

2024-2학기 자바프로그래밍2 5주차 실습

문제 1

```
1 package week5;
2
3 public class Account {
4     private String name;
5     private String accountNumber;
6     protected int balance;
7
8     // 생성자
9     public Account(String name, String accountNumber, int balance) {
10         this.name = name;
11         this.accountNumber = accountNumber;
12         this.balance = balance;
13     }
14
15     // 입금 메서드
16     public void deposit(int amount) {
17         if (amount > 0) {
18             balance += amount;
19             System.out.println(amount + "원이 입금되었습니다. 잔액: " + balance + "원");
20         } else {
21             System.out.println("잘못된 금액을 입력하였습니다.");
22         }
23     }
24
25     // 출금 메서드
26     public void withdraw(int amount) {
27         if (amount > 0 && amount <= balance) {
28             balance -= amount;
29             System.out.println(amount + "원이 출금되었습니다. 잔액: " + balance + "원");
30         } else {
31             System.out.println("잔액이 부족하거나 잘못된 금액을 입력하였습니다.");
32         }
33     }
34
35     @Override
36     public String toString() {
37         return "예금주: " + name + ", 계좌번호: " + accountNumber + ", 잔액: " + balance + "원";
38     }
39 }
40 }
```

```
1 package week5;
2
3 public class CheckingAccount extends Account {
4     private int withdrawLimit; //출금 한도
5
6     // 생성자
7     public CheckingAccount(String name, String accountNumber, int balance, int withdrawLimit) {
8         super(name, accountNumber, balance);
9         this.withdrawLimit = withdrawLimit;
10    }
11
12
13    @Override
14    public void withdraw(int amount) {
15        if (amount > withdrawLimit) {
16            System.out.println("출금 한도를 초과했습니다. 출금 한도: " + withdrawLimit + "원");
17        } else {
18            super.withdraw(amount);
19        }
20    }
21
22    @Override
23    public String toString() {
24        return "입출금 계좌 [" + super.toString() + ", 출금 한도: " + withdrawLimit + "원]";
25    }
26 }
```

```

1 package week5;
2
3 public class SavingsAccount extends Account {
4     private double interestRate; //이자율
5
6     // 생성자
7     public SavingsAccount(String name, String accountNumber, int balance, double interestRate) {
8         super(name, accountNumber, balance);
9         this.interestRate = interestRate;
10    }
11
12    public void applyInterest() {
13        double interest = balance * interestRate;
14        balance += interest;
15        System.out.println("이자가 " + interest + "원 적용되었습니다. 잔액: " + balance + "원");
16    }
17
18    @Override
19    public String toString() {
20        return "저축 계좌 [" + super.toString() + ", 이자율: " + (interestRate * 100) + "%]";
21    }
22 }

```

```

1 package week5;
2
3 public class Exercise01 {
4
5     public static void main(String[] args) {
6
7         // 저축 계좌 생성
8         SavingsAccount savingsAccount = new SavingsAccount("홍길동", "1001", 100000, 0.03);
9         System.out.println(savingsAccount);
10        savingsAccount.deposit(-50000);
11        savingsAccount.applyInterest();
12
13        System.out.println();
14
15        // 입출금 계좌 생성
16        CheckingAccount checkingAccount = new CheckingAccount("홍길동", "1002", 200000, 100000);
17        System.out.println(checkingAccount);
18        checkingAccount.withdraw(120000);
19        checkingAccount.withdraw(100000);
20
21    }
22
23 }

```

문제 2

문제1의 Account에서 추가

```

1 package week5;
2
3 import java.util.Objects;
4
5 public class Account {
6     private String name;
7     private String accountNumber;
8     protected int balance;
9
10    // 생성자
11    public Account(String name, String accountNumber, int balance) {
12        this.name = name;
13        this.accountNumber = accountNumber;
14        this.balance = balance;
15    }
16
17    // 입금 메서드
18    public void deposit(int amount) {
19        if (amount > 0) {
20            balance += amount;
21            System.out.println(amount + "원이 입금되었습니다. 잔액: " + balance + "원");
22        } else {
23            System.out.println("잘못된 금액을 입력하였습니다.");
24        }
25    }
26
27    // 출금 메서드
28    public void withdraw(int amount) {
29        if (amount > 0 && amount <= balance) {
30            balance -= amount;
31            System.out.println(amount + "원이 출금되었습니다. 잔액: " + balance + "원");
32        } else {
33            System.out.println("잔액이 부족하거나 잘못된 금액을 입력하였습니다.");
34        }
35    }
36
37    @Override
38    public String toString() {
39        return "예금주: " + name + ", 계좌번호: " + accountNumber + ", 잔액: " + balance + "원";
40    }
41
42    @Override
43    public boolean equals(Object obj) {
44        if (this == obj) return true;
45        if (!(obj instanceof Account)) return false;
46        return Objects.equals(this.accountNumber, ((Account) obj).accountNumber);
47    }
48
49 }

```

```

1 package week5;
2
3 public class Exercise02 {
4
5     public static void main(String[] args) {
6
7         Account a1 = new Account("홍길동", "20240001", 10000);
8         Account a2 = new Account("홍길동", "20240001", 50000);
9         SavingsAccount a3 = new SavingsAccount("홍길동", "20240001", 100000, 0.03);
10        Account a4 = new Account("홍길동", "20240002", 10000);
11
12
13        if(a1.equals(a2))
14            System.out.println("a1과 a2는 동일한 계좌입니다.");
15        else
16            System.out.println("a1과 a2는 동일한 계좌가 아닙니다.");
17
18        if(a1.equals(a3))
19            System.out.println("a1과 a3은 동일한 계좌입니다.");
20        else
21            System.out.println("a1과 a3은 동일한 계좌가 아닙니다.");
22
23        if(a1.equals(a4))
24            System.out.println("a1과 a4은 동일한 계좌입니다.");
25        else
26            System.out.println("a1과 a4는 동일한 계좌가 아닙니다.");
27
28    }
29
30 }

```

문제 3

```
1 package week5;
2
3 abstract class User {
4     protected String name;
5     private String id;
6     private String password;
7
8     public User(String name, String id, String password) {
9         this.name = name;
10        this.id = id;
11        this.password = password;
12    }
13
14    public abstract void printRole();
15
16    public boolean login(String id, String pwd) {
17        if(id.equals(this.id) && pwd.equals(this.password)) {
18            System.out.println("로그인 성공 : '" + name + "'님 로그인");
19            return true;
20        }
21        return false;
22    }
23 }
24
25 class Administrator extends User {
26     public Administrator(String name, String id, String password) {
27         super(name, id, password);
28     }
29
30     @Override
31     public void printRole() {
32         System.out.println("'" + name + "'님은 관리자입니다.");
33     }
34
35     public void manage() {
36         System.out.println("'" + name + "'님이 시스템을 관리합니다.");
37     }
38 }
39
40 class Student extends User {
41     public Student(String name, String id, String password) {
42         super(name, id, password);
43     }
44
45     @Override
46     public void printRole() {
47         System.out.println("'" + name + "'님은 학생입니다.");
48     }
49
50     public void submit() {
51         System.out.println("'" + name + "'님이 과제를 제출합니다.");
52     }
53 }
54
55 class Professor extends User {
56     public Professor(String name, String id, String password) {
57         super(name, id, password);
58     }
59
60     @Override
61     public void printRole() {
62         System.out.println("'" + name + "'님은 교수입니다.");
63     }
64
65     public void grading() {
66         System.out.println("'" + name + "'님이 채점을 시작합니다.");
67     }
68 }
69
70 }
```

```

1 package week5;
2
3 import java.util.Scanner;
4
5 public class Exercise03 {
6
7     public static void main(String[] args) {
8
9         Scanner input = new Scanner(System.in);
10
11         // 상황형변환으로 User 배열에 다양한 사용자 객체 저장
12         User[] users = new User[3];
13         users[0] = new Administrator("강백호", "admin01", "1234");
14         users[1] = new Student("서태웅", "student01", "0000");
15         users[2] = new Professor("송태섭", "professor01", "1111");
16
17         System.out.print("아이디를 입력하세요: ");
18         String id = input.nextLine();
19         System.out.print("비밀번호를 입력하세요: ");
20         String pwd = input.nextLine();
21
22         User loggedIn = null;
23         for (User user : users) {
24             if(user.login(id, pwd)) loggedIn = user;
25         }
26
27         if (loggedIn != null) {
28             loggedIn.printRole();
29             // 로그인된 사용자가 어떤 타입인지 확인 후 하향형변환하여 고유 동작 실행
30             if (loggedIn instanceof Administrator) {
31                 Administrator admin = (Administrator) loggedIn;
32                 admin.manage();
33             } else if (loggedIn instanceof Student) {
34                 Student student = (Student) loggedIn;
35                 student.submit();
36             } else if (loggedIn instanceof Professor) {
37                 Professor professor = (Professor) loggedIn;
38                 professor.grading();
39             } else {
40                 System.out.println("알 수 없는 사용자입니다.");
41             }
42         }
43         else {
44             System.out.println("로그인 실패: 잘못된 아이디 또는 비밀번호입니다.");
45         }
46
47         input.close();
48     }
49 }
50 }

```