

# 자바기초실습 중간시험 문제 은행

## ▼ (1) 자바 언어 개요

1. 소스 코드 전체를 기계어로 변환하여 실행 파일을 만들어 주는데 사용하는 시스템 소프트웨어는? 4/5 퀴즈
2. 소스 코드를 한 줄씩 읽어서 기계어로 변환해 주는 시스템 소프트웨어는? 4/5 퀴즈
3. 다음 중 프로그래밍 언어가 아닌 것은?
  - a. C++
  - b. Java
  - c. HTML**
  - d. JavaScript
4. JDK와 JRE의 차이점을 설명한 것이다. 틀린 것은?
  - a. JRE는 자바 컴파일러를 포함한다**
  - b. JDK는 JRE를 포함한다
  - c. 자바 프로그램을 개발하려면 JDK가 필요하다
  - d. JRE는 자바가상머신을 포함하므로 자바 프로그램을 실행할 수 있다
5. JDK를 설치할 때 포함된 javac 등 도구를 어느 곳에서는 사용할 수 있게 하려면 무엇을 변경해야 하는가?
6. **자바 컴파일러는 자바 소스 코드를 무엇으로 변환하는가?** 4/5 퀴즈
7. 자바 프로그램의 동작을 명시하고 이러한 동작을 컴퓨터에 알려주는 데 사용되는 문장을 무엇이라고 하는지 영어로 쓰시오
8. 서로 관련된 클래스의 모음으로, 기능 기반으로 클래스를 구성하는데 사용되는 것은?
9. 모든 자바 애플리케이션의 시작점이자 진입점에 해당하는 메서드의 이름을 쓰세요 4/5 퀴즈
10. 다음 빈 칸에 들어갈 기호를 쓰시오
  - a. \_\_\_\_\_ : 한 행을 주석 처리
  - b. \_\_\_\_\_ : 여러 행을 주석 처리
11. 모든 자바 실행문은 무엇으로 끝나야 하는가?

12. 다음은 Hello World가 출력하는 코드이다.

- 다음 코드의 빈칸을 채우시오
- 다음의 소스 코드를 저장하기 위한 자바 파일명을 쓰시오
- 다음의 소스 코드를 컴파일하면 생성되는 파일명을 쓰시오

```
public class MyClass {  
    public static void main(String[] args) {  
        _____ ("Hello World");  
    }  
}
```

## ▼ (2) 기본 출력, 변수, 자료형, 연산자, 형변환

1. println 한 문장을 사용하여 다음과 같은 출력이 되도록 코드를 작성하세요

```
Java  
Program
```

2. 다음 코드의 실행 결과를 쓰세요

```
System.out.print("Welcome to Java");  
System.out.print("객체지향 프로그래밍 언어");
```

3. 다음 빈칸을 채우시오

- \_\_\_\_\_는 컴퓨터 프로그램이 처리할 데이터를 저장하는 위치이다 4/5 퀴즈
- \_\_\_\_\_는 데이터를 저장할 수 있는 메모리 공간이지만, 프로그램이 실행되는 동안 메모리에 저장된 데이터를 변경할 수 없고, 반드시 선언과 동시에 초기화해야 한다. 4/5 퀴즈
- 변수의 유형 중 \_\_\_\_\_는 메서드 내부에 선언된 변수이고, \_\_\_\_\_는 static 키워드로 정의된 클래스 변수이며, \_\_\_\_\_는 클래스 내부에 정의되지만 메서드 내부에는 선언되지 않는 변수이다.

4. 상수를 선언하기 위한 키워드를 쓰시오 4/5 퀴즈

5. 다음은 자바의 기본 데이터 타입을 나타내는 표이다. 빈칸에 적절한 타입은? (단, 적절한 타입이 없을 때는 n/a로 표시하라.)

종류	정수	실수	논리
1바이트			
2바이트			
4바이트			
8바이트			

6. 다음 데이터의 자료형을 쓰시오

- a. true
- b. 50
- c. 'm'
- d. "m"
- e. 50.0

7. 다음 문장에 대해 true/false를 쓰시오

- a. &&는 AND 연산자로 논리 연산을 하는데 사용한다
- b. byte 타입의 연산 결과는 항상 byte 타입이다
- c.  $x > y$ 가 true이거나  $a < b$ 가 true라면,  $x > y \ \&\& \ a < b$ 도 true이다
- d. 모듈로 연산자(%)는 double 타입도 사용할 수 있다

8. 다음 실행문 중 잘못된 것으로 모두 고르고, 잘못된 이유를 쓰세요

- a. long x = 100;
- b. char c = "a";
- c. float f = 100.0;
- d. double d = (double)100;
- e. byte b = 300;

9. float 자료형 변수에 값을 할당하는 올바른 방법을 모두 고르세요.

- a. float a = 100.0
- b. float b = 129

c. float c = 134.78.1

d. float d = 105.3f

10. 다음 코드의 실행 결과를 쓰세요

```
int x = 15;
System.out.println(x < 10 && x < 20);
System.out.println(x < 10 || x < 20);
System.out.println(!(x < 10 && x < 20));
```

11. 다음 코드를 실행한 후의 변수 x와 y의 값을 쓰세요

```
int x = 10;
int y = x++;
y += ++x;
```

12. 다음 연산에서 첫번째로 실행되는 연산을 쓰세요.

```
int a, b, c, d, e, f, g, h;

h = (a + b) * c - d % e + f / g;
```

13. 다음 코드의 실행 결과를 쓰세요

```
int x = 1, y = 2;
System.out.println(x++);
System.out.println(++x + y--);
System.out.println(++x / 3 + x * ++y)
```

14. 다음 코드를 보고 출력 결과를 쓰세요

```
int a = 7, b = 15;
System.out.println(a + " + " + b + " = " + a + b);
System.out.println(a + " + " + b + " = " + (a + b));
```

15. 다음의 코드의 형변환이 가능하면 변환 결과를 쓰고, 가능하지 않다면 false를 쓰세요.

```
float f = 3.14f;
int i = (int)f
```

```
// 문자 a의 정수값을 97이다.
int i = 100;
char c = (char)i;
```

```
int i = 1;
boolean b = (boolean)i;
```

16. 다음 프로그램의 어디에서 오류가 발생하는지 표시하고, 이유를 설명하세요. 4/5 퀴즈

즈

```
public static void main(String[] args) {
    int a;
    System.out.println("변수의 값" + a);
}
```

17. 다음 프로그램의 실행 결과는 무엇인가?

```
public static void main(String[] args) {
    float name = "JavaBook";
    System.out.println(name);
}
```

18. 다음 프로그램의 오류를 설명하세요. 4/5 퀴즈

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

    int a = 4;
    float b = 3;
    int c = ++a + -b;
    System.out.println(c);

}
```

19. 다음 프로그램의 실행 결과를 쓰세요

```
public static void main(String[] args) {  
    int a = 5, b = 4;  
    float c;  
    System.out.println(c = a % b);  
}
```

### ▼ (3) 제어문 (조건문, 반복문, 분기문)

1. 다음 빈칸을 채우세요

- \_\_\_\_\_은 프로그램을 구성하는 코드가 실행될 때 코드의 처리 순서를 변경할 수 있는 문장이다.
- \_\_\_\_\_은 주어진 조건이 참인지 거짓인지 판별하고 그에 따라 특정 명령을 수행하는 문장이다
- \_\_\_\_\_은 일련의 명령문을 반복적으로 실행하는 데 사용된다.
- \_\_\_\_\_은 원할 때마다 반복문에서 벗어나거나 반복문의 조건식으로 이동하도록 실행 흐름을 직접 제어할 수 있는 문장이다.
- \_\_\_\_\_문은 원할 때마다 반복문의 실행을 중지하거나 종료하는데 사용된다.
- \_\_\_\_\_문은 특정 실행문을 건너뛰고 나머지를 계속 실행하려는 경우에 사용된다.

2. 반복문 중 반복 실행문을 우선 실행하고 조건식을 확인하는 것은?

- while문
- for문
- do ~ while문
- switch문

3. 다음 중 switch-case문의 입력이 될 수 있는 것은 무엇인가?

- 정수형
- 문자형
- 불리언형

d. 앞의 보기 모두

4. switch문에서 조건을 만족하는 case 레이블이 없을 경우에 실행될 코드 블록을 지정하는 레이블의 키워드를 쓰세요
5. 두 실행문의 결과가 같은지, 다른지 판단하세요.

```
if (a < 1) b = 1;
else if (a < 3) b = 2;
```

```
if (a < 1) b = 1;
else if (a >= 1 && a < 3) b = 2;
```

6. 다음의 switch문을 수행한 후 b의 값을 쓰세요.

```
int a = 1, b = 2;
switch (a++) {
    case 1 : b += a;
    case 2 : b += 5;
}
```

7. 다음은 변수 num의 숫자가 짝수인지 홀수인지 출력하는 조건문이다. ( )에 들어갈 조건식을 쓰세요

```
if (
    )
    System.out.println("짝수");
else
    System.out.println("홀수");
```

8. 다음 프로그램의 실행 결과를 쓰세요

```
public static void main(String[] args) {
    for (int i = 0; i < 5; i++)
        System.out.print(i + " ");
    System.out.println();
}
```

9. 다음 프로그램의 실행 결과를 쓰세요

```
for (int i = 1; i < 3; i++)
    for (int j = 1; j < 3; j++)
        System.out.println(i + " " + j);
```

10. 다음 반복문을 실행했을 때 sum 값은?

```
int sum = 0;
for (int i = 0; i < 5; i++) {
    if (i % 3 == 0) continue;
    sum += i;
}
```

11. 다음 프로그램의 실행 결과를 쓰세요. 4/5 퀴즈

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

    int a = 5;
    switch(a) {
        case 5:
            System.out.println("숫자 = 5");
        case 6:
            System.out.println("숫자 != 5");
        default:
            System.out.println("숫자가 아님");
    }

}
```

13. 다음 코드의 실행 결과를 쓰세요

```
int total = 0, x = 0, y;

while (++x < 5) {
    y = x * x;
    System.out.println(y);
    total += y;
}
```



```
System.out.println("총합은 " + total);
```

14. 다음 코드에서 오류를 찾아서 수정하시오. 4/5 퀴즈

```
for (i = 0, i > 10, i++)  
    System.out.println(i);
```

15. 다음 코드에서 오류를 찾아서 수정하시오. 4/5 퀴즈

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

#### ▼ (4) 메서드, 배열, 문자열

- 메서드를 사용하는 이유로 옳지 않은 것은?
  - 메서드는 비슷한 성격의 작업을 묶을 수 있다.
  - 메서드는 반복 코드를 줄이기 때문에 읽기가 어렵다
  - 코드를 중복해서 작성하지 않아도 되므로 재사용성이 향상된다.
  - 코드를 더 쉽게 유지할 수 있고, 변경이 필요할 때 쉽게 리팩토링 할 수 있다
- 사용자 정의 메서드에 대한 설명으로 옳은 것을 모두 고르시오
  - 반환 유형이 있는 메서드는 반드시 return이 있어야 한다
  - 반환 유형이 있는 메서드는 메서드 앞에 자료형을 붙여야 한다
  - 반환 유형이 있는 메서드는 return이 없다
  - 반환 유형이 있는 메서드는 메서드명 앞에 반드시 void를 붙여야 한다
- 메서드 오버로딩에 대한 설명으로 옳은 것으로 모두 고르시오
  - 메서드명이 같지만 매개변수가 다른 메서드
  - 메서드명이 같지만 매개변수의 자료형이 다른 메서드

- c. 메서드명이 같지만 매개변수의 개수가 다른 메서드
  - d. 메서드명이 같지만 매개변수의 자료형과 개수가 같은 메서드
4. 메서드명이 같지만 매개변수의 개수나 자료형이 다른 메서드를 하나의 메서드명으로 정의하는 것을 무엇이라고 하는가?
- a. 메서드 오버로딩
  - b. 메서드 오버라이딩
  - c. static 메서드
  - d. 인스턴스 메서드
5. **return 값이 없는 메서드의 메서드명 앞에 추가하는 키워드를 쓰시오.** 4/5 퀴즈
6. 다음 프로그램의 실행 결과를 쓰시오

```
public static void text(int a, String b) {
    System.out.println(a + b);
}

public static void main(String[] args) {
    text(1, " JavaBook");
}
```

7. 다음 코드의 실행 결과를 쓰시오 4/5 퀴즈

```
System.out.println(new String("자바 프로그래밍 기초").substr
```

8. 다음 프로그램의 실행 결과를 쓰시오

```
public static void main(String[] args) {
    System.out.println(addition(32.8f, 7.2f));
}

static int addition(float x, float y) {
    return (int)(x + y);
}
```

9. 다음 실행 결과를 참고하여 프로그램을 완성하시오 4/5 퀴즈

[실행 결과]

숫자 입력 : 5

입력된 숫자 : 5

```
static _____ getNumber() {  
  
    Scanner sc = new Scanner(System.in);  
  
    _____ ("숫자 : ");  
  
    int num = sc.nextInt();  
  
    return num;  
  
}  
  
static _____ displayNumber(int a) {  
  
    System.out.println("입력된 숫자 : " + a);  
  
}  
  
public static void main(String[] args) {  
  
    int a = _____;  
  
    _____;  
  
}
```

10. 자바에서 배열의 요소를 표시하는데 사용되는 루프문은 무엇인가?

- a. switch문
- b. for문
- c. 중첩된 if문
- d. if-else문

11. 다음 중 1차원 배열의 구문이 아닌 것은?

- a. int arr[];
- b. int []arr;
- c. int[] arr;
- d. int[] arr[];

12. 배열에 대한 코드이다. 문법적으로 틀린 것은?

- a. int[] arr = {1, 2, 3};
- b. int arr[3] = {1, 2, 3};
- c. int[] arr; arr = new int[]{1, 2, 3}
- d. int arr[];

13. 다음 프로그램의 실행 결과는 무엇인가? 4/5 퀴즈

```
public static void main(String[] args) {  
  
    int marks[] = new int[4];  
    marks[0] = 65;  
    marks[1] = 70;  
    marks[2] = 74;  
    marks[3] = 90;  
    marks[4] = 87;  
  
    for(int i = 0; i < marks.length; i++) {  
        System.out.println(marks[i]);  
    }  
  
}
```

14. 다음 프로그램의 실행 결과를 쓰세요

```
public static void main(String[] args) {  
  
    int arr[][] = { {100,200}, {300,400} };  
  
    for (int i = 0; i < arr.length; i++) {
```

```

        for (int j = arr.length - 1; j >= 0; j--) {
            System.out.print(arr[i][j] + " ");
        }
    }
}

```

15. 다음 중 문법적으로 틀린 것은?

- a. String s1 = new String("1");
- b. String s1 = "1";
- c. String s1 = "1", s2 = "2";
- d. String s1 = '1';

16. **String** 타입의 변수 **s**에 포함된 문자 개수를 알기 위한 코드를 작성하시오 4/5 퀴즈

17. 문자열을 연결하는 String 클래스의 메서드를 쓰시오

18. 다음 코드의 실행 결과를 쓰시오

```

String s1  = new String("안녕");
String s2 = new String("안녕");
String s3 = "안녕";
String s4 = "안녕";

System.out.println(s1 == s2);
System.out.println(s3 == s4);
System.out.println(s1.equals(s2));
System.out.println(s3.equals(s4));

```

19. 다음 코드의 실행 결과를 쓰시오

```

boolean b = 1 == 2;
System.out.println("java " + b + "1");

```

20. 다음 프로그램의 실행 결과를 쓰세요 4/5 퀴즈

```

public static void main(String[] args) {

```

```
String datas = "Java, Book";  
String data[] = datas.split(",");  
System.out.println(data[0]);  
  
}
```

## ▼ (5) 클래스, 객체지향 프로그래밍

1. 다음의 설명에 대해 true/false를 선택하시오
  - a. 필드는 초기화하지 않아도 된다
  - b. 생성자 없이 클래스를 정의해도 된다
  - c. 생성자는 일반적으로 필드를 초기화한다
  - d. 필드는 생성자보다 먼저 선언해야 한다
  - e. 하나의 클래스는 하나의 객체 인스턴스만 생성할 수 있다
2. 객체지향 프로그래밍에서 객체 간의 상호작용은 무엇을 통해서 하는가?
3. 다음은 객체지향 프로그래밍의 특징을 설명한 글이다. 설명에 해당하는 특징을 쓰세요
  - a. 데이터를 수정하지 못하도록 안전하게 하기 위해 정보의 세부사항을 숨기고, 변수와 메서드를 하나의 단위로 묶는 기법
  - b. 이미 존재하는 한 클래스의 멤버를 다른 클래스에 물려주는 기법
  - c. 변수, 메서드, 객체 등이 여러 형태를 취하는 기법
  - d. 어떤 영역에서 필요로 하는 속성이나 행동을 추출하는 작업, 구체적인 사물들의 공통적인 특징을 파악해서 하나의 개념으로 다루는 수단
4. 다음 설명에 맞는 클래스의 접근 제한자를 쓰세요
  - a. 패키지 등에 관계없이 누구나 사용할 수 있는 경우
  - b. 객체 내부에서만 사용하는 경우
5. 클래스의 구성요소 3가지를 쓰시오
6. 클래스가 다음과 같이 정의되어 있는 경우, 필드를 보호하기 위한 접근제한자를 쓰고, Getter와 Setter를 작성하시오

```

class Rectangle {

    _____ int width;

    _____ int height;

    // 모든 field에 대해 Getter와 Setter 작성

}

```

7. 다음 코드에서 오류가 발생하는 statement를 표시하고, 수정하세요

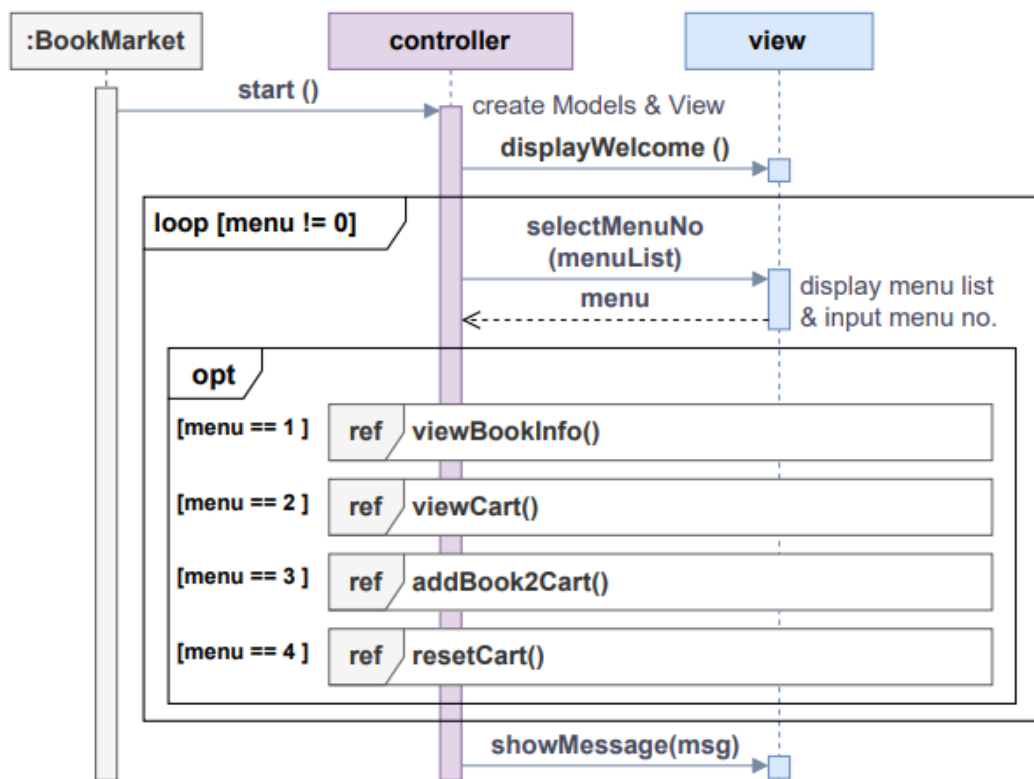
```

class Util {
    public double findPI() {
        return 3.14;
    }
}

public class MathTest {
    public Ststic void main (String args[]) {
        Util u;
        System.out.println(u.findPI());
    }
}

```

8. 다음은 contoller의 클래스 start() 메소드를 아래의 sequence diagram에 따라 작성한 코드이다. 비어있는 부분을 채우시오



```

_____ void start() {

    _____.displayWelcome();

    _____

do {

    menu = _____;

    switch (menu) {
    case 1 : _____

    case 2 : _____

    case 3 : _____

    case 4 : _____

    } while (_____);
  
```

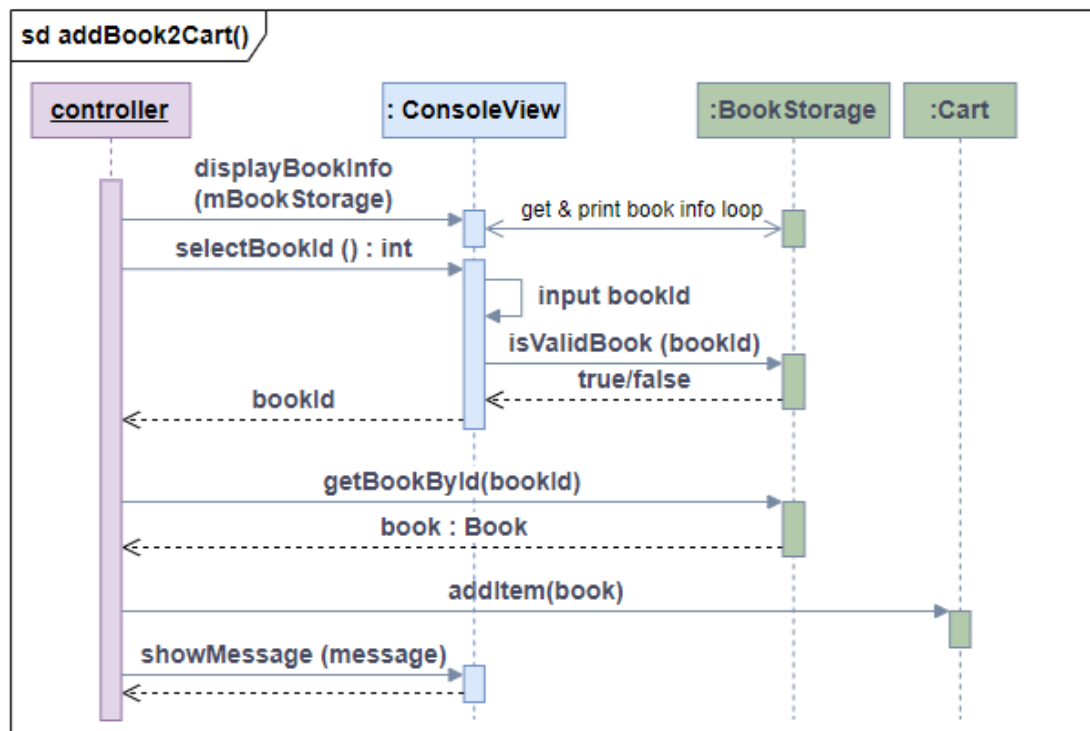


```

        _____("Online Book Market을 종료합니다.")
    }
}

```

9. 다음은 controller의 addBook2Cart() 메소드의 sequence diagram이다.  
 isValidBook(bookId) 메소드를 제공하는 클래스의 일부와 isValidBook() 메소드를 구현한 코드를 채우시오



```

public class _____ {

    private Book[] bookList;

    .... (생략)

    // 입력된 bookId에 해당하는 책이 있는지 여부를 return하는 메
    _____ isValidBook(int bookId) {

        for ( _____ : _____)

```

```
        if (book.id == bookId) return _____;  
  
        return _____;  
  
    }  
  
    ... (생략)  
}
```