alchemyplatform / **learn-solidity-presentations**

Code   Issues   Pull requests 2   Actions   Projects   Security   Ir

learn-solidity-presentations / 1a-value-types / **presentation.md**

Dan-Nolan update presentation 3bd4ba7 · 2 years ago

90 lines (69 loc) · 1.33 KB

Preview   Code   Blame   Raw

marp
true

Solidity Value Types

Let's look at a few important types

```
contract Example {
  uint a;
  int b;
  bool c;

  enum Choice { Up, Down, Left, Right }
  Choice choice = Choice.Up;
}
```

Unsigned Integer: A number without a sign (not positive or negative)

```
contract Example {
  // uint can be declared in steps of 8
  // where the number represents the number of bits
  uint8 x; // 0 -> 255
  uint16 y;

  // uint is an alias for uint256
  uint z1;
  uint256 z2;
}
```

Integer: A number that could be either positive or negative

```
contract Example {
  // uint can be declared in steps of 8
  // where the number represents the number of bits
  int8 x; // -128 -> 127
  int16 y;

  // int is an alias for int256
  int z1;
  int256 z2;
}
```

Boolean: Either `true` or `false`

```
import "forge-std/console.sol";

contract Example {
  constructor(bool myCondition) {
    if(myCondition) {
      // will log yay if myCondition is true
      console.log("yay!");
    }
  }
}
```

Enum: Defining options for a value by name

```
import "forge-std/console.sol";

contract Example {
  enum Choice { Up, Down, Left, Right }

  constructor(Choice choice) {
    if(choice == Choice.Up) {
      console.log("up");
    }
    if(choice == Choice.Down) {
      console.log("down");
    }
  }
}
```