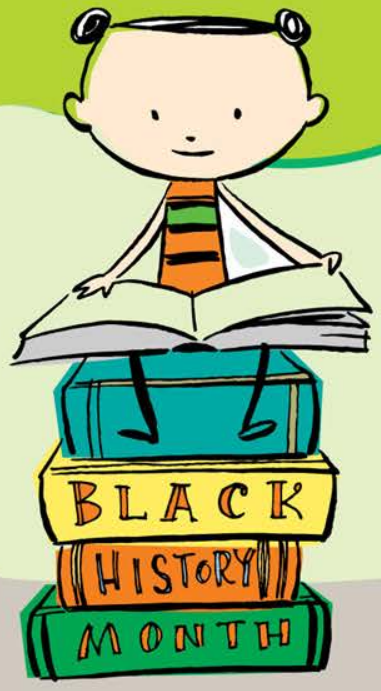
คณิตศาสตร์ ตัวห่อหุ้ม และการจัดรูปแบบ

หัวข้อ

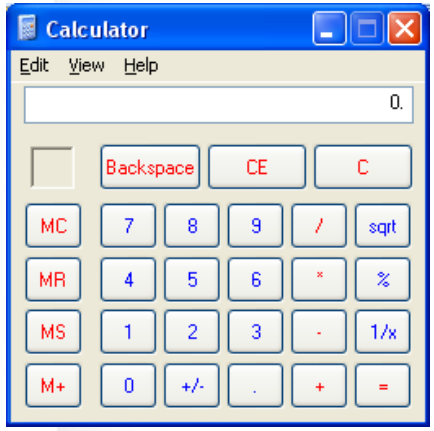
- คลาส Math
- การรับข้อมูลจากผู้ใช้
- ตัวห่อหุ้ม (Wrappers)
- การจัดรูปแบบ

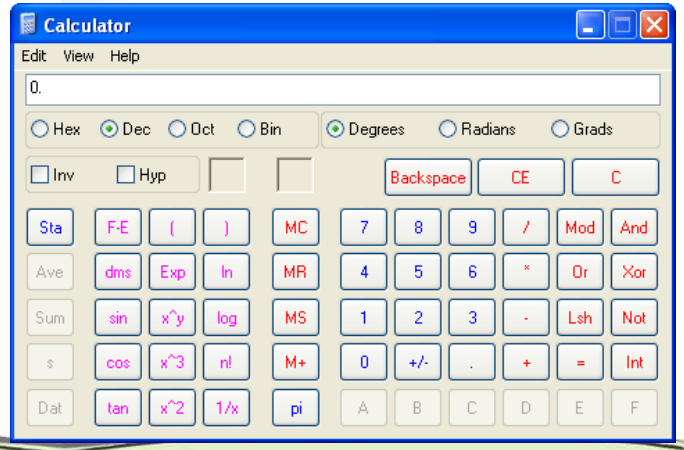
คลาส Math



การคำนวณ

- ง่าย
  - + - * / %
- ซับซ้อน
  - คลาส Math





เมธอดในคลาส Math

- sqrt() ใช้ในการหาค่ารากที่สอง
  - Math.sqrt(9) จะมีค่าเท่ากับ 3
- pow() ใช้ในการหาค่ายกกำลัง
  - Math.pow(2, 4) จะเท่ากับ 2^4 หรือ 16 นั่นเอง
- abs() ใช้ในการหาค่าสัมบูรณ์
  - Math.abs(-4) จะเท่ากับ 4

เมธอดในคลาส Math

- ceil() ใช้ในการหาค่าเลขจำนวนเต็มน้อยที่สุดที่มากกว่าตัวเลขที่ระบุไว้
  - Math.ceil(3.27) จะมีค่าเท่ากับ 4
  - Math.ceil(-3.27) จะมีค่าเท่ากับ -3
- floor() ใช้ในการหาค่าเลขจำนวนเต็มที่มากที่สุดที่น้อยกว่าตัวเลขที่ระบุไว้
  - Math.floor(3.27) จะมีค่าเท่ากับ 3
  - Math.floor(-3.27) จะมีค่าเท่ากับ -4

เมธอดในคลาส Math

- round() ใช้ในการปัดเศษทศนิยมโดยจะหาเลขจำนวนเต็มที่ใกล้กับตัวเลขที่ระบุมากที่สุด
  - Math.round(3.27) จะมีค่าเท่ากับ 3
  - Math.round(-3.27) จะมีค่าเท่ากับ -3
- min() ใช้หาค่าตัวเลขที่น้อยที่สุดระหว่างเลขสองตัวตามที่ระบุไว้
  - Math.min(3, 5) จะได้ค่าเท่ากับ 3
- max() ใช้หาค่าตัวเลขที่มากที่สุดระหว่างเลขสองตัวตามที่ระบุไว้
  - Math.max(3, 5) จะได้ค่าเท่ากับ 5

เมธอดในคลาส Math

- toDegrees() แปลงมุม radian เป็นมุม degree
  - Math.toDegrees(Math.PI) มีค่าเท่ากับ 180.0
- toRadians() แปลงมุม degree เป็นมุม radian
  - Math.toRadians(180) มีค่าเท่ากับ 3.141592653589793
- sin()
  - Math.sin(Math.toRadians(90)) มีค่าเท่ากับ 1

ปริมาตรวัตถุทรงกลม

- ปริมาตร = $(4 \div 3) \times \prod \times$ รัศมี^{3}

```
public class BallVolume {
    public static void main(String[] args) {
        double radius = 10;
        double volume = 4.0 / 3.0 *
                        Math.PI *
                        Math.pow(radius, 3);

        System.out.println(volume);
```

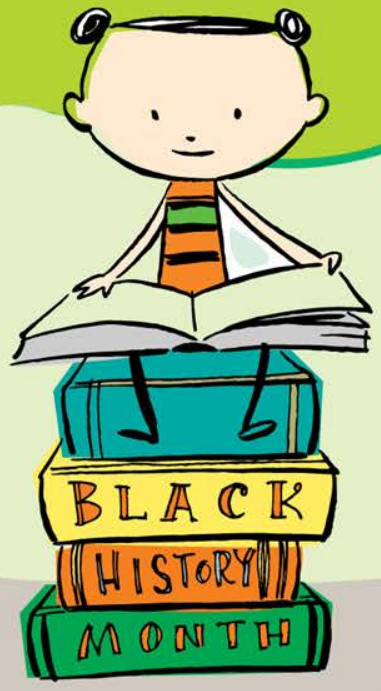
เลขสุ่ม

- เมธอด random()
  - return ค่าสุ่มในช่วง 0 เกือบถึง 1

```
for (int i = 0; i < 10; i++){
    double r = Math.random() * 100;
    int rand = (int) r;

    System.out.println(rand);
```

การรับข้อมูลจากผู้ใช้



คลาส java.util.Scanner

- คอนสตรักเตอร์
  - java.util.Scanner(InputStream src);
- เมธอด
  - nextInt();
  - nextDouble();
  - next();

ตัวอย่าง

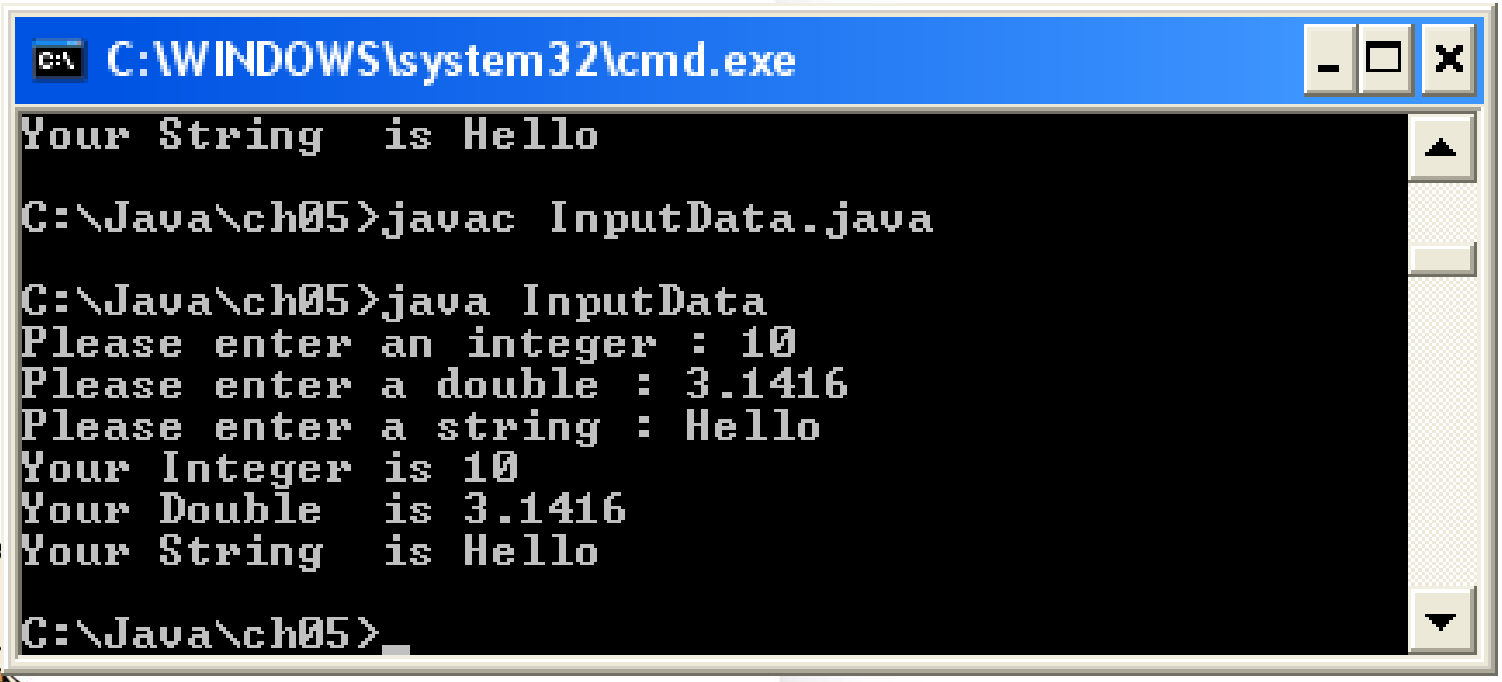
```
java.util.Scanner sc;
sc = new java.util.Scanner(System.in);

System.out.print("Please enter an integer :");
int i = sc.nextInt();

System.out.print("Please enter a double :");
double d = sc.nextDouble();

System.out.print("Please enter a string :");
String s = sc.next();
```

ผลการทำงาน

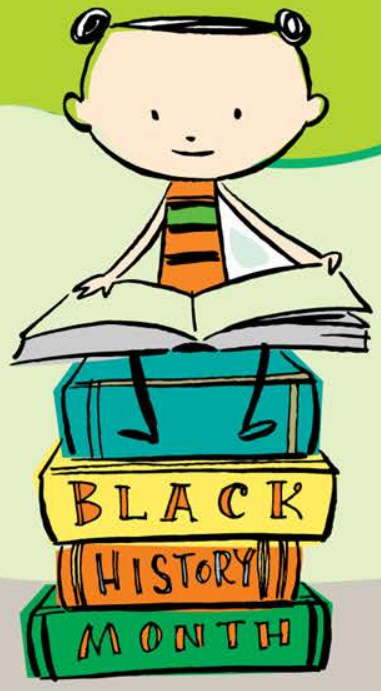

C:\WINDOWS\system32\cmd.exe
Your String is Hello
C:\Java\ch05>javac InputData.java
C:\Java\ch05>java InputData
Please enter an integer : 10
Please enter a double : 3.1416
Please enter a string : Hello
Your Integer is 10
Your Double is 3.1416
Your String is Hello
C:\Java\ch05>

ถ้าไม่อยากใช้ชื่อเต็ม

```
import java.util.Scanner;

class MyClass {
    public static void main(String[] a) {
        Scanner sc = new Scanner(System.in);
        ...
    }
}
```

ตัวห่อหุ้ม (Wrappers)



ตัวห่อหุ้ม

- ห่อชนิดข้อมูลพื้นฐาน
  - เพื่อใส่ใน Collection
- แปลง String เป็นชนิดข้อมูลพื้นฐาน
- แปลงจากชนิดข้อมูลพื้นฐานเป็น String
- มีค่าคงที่ของค่าที่มากสุดและน้อยของชนิดข้อมูลพื้นฐาน

คลาสที่เป็นตัวห่อหุ้ม

ชนิดข้อมูลพื้นฐาน	คลาสในกลุ่ม Wrapper
boolean	Boolean
char	Character
byte	Byte
short	Short
int	Integer
long	Long
float	Float
double	Double

การสร้างวัตถุ Wrapper

```
Boolean     bo = new Boolean(true);
Boolean     bo = new Boolean("true");

Character   c  = new Character('c');

Byte        by = new Byte((byte)20);
Byte        by = new Byte("20");

Short       s  = new Short((short)20);
Short       s  = new Short("20");
```

การแปลง Wrapper กลับเป็นชนิดข้อมูลพื้นฐาน

- รูปแบบ
  - *ตัวแปรชนิดข้อมูลพื้นฐาน* = *ตัวห่อหุ้ม.ชนิดข้อมูลพื้นฐาน*Value();
- ตัวอย่าง

```
String s = "20";
Integer i = new Integer(s);
int number = i.intValue();
```

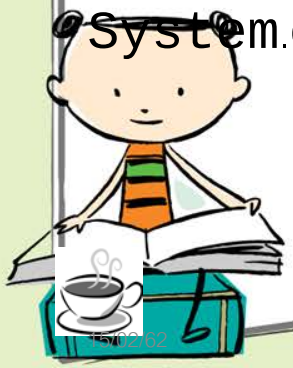
ตัวอย่าง

```
Scanner sc = new Scanner(System.in);

System.out.print("Please enter a number : ");
String s = sc.next();

Integer i = new Integer(s);
int number = i.intValue();

System.out.print("Your number plus 10 equals ");
System.out.println(number + 10);
```



การแปลง Wrapper เป็นสตริง

- รูปแบบ
  - *เรฟเฟอร์เรนซ์สตริง* = *ตัวห่อหุ้ม*.toString();
- ตัวอย่าง

```
int number = 20;
Integer i = new Integer(number);
String s = i.toString();
```

การแปลงสตริงเป็นชนิดข้อมูลพื้นฐาน

- รูปแบบ
  - *ตัวแปรชนิดข้อมูลพื้นฐาน* = *คลาสห่อหุ้ม*.parse*ชนิดข้อมูลพื้นฐาน*();
- ตัวอย่าง

```
Scanner sc = new Scanner(System.in);
System.out.print("Please enter a number : ");
int number = Integer.parseInt(sc.next());
```

การแปลงชนิดข้อมูลพื้นฐาน เป็นสตริง

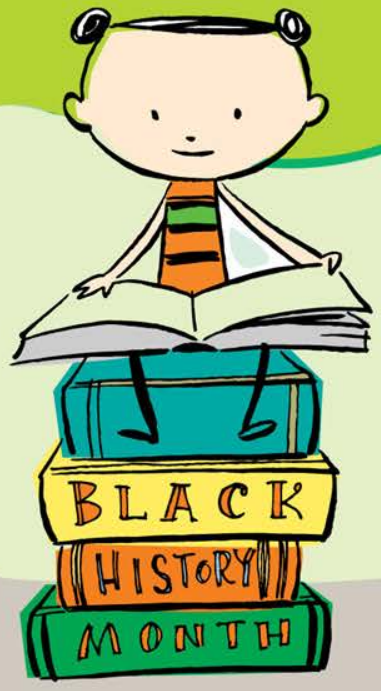
- รูปแบบ
  - *เรฟเฟอร์เรนซ์สตริง* = *คลาสห่อหุ้ม*.toString(*ข้อมูลพื้นฐาน*);
- ตัวอย่าง

```
Scanner sc = new Scanner(System.in);
System.out.print("Please enter a number : ");
String s = Integer.toString(sc.nextInt());
```

ค่าคงที่ในคลาส Wrapper

- Integer
  - Integer.MIN_VALUE = -2147483648
  - Integer.MAX_VALUE = 2147483647
- Double
  - Double.MIN_VALUE = 4.9E-324
  - Double.MAX_VALUE = 1.79769313486231570E308

การจัดรูปแบบ



การจัดรูปแบบด้วยเมธอด printf()

- รูปแบบ
  - *ตัวแสดงผล*.printf("*รูปแบบ*", *ตัวแปร1*, *ตัวแปร2*, ...);
- ตัวอย่าง

```
int     i = 3277;
double  d = 132324.25;
System.out.printf("%d\n", i);
System.out.printf("%x\n", i);
System.out.printf("%f\n", d);
System.out.printf("%16f\n", d);
System.out.printf("%16.2f\n", d);
System.out.printf("%,16.2f\n", d);
```

การจัดรูปแบบด้วยคลาสในกลุ่ม Format

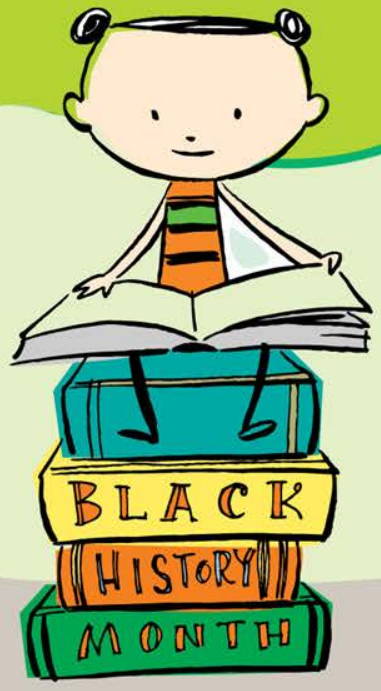
```
double d = 37625.72558;

NumberFormat nf;

nf = NumberFormat.getCurrencyInstance();
System.out.println(nf.format(d)); // ฿37,625.73

nf = NumberFormat.getIntegerInstance();
System.out.println(nf.format(d)); // 37,625
```

สรุป



สรุป

- คลาส Math มีเมธอดสำหรับการคำนวณทางด้านคณิตศาสตร์ชั้นสูง
- คลาสประเภท Wrapper ใช้แปลงตัวแปรชนิดข้อมูลพื้นฐานไปเป็นข้อความ และกลับกัน
- การแปลงวัตถุ String ไปเป็นจำนวนเต็มใช้เมธอด Integer.parseInt()
- การแปลงจำนวนเต็มไปเป็นวัตถุ String ใช้เมธอด Integer.toString()

สรุป

- เมธอด println() ใช้สำหรับแสดงผลลัพธ์ออกทางหน้าจอ
- เมธอด printf() ใช้แสดงผลลัพธ์เช่นเดียวกับเมธอด println() แต่สามารถจัดรูปแบบการแสดงผลได้หลากหลาย
- คลาส NumberFormat ช่วยจัดรูปแบบการแสดงผลได้