build\contracts
> Compiled successfully using:
   - solc: 0.5.16+commit.9c3226ce.Emscripten.clang

truffle(development)>
```

Migrate



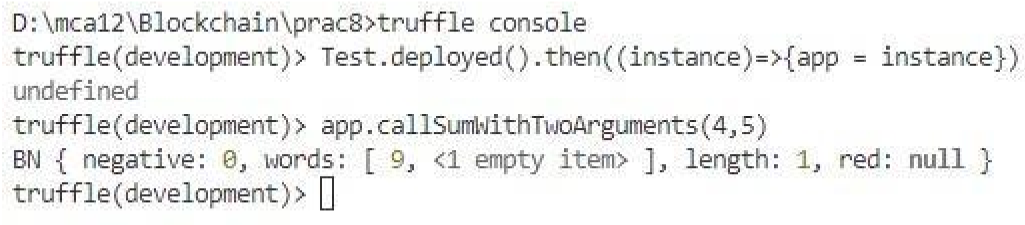
```
D:\mca12\Blockchain\prac8>truffle migrate

Compiling your contracts...
===========================
> Compiling .\contracts\Mycontract.sol
> Compiling .\contracts\Mycontract.sol
> Compiling .\contracts\Test.sol
> Artifacts written to D:\mca12\Blockchain\prac8\build\contracts
> Compiled successfully using:
   - solc: 0.5.16+commit.9c3226ce.Emscripten.clang


Starting migrations...
======================
> Network name:    'development'
> Network id:      5777
> Block gas limit: 6721975 (0x6691b7)

   > block timestamp:     1698267010
   > account:             0x582e1d44f087cd64F140deD3B1e95E24abdd04D7
   > balance:             99.97201244
   > gas used:            145135 (0x236ef)
   > gas price:           20 gwei
   > value sent:          0 ETH
   > total cost:          0.0029027 ETH

   > Saving artifacts
   -------------------------------------
   > Total cost:           0.0029027 ETH

Summary
=======
> Total deployments:   1
> Final cost:          0.0029027 ETH


D:\mca12\Blockchain\prac8>
```

Create instance of Test:



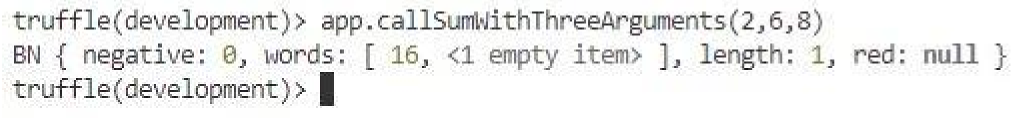
```
D:\mca12\Blockchain\prac8>truffle console
truffle(development)> Test.deployed().then((instance)=>{app = instance})
undefined
truffle(development)>
```

Get sum of 2 numbers:



```
D:\mca12\Blockchain\prac8>truffle console
truffle(development)> Test.deployed().then((instance)=>{app = instance})
undefined
truffle(development)> app.callSumWithTwoArguments(4,5)
BN { negative: 0, words: [ 9, <1 empty item> ], length: 1, red: null }
truffle(development)>
```

Get sum of three numbers:



```
truffle(development)> app.callSumWithThreeArguments(2,6,8)
BN { negative: 0, words: [ 16, <1 empty item> ], length: 1, red: null }
truffle(development)>
```

Conclusion: I have successfully understood how to construct a smart contract using Solidity and Truffle framework.