

박정모 클래스 연습문제

2022-02-10

문제 1 - 다음을 만족하는 Student 클래스를 작성하시오.

1. String형의 학과와 정수형의 학번을 프로퍼티로 선언후 생성자를 통해 주입
2. getter, setter를 정의
3. sayHello() 메서드를 통해 "나는 OOOO학과 OO학번 입니다." 를 출력하는 기능을 구현

```
class Student {
    constructor(depart, studno){
        this._depart = depart;
        this._studno = studno;
    }
    set depart(value){
        if(!value){
            console.log("학과를 입력하세요.");
            return;
        } else {
            this._depart = value;
        }
    }
    get depart(){
        return this._depart;
    }
    set studno(value){
        if(!value){
            console.log("학번을 입력하세요");
            return;
        } else {
            this._studno = value;
        }
    }
    get studno(){
        return this._studno;
    }
    sayHello(){
        console.log("나는 %s학과 %d학번 입니다.", this.depart, this.studno);
    }
}

const a = new Student("건축", 200910649);

a.sayHello();
```

실행결과 스크린샷

```
[Running] node "c:\Users\pjm\studynote\02- JavaScript\10 클래스\연습문제\tempCodeRunnerFile.js"
나는 건축학과 200910649학번 입니다.

[Done] exited with code=0 in 0.058 seconds
```

문제 - 2. 다음을 만족하는 클래스 Account를 작성하시오.

1. 다음의 2 개의 필드를 선언 문자열 owner; (이름) 숫자형 balance; (금액)
2. 위 모든 필드에 대한 getter와 setter의 구현
3. 위 모든 필드를 사용하는 가능한 모든 생성자의 구현
4. 메소드 deposit()의 헤드는 다음과 같으며 인자인 금액을 저축하는 메소드 deposit(amount)
5. 메소드 withdraw()의 헤드는 다음과 같으며 인자인 금액을 인출(리턴)하는 메소드 withdraw(long amount) 인출 상한 금액은 잔액까지로 하며, 이 경우 이러한 상황을 출력

```
class Account {
    constructor(owner, balance){
        this._owner = owner;
        if(balance < 0){
            console.log("금액을 다시 확인하세요");
            return;
        } else {
            this._balance = balance;
        }
    }
    set owner(value){
        this._owner = value;
    }
    get owner(){
        return this._owner;
    }
    set balance(value){
        if(value < 0){
            console.log("금액을 다시 확인하세요.");
            return;
        } else {
            this._balance = value;
        }
    }
    get balance(){
        return this._balance;
    }
    deposit(amount){
        console.log("현재 잔액 = %d", this.balance + amount);
        this.balance += amount;
    }
    withdraw(longAmount){
        if(this.balance < longAmount){
            console.log("인출 상한 금액을 초과하였습니다.");
            return;
        } else {
```

```
        console.log("현재 잔액 = %d", this.balance - longAmount);  
        return (this.balance -= longAmount);  
    }  
}  
  
const b = new Account("Jungmo", 1000);  
  
b.deposit(5000);  
  
b.withdraw(5000);  
  
b.withdraw(4000);  
  
const c = new Account("Someone", -1000);
```

실행결과 스크린샷

```
[Running] node "c:\Users\pjm\studynote\02- JavaScript\10 클래스\연습문제\tempCodeRunnerFile.js"  
현재 잔액 = 6000  
현재 잔액 = 1000  
인출 상한 금액을 초과하였습니다.  
금액을 다시 확인하세요  
  
[Done] exited with code=0 in 0.06 seconds
```