Practice program on solidity

Program 1:

```
pragma solidity >=0.7.0;
contract Variable {
        uint videos=30;
        int playlist=3;
        bool active=true;
        bytes4 symbol='Dk';
        string name="Indian Institute of technology";
}
```

Program 2:

```
pragma solidity >=0.7.0;
contract Variable {
        bool active=false;
        int length=-3;
        uint length2=3;
        string name="Indian Institute of technology";
        bytes2 category='go';
}
```

Program 3:

```
pragma solidity >=0.7.0;
contract Variable {
        bool active=false;
        int length=-3;
        uint length2=3;
        string name="Indian Institute of technology";
        bytes2 category='go';
        address add=0xc538779A628a21D7CCA7b1a3E57E
}
```

Program 4:

```
pragma solidity >0.7.0;
contract functions {
        uint public val=4;
        function add1() public view returns(uint){
        return val;
        }
        function add() public pure returns(uint){
        return 3+5;
        }
        //Pure->> state variables states, edit
        //View ->> state variable view
}
```

Program 5:

```
pragma solidity >0.7.0;
contract functions {
        uint public val=4;
        function add() public pure returns(uint){
        uint val2=3;
        return val2+5;
        }
        function global() public view returns(uint){
        return block.timestamp;
        }
        function global2() public view returns(address){
        return msg.sender;
        }
        function global3() public view returns(uint){
        return msg.sender.balance;
        }
}
```

Program 6:

```
pragma solidity >=0.7.0;
contract varscope{
        uint public val=4;
        function get() public view returns(uint){
        return val;
        }
}
contract varscope2 is varscope{
        function get2() public view returns(uint){
        return val;
        }
}
```

Program 7:

```
pragma solidity>0.7.0;
contract varscope{
        uint public val=4;
        function get() public view returns(uint){
        return val;
        }
}
```

Program 8:

```
contract functions{
        uint public val=4;
        function add(string memory name) public pure returns(string memory){
         //uint val2=3; //Local variable
         return name;
        }
        function global() public view returns(uint){
        return msg.sender.balance;
        }
}
```

Program 9:

```
contract functions{
        uint public val=4;
        function add(uint number) public pure returns(uint){
         return number;
        }
        function global() public view returns(uint){
        return msg.sender.balance;
        }
}
```

Program 10:

```
contract functions{
        uint public val=4;
        function add() public view returns(address){
         return msg.sender;
        }
        function global() public view returns(uint){
        return msg.sender.balance;
        }
}
```

Program 11:

```
contract functions{
        uint public val=4;
        function add() public pure returns(string memory){
         return "IIT";
        }
        function global() public view returns(uint){
        return msg.sender.balance;
        }
}
```