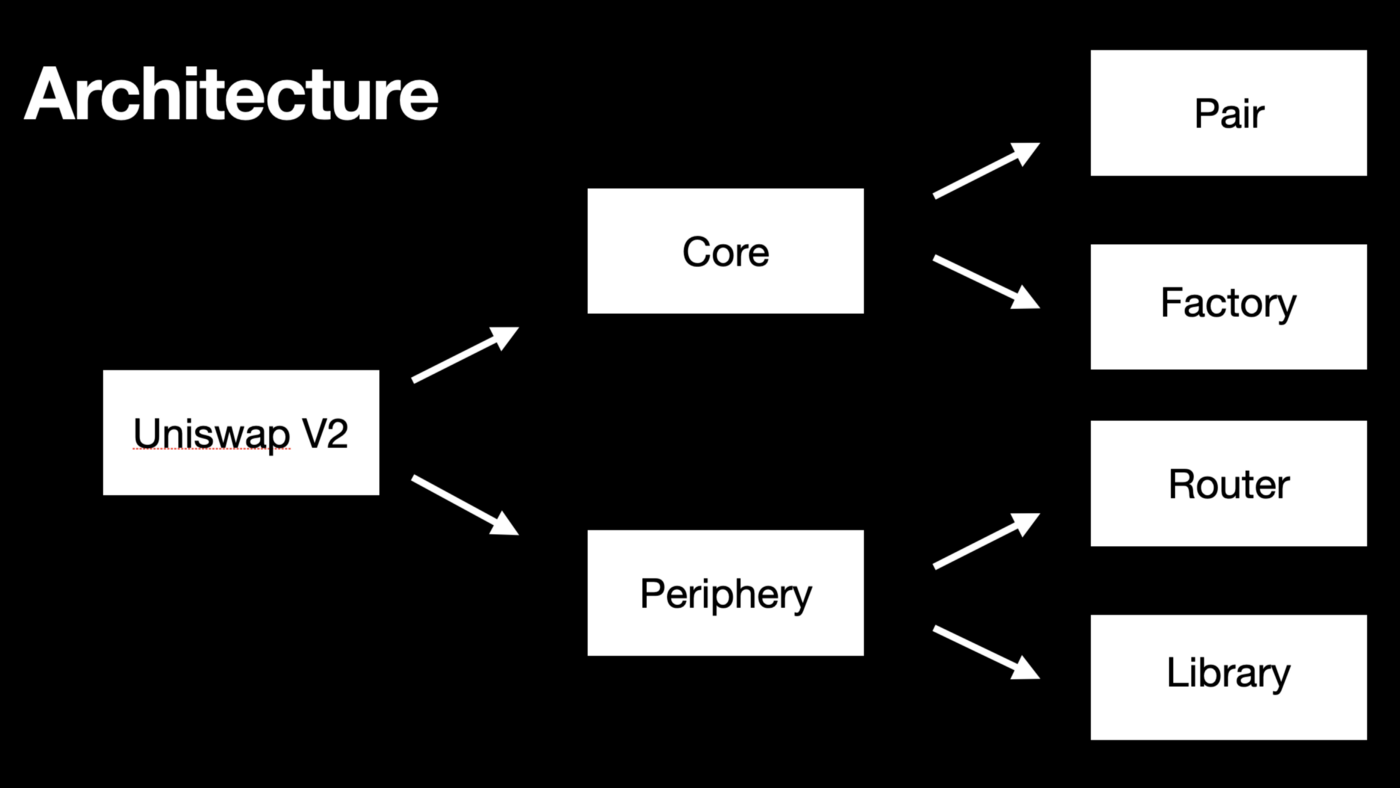
**Uniswap Clone**

**Uniswap have 2 types of contracts core Contracts & periphery Contracts.**

** In order to clone Uniswap we need these contracts to clone Core contract and periphery contract**

**Core Contracts: git clone** [**https://github.com/Uniswap/v2-core.git**](https://github.com/Uniswap/v2-core.git)

**Periphery Contracts:git clone** [**https://github.com/Uniswap/v2-periphery.git**](https://github.com/Uniswap/v2-periphery.git)

**npm i -y**

**Now make 2 separate folders in uniswap main folder with name core and periphery  
install truffle in both of these folders**npm install -g truffle  
truffle init

Next step is to copy all the contracts and paste them in contract folder of truffle do this for both folders  
  
first we deploy core contracts   
make 2 solidity files in contracts folder  
Token1.sol  
Token2.sol  
now paste this code in both files just change name little bit

Token1.sol

pragma solidity ^ 0.5.16;

import '@openzeppelin/contracts/token/ERC20/ERC20Detailed.sol';

import '@openzeppelin/contracts/token/ERC20/ERC20.sol';

contract Token1 is ERC20Detailed,ERC20{

    constructor() ERC20Detailed('Token 1','TK1',18)public{

    }

}

Token2.sol

pragma solidity ^ 0.5.16;

import '@openzeppelin/contracts/token/ERC20/ERC20Detailed.sol';

import '@openzeppelin/contracts/token/ERC20/ERC20.sol';

contract Token2 is ERC20Detailed,ERC20{

    constructor() ERC20Detailed('Token 2','TK2',18)public{

    }

}

Now goto migrations.js and write this code

const Factory = artifacts.require("UniswapV2Factory.sol");

const Token1 = artifacts.require("Token1.sol");

const Token2 = artifacts.require("Token2.sol");

module.exports = async function(deployer,network,addresses) {

 await deployer.deploy(Factory,addresses[0]);

 console.log("######## 1st address",addresses[0]);

 const factory =await Factory.deployed();

 let token1address,token2address;

 if(network === 'mainnet'){

    token1address = '';

    token2address = '';

 }else{

    await deployer.deploy(Token1);

    await deployer.deploy(Token2);

    const token1 = await Token1.deployed();

    const token2 = await Token2.deployed();

    token1address = token1.address;

    token2address = token2.address;

 }

 await factory.createPair(token1address,token2address);

};

Also run this file for importing openzeppellin files

npm install @openzeppelin/contracts@2.5.1

goto truffle-config.js

change solidity version to 0.5.16

and also un comment

    development: {

     host: "127.0.0.1",     // Localhost (default: none)

     port: 8545,            // Standard Ethereum port (default: none)

     network\_id: "\*",       // Any network (default: none)

    },

Now we are ready deploy your contracts  
run a command on another terminal of code  
ganache-cli

And on main terminal of vs code run   
truffle compile  
truffle migrate

It will deploy 3 contracts Factory.sol,Token1.sol,Token2.sol

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Now goto the periphery folder and write

In this Periphery folder goto Migrations and make a new file 2\_deploy\_contract.js   
and write this code there.

const Router = artifacts.require("UniswapV2Router02.sol");

const WETH = artifacts.require("WETH.sol");

module.exports = async function (deployer, network) {

  let weth;

  const FACTORY\_ADDRESS = "0x72cF2FA5a40514ab5F991d44201De45DBEcb8eE4"; // address where factory contract is deployed

  if (network == "mainnet") {

    weth = WETH.at("0xC02aaA39b223FE8D0A0e5C4F27eAD9083C756Cc2");//goto etherscan and and search for WETH and copy that address and paste it here

  } else {

    await deployer.deploy(WETH);

    weth = await WETH.deployed();

  }

  //router takes 2 arguments 1st one is factory address and second one is WETH address

  await deployer.deploy(Router,FACTORY\_ADDRESS,weth.address);

};

Now goto contracts folder and make a new sol file whose name is WETH.sol  
and paste this particular code in there

pragma solidity ^0.6.6;

contract WETH {

    string public name     = "Wrapped Ether";

    string public symbol   = "WETH";

    uint8  public decimals = 18;

    event  Approval(address indexed src, address indexed guy, uint wad);

    event  Transfer(address indexed src, address indexed dst, uint wad);

    event  Deposit(address indexed dst, uint wad);

    event  Withdrawal(address indexed src, uint wad);

    mapping (address => uint)                       public  balanceOf;

    mapping (address => mapping (address => uint))  public  allowance;

    receive() external payable {

        deposit();

    }

    function deposit() public payable {

        balanceOf[msg.sender] += msg.value;

       emit Deposit(msg.sender, msg.value);

    }

    function withdraw(uint wad) public {

        require(balanceOf[msg.sender] >= wad);

        balanceOf[msg.sender] -= wad;

        msg.sender.transfer(wad);

    emit Withdrawal(msg.sender, wad);

    }

    function totalSupply() public view returns (uint) {

        return address(this).balance;

    }

    function approve(address guy, uint wad) public returns (bool) {

        allowance[msg.sender][guy] = wad;

       emit Approval(msg.sender, guy, wad);

        return true;

    }

    function transfer(address dst, uint wad) public returns (bool) {

        return transferFrom(msg.sender, dst, wad);

    }

    function transferFrom(address src, address dst, uint wad)

        public

        returns (bool)

    {

        require(balanceOf[src] >= wad);

        if (src != msg.sender && allowance[src][msg.sender] != uint(-1)) {

            require(allowance[src][msg.sender] >= wad);

            allowance[src][msg.sender] -= wad;

        }

        balanceOf[src] -= wad;

        balanceOf[dst] += wad;

    emit Transfer(src, dst, wad);

        return true;

    }

}

Npm install @uniswap/lib @uniswap/v2-core

Update solidity version to 0.6.6 from truffle-config.js

Update the version of WETH.sol contract to 0.6.6 and also change some syntax like put emit and so on

goto uniswapv2router02.sol and comment from line 318 to 400

Now goto the interface folder and comment out from 23 to 43  
We did this due to gas currently we don’t need these function that’s why we comment out them  
At the end noe run truffle migrate.

This is the complete procedure for the deployment of core and periphery contract deployment