

# Examples

Week 9

# Circle.java

---

```
public class Circle {  
  
    int radius;  
    String name;  
  
    public Circle() { }  
  
    public double getArea() {  
        return 3.14*radius*radius;  
    }  
  
    public static void main(String[] args) {  
        Circle pizza;  
        pizza = new Circle();  
        pizza.radius = 10;  
        pizza.name = "Java-Pizza";  
        double area = pizza.getArea();  
        System.out.println("Area of " + pizza.name + ": " + area);  
    }  
}
```

Area of Java-Pizza: 314.0  
Area of Java-Donut: 12.56

# Circle.java

---

```
    circle donut = new Circle();  
    donut.radius = 2;  
    donut.name = "Java-Donut";  
    area = donut.getArea();  
    System.out.println("Area of " + donut.name + ": " + area);  
}  
}
```

# Rectangle.java

---

```
import java.util.Scanner;

public class Rectangle {
    int width;
    int height;

    public int getArea() {
        return width*height;
    }

    public static void main(String[] args) {
        Rectangle rect = new Rectangle();
        Scanner scanner = new Scanner(System.in);
        System.out.print(">> ");
        rect.width = scanner.nextInt();
        rect.height = scanner.nextInt();
        System.out.println("Area: " + rect.getArea());
        scanner.close();
    }
}
```

```
>> 4 5
Area: 20
```

# BankAccount.java

---

```
import java.util.Scanner;

public class BankAccount
{
    private double balance;

    public BankAccount( double initialBalance )
    {
        if ( initialBalance > 0.0 )
            balance = initialBalance;
    }

    public void credit( double amount )
    {
        balance = balance + amount;
    }
}
```

# BankAccount.java

---

```
public double getBalance()
{
    return balance;
}
```

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

```
    BankAccount account1 = new BankAccount( 50.00 );
    BankAccount account2 = new BankAccount( -7.53 );
```

```
    System.out.printf( "account1 balance: $%.2f\n",
                        account1.getBalance() );
    System.out.printf( "account2 balance: $%.2f\n\n",
                        account2.getBalance() );
```

account1 balance: \$50.00  
account2 balance: \$0.00

Enter deposit amount: 35.7  
account1 balance: \$85.70

Enter deposit amount: 154.8  
account2 balance: \$154.80

# BankAccount.java

---

```
Scanner input = new Scanner( System.in );
double depositAmount;

System.out.print( "Enter deposit amount: ");
depositAmount = input.nextDouble();
account1.credit( depositAmount );
System.out.printf( "account1 balance: $%.2f\n\n",
                   account1.getBalance() );

System.out.print( "Enter deposit amount: ");
depositAmount = input.nextDouble();
account2.credit( depositAmount );
System.out.printf( "account2 balance: $%.2f\n\n",
                   account2.getBalance() );
    }
}
```

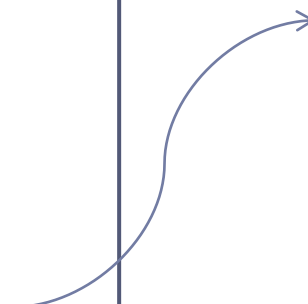
# Random 클래스

---

- ▶ import java.util.Random 필요
- ▶ 무작위 수(난수, random number)를 만들어 줌

```
public class Random
{
    public Random( ) { ... }
    public boolean nextBoolean( ) { ... }
    public double nextDouble( ) { ... }
    public float nextFloat( ) { ... }
    public long nextLong( ) { ... }
    public int nextInt( ) { ... }
    public int nextInt( int n ) { ... }
    ...
}
```

0 ~ (n-1)의  
무작위 수 반환





# RandomIntegers.java

---

```
import java.util.Random;

public class RandomIntegers
{
    public static void main( String[] args )
    {
        Random randomNumbers = new Random();
        int face;

        for( int count=1; count <= 20; count++)
        {
            face = 1 + randomNumbers.nextInt(6);
            System.out.printf( "%d ", face );
        }
    }
}
```

4	3	1	4	2	2	5	4	1
5	4	2	2	1	6	1	1	3
4	6							

# Ex09\_1.java

---

## ▶ 주사위 굴리기 모의실험

- ▶ 주사위를 6000번 굴리는 모의실험을 하고, 각 숫자가 몇 번 발생했는지 출력 하시오.
- ▶ Random 클래스를 사용하여 무작위 수를 발생시키시오.

```
1: 978  
2: 997  
3: 1021  
4: 991  
5: 990  
6: 1023
```



# Ex09\_2.java

## ▶ 성적분석 모의실험

- ▶ 0~100의 점수 50개를 무작위로 만들어 배열에 저장하고 출력한다.
- ▶ 50개의 평균을 계산하여 출력한다.
- ▶ 점수대 별 인원 수를 세어, 가로막대그래프 형태로 출력한다.
- ▶ 개수를 300개로 늘려서 다시 실험한다.

```
score[]: 38 85 31 8 18 13 53 33
25 2 73 32 53 14 77 74 69 34 79
27 31 67 81 62 81 29 92 22 65 20
50 36 70 50 79 23 45 17 84 26 68
54 2 78 64 5 83 41 75 28
```

```
Average: 47.32
```

```
100: 0
99-90: 1 *
89-80: 5 *****
79-70: 8 *****
69-60: 6 *****
59-50: 5 *****
49-40: 2 **
39-30: 7 *****
29-20: 8 *****
19-10: 4 *****
09-00: 4 *****
```