

**WHAT WOULD SOLIDITY LOOK LIKE
IF IT WAS BUILT TODAY?**



CODE EXAMPLE

```
1 interface Mintooooor {
2     function mint(uint8 v, bytes32 r, bytes32 s) external payable;
3 }
4
5 contract Foo is Mintooooor {
6     function mint(uint8 v, bytes32 r, bytes32 s) external payable override {
7         address a = ecrecover(MAGIC, v, r, s);
8         // mint to `a`
9     }
10 }
```

```
1 interface Mintooooor {
2     function mint(uint8 v, bytes32 r, bytes32 s) external payable;
3 }
4
5 contract Foo is Mintooooor {
6     function mint(uint8 v, bytes32 r, bytes32 s) external payable override {
7         address a = ecrecover(MAGIC, v, r, s);
8         // mint to `a`
9     }
10 }
```

```
1 abi Mintooor {
2     function mint(uint8 v, bytes32 r, bytes32 s) external payable;
3 }
4
5 impl Mintooor for Foo {
6     function mint(uint8 v, bytes32 r, bytes32 s) external payable {
7         address a = ecrecover(MAGIC, v, r, s);
8         // mint to `a`
9     }
10 }
11
12 impl Foo {
13 }
```



```
1 abi Mintooor {
2     #[payability(nonpayable)]
3     function mint(uint8 v, bytes32 r, bytes32 s) external;
4 }
5
6 impl Mintooor for Foo {
7     #[payability(nonpayable)]
8     function mint(uint8 v, bytes32 r, bytes32 s) external {
9         address a = ecrecover(MAGIC, v, r, s);
10        // mint to `a`
11    }
12 }
```



```
1 abi Mintooor {
2     function mint(uint8 v, bytes32 r, bytes32 s) external;
3 }
4
5 impl Mintooor for Foo {
6     function mint(uint8 v, bytes32 r, bytes32 s) external {
7         address a = ecrecover(MAGIC, v, r, s);
8         // mint to `a`
9     }
10 }
```



```
1 abi Mintooor {
2     #[visibility(external)]
3     function mint(uint8 v, bytes32 r, bytes32 s);
4 }
5
6 impl Mintooor for Foo {
7     #[visibility(external)]
8     function mint(uint8 v, bytes32 r, bytes32 s) {
9         address a = ecrecover(MAGIC, v, r, s);
10        // mint to `a`
11    }
12 }
```



```
1 use std::crypto::ecrecover;
2
3 abi Mintooooor {
4     #[visibility(external)]
5     function mint(uint8 v, bytes32 r, bytes32 s);
6 }
7
8 impl Mintooooor for Foo {
9     #[visibility(external)]
10    function mint(uint8 v, bytes32 r, bytes32 s) {
11        address a = ecrecover(MAGIC, v, r, s);
12        // mint to `a`
13    }
14 }
```



```
1 use std::crypto::ecrecover;
2
3 abi Mintooooor {
4     #[visibility(external)]
5     function mint(uint8 v, bytes32 r, bytes32 s);
6 }
7
8 impl Mintooooor for Foo {
9     #[visibility(external)]
10    function mint(uint8 v, bytes32 r, bytes32 s) {
11        Result r = ecrecover(MAGIC, v, r, s);
12        address a = r.unwrap();
13    }
14 }
```



```
1 use std::crypto::ecrecover;
2
3 abi Mintooooor {
4     #[visibility(external)]
5     function mint(uint8 v, bytes32 r, bytes32 s);
6 }
7
8 impl Mintooooor for Foo {
9     #[visibility(external)]
10    function mint(uint8 v, bytes32 r, bytes32 s) {
11        address a = ecrecover(MAGIC, v, r, s).unwrap();
12    }
13 }
```



```
1 use std::crypto::ecrecover;
2
3 abi Mintooooor {
4     #[visibility(external)]
5     function mint(v: uint8, r: bytes32, s: bytes32);
6 }
7
8 impl Mintooooor for Foo {
9     #[visibility(external)]
10    function mint(v: uint8, r: bytes32, s: bytes32) {
11        let a: address = ecrecover(MAGIC, v, r, s).unwrap();
12    }
13 }
```



```
1 use std::crypto::ecrecover;
2
3 abi Mintooooor {
4     #[visibility(external)]
5     function mint(v: uint8, r: bytes32, s: bytes32);
6 }
7
8 impl Mintooooor for Foo {
9     #[visibility(external)]
10    function mint(v: uint8, r: bytes32, s: bytes32) {
11        let a = ecrecover(MAGIC, v, r, s).unwrap();
12    }
13 }
```

```
1 use std::crypto::ecrecover;
2
3 abi Mintooooor {
4     #[visibility(external)]
5     fn mint(v: u8, r: b256, s: b256);
6 }
7
8 impl Mintooooor for Foo {
9     #[visibility(external)]
10    fn mint(v: u8, r: b256, s: b256) {
11        let a = ecrecover(MAGIC, v, r, s).unwrap();
12    }
13 }
```

FIN