

# Lottery Smart Contract

K.Y. Shen

## 1.Rinkerby Faucet

<https://faucet.rinkeby.io/>

## 2.Deploy contract using Infura

<https://infura.io/>

Sign up account and get your link and mnemonic words

`npm install`

`node deploy.js`

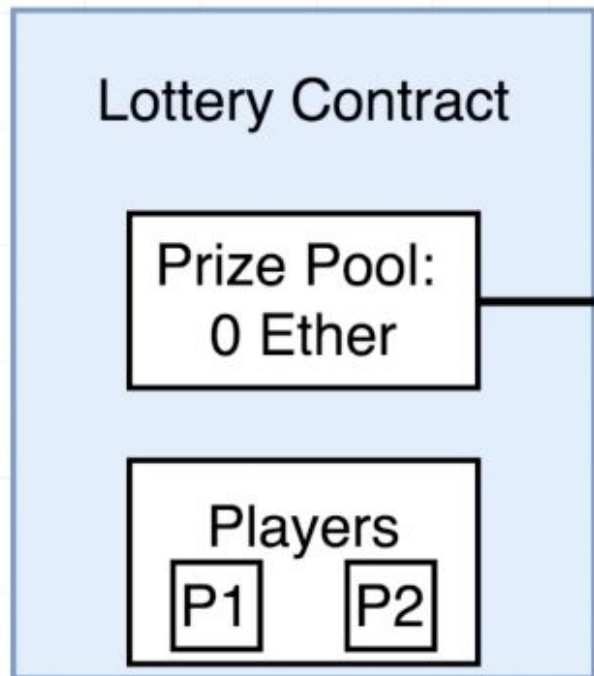
## 3.Deploy frontend on Netlify

`npm install`

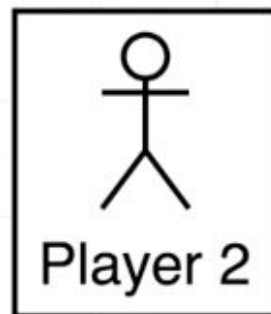
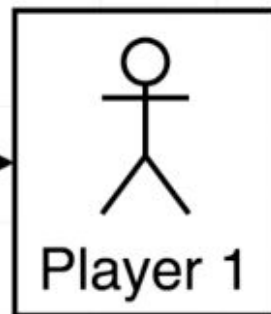
`npm run start`

`npm run build`

Note: the version of solidity inside package.json should be equal to Lottery.sol(0.4.17)



2 Ether

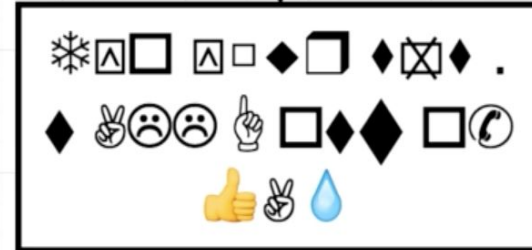


Our Javascript  
Code



ABI

Application  
Binary  
Interface



Bytecode

Solidity Smart Contract



# Lottery Contract

## Variables

Name	Purpose
manager	Address of person who created the contract
players	Array of addresses of people who have entered

## Functions

Name	Purpose
enter	Enters a player into the lottery
pickWinner	Randomly picks a winner and sends them the prize pool

## Common Function Types

Can only use  
one per function

public

Anyone can call this function

private

Only this contract can call this function.

view

This function returns data and does *not* modify the contract's data

constant

This function returns data and does *not* modify the contract's data

pure

Function will not modify or even *read* the contract's data

payable

When someone call this function they might send ether along

They mean the  
same thing

## Basic Types

Name	Notes	Examples		
string	Sequence of characters	"Hi there!"	"Chocolate"	
bool	Boolean value	true	false	
int	Integer, positive or negative. Has no decimal	0	-30000	59158
uint	'Unsigned' integer, positive number. Has no decimal	0	30000	999910
fixed/ufixed	'Fixed' point number. Number with a decimal after it	20.001	-42.4242	3.14
address	Has methods tied to it for sending money	0x18bae199c8dbae199c8d		

## Integer Ranges

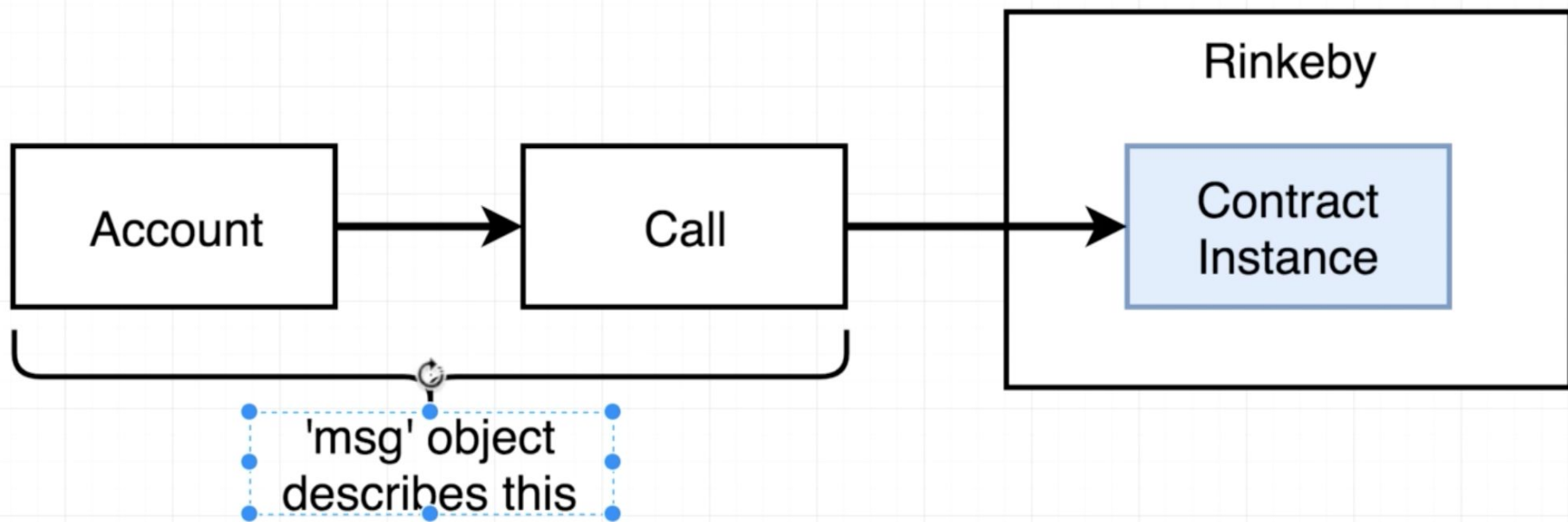
Name	Lower Bound	Upper Bound
int8	-128	127
int16	-32,768	32,767
int32	-2,147,483,648	2,147,483,647
...	...	...
int256	Really, really negative	Really, really big

int

==

int256





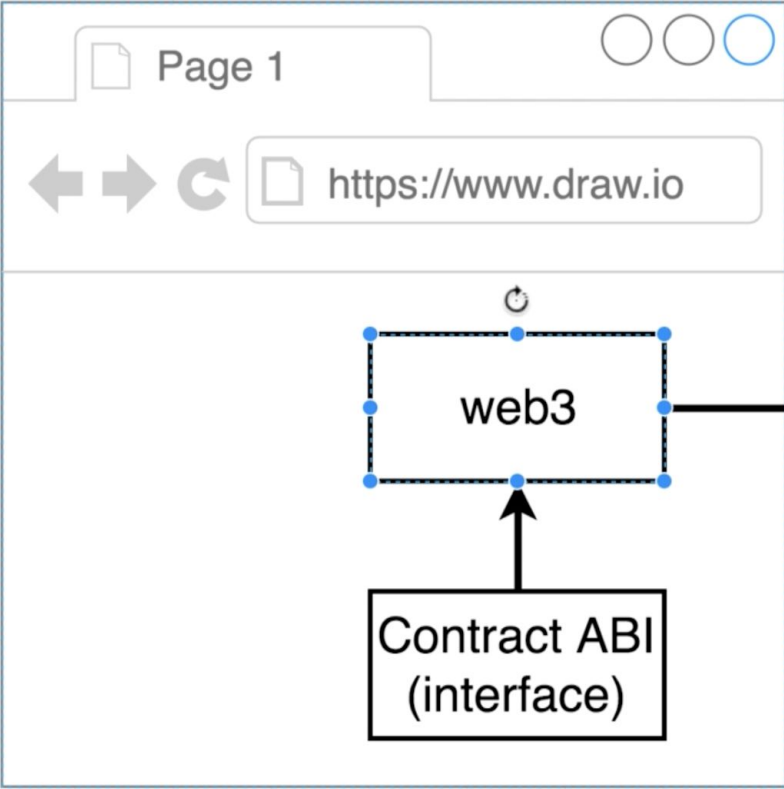
## The 'msg' Global Variable

Property Name	Property Name
msg.data	'Data' field from the call or transaction that invoked the current function
msg.gas	Amount of gas the current function invocation has available
msg.sender	Address of the account that started the current function invocation
msg.value	Amount of ether (in wei) that was sent along with the function invocation

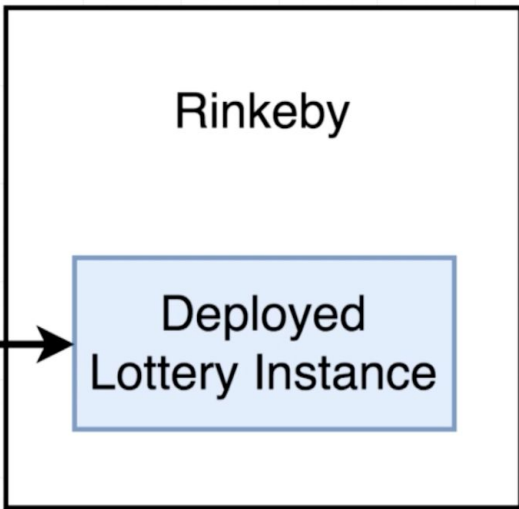
## Reference Types

Name	Notes	Examples
fixed array	Array that contains a <i>single type</i> of element. Has an unchanging length	<div>int[3] --&gt; [1, 2, 3]</div> <div>bool[2] --&gt; [true, false]</div>
dynamic array <sup>†</sup>	Array that contains a <i>single type</i> of element. Can change in size over time	<div>int[] --&gt; [1,2,3]</div> <div>bool[] --&gt; [true, false]</div>
mapping	Collection of key value pairs. Think of Javascript objects, Ruby hashes, or Python dictionary. All keys must be of the same type, and all values must be of the same type	<div>mapping(string =&gt; string)</div> <div>mapping(int =&gt; bool)</div>
struct	Collection of key value pairs that can have different types.	<div> <pre>struct Car {   string make;   string model;   uint value; }</pre> </div>

```
1 |pragma solidity ^0.4.17;
2
3 contract Lottery {
4     address public manager;
5     address[] public players;
6
7     function Lottery() public {
8         manager = msg.sender;
9     }
10
11     function enter() public payable {
12         require(msg.value > .01 ether);
13
14         players.push(msg.sender);
15     }
16
17     function random() private view returns (uint) {
18         return uint(keccak256(block.difficulty, now, players));
19     }
20
21     function pickWinner() public restricted {
22         uint index = random() % players.length;
23         players[index].transfer(this.balance);
24         players = new address[](0);
25     }
26
27     modifier restricted() {
28         require(msg.sender == manager);
29         _;
30     }
31
32     function getPlayers() public view returns (address[]) {
33         return players;
34     }
35 }
```



*Find the contract at  
address 0x1b8a92*





Rinkeby Test Network



○ Not connected

**Account 1**

0xC1bf...F124



2.999 ETH

### Connected sites



Account 1 is not connected to any sites.

[Manually connect to current site](#)



2.999 ETH



[Add Token](#)

# Reference

<https://www.udemy.com/course/ethereum-and-solidity-the-complete-developers-guide/learn/lecture/9020602#overview>

<https://solidity.readthedocs.io/>

<https://github.com/kuoyehs/tiba-lottery>