

2025년 상반기 K-디지털 트레이닝

예외 처리 (심화1)

[KB] IT's Your Life



예외 처리

◎ 계좌 클래스를 정의하고자 한다. 잔액이 부족할 때 발생할 사용자 정의 예외를 정의하세요.

```
package ch11.sec06;
public class InsufficientException {
}
```

InsufficientException.java

```
package ch11.sec06;

public class InsufficientException extends Exception {
   public InsufficientException() {
   }

   public InsufficientException(String message) {
      super(message);
   }
}
```

- ♡ 계좌 클래스의 기본 코드이다. withdraw() 메서드를 완성하세요.
 - 인출 금액(money)가 잔액보다 크면 InsufficientException 예외 발생
 - 인출 금액(money)가 잔액보다 작으면 잔액에서 차감

```
package ch11.sec06;
public class Account {
 private long balance;
 public Account() { }
 public long getBalance() {
   return balance;
 public void deposit(int money) {
   balance += money;
 public void withdraw(int money) {
```

Account.java

```
package ch11.sec06;
public class Account {
 private long balance;
 public Account() { }
 public long getBalance() {
   return balance;
 public void deposit(int money) {
   balance += money;
 public void withdraw(int money) throws InsufficientException {
   if(balance < money) {</pre>
    throw new InsufficientException("잔고 부족: "+(money-balance)+" 모자람");
   balance -= money;
```

♡ 다음 순서로 Account 클래스 운영하세요.

- o Account 클래스의 인스턴스 생성
- 10000원 예금
- 5000원 인출
- 7000원 인출

```
package ch11.sec06;

public class AccountExample {
   public static void main(String[] args) {
   }
}
```

예외 처리

AccountExample.java

잔고 부족: 20000 모자람

```
package ch11.sec06;
public class AccountExample {
 public static void main(String[] args) {
   Account account = new Account();
   //예금하기
   account.deposit(10000);
   System.out.println("예금액: " + account.getBalance());
   //출금하기
   try {
    account.withdraw(5000);
    account.withdraw(7000);
   } catch(InsufficientException e) {
    String message = e.getMessage();
    System.out.println(message);
예금액: 10000
```

7



2025년 상반기 K-디지털 트레이닝

java.base 모듈 (심화2)

[KB] IT's Your Life



java.base 모듈

💟 다음 조건을 만족하는 프로그램을 작성하세요.

- 난수 추출은 Random 클래스를 이용
- 1~45 범위의 랜덤한 숫자 6개 선택한 후 출력(선택한 번호)
- 1~45 범위의 랜덤한 숫자 6개 선택한 후 출력(당첨 번호)
- 두 숫자 그룹을 각각 정렬
- 두 그룹 내용의 동일 여부에 따라 1등 당첨여부를 출력하세요.

```
package ch12.sec07;

public class RandomExample {
   public static void main(String[] args) {
   }
}
```

java.base 모듈

RandomExample.java

```
package ch12.sec07;
import java.util.Arrays;
import java.util.Random;
public class RandomExample {
 public static void main(String[] args) {
  //선택번호
  int[] selectNumber = new int[6];
  Random random = new Random(3);
  System.out.print("선택번호: ");
  for(int i=0; i<6; i++) {
    selectNumber[i] = random.nextInt(45) + 1;
    System.out.print(selectNumber[i] + " ");
  System.out.println();
```

선택번호: 15 21 16 17 34 28

RandomExample.java

```
//당첨번호
int[] winningNumber = new int[6];
random = new Random(5);
System.out.print("당첨번호: ");
for(int i=0; i<6; i++) {
 winningNumber[i] = random.nextInt(45) + 1;
 System.out.print(winningNumber[i] + " ");
System.out.println();
//당첨여부, 비교하기 전에 배열 항목을 정렬
Arrays.sort(selectNumber);
Arrays.sort(winningNumber);
boolean result = Arrays.equals(selectNumber, winningNumber); // 배열 항목 비교
System.out.print("당첨여부: ");
if(result) {
 System.out.println("1등에 당첨되셨습니다.");
                                            당첨번호: 18 38 45 15 22 36
} else {
                                            당첨여부: 당첨되지 않았습니다.
 System.out.println("당첨되지 않았습니다.");
```

java.base 모듈

- ☑ SimpleDateFormat을 이용하여 다음과 같이 출력되도록 하세요.
 - 실제 시간/날짜는 현재 시간/날짜로 한다.

```
package ch12.sec09;

public class SimpleDateFormatExample {
   public static void main(String[] args) {
      Date now;
   }
}
```

```
2024-01-18
2024년 01월 18일
2024.01.18 오후 12:38:11
오늘은 목요일
```

SimpleDateFormatExample.java

```
package ch12.sec09;
import java.text.SimpleDateFormat;
import java.util.Date;
public class SimpleDateFormatExample {
 public static void main(String[] args) {
  Date now = new Date();
   SimpleDateFormat sdf = new SimpleDateFormat("yyyy-MM-dd");
   System.out.println( sdf.format(now) );
   sdf = new SimpleDateFormat("yyyy년 MM월 dd일");
   System.out.println( sdf.format(now) );
   sdf = new SimpleDateFormat("yyyy.MM.dd a HH:mm:ss");
   System.out.println( sdf.format(now) );
                                             2024-01-18
   sdf = new SimpleDateFormat("오늘은 E요일"):
                                             2024년 01월 18일
   System.out.println( sdf.format(now) );
                                             2024.01.18 오후 12:38:11
                                             오늘은 목요일
```