연습문제 풀이

문제1 : 교재 6장 클래스 확인문제 1~20번

1. 3
2. 4
3. 4
4. 3
5. 1
6. 4
7. 2
8. 2

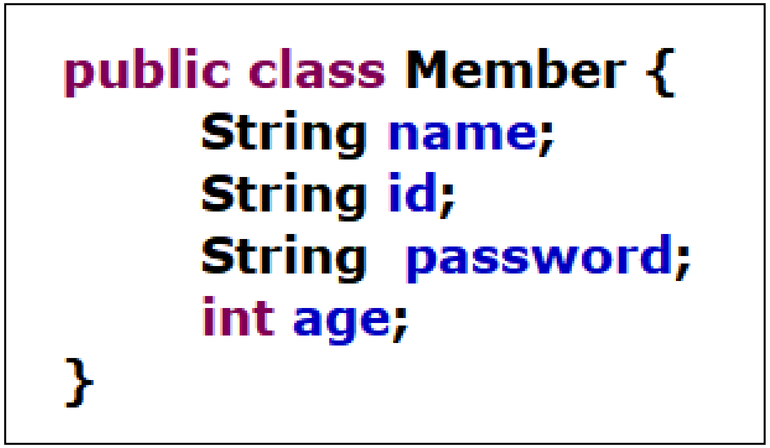
9. 2 (final 필드는 생성자에서 초기화 가능(인스턴스 레벨)
static final 필드는 생성자에서 초기화 불가
- 클래스 레벨로 한 번만 초기화 가능
- 객체 생성할 때마다 초기화할 수 없음)

10. 4.

11. 3 (동일 패키지 내의 모든 클래스에서 접근 가능)

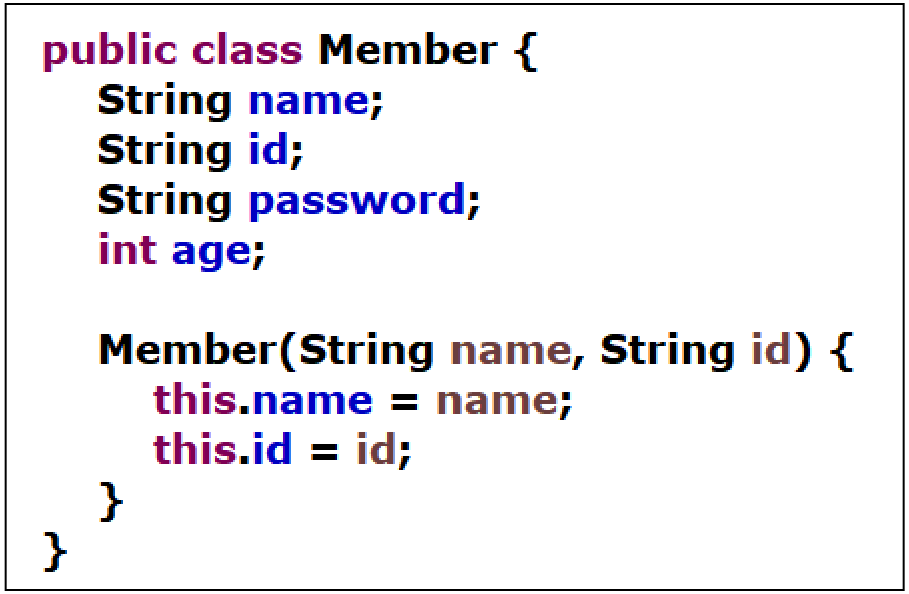
12. (1. 필드, 2. 생성자, 3.메소드)

13.



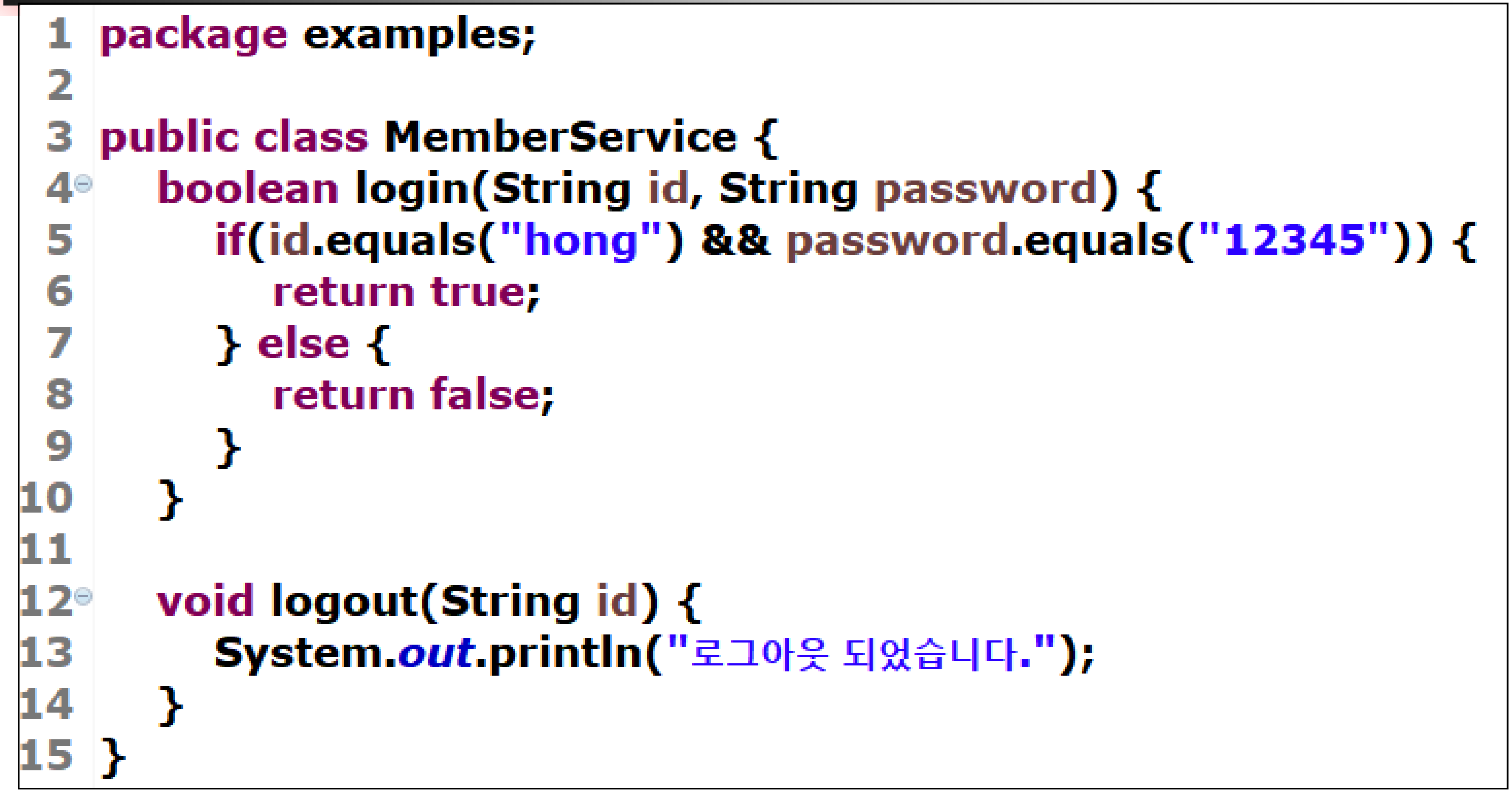
```
public class Member {
        String name;
        String id;
        String  password;
        int age;
}
```

14.

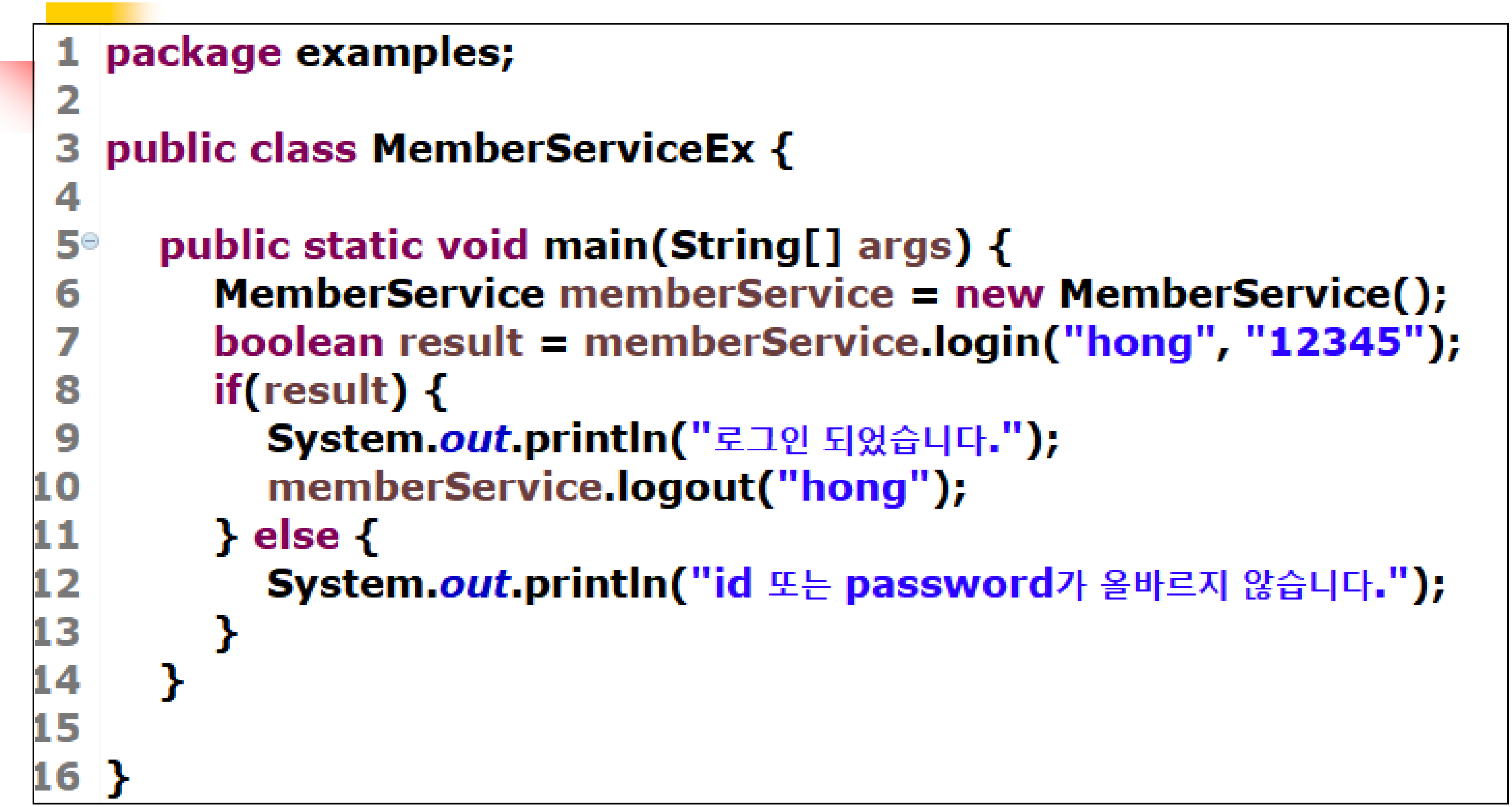


```
public class Member {
    String name;
    String id;
    String password;
    int age;

    Member(String name, String id) {
        this.name = name;
        this.id = id;
    }
}
```

15

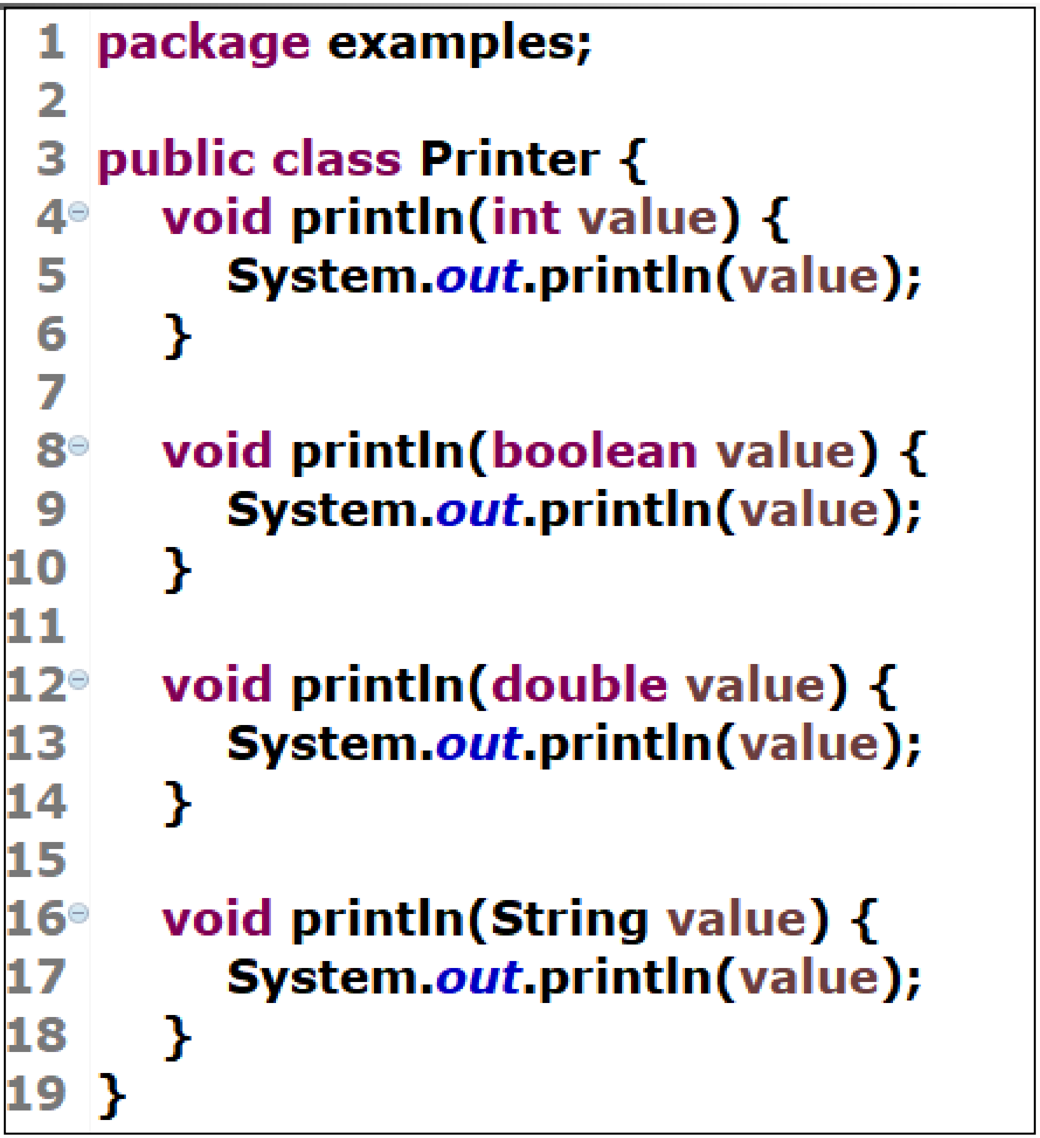


```java
package examples;

public class MemberService {
    boolean login(String id, String password) {
        if(id.equals("hong") && password.equals("12345")) {
            return true;
        } else {
            return false;
        }
    }

    void logout(String id) {
        System.out.println("로그아웃 되었습니다.");
    }
}
```

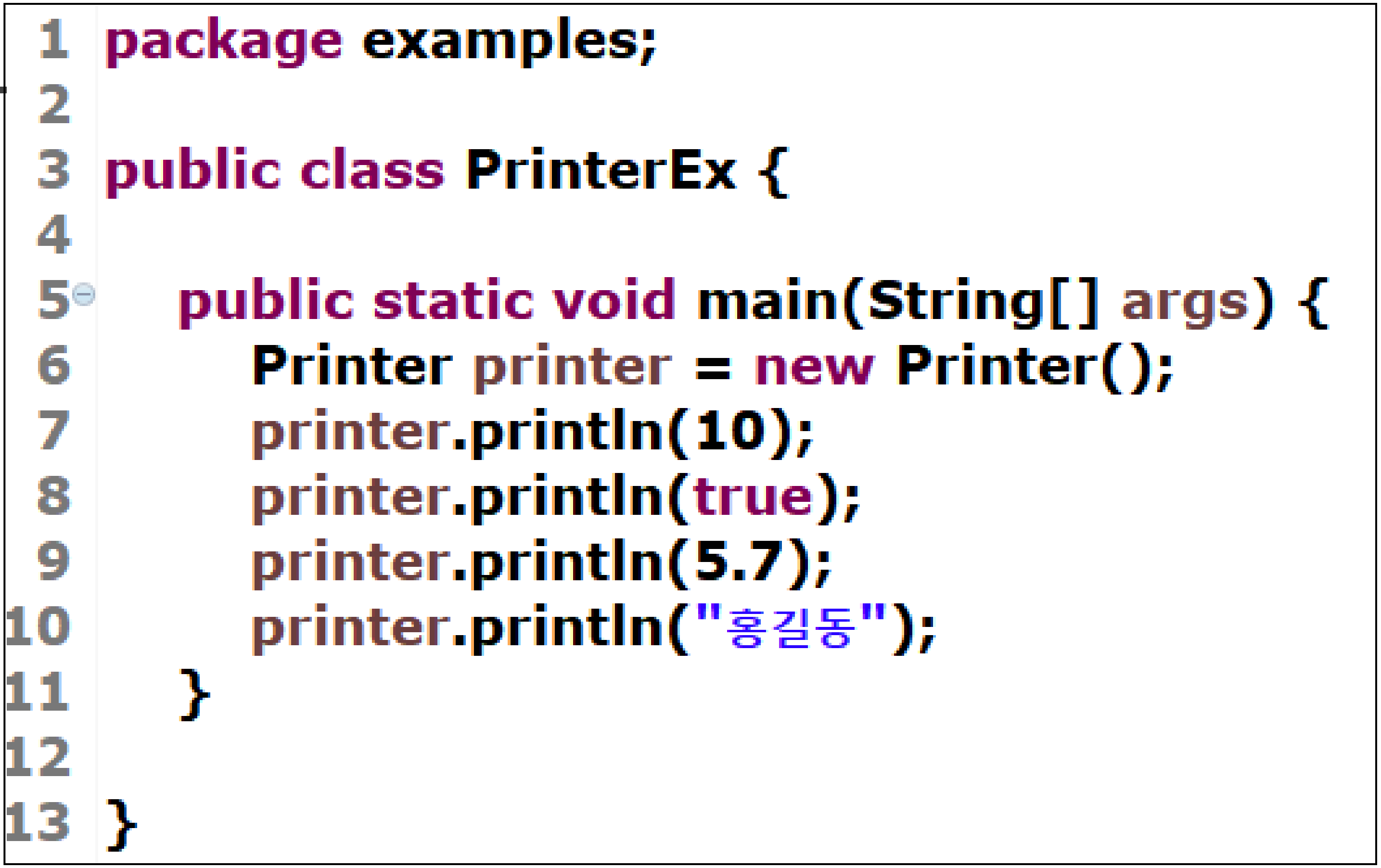


```java
package examples;

public class MemberServiceEx {

    public static void main(String[] args) {
        MemberService memberService = new MemberService();
        boolean result = memberService.login("hong", "12345");
        if(result) {
            System.out.println("로그인 되었습니다.");
            memberService.logout("hong");
        } else {
            System.out.println("id 또는 password가 올바르지 않습니다.");
        }
    }

}
```

16

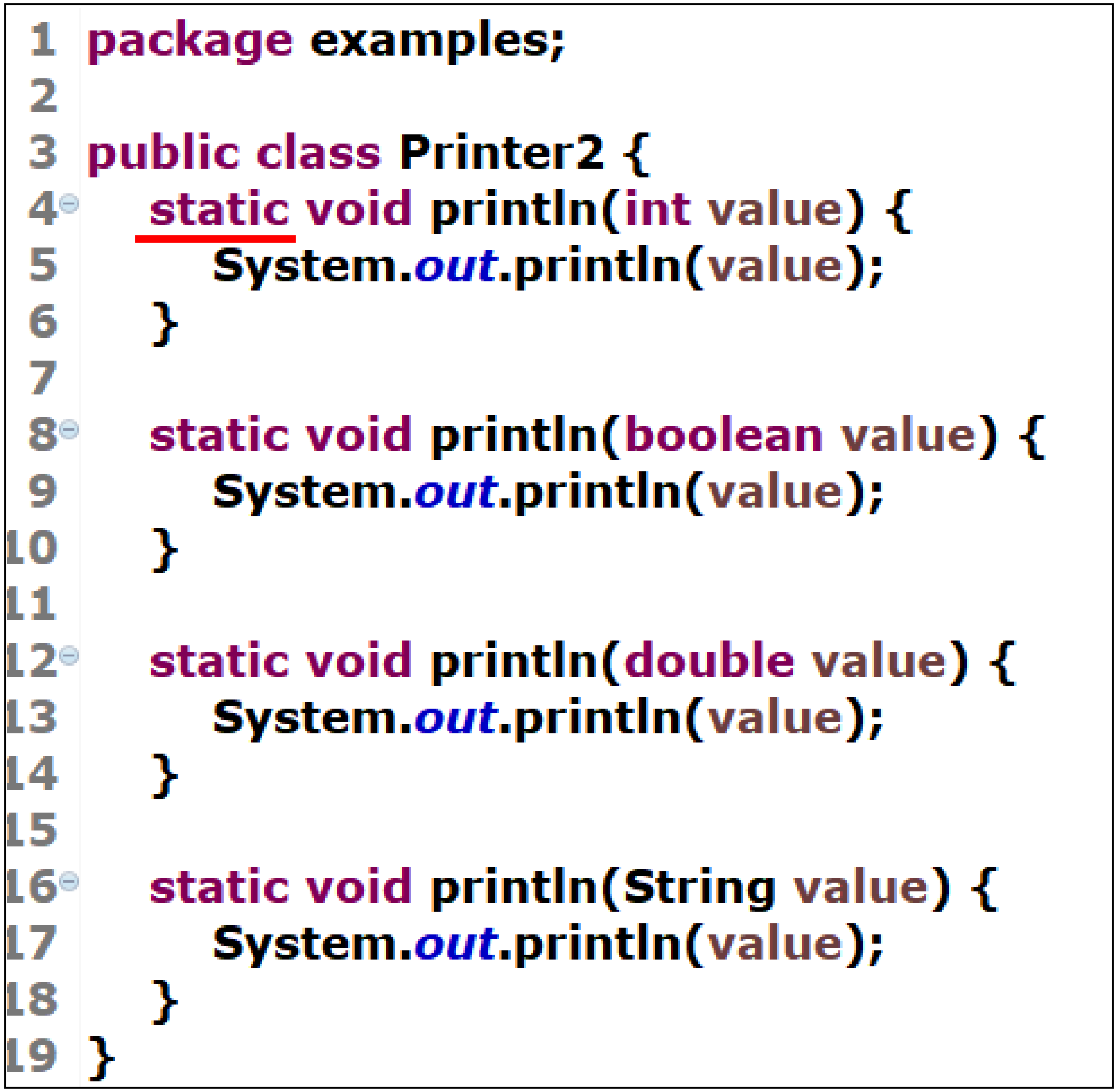


```java
package examples;

public class Printer {
    void println(int value) {
        System.out.println(value);
    }

    void println(boolean value) {
        System.out.println(value);
    }

    void println(double value) {
        System.out.println(value);
    }

    void println(String value) {
        System.out.println(value);
    }
}
```

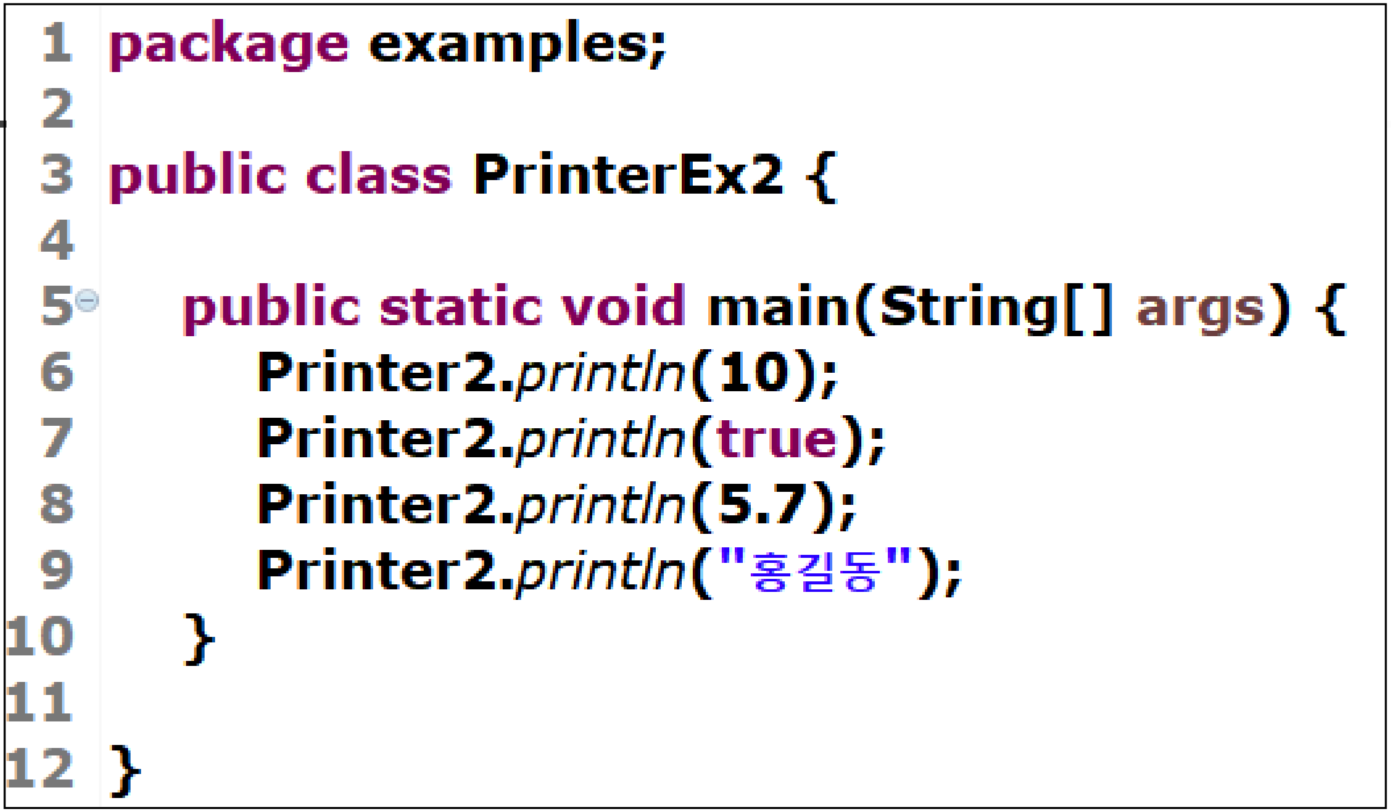


```
package examples;

public class PrinterEx {

    public static void main(String[] args) {
        Printer printer = new Printer();
        printer.println(10);
        printer.println(true);
        printer.println(5.7);
        printer.println("홍길동");
    }

}
```

객체 생성하지 않고 호출 static 메소드



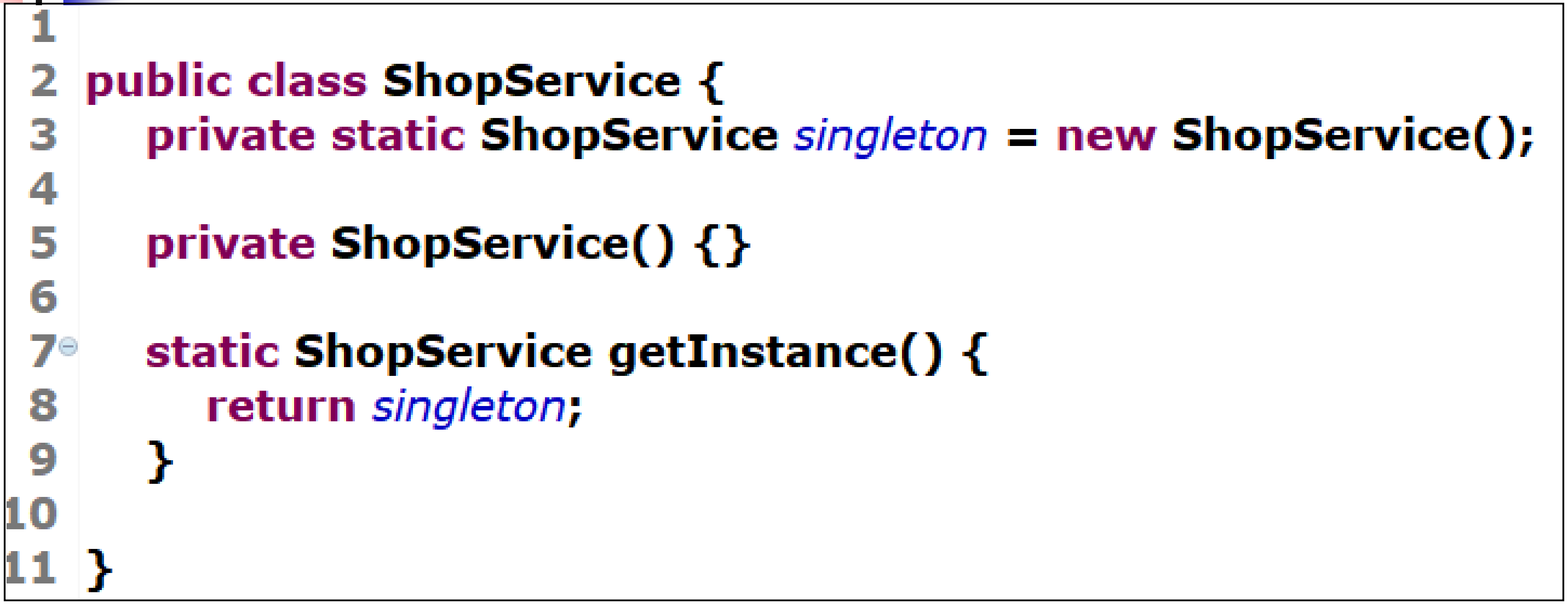
```
package examples;

public class Printer2 {
    static void println(int value) {
        System.out.println(value);
    }

    static void println(boolean value) {
        System.out.println(value);
    }

    static void println(double value) {
        System.out.println(value);
    }

    static void println(String value) {
        System.out.println(value);
    }
}
```



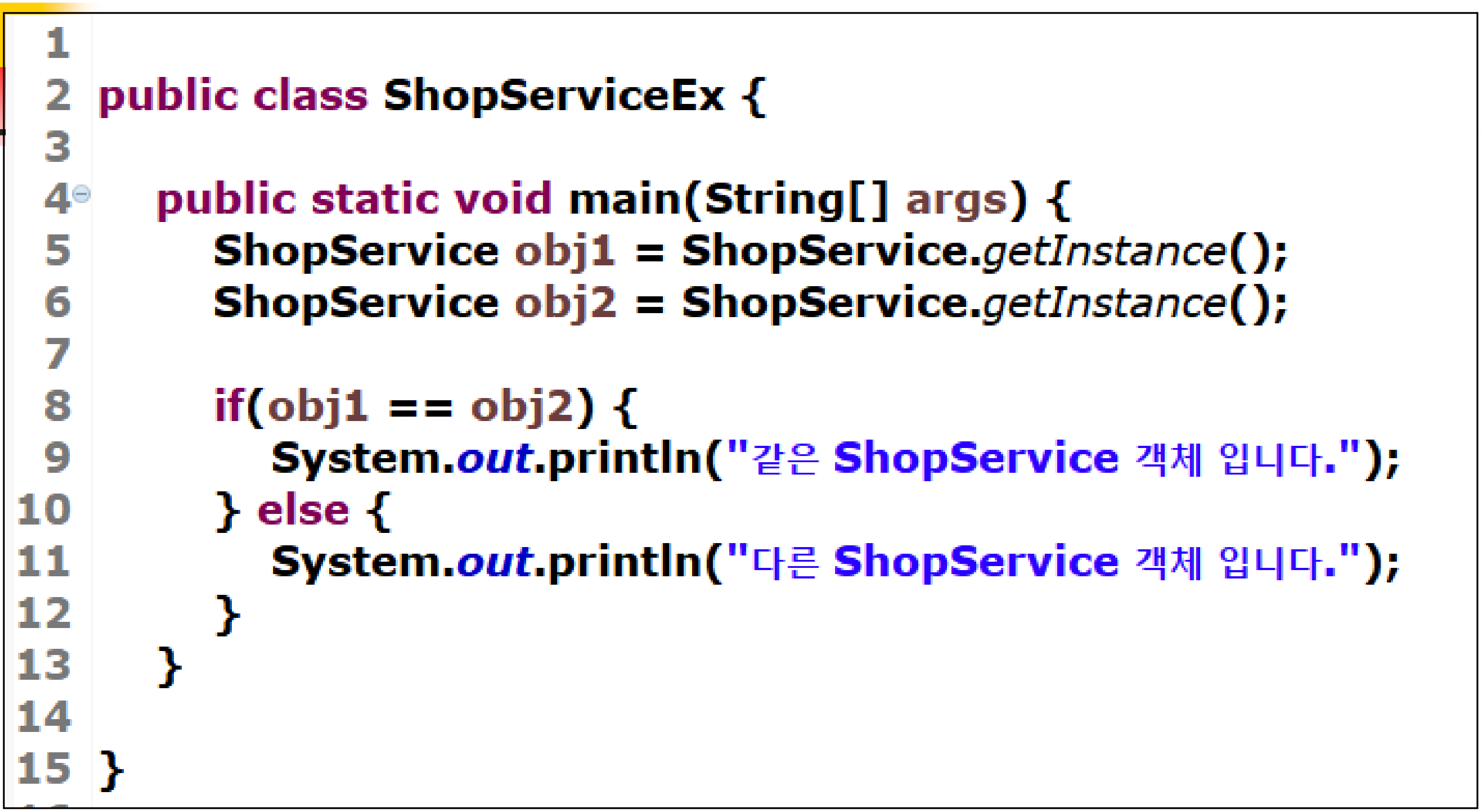
```java
package examples;

public class PrinterEx2 {

    public static void main(String[] args) {
        Printer2.println(10);
        Printer2.println(true);
        Printer2.println(5.7);
        Printer2.println("홍길동");
    }

}
```

static 메소드 호출 : 클래스명.메소드

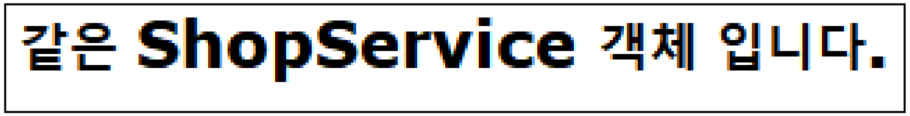
18



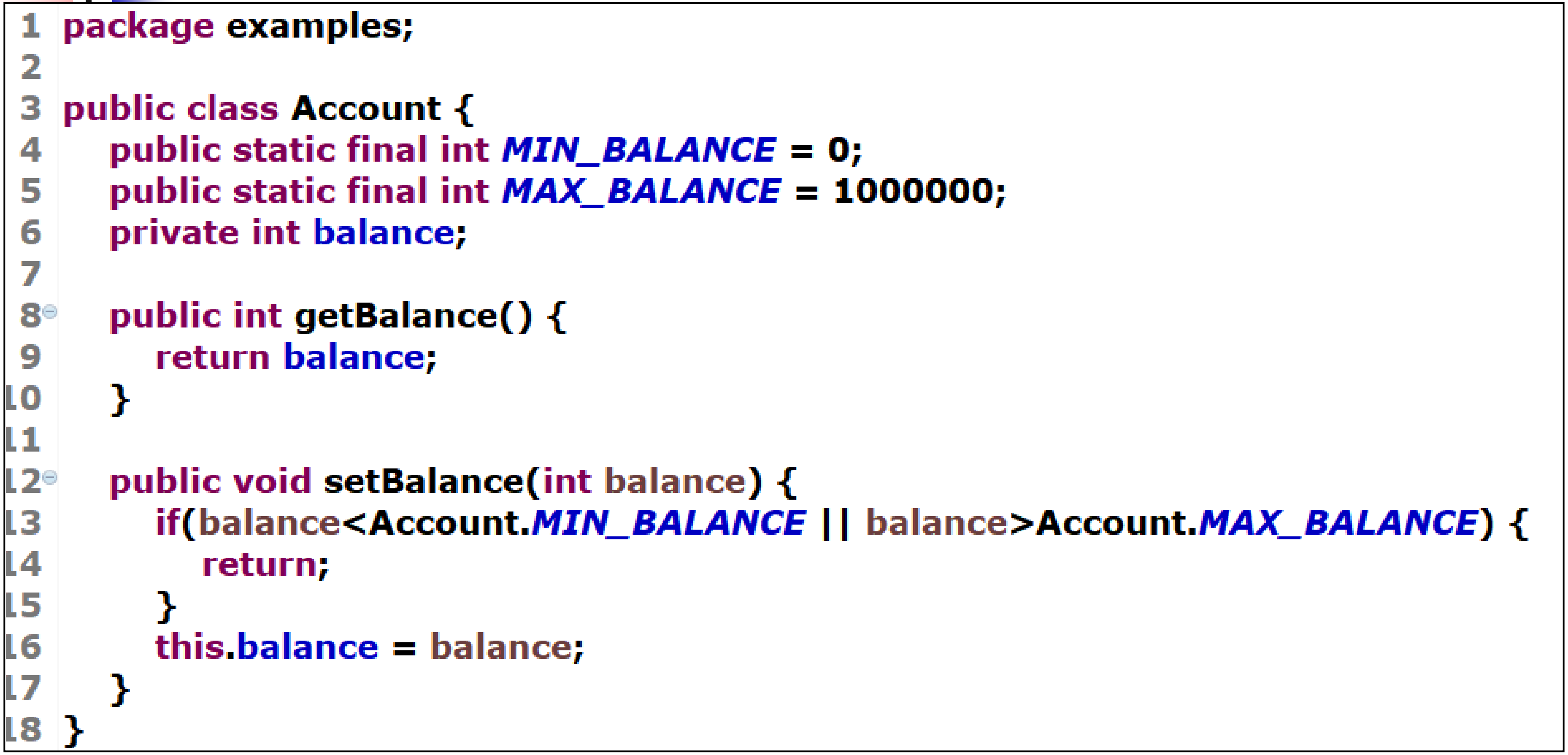
```java

public class ShopService {
    private static ShopService singleton = new ShopService();

    private ShopService() {}

    static ShopService getInstance() {
        return singleton;
    }

}
```



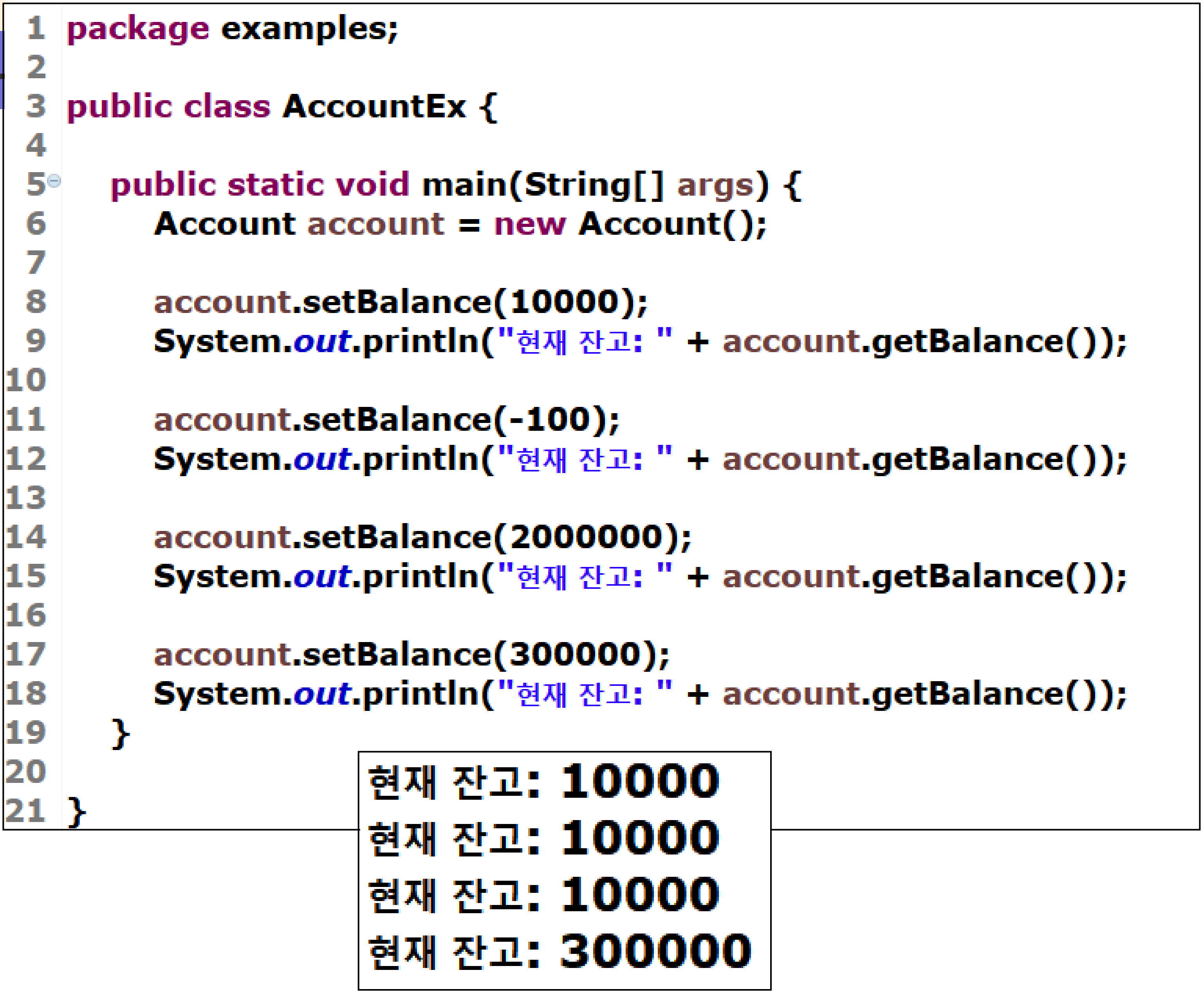
```

public class ShopServiceEx {

    public static void main(String[] args) {
        ShopService obj1 = ShopService.getInstance();
        ShopService obj2 = ShopService.getInstance();

        if(obj1 == obj2) {
            System.out.println("같은 ShopService 객체 입니다.");
        } else {
            System.out.println("다른 ShopService 객체 입니다.");
        }
    }

}
```



```
같은 ShopService 객체 입니다.
```

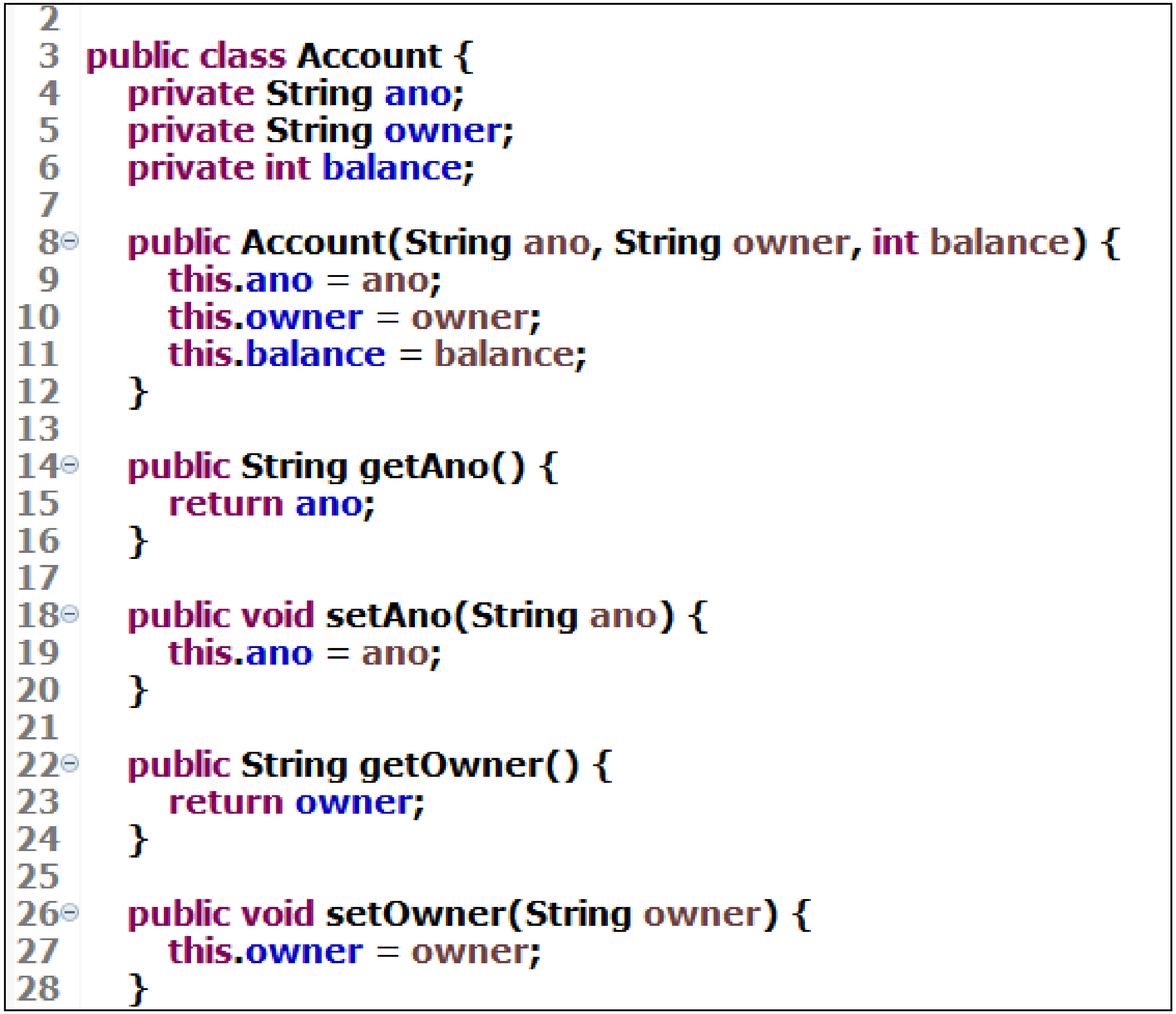


```java
package examples;

public class Account {
    public static final int MIN_BALANCE = 0;
    public static final int MAX_BALANCE = 1000000;
    private int balance;

    public int getBalance() {
        return balance;
    }

    public void setBalance(int balance) {
        if(balance<Account.MIN_BALANCE || balance>Account.MAX_BALANCE) {
            return;
        }
        this.balance = balance;
    }
}
```

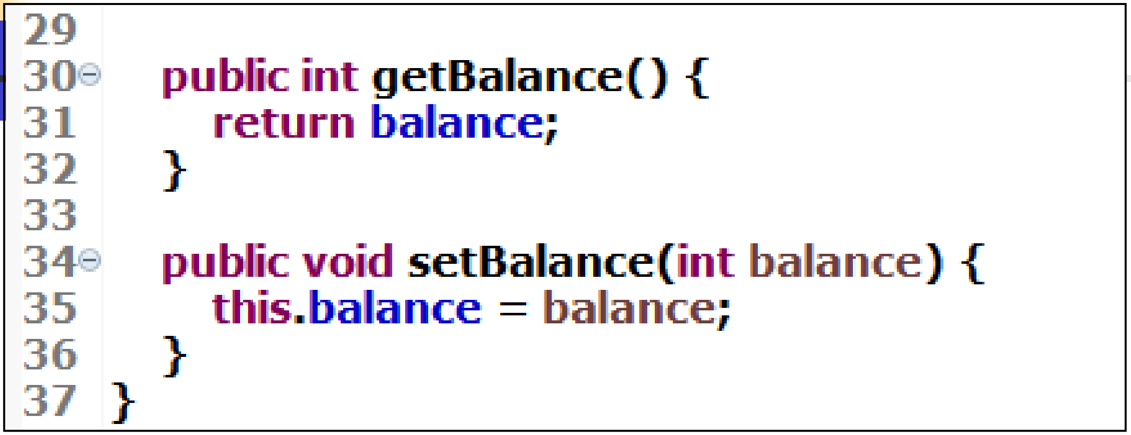


```java
package examples;

public class AccountEx {

    public static void main(String[] args) {
        Account account = new Account();

        account.setBalance(10000);
        System.out.println("현재 잔고: " + account.getBalance());

        account.setBalance(-100);
        System.out.println("현재 잔고: " + account.getBalance());

        account.setBalance(2000000);
        System.out.println("현재 잔고: " + account.getBalance());

        account.setBalance(300000);
        System.out.println("현재 잔고: " + account.getBalance());
    }

}
```

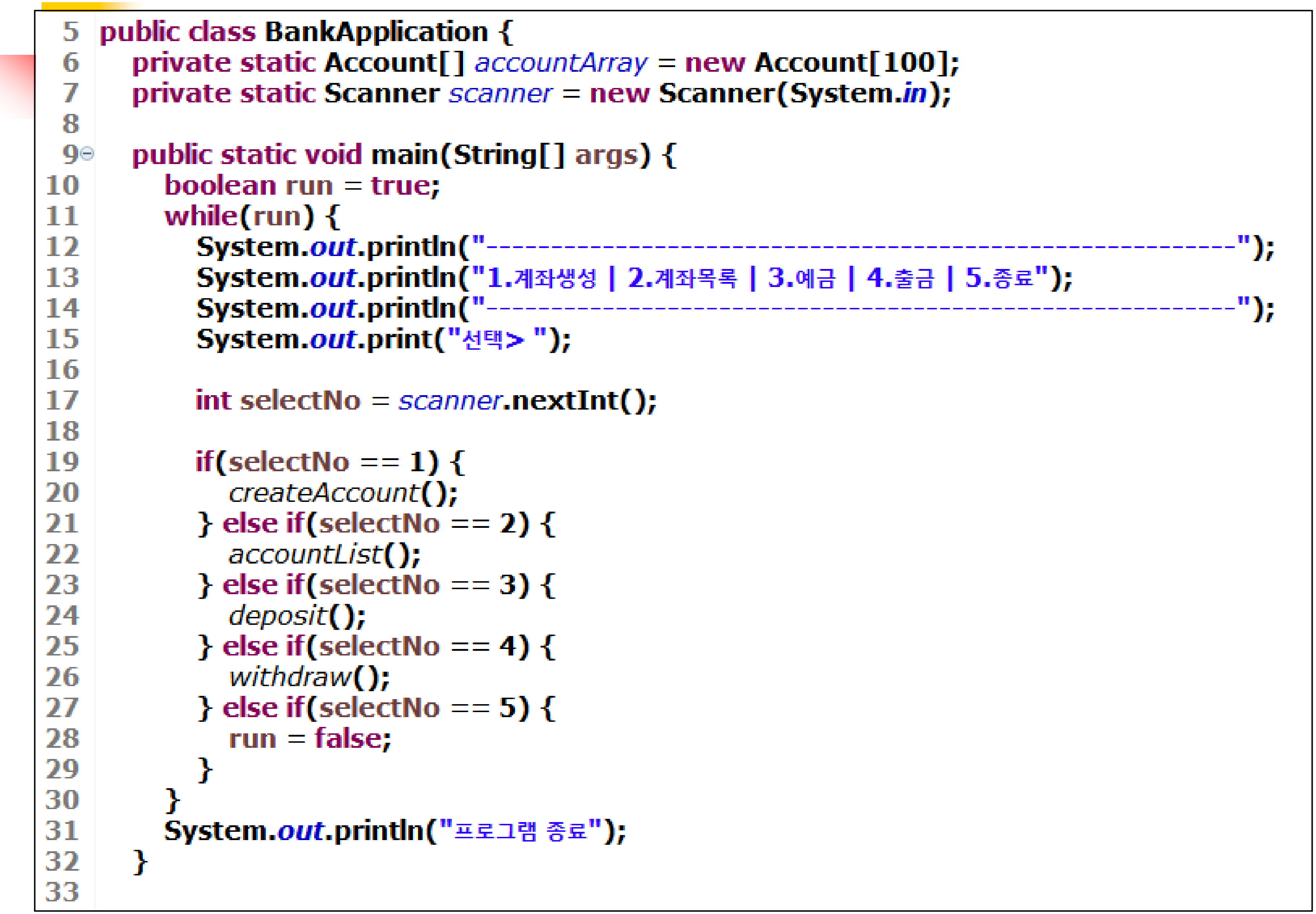
```
현재 잔고: 10000
현재 잔고: 10000
현재 잔고: 10000
현재 잔고: 300000
```



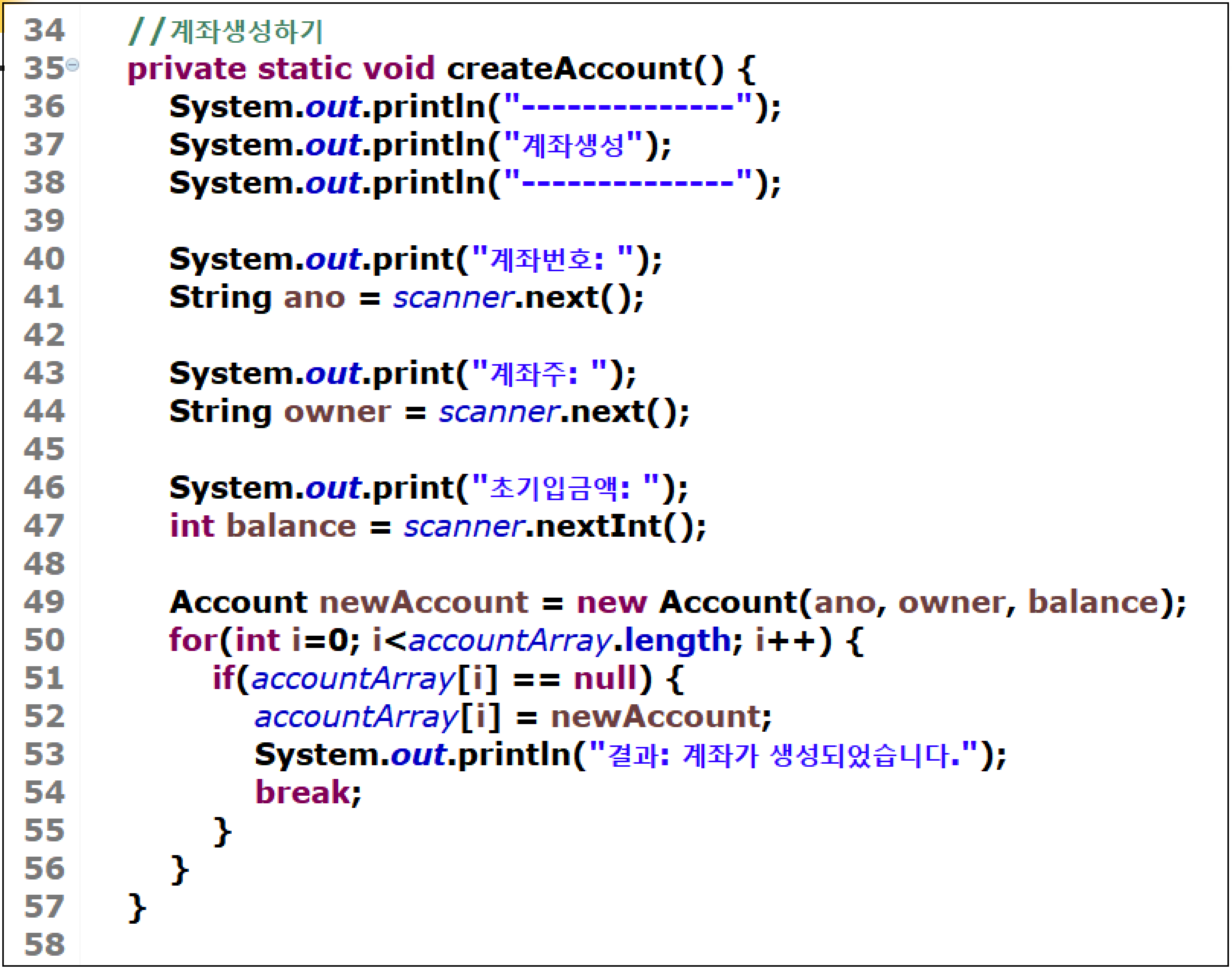
```java
public class Account {
    private String ano;
    private String owner;
    private int balance;

    public Account(String ano, String owner, int balance) {
        this.ano = ano;
        this.owner = owner;
        this.balance = balance;
    }

    public String getAno() {
        return ano;
    }

    public void setAno(String ano) {
        this.ano = ano;
    }

    public String getOwner() {
        return owner;
    }

    public void setOwner(String owner) {
        this.owner = owner;
    }
```



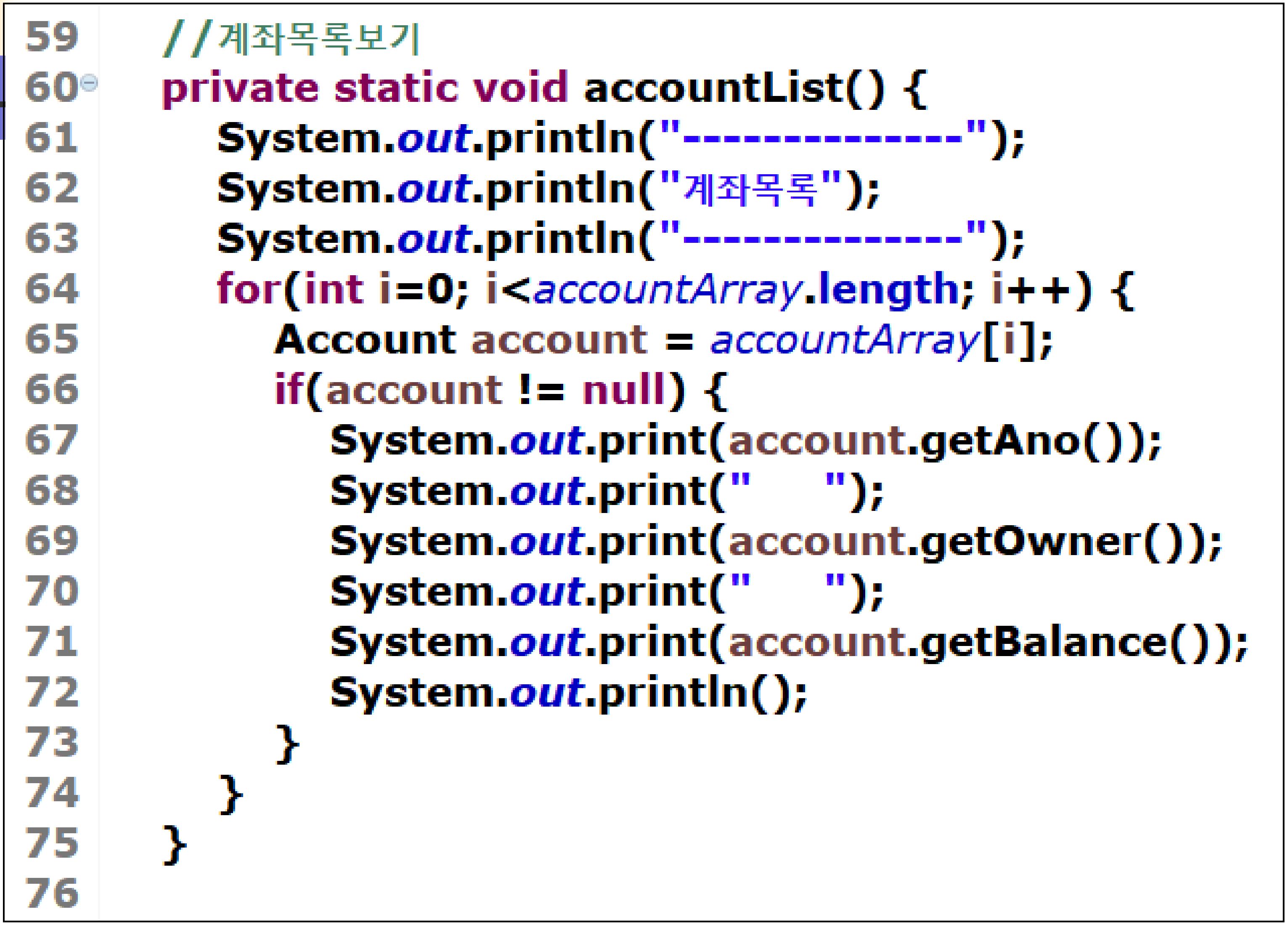
```java

    public int getBalance() {
        return balance;
    }

    public void setBalance(int balance) {
        this.balance = balance;
    }
}
```



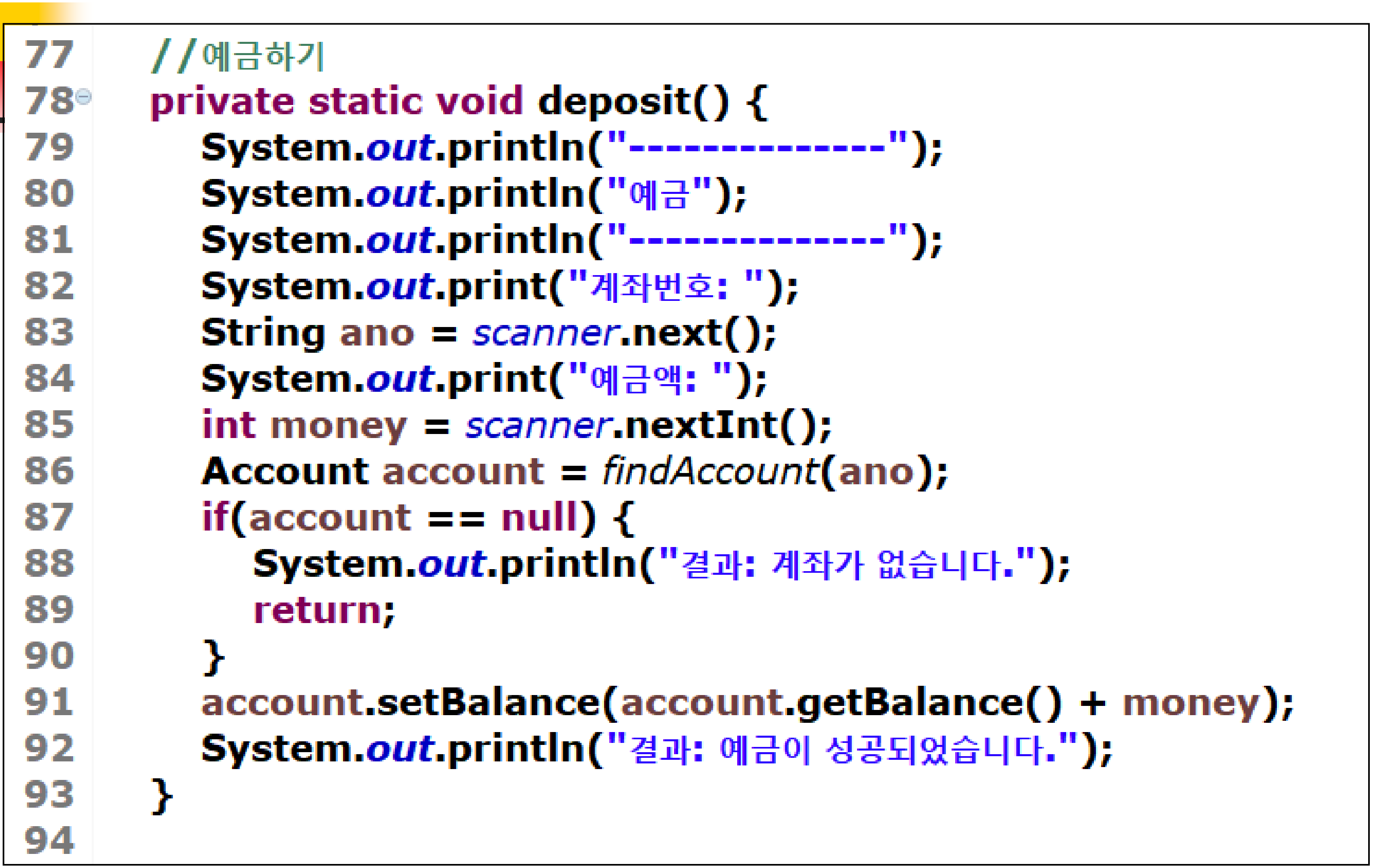
```java
public class BankApplication {
    private static Account[] accountArray = new Account[100];
    private static Scanner scanner = new Scanner(System.in);

    public static void main(String[] args) {
        boolean run = true;
        while(run) {
            System.out.println("------------------------------------------------------------");
            System.out.println("1.계좌생성 | 2.계좌목록 | 3.예금 | 4.출금 | 5.종료");
            System.out.println("------------------------------------------------------------");
            System.out.print("선택> ");

            int selectNo = scanner.nextInt();

            if(selectNo == 1) {
                createAccount();
            } else if(selectNo == 2) {
                accountList();
            } else if(selectNo == 3) {
                deposit();
            } else if(selectNo == 4) {
                withdraw();
            } else if(selectNo == 5) {
                run = false;
            }
        }
        System.out.println("프로그램 종료");
    }

```



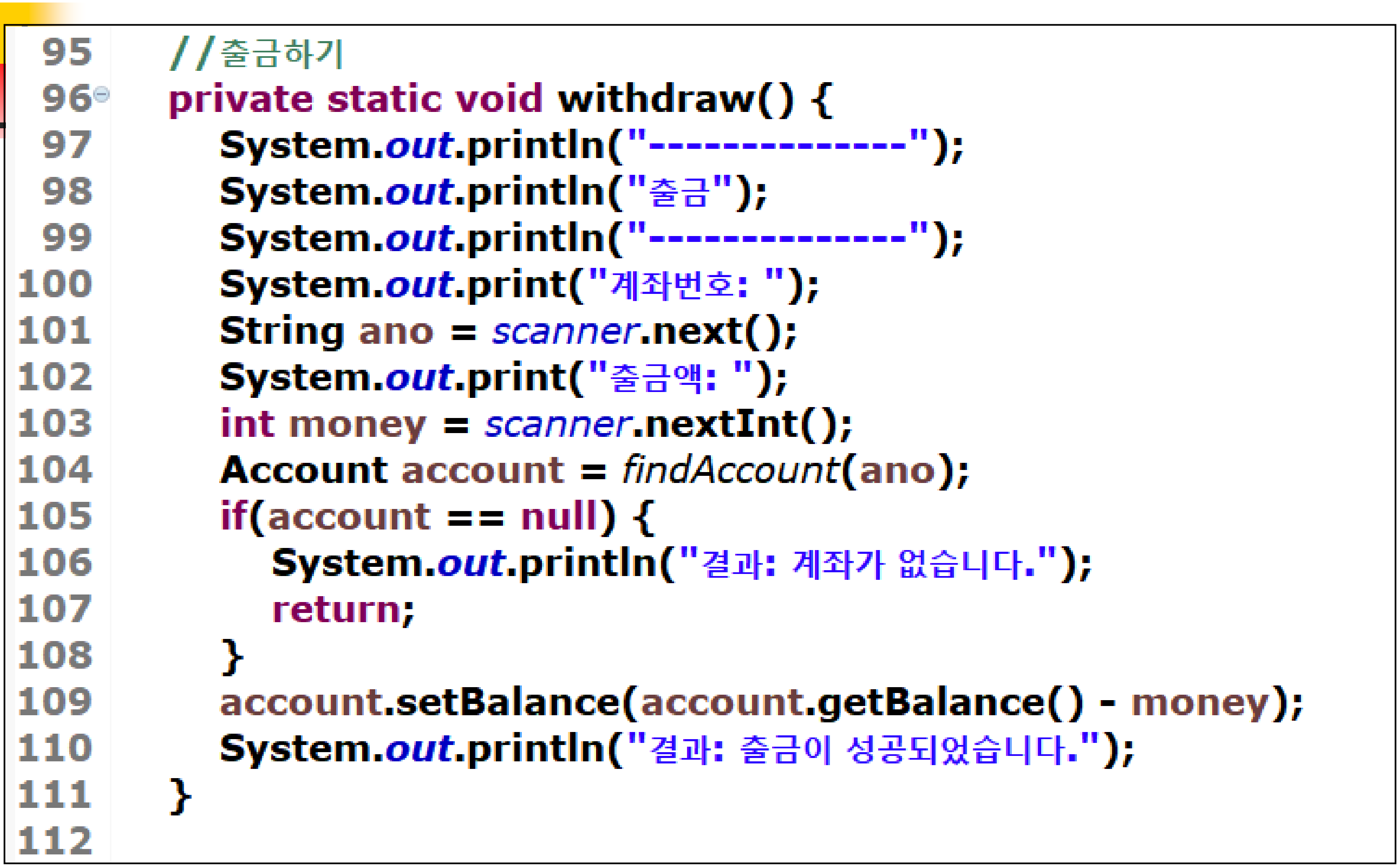
```java
//계좌생성하기
private static void createAccount() {
    System.out.println("--------------");
    System.out.println("계좌생성");
    System.out.println("--------------");

    System.out.print("계좌번호: ");
    String ano = scanner.next();

    System.out.print("계좌주: ");
    String owner = scanner.next();

    System.out.print("초기입금액: ");
    int balance = scanner.nextInt();

    Account newAccount = new Account(ano, owner, balance);
    for(int i=0; i<accountArray.length; i++) {
        if(accountArray[i] == null) {
            accountArray[i] = newAccount;
            System.out.println("결과: 계좌가 생성되었습니다.");
            break;
        }
    }
}

```



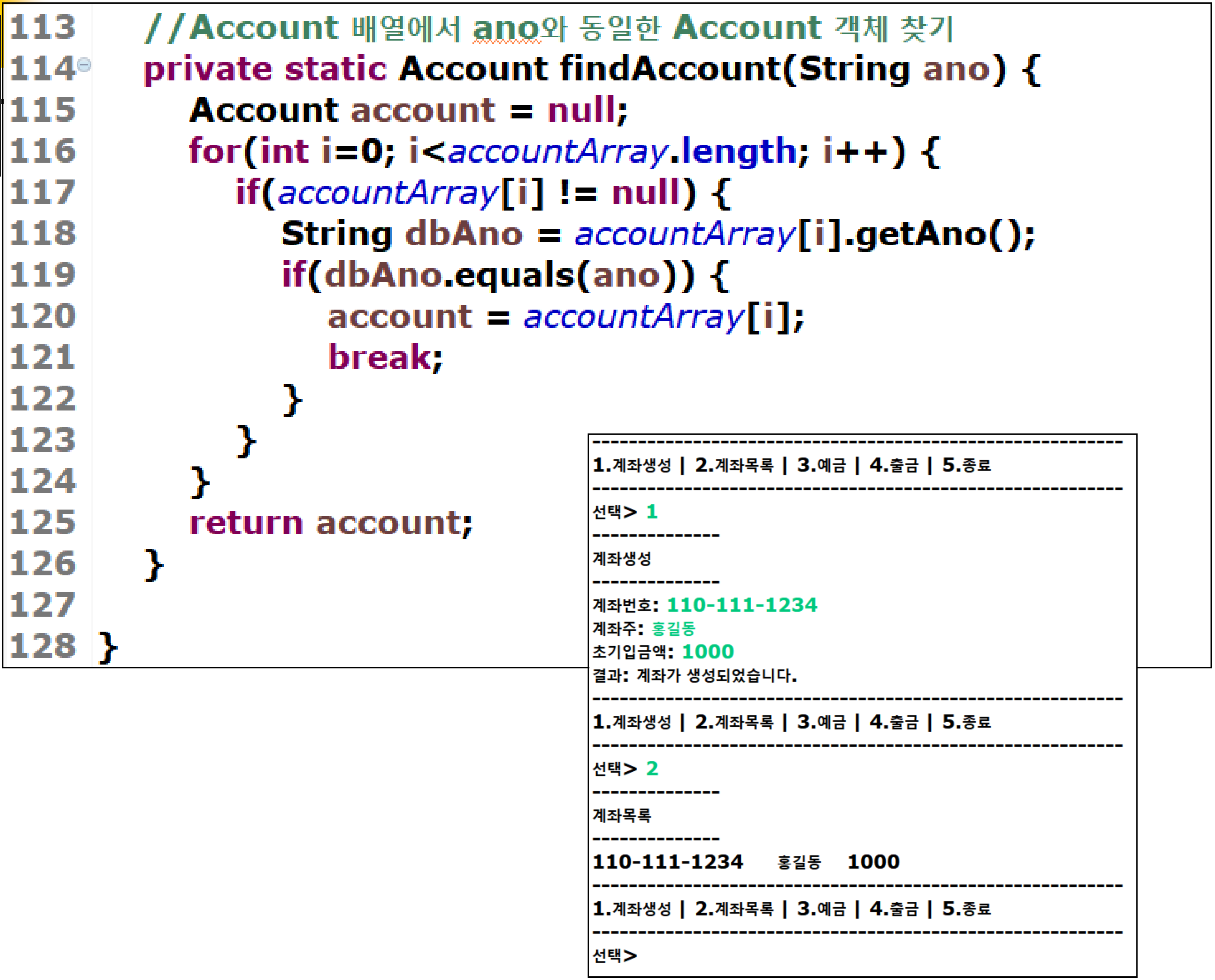
```java
    //계좌목록보기
    private static void accountList() {
        System.out.println("---------------");
        System.out.println("계좌목록");
        System.out.println("---------------");
        for(int i=0; i<accountArray.length; i++) {
            Account account = accountArray[i];
            if(account != null) {
                System.out.print(account.getAno());
                System.out.print("    ");
                System.out.print(account.getOwner());
                System.out.print("    ");
                System.out.print(account.getBalance());
                System.out.println();
            }
        }
    }

```



```java
    //예금하기
    private static void deposit() {
        System.out.println("---------------");
        System.out.println("예금");
        System.out.println("---------------");
        System.out.print("계좌번호: ");
        String ano = scanner.next();
        System.out.print("예금액: ");
        int money = scanner.nextInt();
        Account account = findAccount(ano);
        if(account == null) {
            System.out.println("결과: 계좌가 없습니다.");
            return;
        }
        account.setBalance(account.getBalance() + money);
        System.out.println("결과: 예금이 성공되었습니다.");
    }

```



```java
    //출금하기
    private static void withdraw() {
        System.out.println("--------------");
        System.out.println("출금");
        System.out.println("--------------");
        System.out.print("계좌번호: ");
        String ano = scanner.next();
        System.out.print("출금액: ");
        int money = scanner.nextInt();
        Account account = findAccount(ano);
        if(account == null) {
            System.out.println("결과: 계좌가 없습니다.");
            return;
        }
        account.setBalance(account.getBalance() - money);
        System.out.println("결과: 출금이 성공되었습니다.");
    }

```



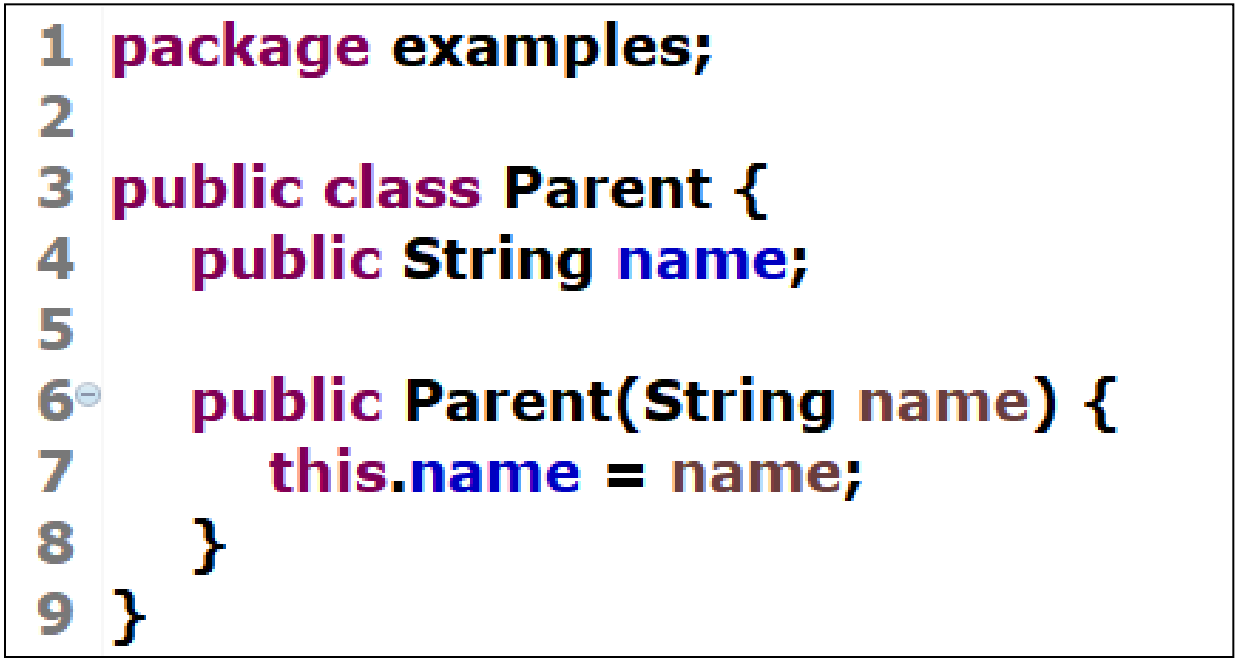
```java
    //Account 배열에서 ano와 동일한 Account 객체 찾기
    private static Account findAccount(String ano) {
        Account account = null;
        for(int i=0; i<accountArray.length; i++) {
            if(accountArray[i] != null) {
                String dbAno = accountArray[i].getAno();
                if(dbAno.equals(ano)) {
                    account = accountArray[i];
                    break;
                }
            }
        }
        return account;
    }

}
```

```
------------------------------------------------------------
1.계좌생성 | 2.계좌목록 | 3.예금 | 4.출금 | 5.종료
------------------------------------------------------------
선택> 1
--------------
계좌생성
--------------
계좌번호: 110-111-1234
계좌주: 홍길동
초기입금액: 1000
결과: 계좌가 생성되었습니다.
------------------------------------------------------------
1.계좌생성 | 2.계좌목록 | 3.예금 | 4.출금 | 5.종료
------------------------------------------------------------
선택> 2
--------------
계좌목록
--------------
110-111-1234    홍길동    1000
------------------------------------------------------------
1.계좌생성 | 2.계좌목록 | 3.예금 | 4.출금 | 5.종료
------------------------------------------------------------
선택>
```

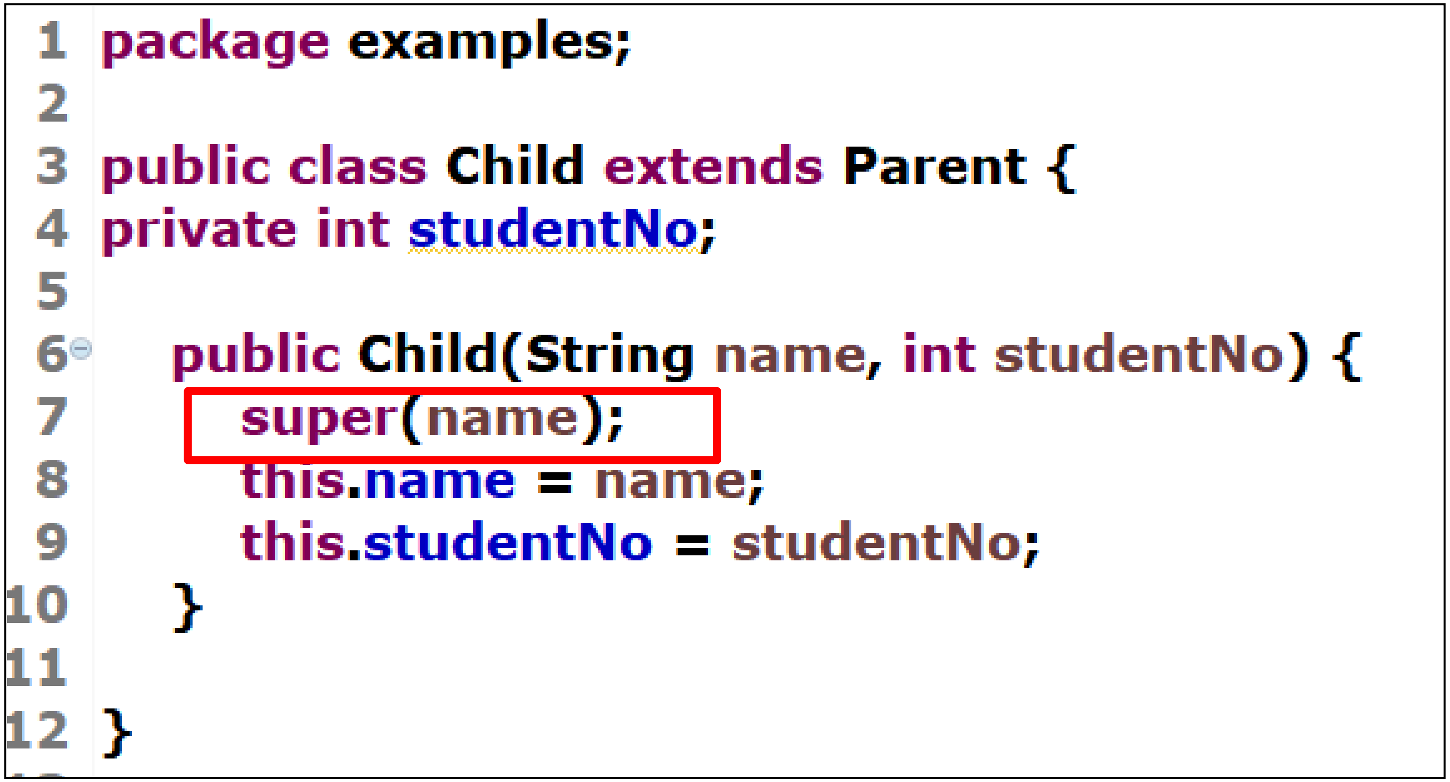
문제2 : 교재 7장 인터페이스 확인문제 1~8번

1. 1
2. 2
3. 1
4. 4

5



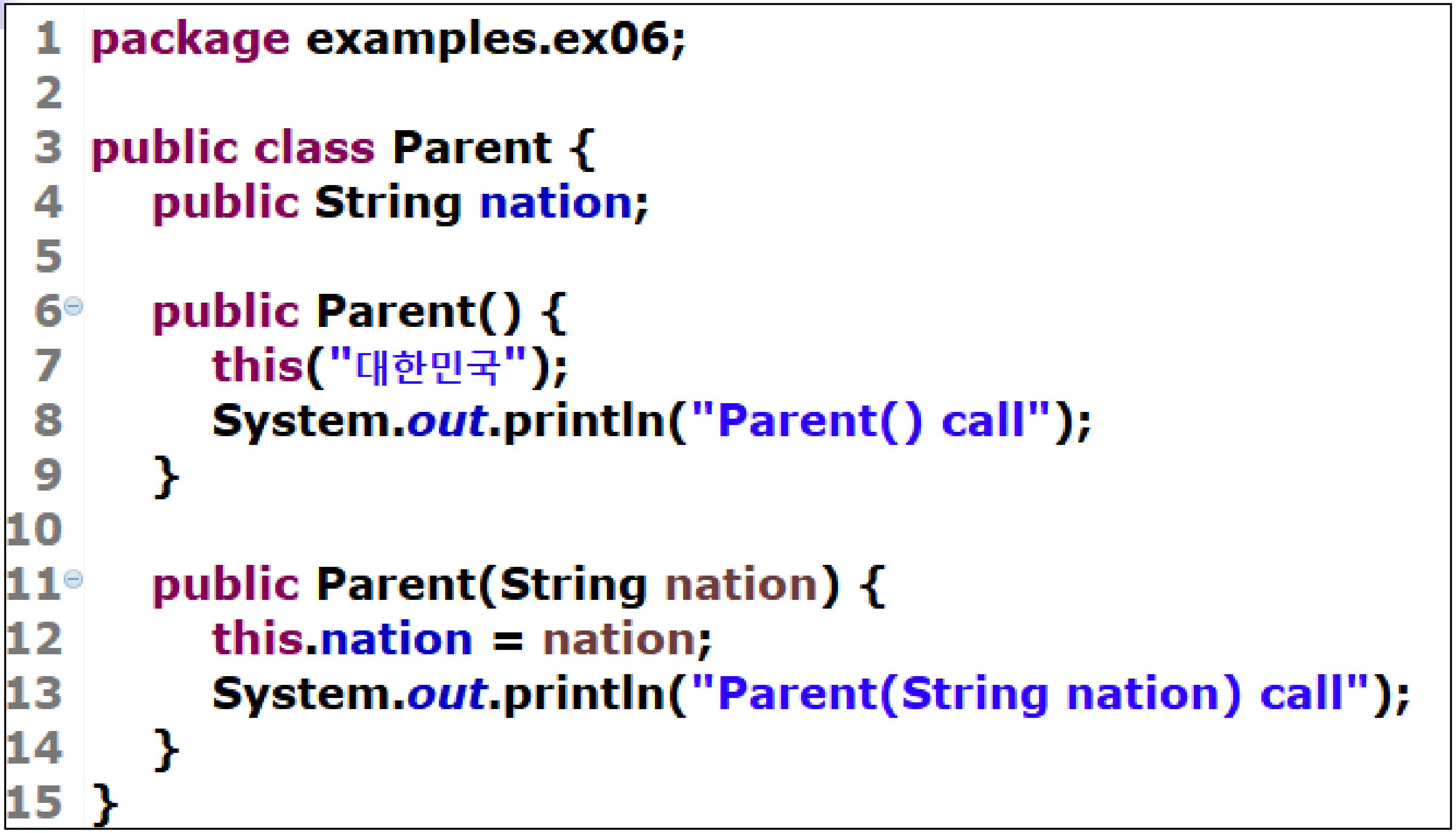
```
package examples;

public class Parent {
    public String name;

    public Parent(String name) {
        this.name = name;
    }
}
```



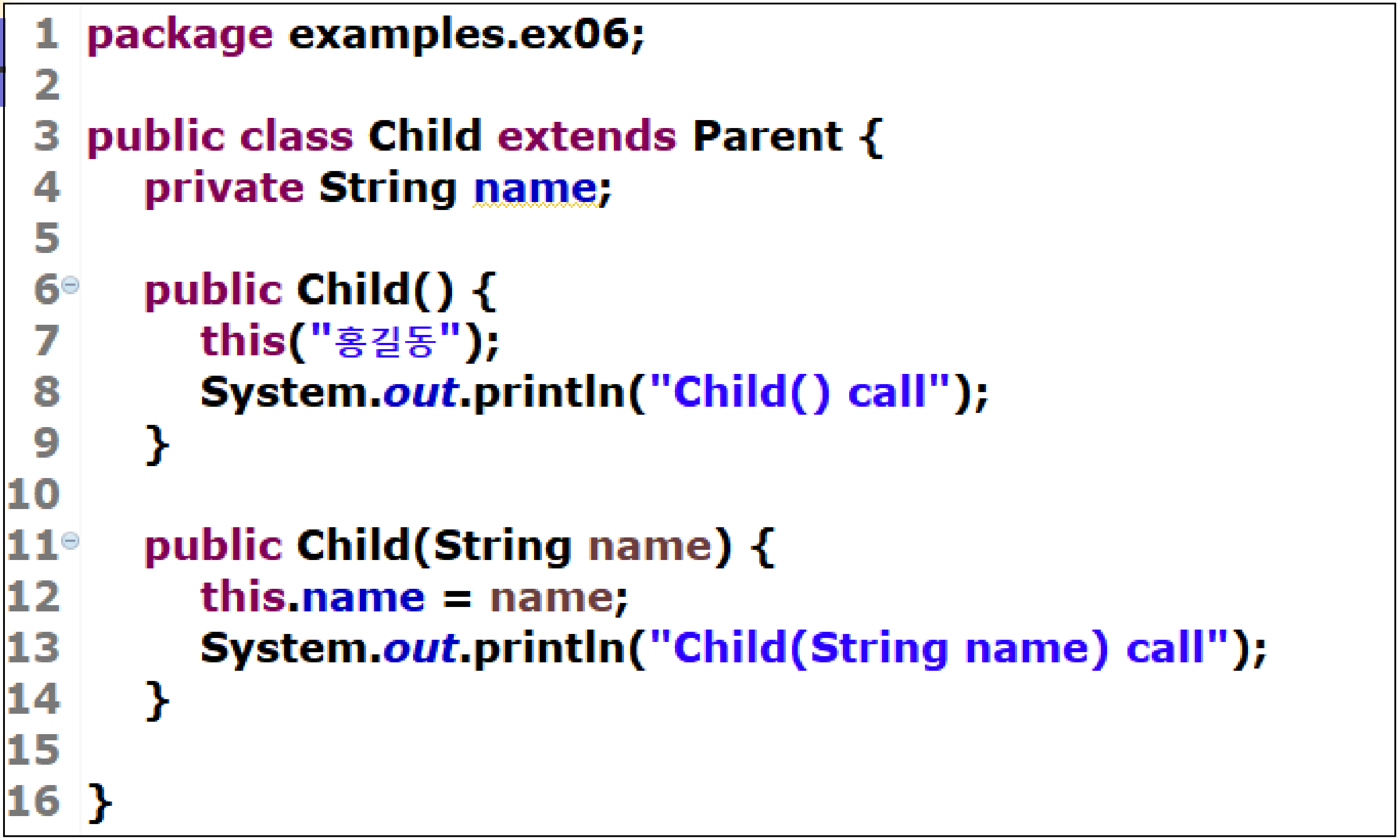
```
package examples;

public class Child extends Parent {
private int studentNo;

    public Child(String name, int studentNo) {
        super(name);
        this.name = name;
        this.studentNo = studentNo;
    }

}
```

5. 자식 클래스에서 부모 생성자를 호출하지 않았기 때문

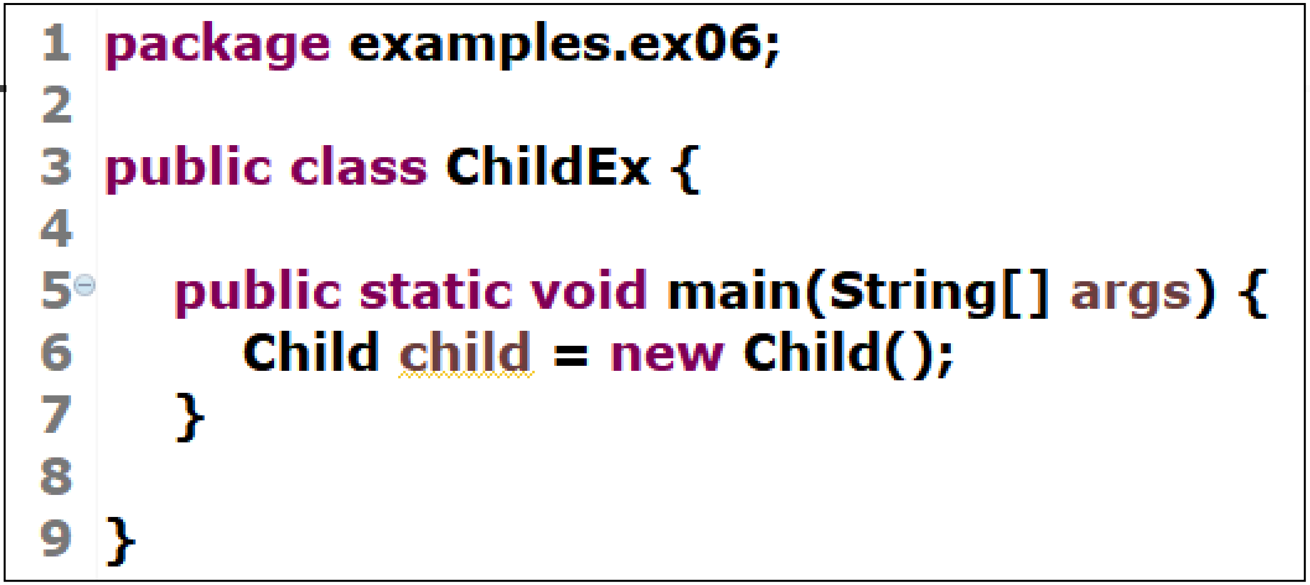
6



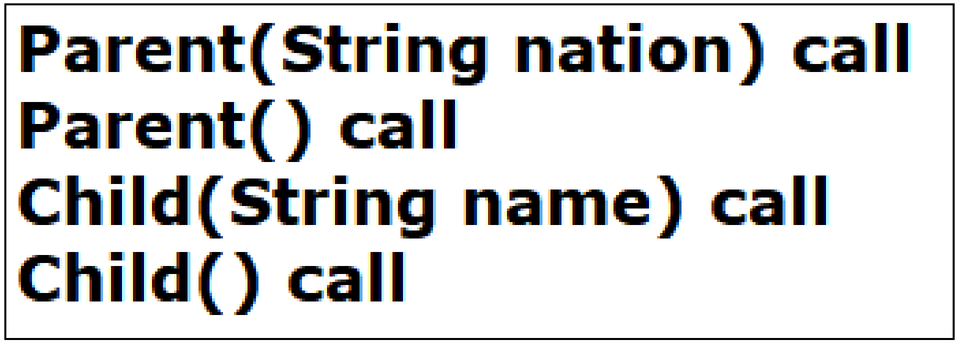
```java
package examples.ex06;

public class Parent {
    public String nation;

    public Parent() {
        this("대한민국");
        System.out.println("Parent() call");
    }

    public Parent(String nation) {
        this.nation = nation;
        System.out.println("Parent(String nation) call");
    }
}
```



```java
package examples.ex06;

public class Child extends Parent {
    private String name;

    public Child() {
        this("홍길동");
        System.out.println("Child() call");
    }

    public Child(String name) {
        this.name = name;
        System.out.println("Child(String name) call");
    }

}
```

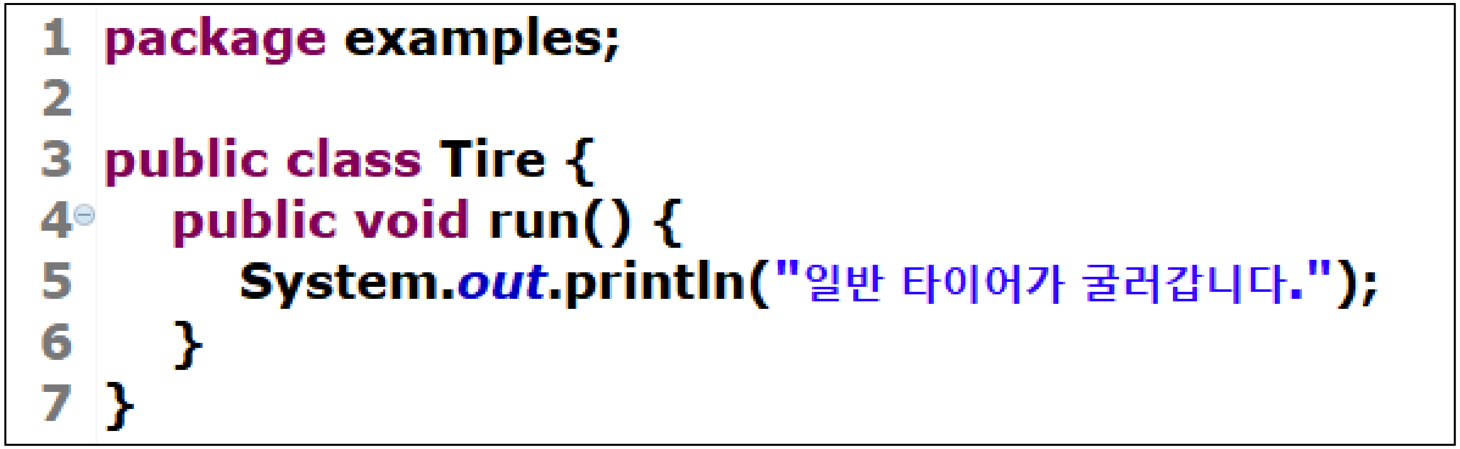


```java
package examples.ex06;

public class ChildEx {

    public static void main(String[] args) {
        Child child = new Child();
    }

}
```

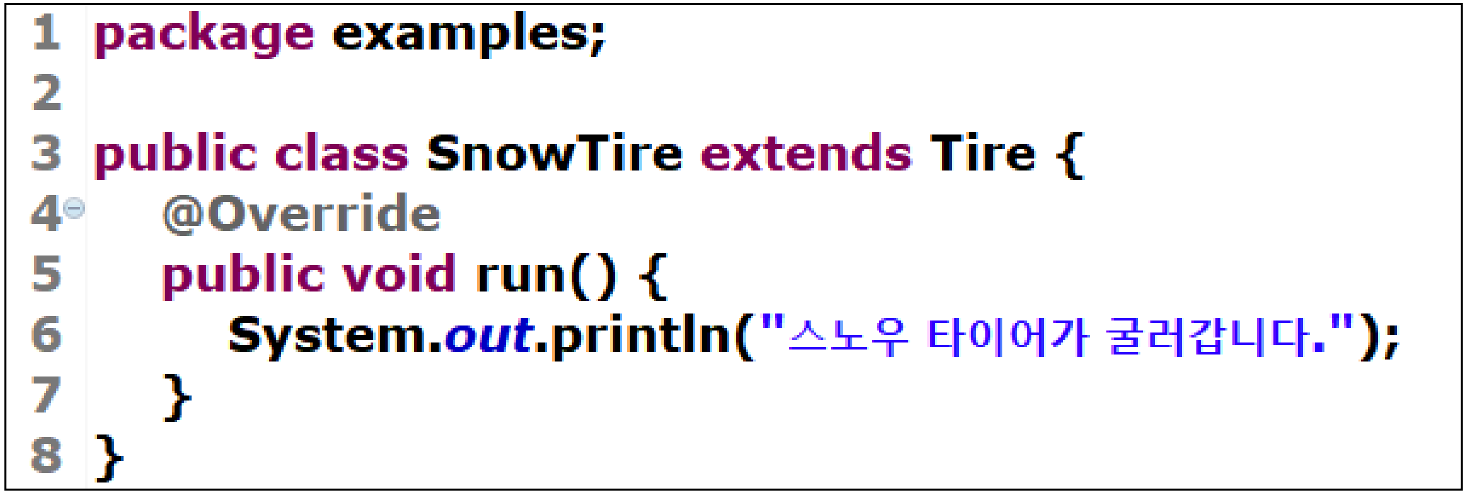


```
Parent(String nation) call
Parent() call
Child(String name) call
Child() call
```

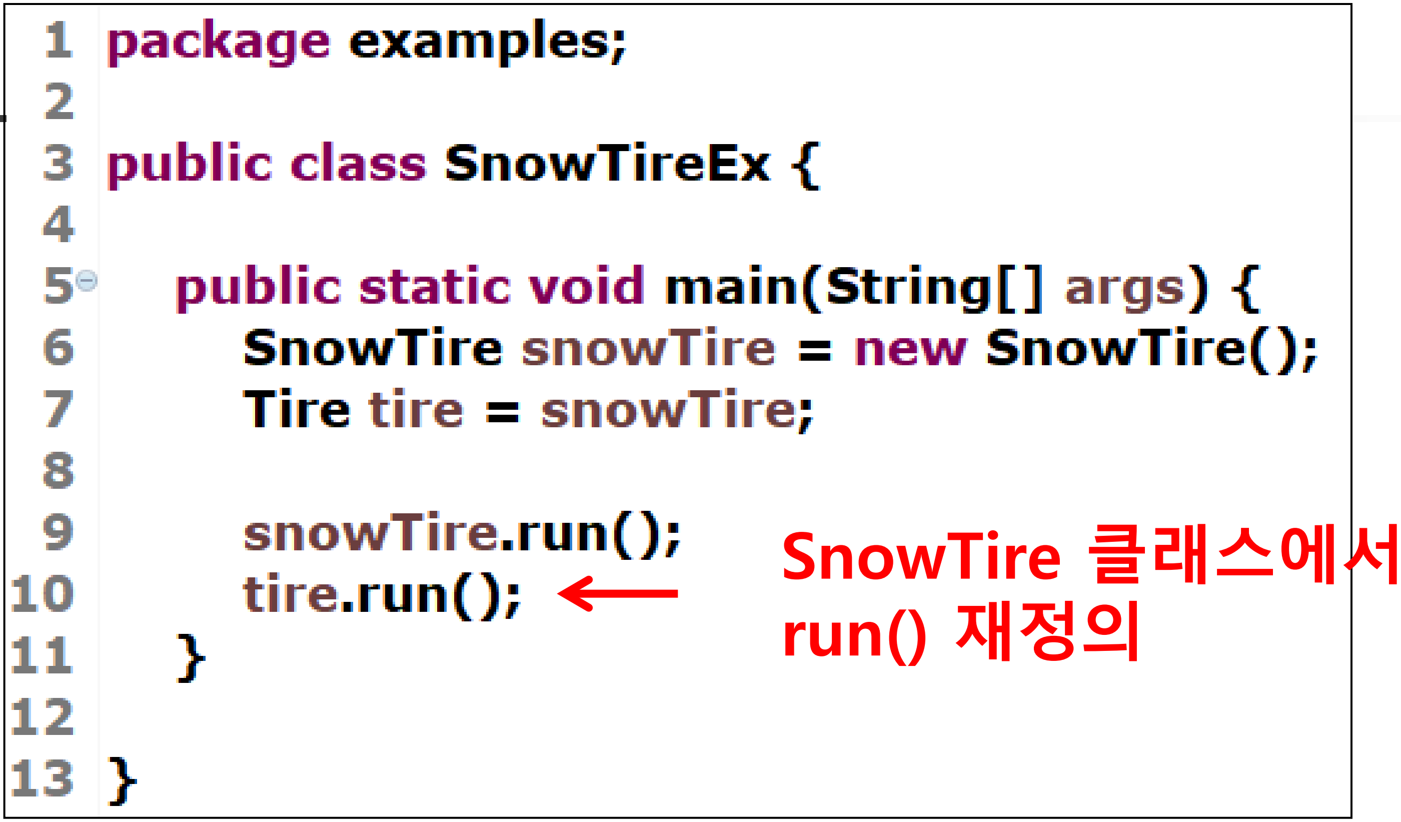
7



```java
package examples;

public class Tire {
    public void run() {
        System.out.println("일반 타이어가 굴러갑니다.");
    }
}
```

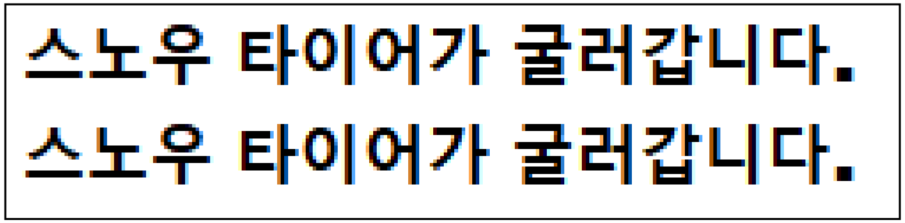


```java
package examples;

public class SnowTire extends Tire {
    @Override
    public void run() {
        System.out.println("스노우 타이어가 굴러갑니다.");
    }
}
```



```java
package examples;

public class SnowTireEx {

    public static void main(String[] args) {
        SnowTire snowTire = new SnowTire();
        Tire tire = snowTire;

        snowTire.run();
        tire.run();
    }

}
```

SnowTire 클래스에서 run() 재정의



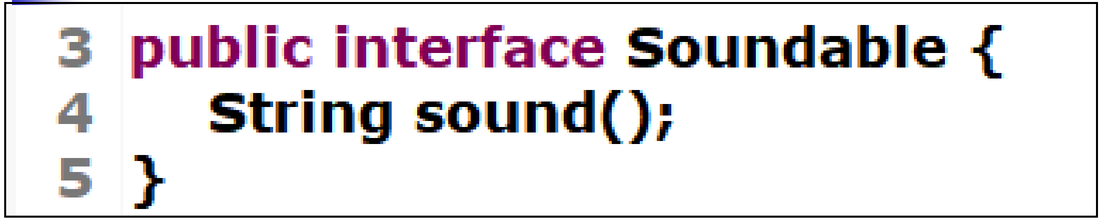
```
스노우 타이어가 굴러갑니다.
스노우 타이어가 굴러갑니다.
```

8. 2

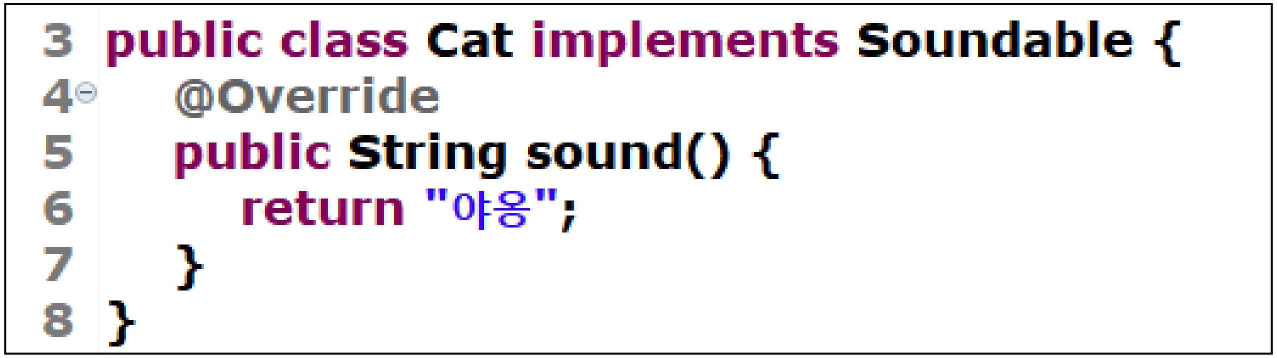
문제3 : 교재 8장 인터페이스 확인문제 1~5번

1. 3
2. 4

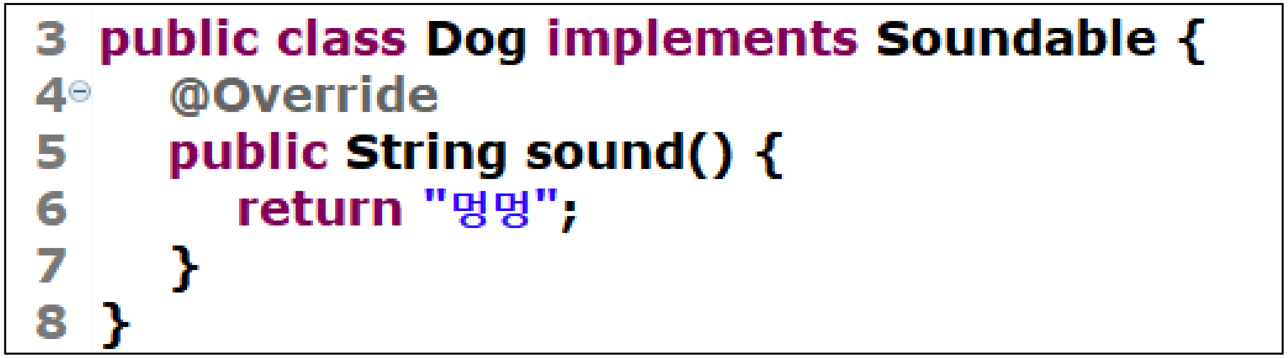
3



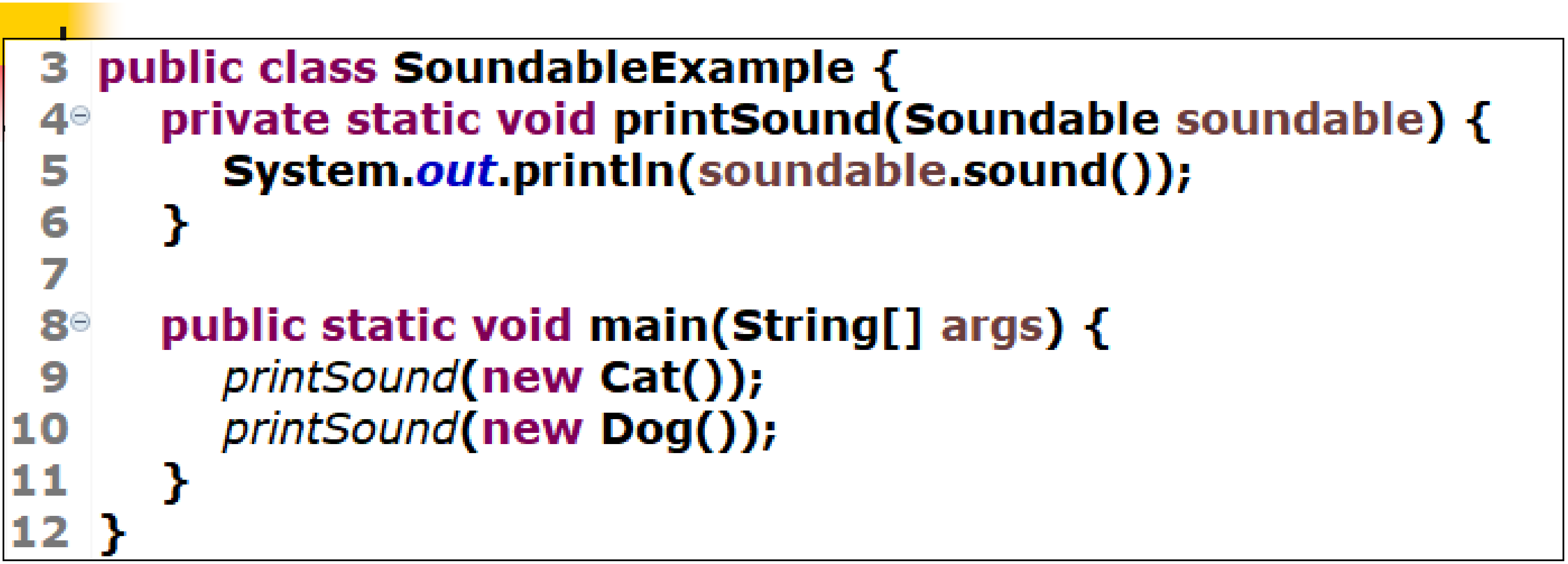
```java
public interface Soundable {
    String sound();
}
```



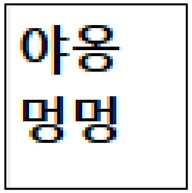
```java
public class Cat implements Soundable {
    @Override
    public String sound() {
        return "야옹";
    }
}
```



```java
public class Dog implements Soundable {
    @Override
    public String sound() {
        return "멍멍";
    }
}
```

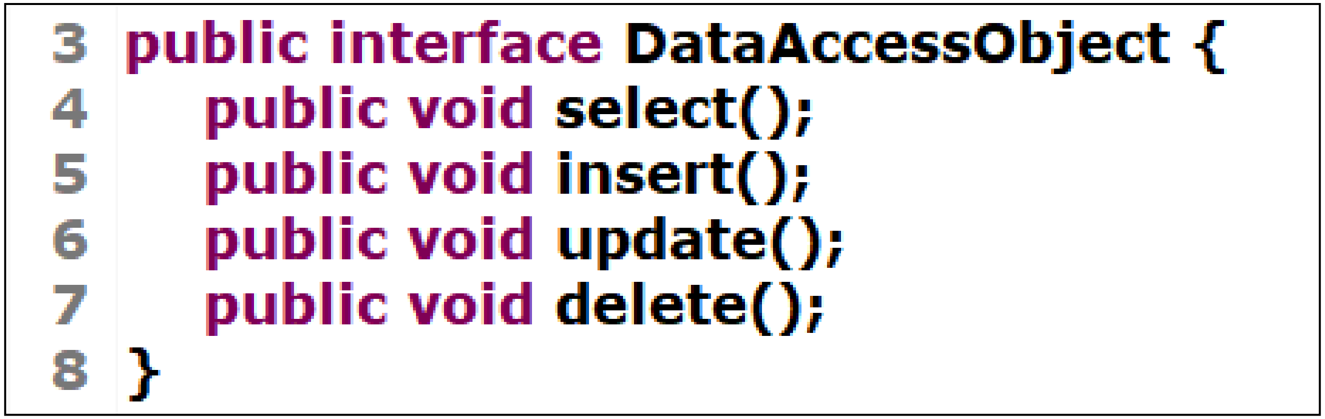


```java
public class SoundableExample {
    private static void printSound(Soundable soundable) {
        System.out.println(soundable.sound());
    }

    public static void main(String[] args) {
        printSound(new Cat());
        printSound(new Dog());
    }
}
```

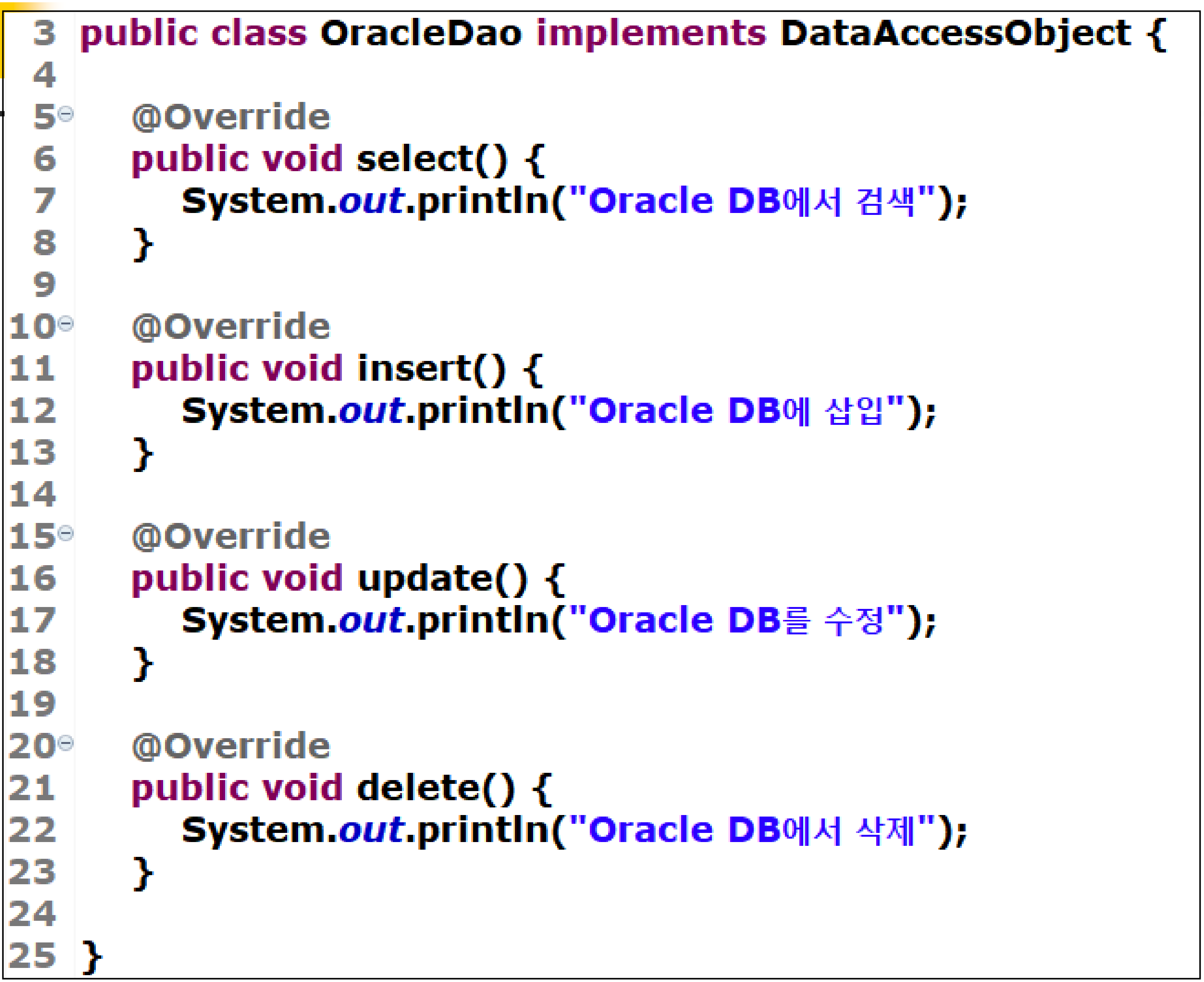


```
야옹
멍멍
```

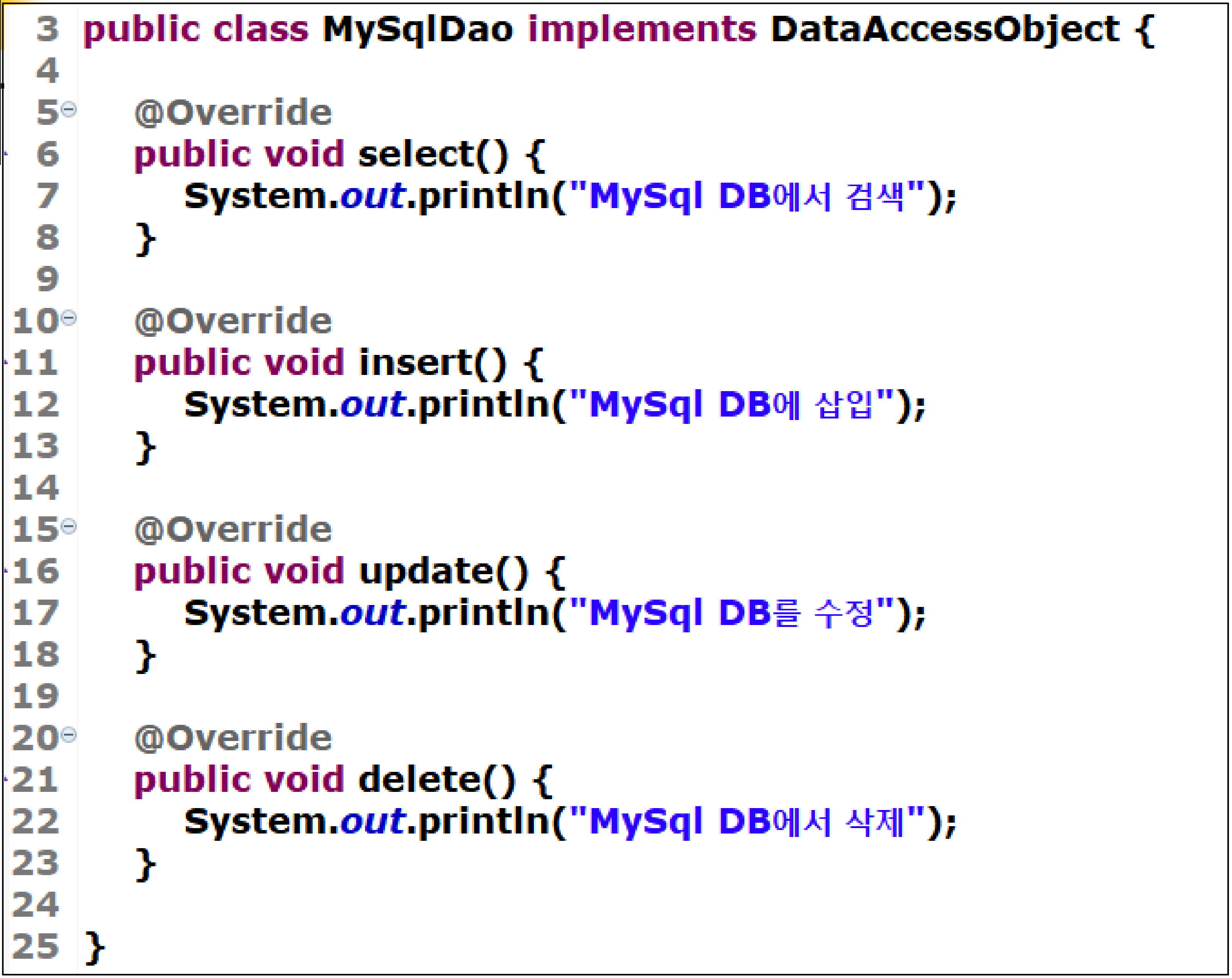
4



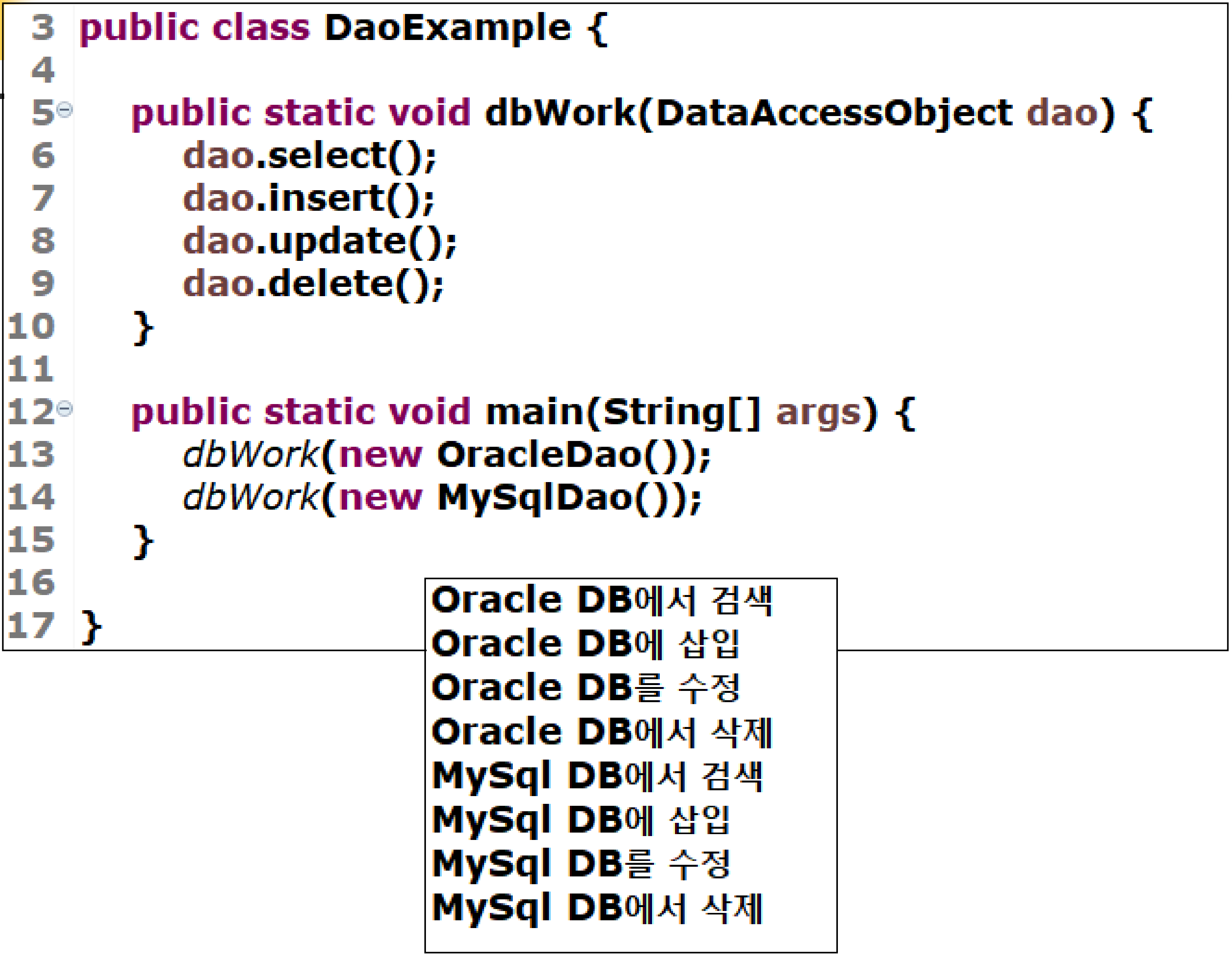
```java
public interface DataAccessObject {
    public void select();
    public void insert();
    public void update();
    public void delete();
}
```



```java
public class OracleDao implements DataAccessObject {

    @Override
    public void select() {
        System.out.println("Oracle DB에서 검색");
    }

    @Override
    public void insert() {
        System.out.println("Oracle DB에 삽입");
    }

    @Override
    public void update() {
        System.out.println("Oracle DB를 수정");
    }

    @Override
    public void delete() {
        System.out.println("Oracle DB에서 삭제");
    }

}
```



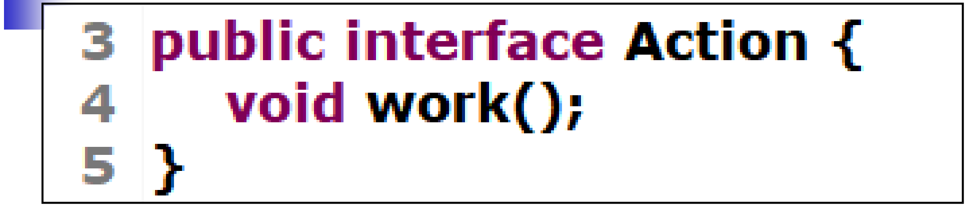
```java
public class MySqlDao implements DataAccessObject {

    @Override
    public void select() {
        System.out.println("MySql DB에서 검색");
    }

    @Override
    public void insert() {
        System.out.println("MySql DB에 삽입");
    }

    @Override
    public void update() {
        System.out.println("MySql DB를 수정");
    }

    @Override
    public void delete() {
        System.out.println("MySql DB에서 삭제");
    }

}
```



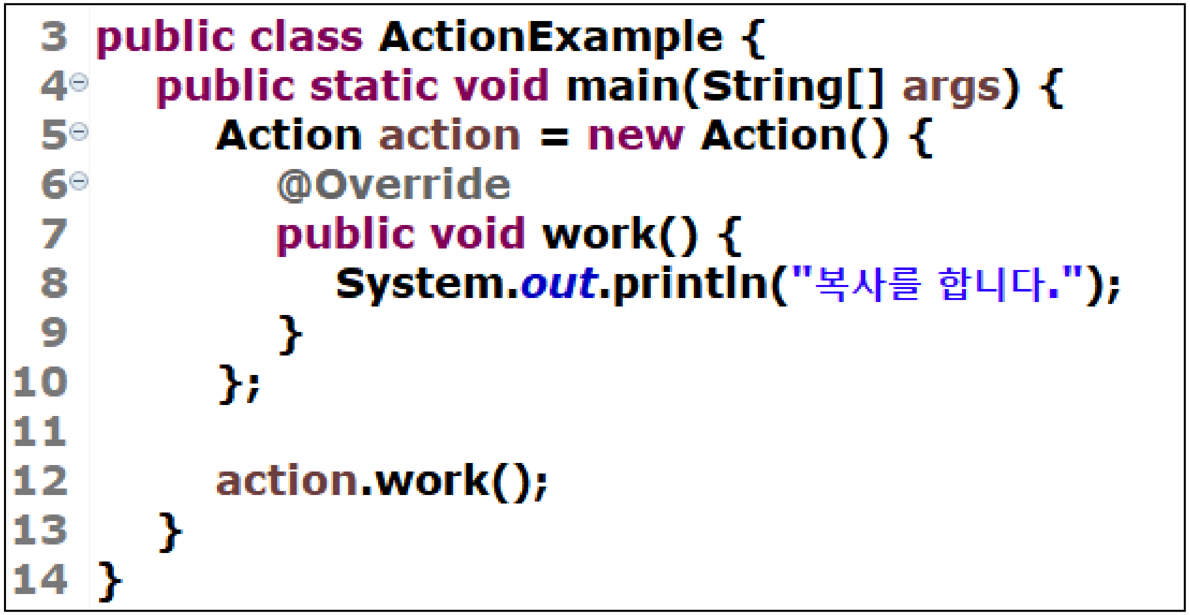
```java
public class DaoExample {

    public static void dbWork(DataAccessObject dao) {
        dao.select();
        dao.insert();
        dao.update();
        dao.delete();
    }

    public static void main(String[] args) {
        dbWork(new OracleDao());
        dbWork(new MySqlDao());
    }

}
```

```
Oracle DB에서 검색
Oracle DB에 삽입
Oracle DB를 수정
Oracle DB에서 삭제
MySql DB에서 검색
MySql DB에 삽입
MySql DB를 수정
MySql DB에서 삭제
```

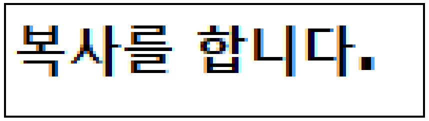
5



```java
public interface Action {
    void work();
}
```



```java
public class ActionExample {
    public static void main(String[] args) {
        Action action = new Action() {
            @Override
            public void work() {
                System.out.println("복사를 합니다.");
            }
        };

        action.work();
    }
}
```



```
복사를 합니다.
```

문제4

- 다음 프로그램의 출력 결과는?

```
int x = 0;
while(++x < 10){
    System.out.println(x);
}
```

```
1
2
3
4
5
6
7
8
9
```

x값 증가 후 비교

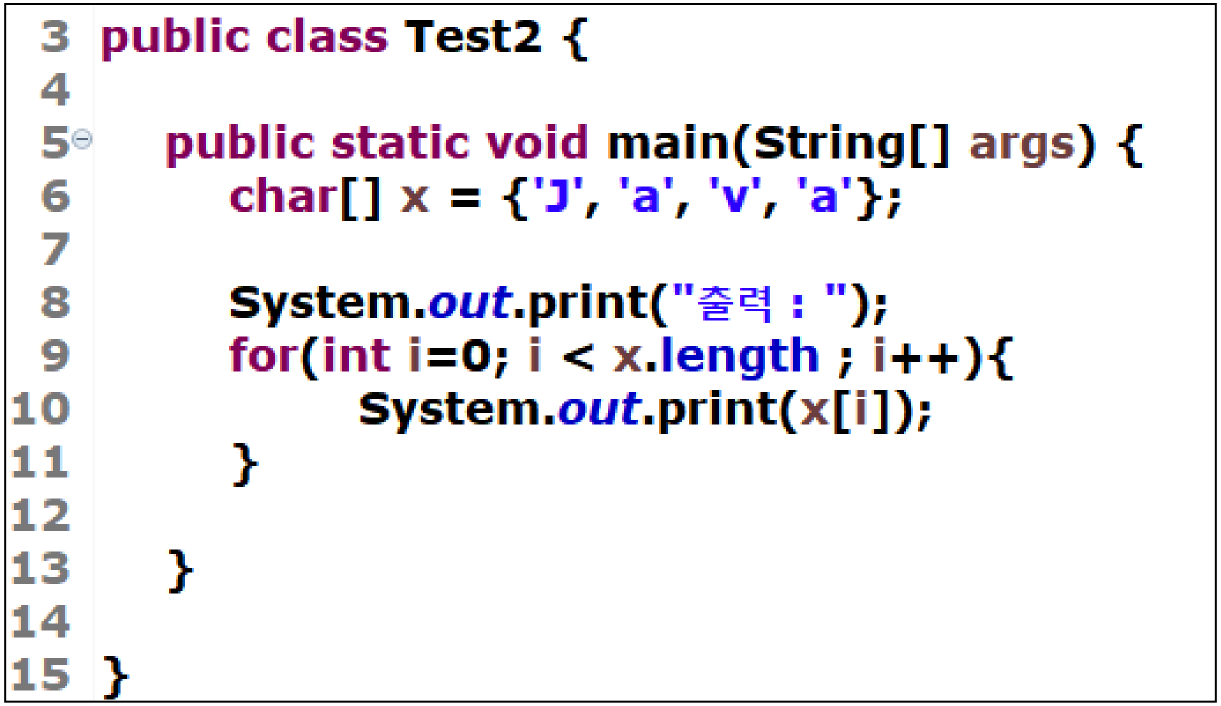
```
int x = 0;
while(x++ < 10){
    System.out.println(x);
}
```

```
1
2
3
4
5
6
7
8
9
10
```

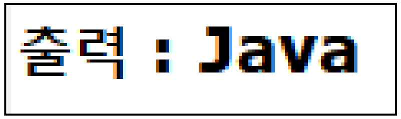
x값 비교 후 증가

문제5

- 크기가 4인 char 타입의 배열 x를 선언하고 다음과 같이 출력되도록 값 초기화
  - 출력 : Java



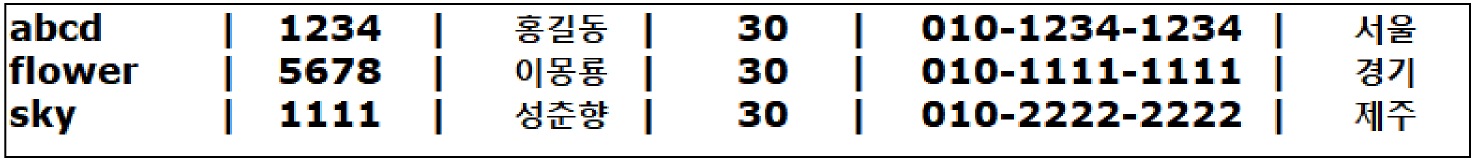
```java
public class Test2 {

    public static void main(String[] args) {
        char[] x = {'J', 'a', 'v', 'a'};

        System.out.print("출력 : ");
        for(int i=0; i < x.length ; i++){
            System.out.print(x[i]);
        }

    }

}
```



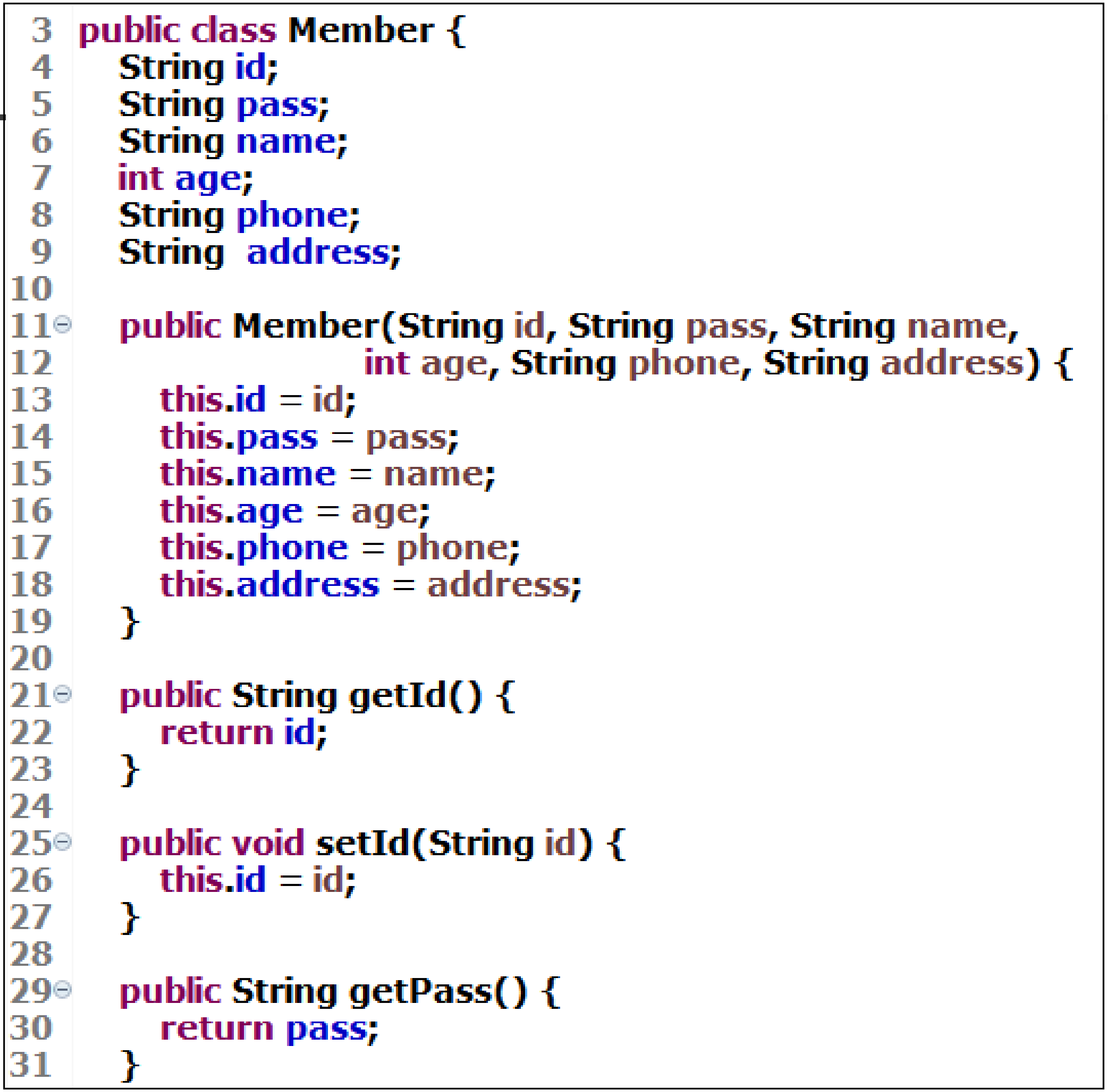
```
출력 : Java
```

문제6

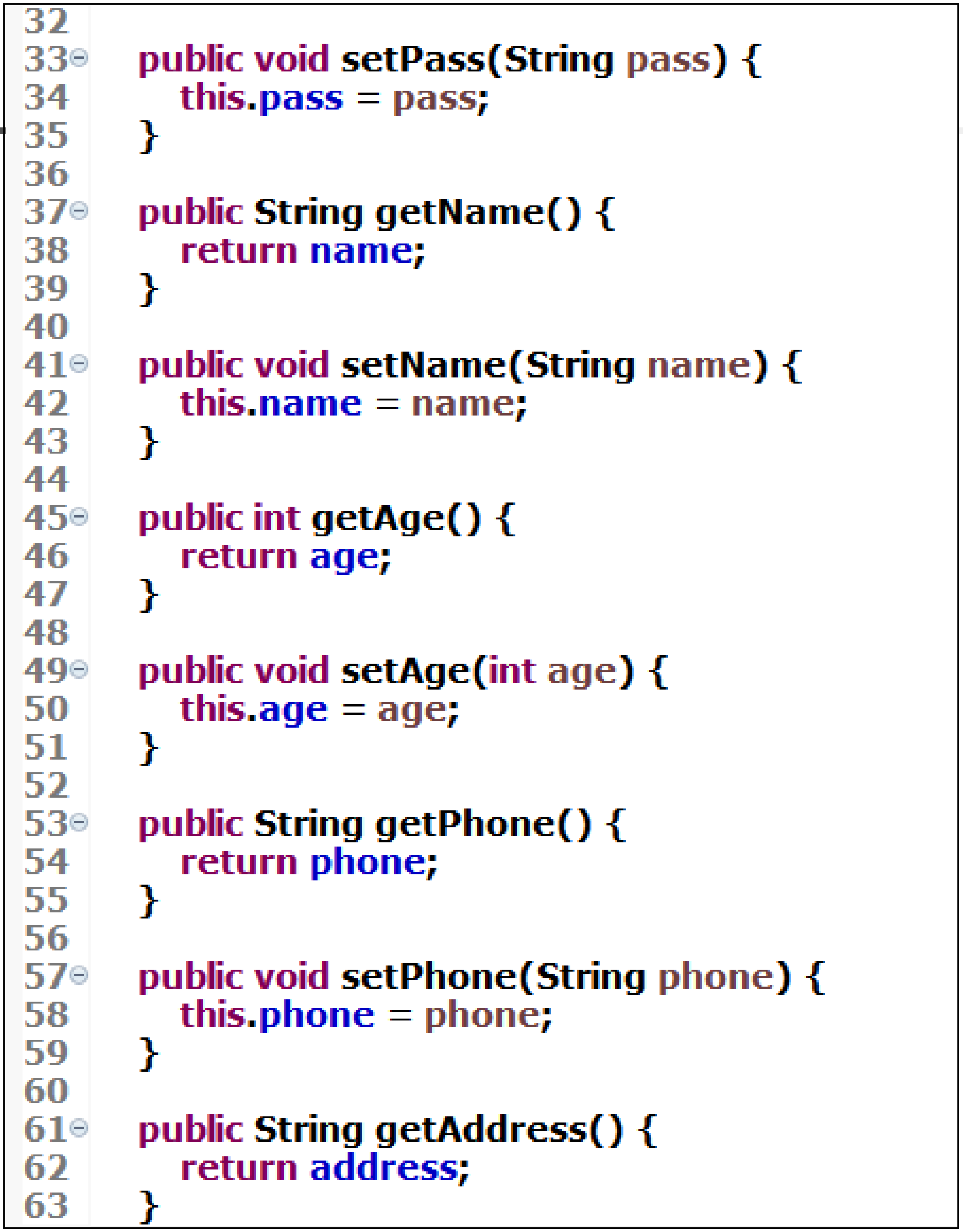
- 다음과 같이 회원 정보를 저장하고 출력하는 프로그램 작성
  - 회원 클래스 : Member
  - main() : 데이터 생성자에게 전달하고 출력
  - Getters/Setters, toString() 사용



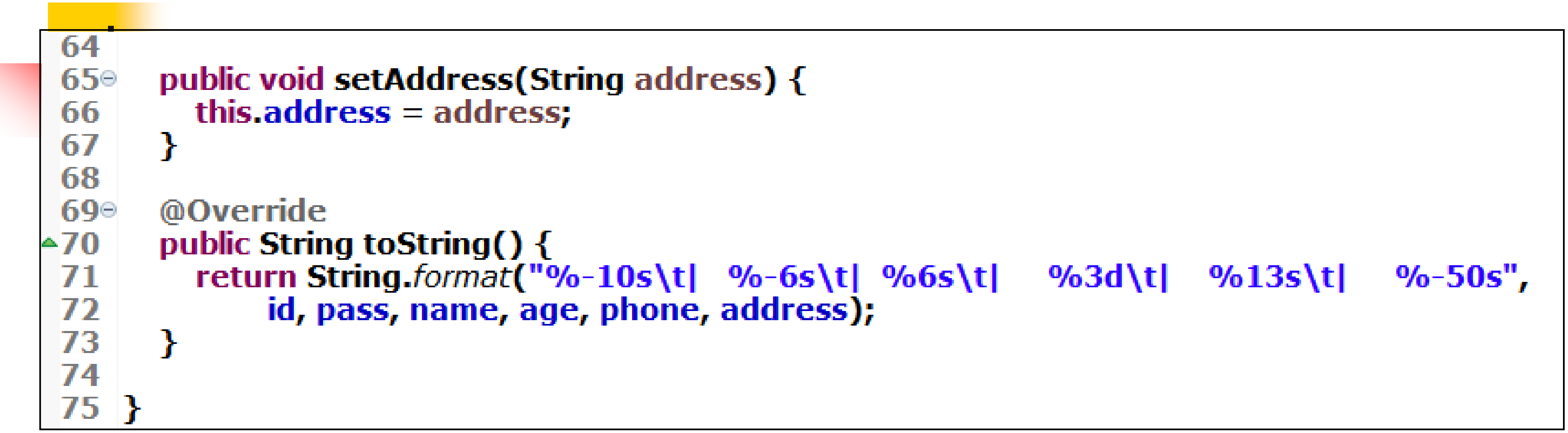
```
abcd      |  1234  |  홍길동 |   30   |  010-1234-1234 |   서울
flower    |  5678  |  이몽룡 |   30   |  010-1111-1111 |   경기
sky       |  1111  |  성춘향 |   30   |  010-2222-2222 |   제주
```



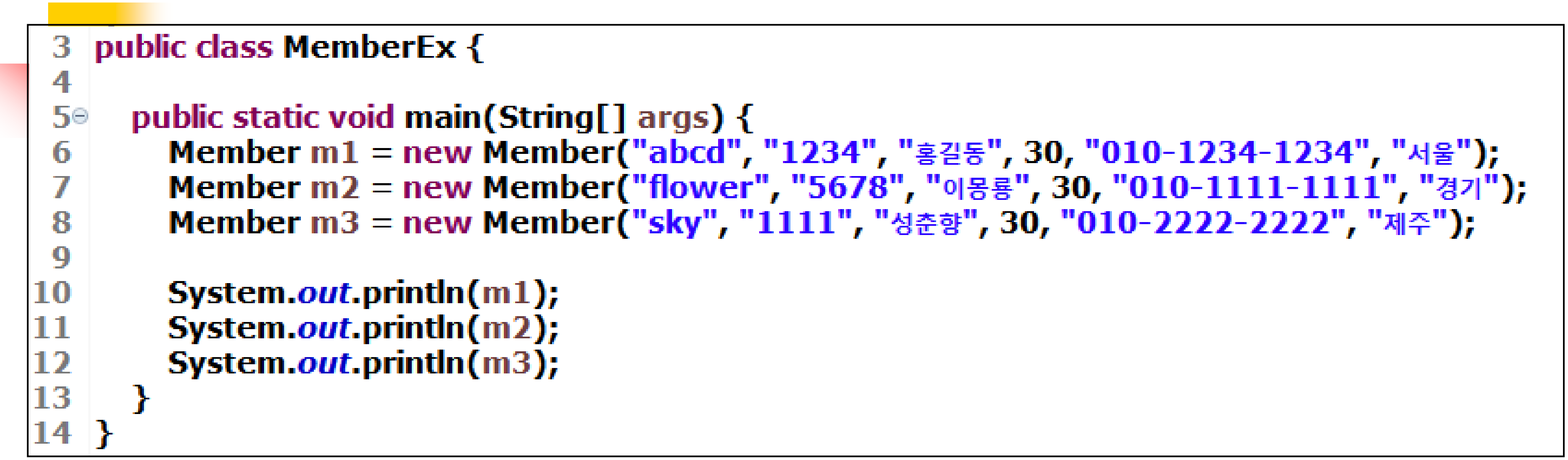
```java
public class Member {
    String id;
    String pass;
    String name;
    int age;
    String phone;
    String  address;

    public Member(String id, String pass, String name,
                            int age, String phone, String address) {
        this.id = id;
        this.pass = pass;
        this.name = name;
        this.age = age;
        this.phone = phone;
        this.address = address;
    }

    public String getId() {
        return id;
    }

    public void setId(String id) {
        this.id = id;
    }

    public String getPass() {
        return pass;
    }
```



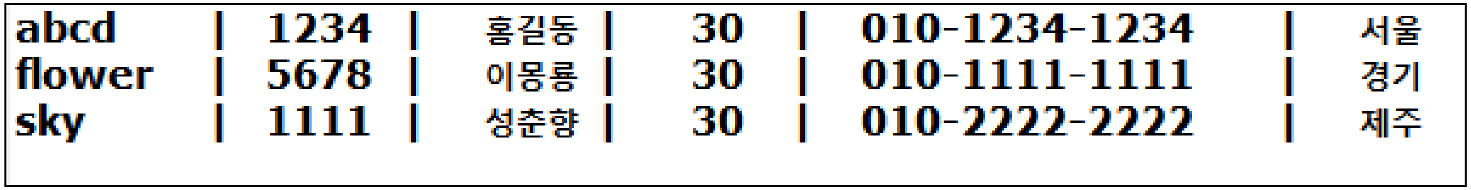
```java

    public void setPass(String pass) {
        this.pass = pass;
    }

    public String getName() {
        return name;
    }

    public void setName(String name) {
        this.name = name;
    }

    public int getAge() {
        return age;
    }

    public void setAge(int age) {
        this.age = age;
    }

    public String getPhone() {
        return phone;
    }

    public void setPhone(String phone) {
        this.phone = phone;
    }

    public String getAddress() {
        return address;
    }
```



```java

    public void setAddress(String address) {
        this.address = address;
    }

    @Override
    public String toString() {
        return String.format("%-10s\t|  %-6s\t| %6s\t|   %3d\t|  %13s\t|   %-50s",
                id, pass, name, age, phone, address);
    }

}
```



```java
public class MemberEx {

    public static void main(String[] args) {
        Member m1 = new Member("abcd", "1234", "홍길동", 30, "010-1234-1234", "서울");
        Member m2 = new Member("flower", "5678", "이몽룡", 30, "010-1111-1111", "경기");
        Member m3 = new Member("sky", "1111", "성춘향", 30, "010-2222-2222", "제주");

        System.out.println(m1);
        System.out.println(m2);
        System.out.println(m3);
    }
}
```



```
abcd      |  1234  |   홍길동 |   30   |  010-1234-1234   |   서울
flower    |  5678  |   이몽룡 |   30   |  010-1111-1111   |   경기
sky       |  1111  |   성춘향 |   30   |  010-2222-2222   |   제주
```