**문제1. 다음 주어진 배열 데이터를 이용하여 3의 배수의 개수와 배수의 합을 구하여 출력형태와 같이 출력하세요.**

**<<실행결과>>**

**주어진 배열에서 3의 배수의 개수=> 6**

**주어진 배열에서 3의 배수의 합=> 108**

**주의 사항 :**

1. 주어진 main메소드를 이용하여 테스트 합니다.
2. 주어진 메소드의 signature는 변경하지 않습니다. 주어진 signature를 변경하면 해당 메소드는 오답으로 처리합니다.
3. 컴파일이 안되거나 Exception 이 발생하는 경우는 무조건 **오답으로 처리**됩니다.

**문제2. 키보드에서 정수로 된 돈의 액수를 입력 받아 오만 원권, 만원 권, 오천 운권, 천원 권,**

**500원짜리 동전, 100원짜리 동전, 50원짜리 동전, 10원짜리 동전, 1원짜리 동전 각 몇**

**개로 변환 되는지 작성하시오.**

Prob2.java

|  |
| --- |
| **package** prob2;  **public** **class** Prob2 {  **public** **static** **void** main(String[] args) {  }  } |

**<<실행결과>>**

|  |
| --- |
| **금액: 67879**  **오만 원권: 1매**  **만 원권: 1매**  **오천 원권: 1매**  **천 원권: 2매**  **500원 동전: 1개**  **100원 동전: 3개**  **50원 동전:1개**  **10원 동전:2개**  **1원 동전:9개** |

**문제 3. 문자열(String)을 입력 받아, 해당 문자열을 문자 순서를 뒤집어서 char[]로 반환하는 메소드를 만들고, char[]을 입력 받아 출력하는 메소드를 만드시오.**

**public static char[] reverse(String str)**

* **문자열의 순서를 뒤집어서 char 배열로 리턴하는 메소드를 구현한다**
* **예를 들어 “Hi!” 라는 문자열을 파라미터로 전달 하면 {‘!’, ‘i’, ‘H’ } 배열을 반환한다.**

**public static void printCharArray(char[] array)**

* **char[]을 받아들여 출력한다.**
* **{‘!’, ‘i’, ‘H’} 배열을 파라미터로 전달하면 다음과 같이 출력된다.**

**!iH**

**<<실행결과>>**

**dlroW olleH**

**!gnimmargorP avaJ**

**주의 사항 :**

1. 주어진 main메소드를 이용하여 테스트 합니다.
2. 주어진 메소드의 signature는 변경하지 않습니다. 주어진 signature를 변경하면 해당 메소드는 오답으로 처리합니다.
3. 컴파일이 안되거나 Exception 이 발생하는 경우는 무조건 **오답으로 처리**됩니다.

**문제4. 배열 크기만큼 키보드에서 정수를 입력받고 평균을 구하는 프로그램을 작성하시오**

키보드에서 문자열을 입력 받을 때에는 java.util.Scanner 클래스를 사용합니다.

주어진 소스를 수정하면 안됩니다.

Prob4.java

|  |
| --- |
| **package** prob4;  import java.util.Scanner;  **public** **class** Prob4 {  **public** **static** **void** main(String[] args) {  Scanner in = new Scanner( System.in );  int intArray[] = new int[5];  double sum = 0;  /\* 키보드에서 배열 크기만큼 입력 받아 배열에 정장하는 코드 \*/  /\* 배열에 저장된 정수 값 더하기 \*/  /\* 출력 \*/  }  } |

**<<실행결과>>**

|  |
| --- |
| **5개의 숫자를 입력하세요.**  10  20  30  40  50  **평균은 30.0 입니다.** |

**문제 5. 다음과 같은 2가지 클래스가 제공된다. Prob2 클래스를 실행했을 때 실행결과와 같이 나올 수 있도록 MyBase클래스를 작성하시오.**

|  |
| --- |
| **Prob5.java** |
| **public class Prob5 {**  **public static void main(String args[]){**  **Base base = new MyBase();**  **base.service("낮");**  **base.service("밤");**  **base.service("오후");**  **}**  **}** |

|  |
| --- |
| **Base.java** |
| **public class Base {**  **public void service(String state){**  **if(state.equals("낮")) {**  **day();**  **} else {**  **night();**  **}**  **}**    **public void day() {**  **System.*out*.println("낮");**  **}**    **public void night() {**  **System.*out*.println("night");**  **}**  **}** |

**<<실행결과>>**

낮에는 열심히 일하자!

night

오후도 낮과 마찬가지로 일해야 합니다.

**주의 사항 :**

1. 주어진 main메소드를 이용하여 테스트 합니다.
2. 주어진 Class Base는 절대로 수정하면 안됩니다.
3. 컴파일이 안되거나 Exception 이 발생하는 경우는 무조건 **오답으로 처리**됩니다.

**문제6. 다음은 Employee클래스를 상속받아 Depart클래스를 구현합니다. Depart클래스를 구현할 때 실행결과를 참조합니다.**

Employee.java

|  |
| --- |
| **package** prob5;  **public** **class** Employee {  **private** String name;  **private** **int** salary;  **public** Employee() {  }  Employee(String name, **int** salary) {  **this**.name = name;  **this**.salary = salary;  }    **public** String getName() {  **return** name;  }  **public** **void** setName(String name) {  **this**.name = name;  }  **public** **int** getSalary() {  **return** salary;  }  **public** **void** setSalary(**int** salary) {  **this**.salary = salary;  }  **public** **void** getInformation() {  System.*out*.printf("이름: %s 연봉: %d\n", name, salary);  }  } |

Prob6.java

|  |
| --- |
| **package** prob6;  **public** **class** Prob6 {  **public** **static** **void** main(String[] args) {  Employee pt = **new** Depart("홍길동", 5000, "개발부");  pt.getInformation();  }  } |

**<<실행결과>>**

|  |
| --- |
| 이름: 홍길동 연봉: 3000 부서:기획부 |

**<<주의사항>>**

1. Employee.java 와 Prob6.java클래스는 수정하지 않습니다.
2. Depart클래스에 멤버변수를 String department변수외에 추가하지 않습니다.
3. Depart클래스에서 생성자를 구현하지 않으면 감점, getInformation()메서드를 구현하지 않으면 감점. 감점요소가 2개면 오답으로 처리됩니다.
4. 컴파일이 안되거나 Exception이 발생하는 경우도 무조건 오답으로 처리됩니다.

**문제7. PlayBird 클래스를 이용하여 두 클래스의 메소드를 호출시켰을 때 콘솔에 출력되는 결과가 아래와같도록 Duck 클래스와 Sparrow 클래스를 완성하세요.**

오리(Duck) 클래스와 참새(Sparrow) 클래스는

이름을 나타내는 name과

다리의 수를 나타내는 legs,

몸길이를 나타내는 length 데이터를 가지고 있습니다.

메소드로는,

날다 fly,

울다 sing,

이름기록하기 setName,

문자열출력하기 toString 를 가지고 있습니다.

**<<실행결과>>**

|  |
| --- |
| **오리(꽥꽥이)는 날지 않습니다.**  **오리(꽥꽥이)가 소리내어 웁니다.**  **오리의 이름은 꽥꽥이 입니다.**  **참새(짹짹)가 날아다닙니다.**  **참새(짹짹)가 소리내어 웁니다.**  **참새의 이름은 짹짹 입니다.** |