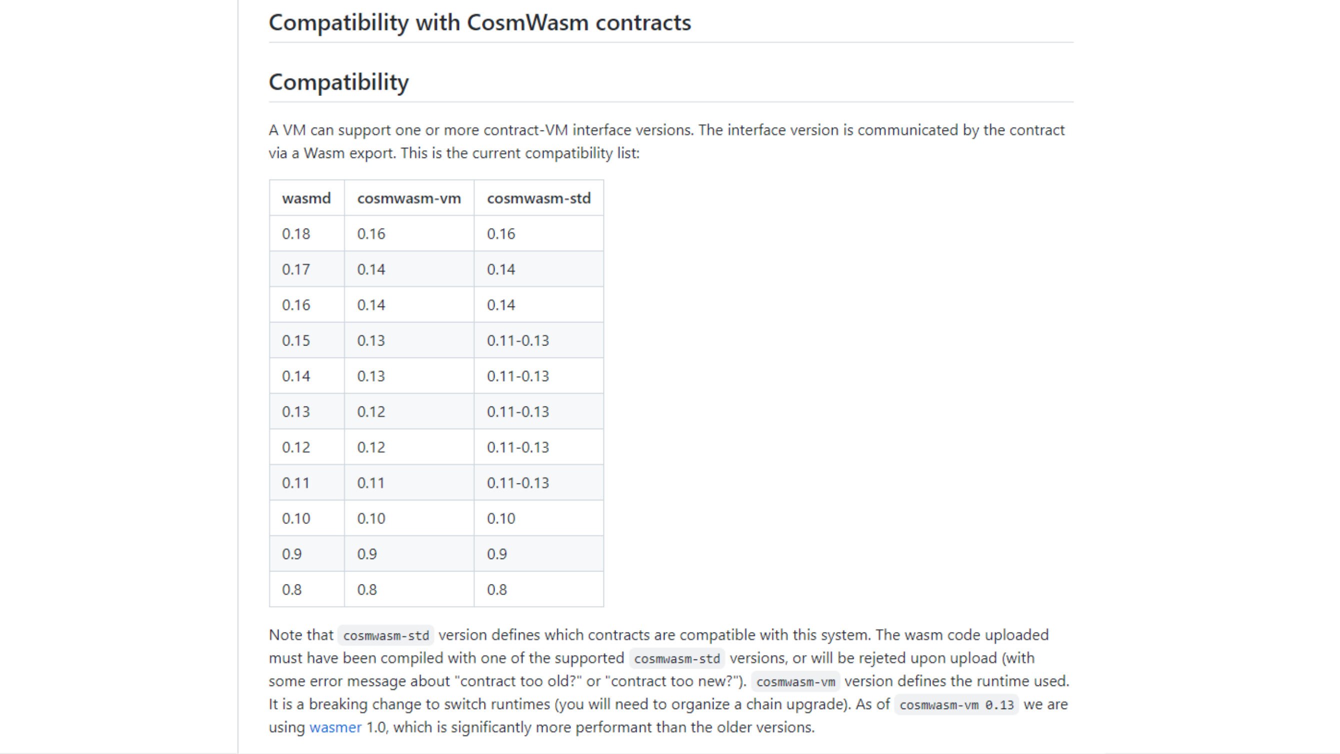
ARCHWAY CLI 1 - COSMWASM TEMPLATE



Compatibility with CosmWasm contracts

Compatibility

A VM can support one or more contract-VM interface versions. The interface version is communicated by the contract via a Wasm export. This is the current compatibility list:

wasmd	cosmwasm-vm	cosmwasm-std
0.18	0.16	0.16
0.17	0.14	0.14
0.16	0.14	0.14
0.15	0.13	0.11-0.13
0.14	0.13	0.11-0.13
0.13	0.12	0.11-0.13
0.12	0.12	0.11-0.13
0.11	0.11	0.11-0.13
0.10	0.10	0.10
0.9	0.9	0.9
0.8	0.8	0.8

Note that `cosmwasm-std` version defines which contracts are compatible with this system. The wasm code uploaded must have been compiled with one of the supported `cosmwasm-std` versions, or will be rejeted upon upload (with some error message about "contract too old?" or "contract too new?"). `cosmwasm-vm` version defines the runtime used. It is a breaking change to switch runtimes (you will need to organize a chain upgrade). As of `cosmwasm-vm 0.13` we are using wasmer 1.0, which is significantly more performant than the older versions.

SCRIPT

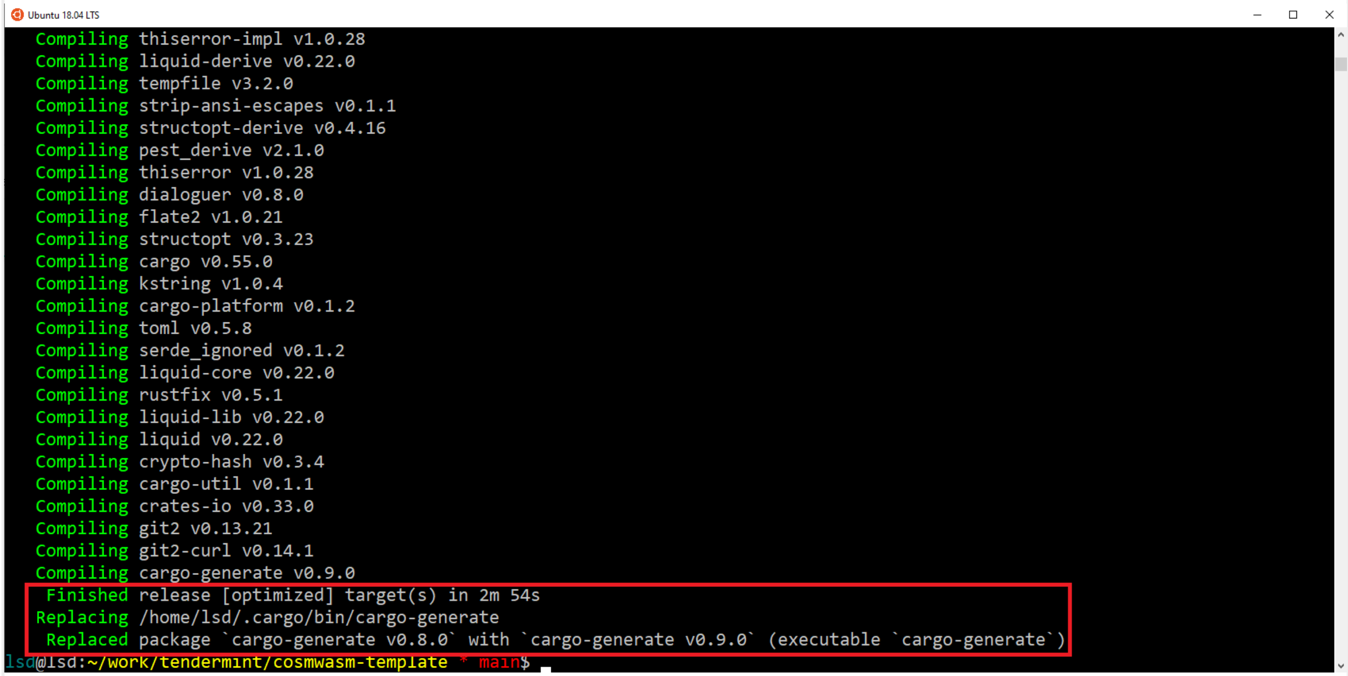
N/A

ACTION

Before beginning, it may be wise to check your compatability settings.

The above presents a compatibility chart between CosmWasm and CosmWasm contracts

Pre-requisites for this tutorial:
- rustc
- cargo
- wasmd
- docker



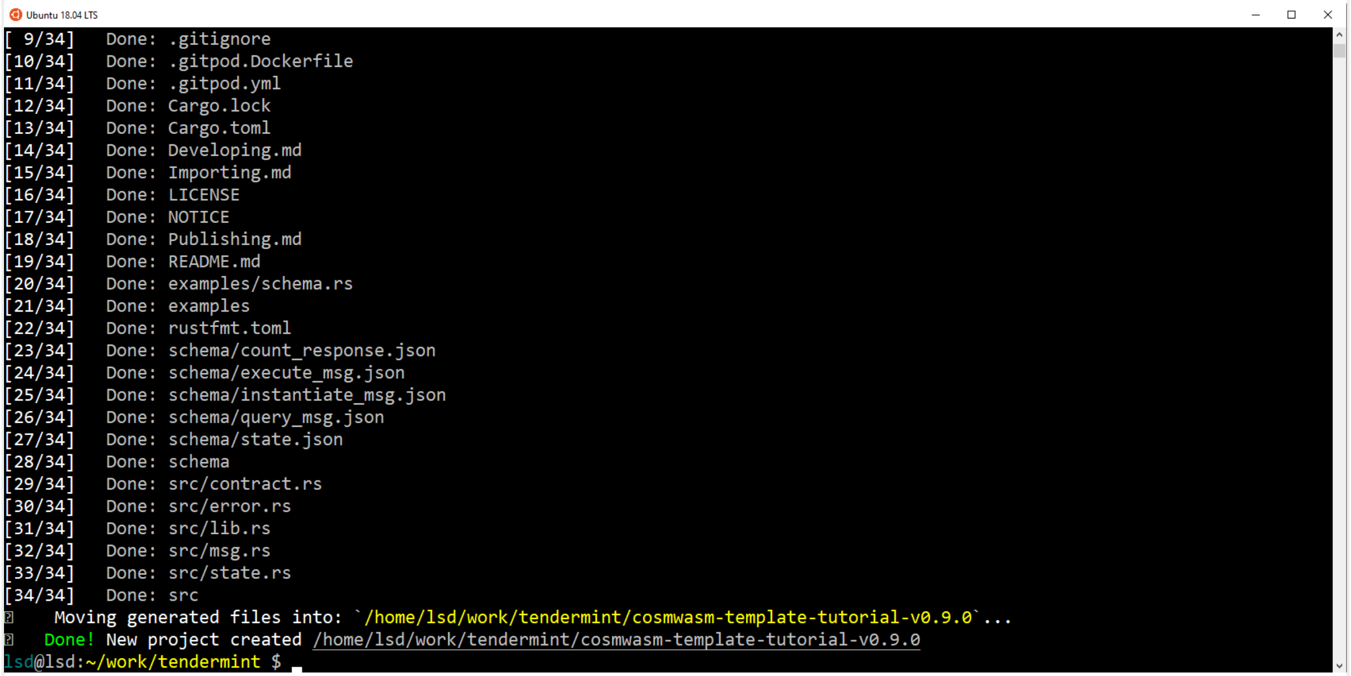
```
   Compiling thiserror-impl v1.0.28
   Compiling liquid-derive v0.22.0
   Compiling tempfile v3.2.0
   Compiling strip-ansi-escapes v0.1.1
   Compiling structopt-derive v0.4.16
   Compiling pest_derive v2.1.0
   Compiling thiserror v1.0.28
   Compiling dialoguer v0.8.0
   Compiling flate2 v1.0.21
   Compiling structopt v0.3.23
   Compiling cargo v0.55.0
   Compiling kstring v1.0.4
   Compiling cargo-platform v0.1.2
   Compiling toml v0.5.8
   Compiling serde_ignored v0.1.2
   Compiling liquid-core v0.22.0
   Compiling rustfix v0.5.1
   Compiling liquid-lib v0.22.0
   Compiling liquid v0.22.0
   Compiling crypto-hash v0.3.4
   Compiling cargo-util v0.1.1
   Compiling crates-io v0.33.0
   Compiling git2 v0.13.21
   Compiling git2-curl v0.14.1
   Compiling cargo-generate v0.9.0
    Finished release [optimized] target(s) in 2m 54s
   Replacing /home/lsd/.cargo/bin/cargo-generate
    Replaced package `cargo-generate v0.8.0` with `cargo-generate v0.9.0` (executable `cargo-generate`)
lsd@lsd:~/work/tendermint/cosmwasm-template * main$
```

SCRIPT

$ cargo install cargo-generate --features vendored-openssl

ACTION

Install or update `cargo-generate` dependency



```
[ 9/34]   Done: .gitignore
[10/34]   Done: .gitpod.Dockerfile
[11/34]   Done: .gitpod.yml
[12/34]   Done: Cargo.lock
[13/34]   Done: Cargo.toml
[14/34]   Done: Developing.md
[15/34]   Done: Importing.md
[16/34]   Done: LICENSE
[17/34]   Done: NOTICE
[18/34]   Done: Publishing.md
[19/34]   Done: README.md
[20/34]   Done: examples/schema.rs
[21/34]   Done: examples
[22/34]   Done: rustfmt.toml
[23/34]   Done: schema/count_response.json
[24/34]   Done: schema/execute_msg.json
[25/34]   Done: schema/instantiate_msg.json
[26/34]   Done: schema/query_msg.json
[27/34]   Done: schema/state.json
[28/34]   Done: schema
[29/34]   Done: src/contract.rs
[30/34]   Done: src/error.rs
[31/34]   Done: src/lib.rs
[32/34]   Done: src/msg.rs
[33/34]   Done: src/state.rs
[34/34]   Done: src
□    Moving generated files into: `/home/lsd/work/tendermint/cosmwasm-template-tutorial-v0.9.0`...
□   Done! New project created /home/lsd/work/tendermint/cosmwasm-template-tutorial-v0.9.0
lsd@lsd:~/work/tendermint $
```

SCRIPT

3

$ cargo generate --git https://github.com/CosmWasm/cosmwasm-template.git --name YOUR_PROJECT_NAME

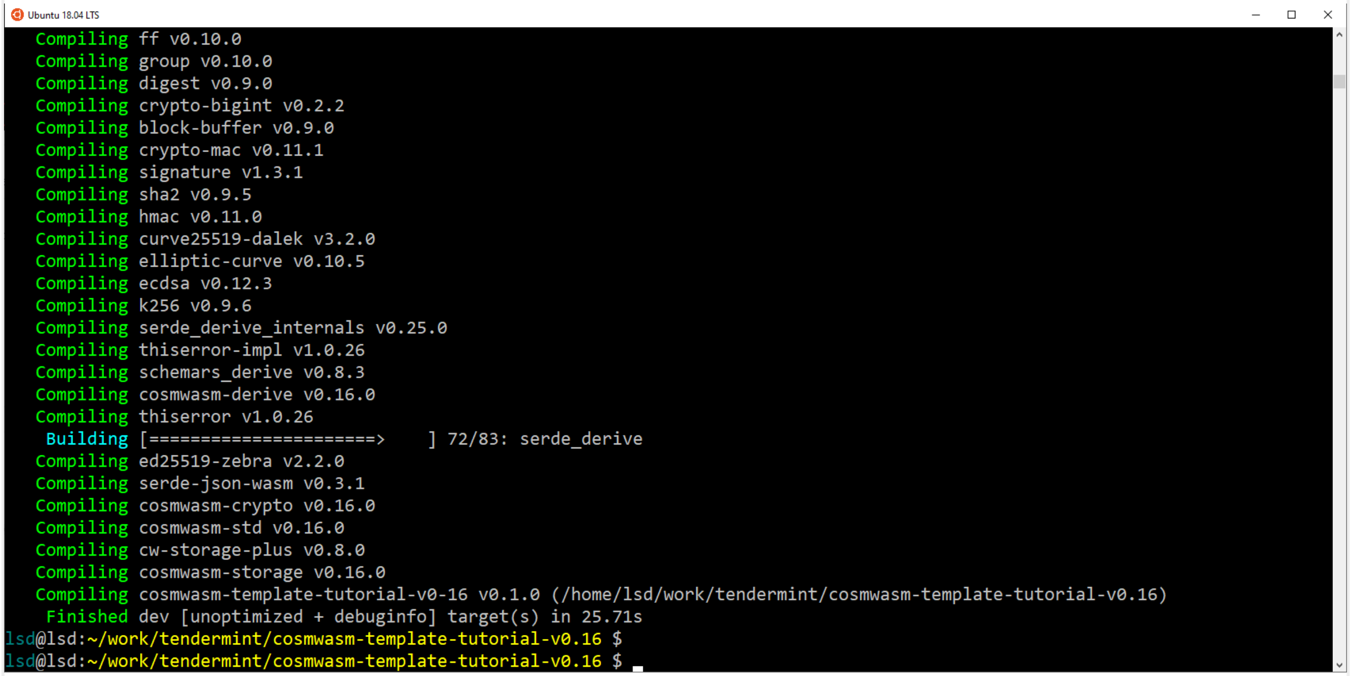
(Or to use a specific version of CosmWasm for contract compilation:

$ cargo generate --git https://github.com/CosmWasm/cosmwasm-template.git --branch 0.14 --name YOUR_PROJECT_NAME)

ACTION

Install a blank project template for creating a DApp contract.

To create your project with a specific version of CosmWasm you can pass the flag and value; e.g. `--branch BRANCH_VERSION`



```
   Compiling ff v0.10.0
   Compiling group v0.10.0
   Compiling digest v0.9.0
   Compiling crypto-bigint v0.2.2
   Compiling block-buffer v0.9.0
   Compiling crypto-mac v0.11.1
   Compiling signature v1.3.1
   Compiling sha2 v0.9.5
   Compiling hmac v0.11.0
   Compiling curve25519-dalek v3.2.0
   Compiling elliptic-curve v0.10.5
   Compiling ecdsa v0.12.3
   Compiling k256 v0.9.6
   Compiling serde_derive_internals v0.25.0
   Compiling thiserror-impl v1.0.26
   Compiling schemars_derive v0.8.3
   Compiling cosmwasm-derive v0.16.0
   Compiling thiserror v1.0.26
    Building [======================>    ] 72/83: serde_derive
   Compiling ed25519-zebra v2.2.0
   Compiling serde-json-wasm v0.3.1
   Compiling cosmwasm-crypto v0.16.0
   Compiling cosmwasm-std v0.16.0
   Compiling cw-storage-plus v0.8.0
   Compiling cosmwasm-storage v0.16.0
   Compiling cosmwasm-template-tutorial-v0-16 v0.1.0 (/home/lsd/work/tendermint/cosmwasm-template-tutorial-v0.16)
    Finished dev [unoptimized + debuginfo] target(s) in 25.71s
lsd@lsd:~/work/tendermint/cosmwasm-template-tutorial-v0.16 $
lsd@lsd:~/work/tendermint/cosmwasm-template-tutorial-v0.16 $
```

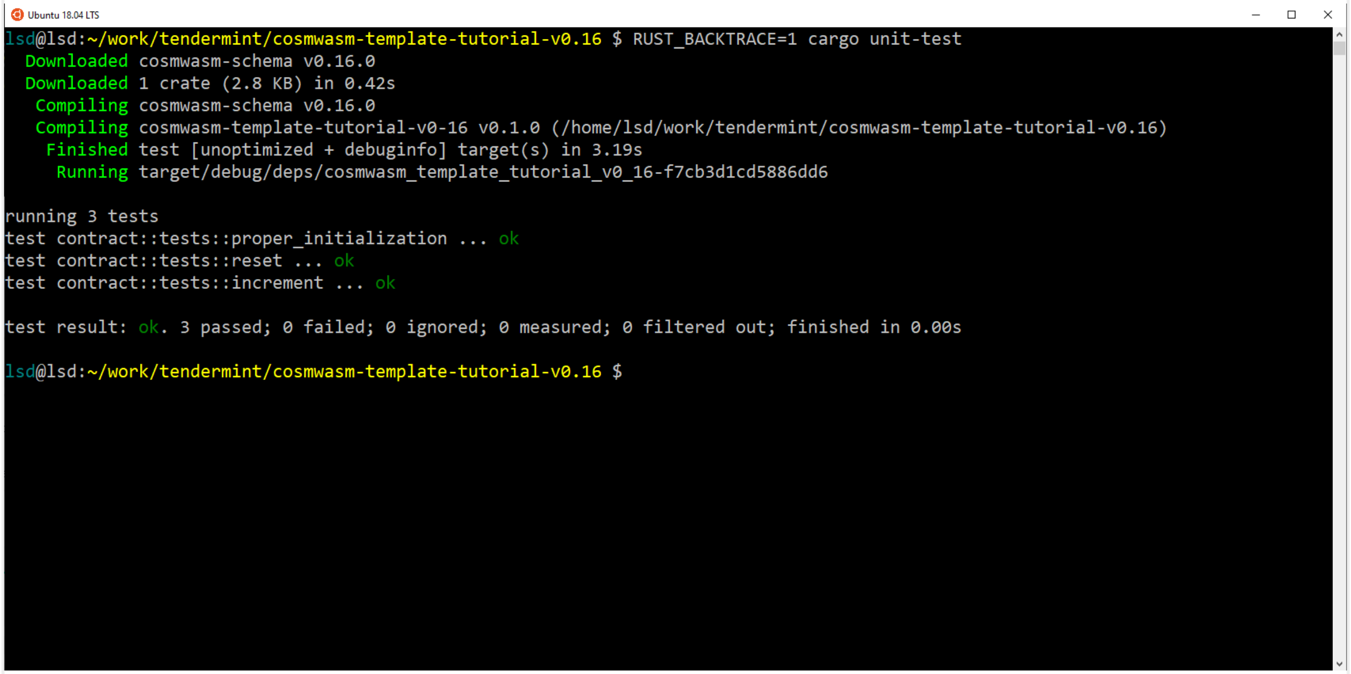
4

SCRIPT

```
$ cd YOUR_PROJECT_NAME
$ cargo build
```

ACTION

Compile your new project



```
lsd@lsd:~/work/tendermint/cosmwasm-template-tutorial-v0.16 $ RUST_BACKTRACE=1 cargo unit-test
  Downloaded cosmwasm-schema v0.16.0
  Downloaded 1 crate (2.8 KB) in 0.42s
   Compiling cosmwasm-schema v0.16.0
   Compiling cosmwasm-template-tutorial-v0-16 v0.1.0 (/home/lsd/work/tendermint/cosmwasm-template-tutorial-v0.16)
    Finished test [unoptimized + debuginfo] target(s) in 3.19s
     Running target/debug/deps/cosmwasm_template_tutorial_v0_16-f7cb3d1cd5886dd6

running 3 tests
test contract::tests::proper_initialization ... ok
test contract::tests::reset ... ok
test contract::tests::increment ... ok

test result: ok. 3 passed; 0 failed; 0 ignored; 0 measured; 0 filtered out; finished in 0.00s

lsd@lsd:~/work/tendermint/cosmwasm-template-tutorial-v0.16 $
```

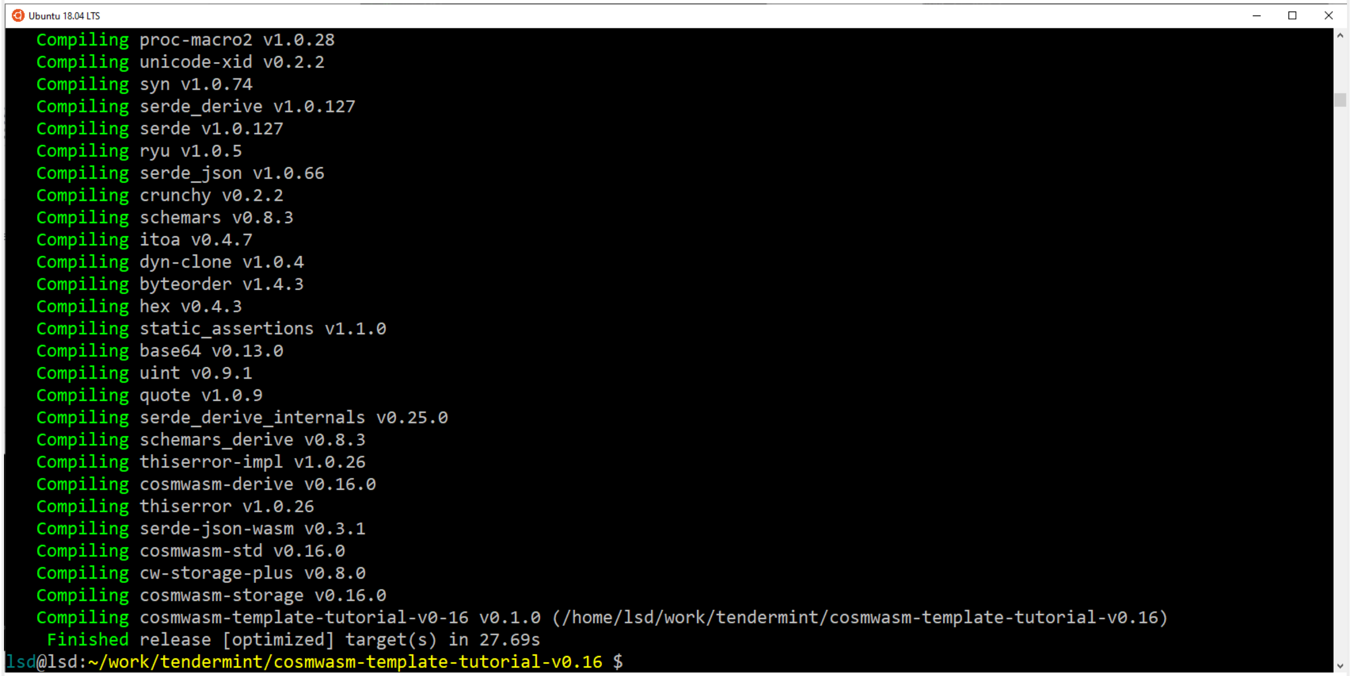
5

SCRIPT

$ RUST_BACKTRACE=1 cargo unit-test

ACTION

Unit test build



```
   Compiling proc-macro2 v1.0.28
   Compiling unicode-xid v0.2.2
   Compiling syn v1.0.74
   Compiling serde_derive v1.0.127
   Compiling serde v1.0.127
   Compiling ryu v1.0.5
   Compiling serde_json v1.0.66
   Compiling crunchy v0.2.2
   Compiling schemars v0.8.3
   Compiling itoa v0.4.7
   Compiling dyn-clone v1.0.4
   Compiling byteorder v1.4.3
   Compiling hex v0.4.3
   Compiling static_assertions v1.1.0
   Compiling base64 v0.13.0
   Compiling uint v0.9.1
   Compiling quote v1.0.9
   Compiling serde_derive_internals v0.25.0
   Compiling schemars_derive v0.8.3
   Compiling thiserror-impl v1.0.26
   Compiling cosmwasm-derive v0.16.0
   Compiling thiserror v1.0.26
   Compiling serde-json-wasm v0.3.1
   Compiling cosmwasm-std v0.16.0
   Compiling cw-storage-plus v0.8.0
   Compiling cosmwasm-storage v0.16.0
   Compiling cosmwasm-template-tutorial-v0-16 v0.1.0 (/home/lsd/work/tendermint/cosmwasm-template-tutorial-v0.16)
    Finished release [optimized] target(s) in 27.69s
lsd@lsd:~/work/tendermint/cosmwasm-template-tutorial-v0.16 $
```

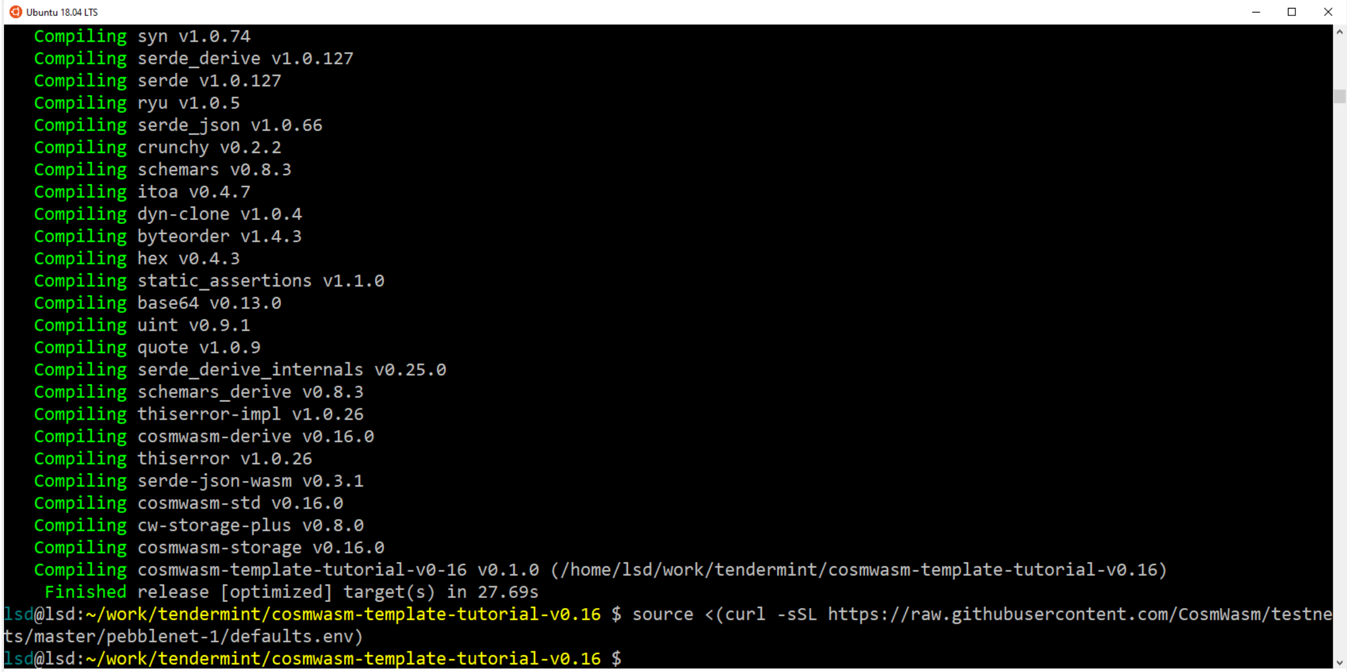
6

SCRIPT

$ cargo wasm

ACTION

Produce an unoptimized wasm executable



```
   Compiling syn v1.0.74
   Compiling serde_derive v1.0.127
   Compiling serde v1.0.127
   Compiling ryu v1.0.5
   Compiling serde_json v1.0.66
   Compiling crunchy v0.2.2
   Compiling schemars v0.8.3
   Compiling itoa v0.4.7
   Compiling dyn-clone v1.0.4
   Compiling byteorder v1.4.3
   Compiling hex v0.4.3
   Compiling static_assertions v1.1.0
   Compiling base64 v0.13.0
   Compiling uint v0.9.1
   Compiling quote v1.0.9
   Compiling serde_derive_internals v0.25.0
   Compiling schemars_derive v0.8.3
   Compiling thiserror-impl v1.0.26
   Compiling cosmwasm-derive v0.16.0
   Compiling thiserror v1.0.26
   Compiling serde-json-wasm v0.3.1
   Compiling cosmwasm-std v0.16.0
   Compiling cw-storage-plus v0.8.0
   Compiling cosmwasm-storage v0.16.0
   Compiling cosmwasm-template-tutorial-v0-16 v0.1.0 (/home/lsd/work/tendermint/cosmwasm-template-tutorial-v0.16)
    Finished release [optimized] target(s) in 27.69s
lsd@lsd:~/work/tendermint/cosmwasm-template-tutorial-v0.16 $ source <(curl -sSL https://raw.githubusercontent.com/CosmWasm/testne
ts/master/pebblenet-1/defaults.env)
lsd@lsd:~/work/tendermint/cosmwasm-template-tutorial-v0.16 $
```

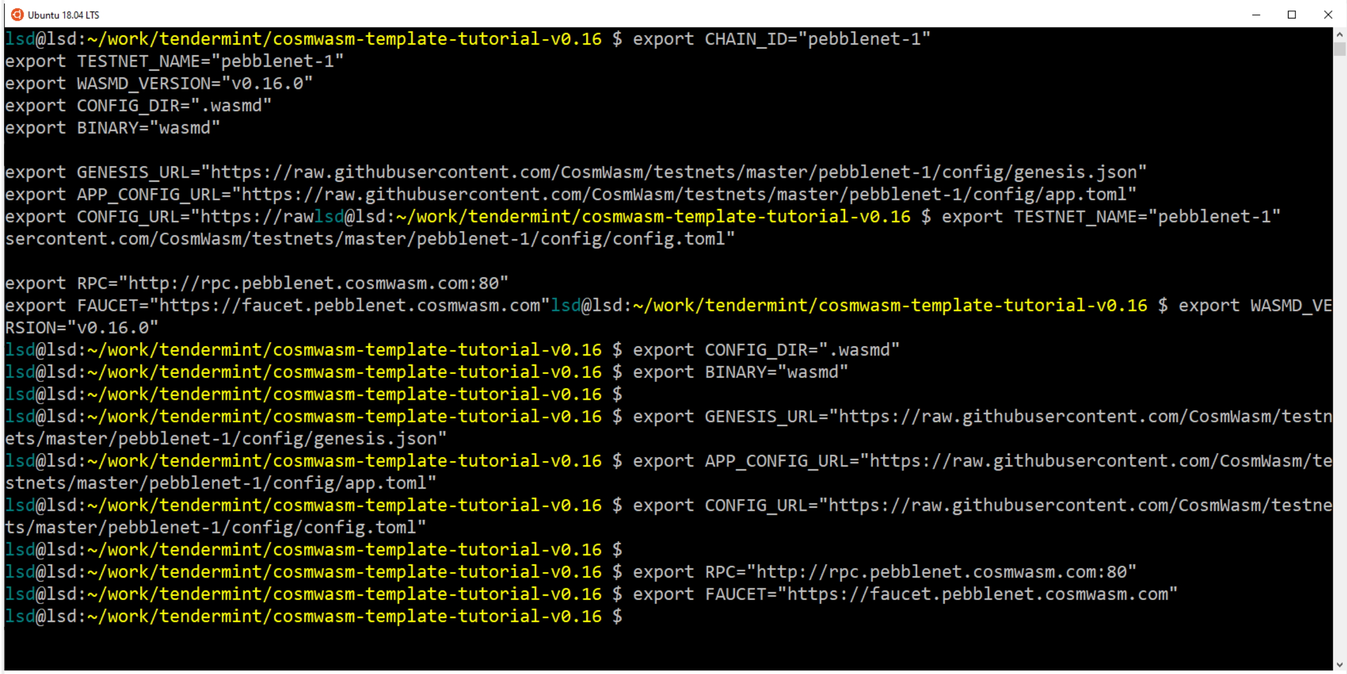
7

SCRIPT

$ source <(curl -sSL https://raw.githubusercontent.com/CosmWasm/testnets/master/pebblenet-1/defaults.env)

ACTION

Source the pebble testnet configuration



```
lsd@lsd:~/work/tendermint/cosmwasm-template-tutorial-v0.16 $ export CHAIN_ID="pebblenet-1"
export TESTNET_NAME="pebblenet-1"
export WASMD_VERSION="v0.16.0"
export CONFIG_DIR=".wasmd"
export BINARY="wasmd"

export GENESIS_URL="https://raw.githubusercontent.com/CosmWasm/testnets/master/pebblenet-1/config/genesis.json"
export APP_CONFIG_URL="https://raw.githubusercontent.com/CosmWasm/testnets/master/pebblenet-1/config/app.toml"
export CONFIG_URL="https://rawlsd@lsd:~/work/tendermint/cosmwasm-template-tutorial-v0.16 $ export TESTNET_NAME="pebblenet-1"
sercontent.com/CosmWasm/testnets/master/pebblenet-1/config/config.toml"

export RPC="http://rpc.pebblenet.cosmwasm.com:80"
export FAUCET="https://faucet.pebblenet.cosmwasm.com"lsd@lsd:~/work/tendermint/cosmwasm-template-tutorial-v0.16 $ export WASMD_VE
RSION="v0.16.0"
lsd@lsd:~/work/tendermint/cosmwasm-template-tutorial-v0.16 $ export CONFIG_DIR=".wasmd"
lsd@lsd:~/work/tendermint/cosmwasm-template-tutorial-v0.16 $ export BINARY="wasmd"
lsd@lsd:~/work/tendermint/cosmwasm-template-tutorial-v0.16 $
lsd@lsd:~/work/tendermint/cosmwasm-template-tutorial-v0.16 $ export GENESIS_URL="https://raw.githubusercontent.com/CosmWasm/testn
ets/master/pebblenet-1/config/genesis.json"
lsd@lsd:~/work/tendermint/cosmwasm-template-tutorial-v0.16 $ export APP_CONFIG_URL="https://raw.githubusercontent.com/CosmWasm/te
stnets/master/pebblenet-1/config/app.toml"
lsd@lsd:~/work/tendermint/cosmwasm-template-tutorial-v0.16 $ export CONFIG_URL="https://raw.githubusercontent.com/CosmWasm/testne
ts/master/pebblenet-1/config/config.toml"
lsd@lsd:~/work/tendermint/cosmwasm-template-tutorial-v0.16 $
lsd@lsd:~/work/tendermint/cosmwasm-template-tutorial-v0.16 $ export RPC="http://rpc.pebblenet.cosmwasm.com:80"
lsd@lsd:~/work/tendermint/cosmwasm-template-tutorial-v0.16 $ export FAUCET="https://faucet.pebblenet.cosmwasm.com"
lsd@lsd:~/work/tendermint/cosmwasm-template-tutorial-v0.16 $
```

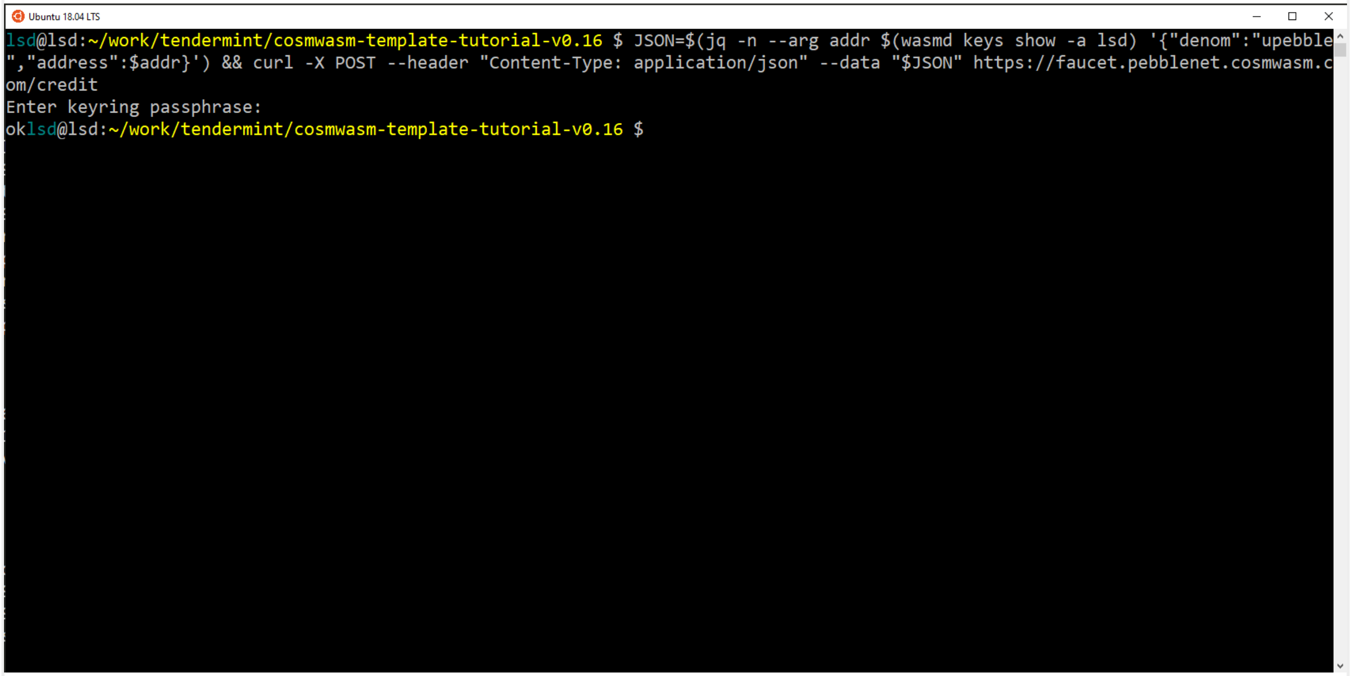
SCRIPT

8

```
$ export CHAIN_ID="pebblenet-1"
$ export TESTNET_NAME="pebblenet-1"
$ export WASMD_VERSION="v0.16.0"
$ export CONFIG_DIR=".wasmd"
$ export BINARY="wasmd"
$ export GENESIS_URL="https://raw.githubusercontent.com/CosmWasm/testnets/master/pebblenet-1/config/genesis.json"
$ export APP_CONFIG_URL="https://raw.githubusercontent.com/CosmWasm/testnets/master/pebblenet-1/config/app.toml"
$ export  CONFIG_URL="https://raw.githubusercontent.com/CosmWasm/testnets/master/pebblenet-1/config/config.toml"
$ export RPC="https://rpc.pebblenet.cosmwasm.com:443"
$ export FAUCET="https://faucet.pebblenet.cosmwasm.com"
```

ACTION

Testnet environment settings



```
lsd@lsd:~/work/tendermint/cosmwasm-template-tutorial-v0.16 $ JSON=$(jq -n --arg addr $(wasmd keys show -a lsd) '{"denom":"upebble
","address":$addr}') && curl -X POST --header "Content-Type: application/json" --data "$JSON" https://faucet.pebblenet.cosmwasm.c
om/credit
Enter keyring passphrase:
oklsd@lsd:~/work/tendermint/cosmwasm-template-tutorial-v0.16 $
```

9

SCRIPT

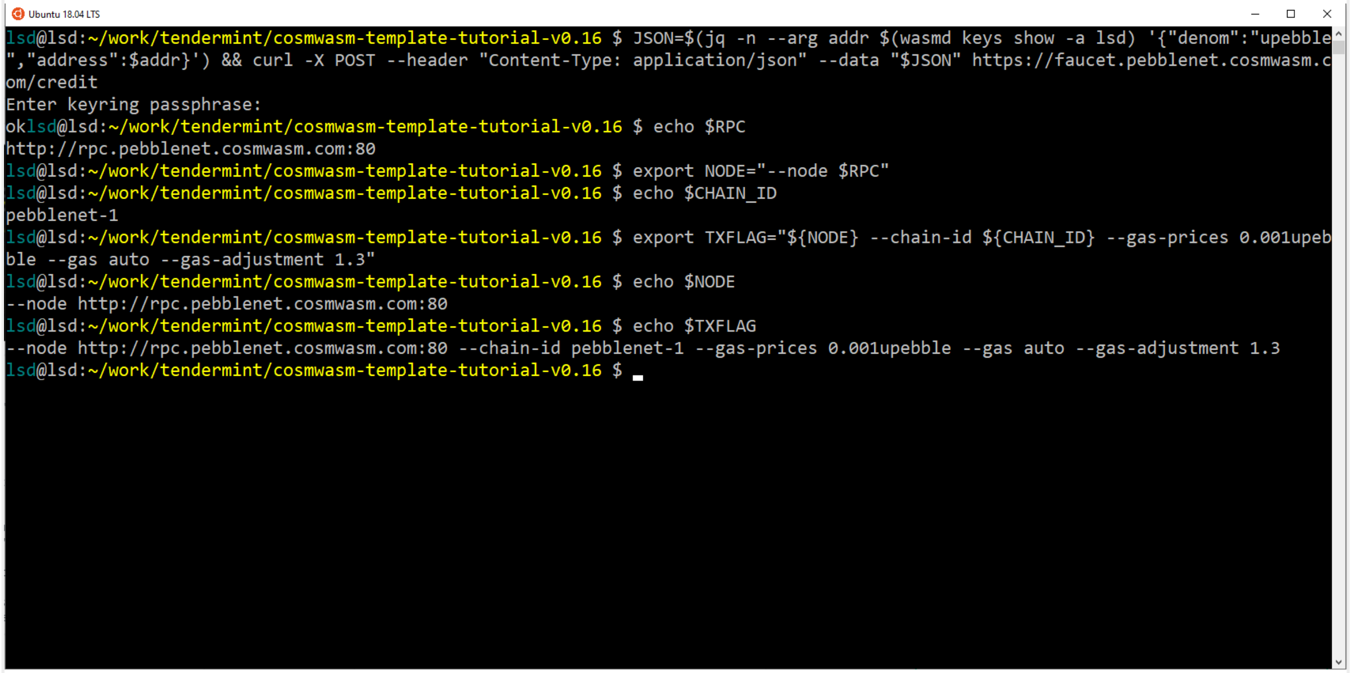
```
$ JSON=$(jq -n --arg addr $(wasmd keys show -a YOUR_WALLET_NAME)
'{"denom":"upebble","address":$addr}') && curl -X POST --header "Content-Type: application/json" --data
"$JSON" https://faucet.pebblenet.cosmwasm.com/credit
```

ACTION

Request testnet funds

(If you don't already have a wallet setup in wasmd CLI, run this command first):

```
$ wasmd keys add YOUR_WALLET_NAME
```



```
lsd@lsd:~/work/tendermint/cosmwasm-template-tutorial-v0.16 $ JSON=$(jq -n --arg addr $(wasmd keys show -a lsd) '{"denom":"upebble
","address":$addr}') && curl -X POST --header "Content-Type: application/json" --data "$JSON" https://faucet.pebblenet.cosmwasm.c
om/credit
Enter keyring passphrase:
oklsd@lsd:~/work/tendermint/cosmwasm-template-tutorial-v0.16 $ echo $RPC
http://rpc.pebblenet.cosmwasm.com:80
lsd@lsd:~/work/tendermint/cosmwasm-template-tutorial-v0.16 $ export NODE="--node $RPC"
lsd@lsd:~/work/tendermint/cosmwasm-template-tutorial-v0.16 $ echo $CHAIN_ID
pebblenet-1
lsd@lsd:~/work/tendermint/cosmwasm-template-tutorial-v0.16 $ export TXFLAG="${NODE} --chain-id ${CHAIN_ID} --gas-prices 0.001upeb
ble --gas auto --gas-adjustment 1.3"
lsd@lsd:~/work/tendermint/cosmwasm-template-tutorial-v0.16 $ echo $NODE
--node http://rpc.pebblenet.cosmwasm.com:80
lsd@lsd:~/work/tendermint/cosmwasm-template-tutorial-v0.16 $ echo $TXFLAG
--node http://rpc.pebblenet.cosmwasm.com:80 --chain-id pebblenet-1 --gas-prices 0.001upebble --gas auto --gas-adjustment 1.3
lsd@lsd:~/work/tendermint/cosmwasm-template-tutorial-v0.16 $ _
```

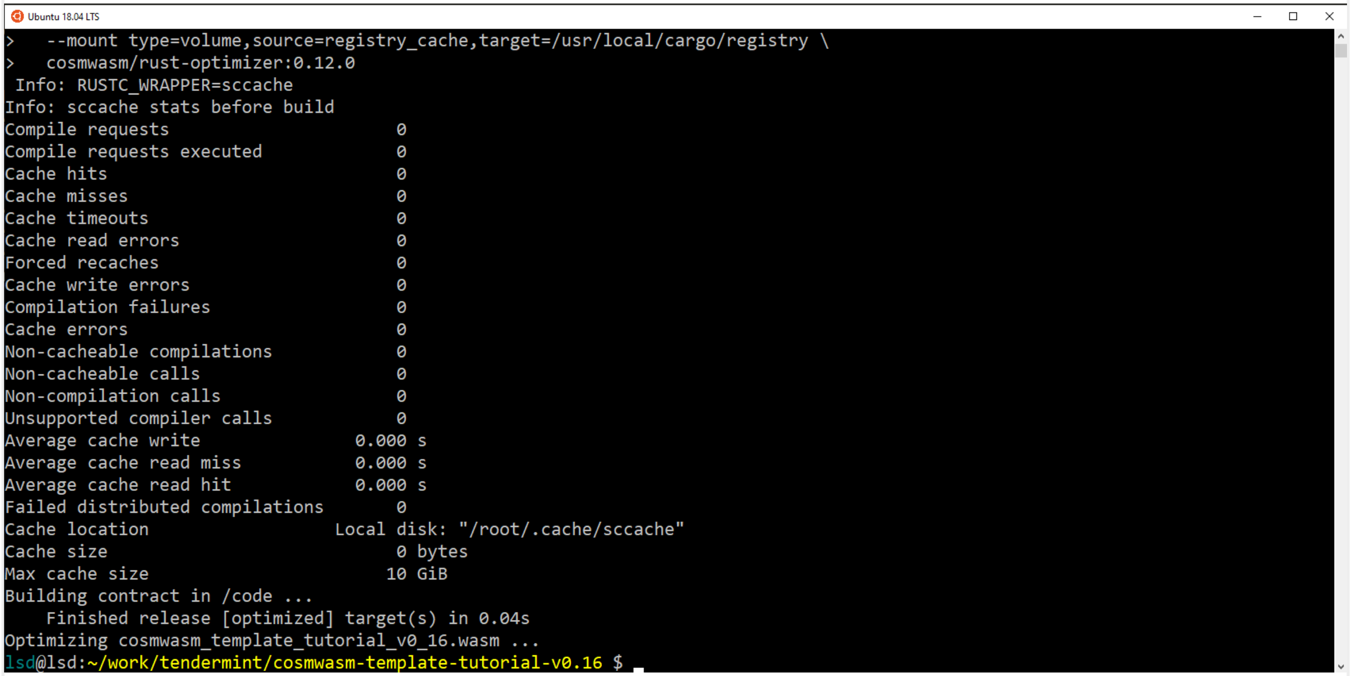
SCRIPT

$ export NODE="--node $RPC"
$ export TXFLAG="${NODE} --chain-id ${CHAIN_ID} --gas-prices 0.001upebble --gas auto --gas-adjustment 1.3"

ACTION

(* Optional *) wasmd environment settings

Setting these params saves a lot of typing, and we'll be using them as shorthand later



```
>   --mount type=volume,source=registry_cache,target=/usr/local/cargo/registry \
>   cosmwasm/rust-optimizer:0.12.0
 Info: RUSTC_WRAPPER=sccache
Info: sccache stats before build
Compile requests                      0
Compile requests executed             0
Cache hits                            0
Cache misses                          0
Cache timeouts                        0
Cache read errors                     0
Forced recaches                       0
Cache write errors                    0
Compilation failures                  0
Cache errors                          0
Non-cacheable compilations            0
Non-cacheable calls                   0
Non-compilation calls                 0
Unsupported compiler calls            0
Average cache write               0.000 s
Average cache read miss           0.000 s
Average cache read hit            0.000 s
Failed distributed compilations       0
Cache location                  Local disk: "/root/.cache/sccache"
Cache size                            0 bytes
Max cache size                       10 GiB
Building contract in /code ...
    Finished release [optimized] target(s) in 0.04s
Optimizing cosmwasm_template_tutorial_v0_16.wasm ...
lsd@lsd:~/work/tendermint/cosmwasm-template-tutorial-v0.16 $ _
```

11

SCRIPT

```
$ export pwd=YOUR_WASM_KEYCHAIN_PASSWORD
$ docker run --rm -v "$(pwd)":/code \
  --mount type=volume,source="$(basename "$(pwd)")_cache",target=/code/target \
  --mount type=volume,source=registry_cache,target=/usr/local/cargo/registry \
  cosmwasm/rust-optimizer:0.12.0
```

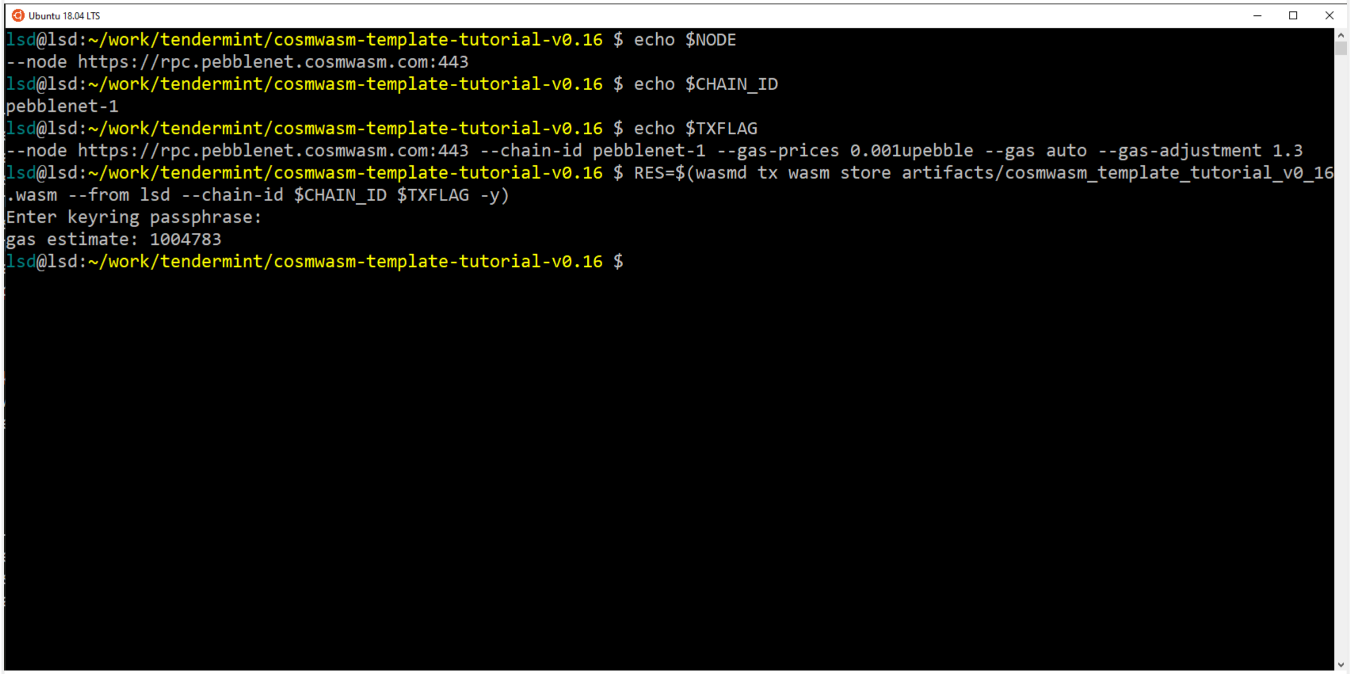
ACTION

Produce an optimized CosmWasm wasm executable that can be deployed on-chain

NOTE:
Docker won't stop automatically and when you see the last message check for the build output in /artifacts.

If Docker finished building the optimized wasm file, it will be in that path and you can safely stop the container. Open another terminal prompt to stop the Docker container, this will exit you from the hanging state of your original terminal. E.g.:

```
$ docker ps -q
```



```
lsd@lsd:~/work/tendermint/cosmwasm-template-tutorial-v0.16 $ echo $NODE
--node https://rpc.pebblenet.cosmwasm.com:443
lsd@lsd:~/work/tendermint/cosmwasm-template-tutorial-v0.16 $ echo $CHAIN_ID
pebblenet-1
lsd@lsd:~/work/tendermint/cosmwasm-template-tutorial-v0.16 $ echo $TXFLAG
--node https://rpc.pebblenet.cosmwasm.com:443 --chain-id pebblenet-1 --gas-prices 0.001upebble --gas auto --gas-adjustment 1.3
lsd@lsd:~/work/tendermint/cosmwasm-template-tutorial-v0.16 $ RES=$(wasmd tx wasm store artifacts/cosmwasm_template_tutorial_v0_16
.wasm --from lsd --chain-id $CHAIN_ID $TXFLAG -y)
Enter keyring passphrase:
gas estimate: 1004783
lsd@lsd:~/work/tendermint/cosmwasm-template-tutorial-v0.16 $
```

SCRIPT

```
$ RES=$(wasmd tx wasm store artifacts/YOUR_DOCKER_WASM_OUTPUT_FILE.wasm --from
YOUR_WALLET_NAME --chain-id $CHAIN_ID $TXFLAG -y)
$ CODE_ID=$(echo $RES | jq -r '.logs[0].events[-1].attributes[0].value')
$ wasmd query wasm list-contract-by-code $CODE_ID $NODE --output json
$ wasmd query wasm code $CODE_ID $NODE download.wasm
$ diff artifacts/YOUR_DOCKER_WASM_OUTPUT_FILE.wasm download.wasm
```

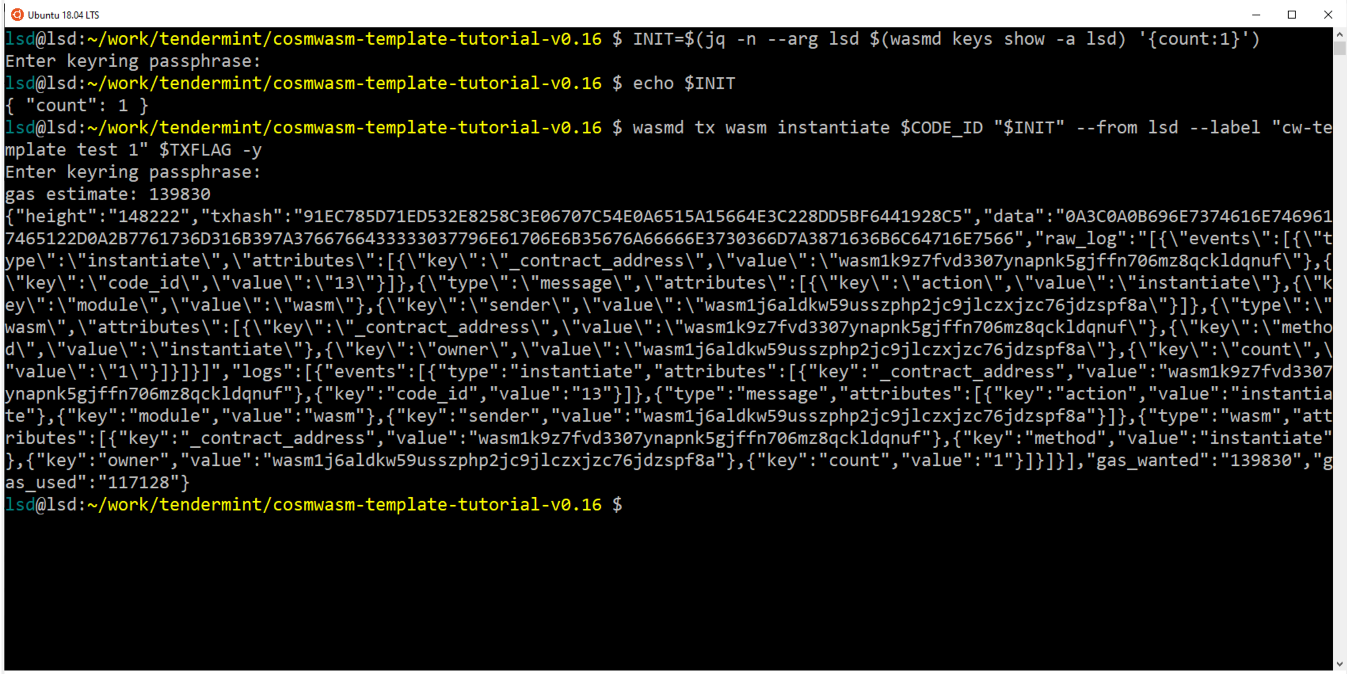
ACTION

Produce archived executable on CosmWasm network

Using an archive of your build reduces on-chain size and gas cost for deployment.

If we query the chain we still won't find any instance of our contract (returns null pagination) because we haven't yet deployed an instance.

We can also now query the network for our code and check it matches our locally built wasm archive.



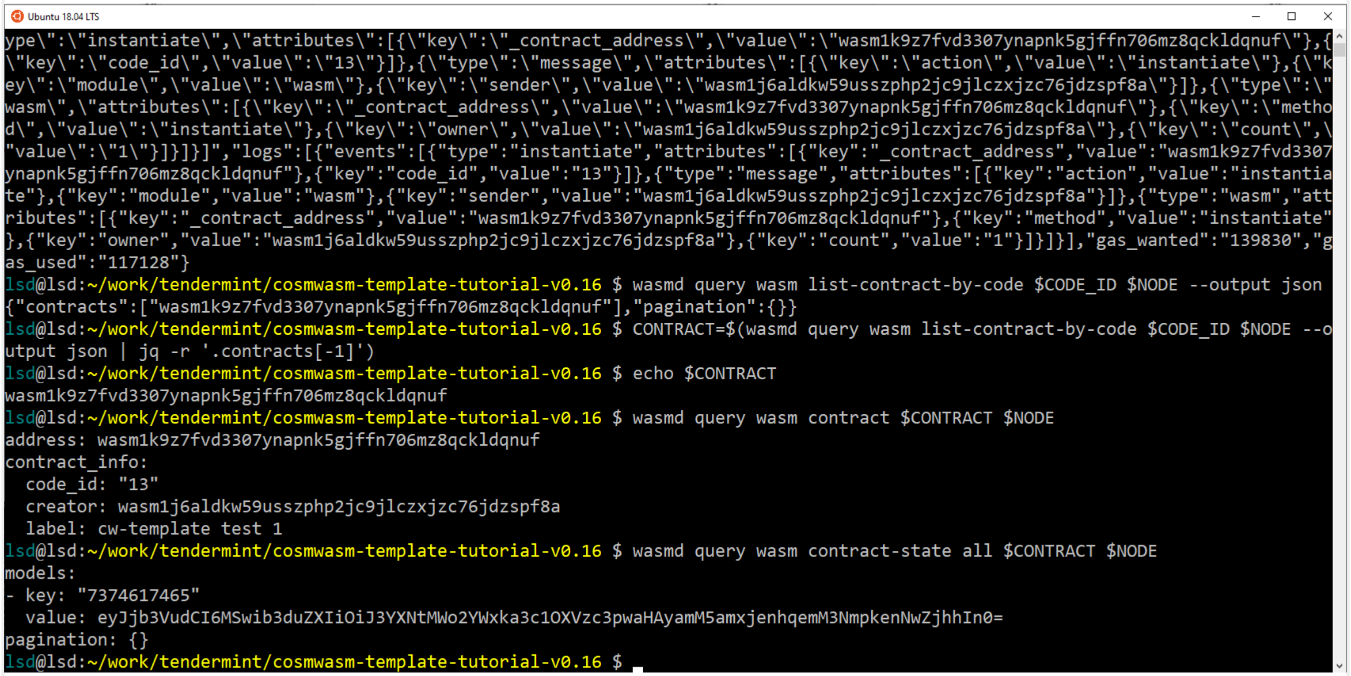
```
lsd@lsd:~/work/tendermint/cosmwasm-template-tutorial-v0.16 $ INIT=$(jq -n --arg lsd $(wasmd keys show -a lsd) '{count:1}')
Enter keyring passphrase:
lsd@lsd:~/work/tendermint/cosmwasm-template-tutorial-v0.16 $ echo $INIT
{ "count": 1 }
lsd@lsd:~/work/tendermint/cosmwasm-template-tutorial-v0.16 $ wasmd tx wasm instantiate $CODE_ID "$INIT" --from lsd --label "cw-te
mplate test 1" $TXFLAG -y
Enter keyring passphrase:
gas estimate: 139830
{"height":"148222","txhash":"91EC785D71ED532E8258C3E06707C54E0A6515A15664E3C228DD5BF6441928C5","data":"0A3C0A0B696E7374616E746961
7465122D0A2B7761736D316B397A376676643333037796E61706E6B35676A66666E3730366D7A3871636B6C64716E7566","raw_log":"[{\"events\":[{\"t
ype\":\"instantiate\",\"attributes\":[{\"key\":\"_contract_address\",\"value\":\"wasm1k9z7fvd3307ynapnk5gjffn706mz8qckldqnuf\"},{
\"key\":\"code_id\",\"value\":\"13\"}]},{\"type\":\"message\",\"attributes\":[{\"key\":\"action\",\"value\":\"instantiate\"},{\"k
ey\":\"module\",\"value\":\"wasm\"},{\"key\":\"sender\",\"value\":\"wasm1j6aldkw59usszphp2jc9jlczxjzc76jdzspf8a\"}]},{\"type\":\"
wasm\",\"attributes\":[{\"key\":\"_contract_address\",\"value\":\"wasm1k9z7fvd3307ynapnk5gjffn706mz8qckldqnuf\"},{\"key\":\"metho
d\",\"value\":\"instantiate\"},{\"key\":\"owner\",\"value\":\"wasm1j6aldkw59usszphp2jc9jlczxjzc76jdzspf8a\"},{\"key\":\"count\",\
"value\":\"1\"}]}]}]","logs":[{"events":[{"type":"instantiate","attributes":[{"key":"_contract_address","value":"wasm1k9z7fvd3307
ynapnk5gjffn706mz8qckldqnuf"},{"key":"code_id","value":"13"}]},{"type":"message","attributes":[{"key":"action","value":"instantia
te"},{"key":"module","value":"wasm"},{"key":"sender","value":"wasm1j6aldkw59usszphp2jc9jlczxjzc76jdzspf8a"}]},{"type":"wasm","att
ributes":[{"key":"_contract_address","value":"wasm1k9z7fvd3307ynapnk5gjffn706mz8qckldqnuf"},{"key":"method","value":"instantiate"
},{"key":"owner","value":"wasm1j6aldkw59usszphp2jc9jlczxjzc76jdzspf8a"},{"key":"count","value":"1"}]}]}],"gas_wanted":"139830","g
as_used":"117128"}
lsd@lsd:~/work/tendermint/cosmwasm-template-tutorial-v0.16 $
```

SCRIPT

```
$ INIT=$(jq -n --arg lsd $(wasmd keys show -a lsd) '{count:1}')
$ wasmd tx wasm instantiate $CODE_ID "$INIT" --from lsd --label "cw-template test 1" $TXFLAG -y
```

ACTION

Create an instance of our wasm contract with our counter variable start at 1.



```
ype\":\"instantiate\",\"attributes\":[{\"key\":\"_contract_address\",\"value\":\"wasm1k9z7fvd3307ynapnk5gjffn706mz8qckldqnuf\"},{
\"key\":\"code_id\",\"value\":\"13\"}]},{\"type\":\"message\",\"attributes\":[{\"key\":\"action\",\"value\":\"instantiate\"},{\"k
ey\":\"module\",\"value\":\"wasm\"},{\"key\":\"sender\",\"value\":\"wasm1j6aldkw59usszphp2jc9jlczxjzc76jdzspf8a\"}]},{\"type\":\"
wasm\",\"attributes\":[{\"key\":\"_contract_address\",\"value\":\"wasm1k9z7fvd3307ynapnk5gjffn706mz8qckldqnuf\"},{\"key\":\"metho
d\",\"value\":\"instantiate\"},{\"key\":\"owner\",\"value\":\"wasm1j6aldkw59usszphp2jc9jlczxjzc76jdzspf8a\"},{\"key\":\"count\",\
"value\":\"1\"}]}]}]","logs":[{"events":[{"type":"instantiate","attributes":[{"key":"_contract_address","value":"wasm1k9z7fvd3307
ynapnk5gjffn706mz8qckldqnuf"},{"key":"code_id","value":"13"}]},{"type":"message","attributes":[{"key":"action","value":"instantia
te"},{"key":"module","value":"wasm"},{"key":"sender","value":"wasm1j6aldkw59usszphp2jc9jlczxjzc76jdzspf8a"}]},{"type":"wasm","att
ributes":[{"key":"_contract_address","value":"wasm1k9z7fvd3307ynapnk5gjffn706mz8qckldqnuf"},{"key":"method","value":"instantiate"
},{"key":"owner","value":"wasm1j6aldkw59usszphp2jc9jlczxjzc76jdzspf8a"},{"key":"count","value":"1"}]}]}],"gas_wanted":"139830","g
as_used":"117128"}
lsd@lsd:~/work/tendermint/cosmwasm-template-tutorial-v0.16 $ wasmd query wasm list-contract-by-code $CODE_ID $NODE --output json
{"contracts":["wasm1k9z7fvd3307ynapnk5gjffn706mz8qckldqnuf"],"pagination":{}}
lsd@lsd:~/work/tendermint/cosmwasm-template-tutorial-v0.16 $ CONTRACT=$(wasmd query wasm list-contract-by-code $CODE_ID $NODE --o
utput json | jq -r '.contracts[-1]')
lsd@lsd:~/work/tendermint/cosmwasm-template-tutorial-v0.16 $ echo $CONTRACT
wasm1k9z7fvd3307ynapnk5gjffn706mz8qckldqnuf
lsd@lsd:~/work/tendermint/cosmwasm-template-tutorial-v0.16 $ wasmd query wasm contract $CONTRACT $NODE
address: wasm1k9z7fvd3307ynapnk5gjffn706mz8qckldqnuf
contract_info:
  code_id: "13"
  creator: wasm1j6aldkw59usszphp2jc9jlczxjzc76jdzspf8a
  label: cw-template test 1
lsd@lsd:~/work/tendermint/cosmwasm-template-tutorial-v0.16 $ wasmd query wasm contract-state all $CONTRACT $NODE
models:
- key: "7374617465"
  value: eyJjb3VudCI6MSwib3duZXIiOiJ3YXNtMWo2YWxka3c1OXVzc3pwaHAyamM5amxjenhqemM3NmpkenNwZjhhIn0=
pagination: {}
lsd@lsd:~/work/tendermint/cosmwasm-template-tutorial-v0.16 $
```

SCRIPT

$ wasmd query wasm list-contract-by-code $CODE_ID $NODE --output json

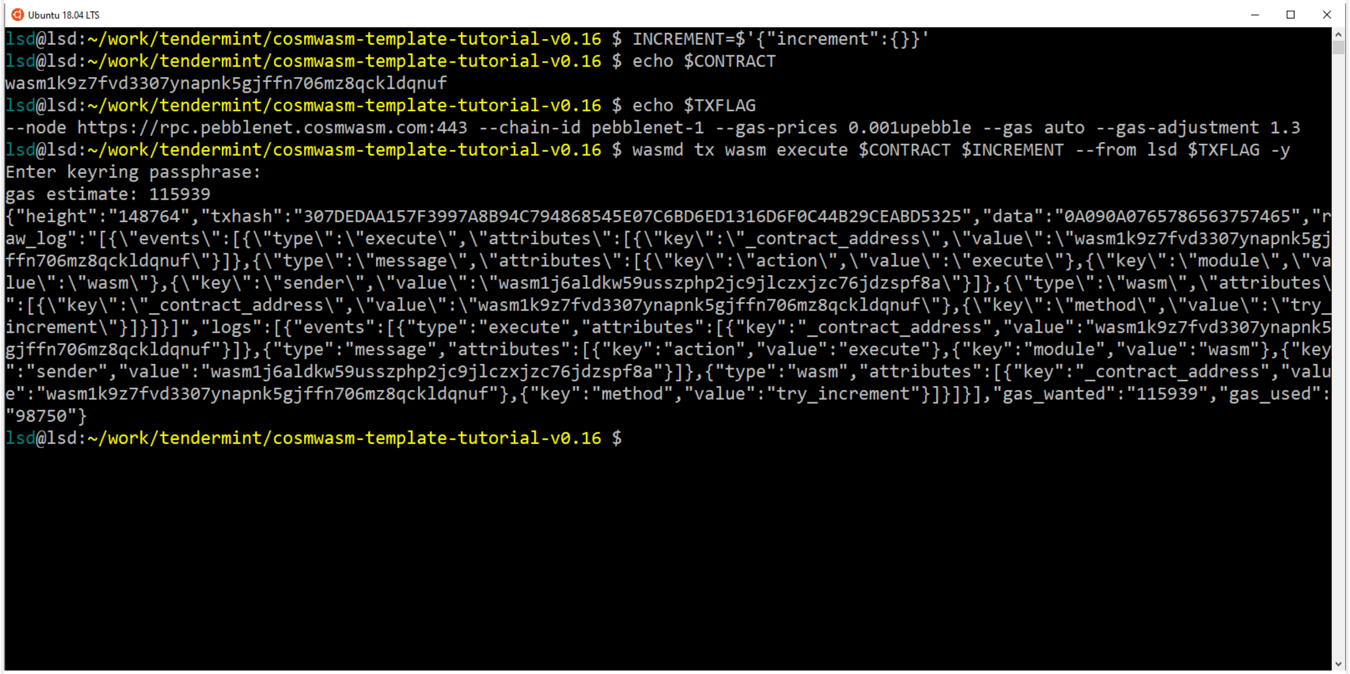
$ CONTRACT=$(wasmd query wasm list-contract-by-code $CODE_ID $NODE --output json | jq -r '.contracts[-1]')

$ wasmd query wasm contract $CONTRACT $NODE

$ wasmd query wasm contract-state all $CONTRACT $NODE

ACTION

Now that our code is on chain, we can query the network for its state.



```
lsd@lsd:~/work/tendermint/cosmwasm-template-tutorial-v0.16 $ INCREMENT=$'{"increment":{}}'
lsd@lsd:~/work/tendermint/cosmwasm-template-tutorial-v0.16 $ echo $CONTRACT
wasm1k9z7fvd3307ynapnk5gjffn706mz8qckldqnuf
lsd@lsd:~/work/tendermint/cosmwasm-template-tutorial-v0.16 $ echo $TXFLAG
--node https://rpc.pebblenet.cosmwasm.com:443 --chain-id pebblenet-1 --gas-prices 0.001upebble --gas auto --gas-adjustment 1.3
lsd@lsd:~/work/tendermint/cosmwasm-template-tutorial-v0.16 $ wasmd tx wasm execute $CONTRACT $INCREMENT --from lsd $TXFLAG -y
Enter keyring passphrase:
gas estimate: 115939
{"height":"148764","txhash":"307DEDAA157F3997A8B94C794868545E07C6BD6ED1316D6F0C44B29CEABD5325","data":"0A090A0765786563757465","r
aw_log":"[{\"events\":[{\"type\":\"execute\",\"attributes\":[{\"key\":\"_contract_address\",\"value\":\"wasm1k9z7fvd3307ynapnk5gj
ffn706mz8qckldqnuf\"}]},{\"type\":\"message\",\"attributes\":[{\"key\":\"action\",\"value\":\"execute\"},{\"key\":\"module\",\"va
lue\":\"wasm\"},{\"key\":\"sender\",\"value\":\"wasm1j6aldkw59usszphp2jc9jlczxjzc76jdzspf8a\"}]},{\"type\":\"wasm\",\"attributes\
":[{\"key\":\"_contract_address\",\"value\":\"wasm1k9z7fvd3307ynapnk5gjffn706mz8qckldqnuf\"},{\"key\":\"method\",\"value\":\"try_
increment\"}]}]}]","logs":[{"events":[{"type":"execute","attributes":[{"key":"_contract_address","value":"wasm1k9z7fvd3307ynapnk5
gjffn706mz8qckldqnuf"}]},{"type":"message","attributes":[{"key":"action","value":"execute"},{"key":"module","value":"wasm"},{"key
":"sender","value":"wasm1j6aldkw59usszphp2jc9jlczxjzc76jdzspf8a"}]},{"type":"wasm","attributes":[{"key":"_contract_address","valu
e":"wasm1k9z7fvd3307ynapnk5gjffn706mz8qckldqnuf"},{"key":"method","value":"try_increment"}]}]}],"gas_wanted":"115939","gas_used":
"98750"}
lsd@lsd:~/work/tendermint/cosmwasm-template-tutorial-v0.16 $
```

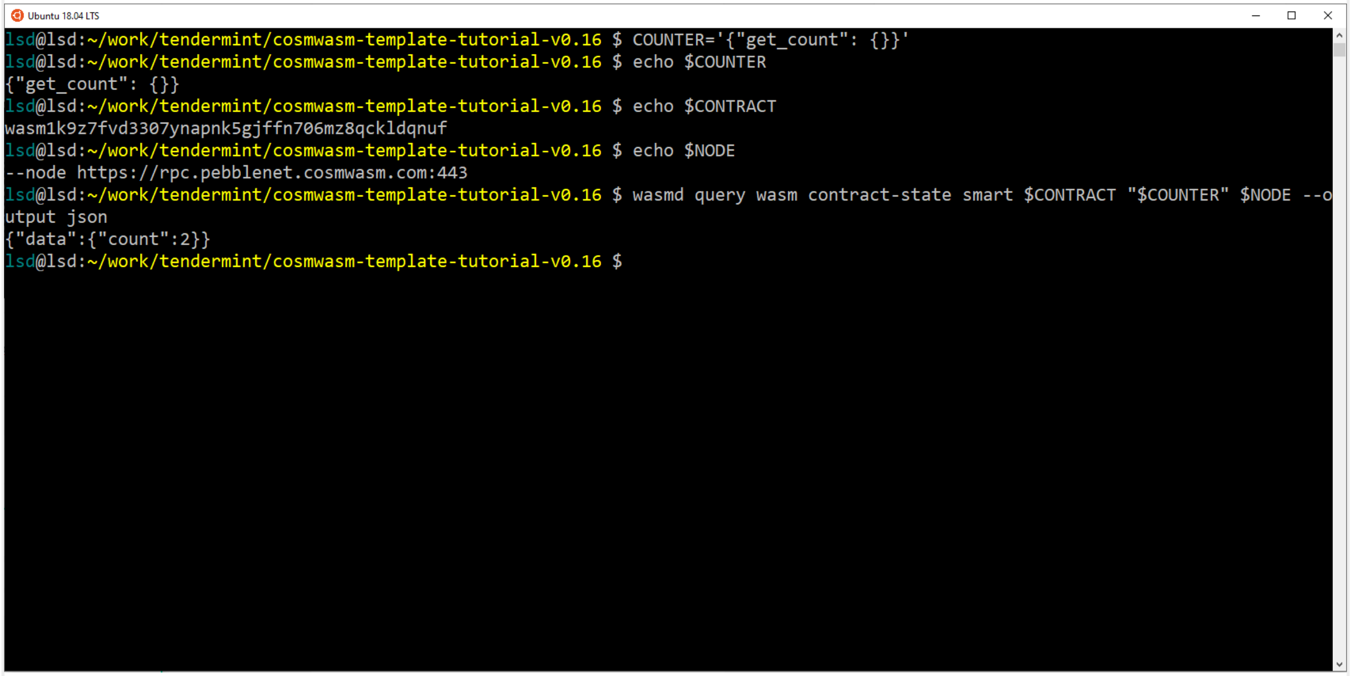
SCRIPT

```
$ INCREMENT=$'{"increment":{}}'
$ wasmd tx wasm execute $CONTRACT $INCREMENT --from lsd $TXFLAG -y
```

ACTION

Make a transaction to the contract.

The default cosmwasm-template code is a counter contract, so let's increment the counter.



```
lsd@lsd:~/work/tendermint/cosmwasm-template-tutorial-v0.16 $ COUNTER='{"get_count": {}}'
lsd@lsd:~/work/tendermint/cosmwasm-template-tutorial-v0.16 $ echo $COUNTER
{"get_count": {}}
lsd@lsd:~/work/tendermint/cosmwasm-template-tutorial-v0.16 $ echo $CONTRACT
wasm1k9z7fvd3307ynapnk5gjffn706mz8qckldqnuf
lsd@lsd:~/work/tendermint/cosmwasm-template-tutorial-v0.16 $ echo $NODE
--node https://rpc.pebblenet.cosmwasm.com:443
lsd@lsd:~/work/tendermint/cosmwasm-template-tutorial-v0.16 $ wasmd query wasm contract-state smart $CONTRACT "$COUNTER" $NODE --o
utput json
{"data":{"count":2}}
lsd@lsd:~/work/tendermint/cosmwasm-template-tutorial-v0.16 $
```

SCRIPT

```
$ COUNTER='{"get_count": {}}'
$ wasmd query wasm contract-state smart $CONTRACT "$COUNTER" $NODE --output json
```

ACTION

Check the on-chain counter was correctly implemented (should now be 2)