**성명: 무키모프 함담**

**학번: 2019117366**

**학년: 2학년**

**학과: 과학기술대학 소프트웨어학과**

**Chapter 3**

**- Summary**

2. **continue**문은 반복문을 빠져나가지 않고 다음 반복으로 진행한다.

3. **break**문은 하나의 반복문을 빠져나갈 때 사용한다.

4. 배열 만드는 과정은 배열 레퍼런스 변수 선언과 **배열 생성의** 두 과정으로 이루어진다. 배열 레퍼런스 변수의 선언은 배열 메모리를 가리키는 레퍼런스(주소와 유사) 값을 가질 변수에 대한 선언이고 배열 생성은 배열 저장 공간을 확보하는 것이다.

5. 배열의 크기는 배열 객체의 **length** 필드에 저장되어 있으며, 배열의 인덱스는 0부터 시작한다. 배열은 그 크기를 넘어서 전근하면 실행 중 오류(예외)가 발생한다.

6. **for-each**문은 배열의 각 원소를 순차적으로 접근하는 특별한 for 문으로서, 정수 배열의 모든 원소를 출력하는 코드 사례는 다음과 같다.

int [] array = { 1,2,3,4 };

for(int k : array) { System.out.println(k); }

7. 실행 중 오동작이나 결과에 악영향을 미치는 예상치 못한 오류를 **예외**라고 한다.

8. 예가 발생할 가능성이 있는 코드는 **try { }** 블록으로 싸고, 예외가 발생할 때 처리하는 코드는 catch () { } 블록으로 작성하고, 예외가 발생하든 안 든 최종적으로 실행할 코드는 finally { } 블록에 작성한다. finally { } 블록은 생략 가능하다.

**- 이론문제**

2. 다음 for문에 대해 물음에 답하라.

double sum = 0.0;

double d[] = { 1.0, 2.3, 3.4, 5.5 };

for( int i = 0; i<4; i++) sum += d[i];

System.out.println(sum);

(1) 예상되는 실행 결과는 무엇인가?

12.2

(2) for(int i=0; i<4; i++) 부분을 배열의 length 필드를 이용하여 수정하라.

for (int i = 0; i < d.length; i++) {

sum += d[i];

}

(3) while 문으로 바꾸어 작성하라.

double sum = 0.0;

double[] d = { 1.0, 2.3, 3.4, 5.5 };

int i = 0;

while (i < 4) {

sum += d[i];

i++;

}

System.out.println(sum);

(4) do-while 문으로 바꾸어 작성하라.

double sum = 0.0;

double[] d = { 1.0, 2.3, 3.4, 5.5 };

int i = 0;

do {

sum += d[i];

i++;

} while (i < 4);

System.out.println(sum);

(5) for-each 문으로 바꾸어 작성하라.

double sum = 0.0;

double[] d = { 1.0, 2.3, 3.4, 5.5 };

for(double i : d) {

sum +=i;

}

System.out.println(sum);

4. 다음 코드에 대해 실행 중 오류가 발생하는 보기는 ?

int myArray [] = { 1,2,3,4,5 };

(2) myArray[myArray.length] = 100;

6. for-each 문을 이용하여 배열 b를 모두 출력하고자 한다. 빈칸에 적절한 코드를 삽입하라.

boolean [ ] b = { true, false, true, true };

for ( boolean boo : b )

System.out.println(boo)

7. 다음은 사용자로부터 배열의 개수를 얻어서 배열을 생성하여 리턴하는 allocArray( ) 메소드를 작성하고자 한다.

(1) 빈칸을 적절한 코드로 채워라.

double [ ] allocArray ( ) {

Scanner scanner = new Scanner(System.in);

double [ ] n = new double [scanner.nextInt()];

return n;

}

(2) allocArray( )를 호출하여 배열을 전달받는 한 줄의 코드를 작성하라.

double d [ ] = allocArray( );

8. 다음 코드에 대해 물음에 답하라.

String s = “123”;

try {

int n = Integer.parseInt(s);

n++;

System.out.println(n);

} catch ( NumberFormatException e ) {

System.out.println( s + “를 정수로 변환할 수 없습니다.” );

} finally {

System.out.println(“계산을 끝냅니다.”);

}

Integer.parseInt( String s )는 문자열 s를 정수로 변환하여 리턴하는 메소드이다. 만일 s가 정수로 변환할 수 없는 문자열이면 NumberFormatException 예외가 발생한다.

(1) 코드를 실행한 결과 출력되는 내용은.

124

계산을 끝냅니다.

(2) s가 “23.5”일 대 앞의 코드를 실행한 결과 출력되는 내용은.

23.5를 정수로 변환할 수 없습니다.

계산을 끝냅니다.