

RECITATION 3

Brent McCorvey
Son Tran

PRINTF

- Conversion characters:
 - d: decimal integer (int)
 - f: floating-point number (float, double)
 - s: string
 - S: uppercase all the letters in the string
 - n: newline

PRINTF

- Examples: salary = 30.0, name = "Kevin"
 - `System.out.printf("Total is: %d", salary);`
 - Total is: 30.0
 - `System.out.printf("%s makes %d per hour", name, salary);`
 - Kevin makes 30.0 per hour

NUMBER FORMATTING

- `NumberFormat us = NumberFormat.getCurrencyInstance()`
- `us.format(aDouble)` returns a **String**
- Ex: `us.format(200)` returns \$200

INPUT (STRING)

- `Scanner keyboard = new Scanner(System.in);`
- `keyboard.next()`: can't read two words separated by space
- ***It places the cursor in the same line after reading the input***
- However, to parse more than one string, use `keyboard.nextLine()`
- Sample code I

INPUT (INTEGER)

- You can use: `keyboard.nextInt();`
- *It also places the cursor in the same line after reading the input*
- If you only have one input:
 - `Integer.parseInt(keyboard.nextLine())` : returns int
 - What's the difference? (Next slide)

INPUT (INTEGER)

- Sample code 2

INPUT (DOUBLE)

- `Double.parseDouble(keyboard.nextLine())`

BOOLEAN EXPRESSION

- `isFull()` is a method that returns a boolean
- **NEVER EVER** write `if (isFull() == true)`
- *if (isFull())* is correct
- if you want to check for false, write: *if (! isFull())*

COMBINING BOOLEAN EXPRESSION

- `true && true` returns `true`
- `true && false` returns `false`
- `false && false` returns `false`

COMBINING BOOLEAN EXPRESSION

- `true || true` returns `true`
- `true || false` returns `true`
- `false || false` returns `false`

COMBINING BOOLEAN EXPRESSION

- Let $\text{total} = 10, \text{num} = 5$
- $\text{num} \geq 5 \ \&\& \ \text{total} == 20$ returns _____
- $(\text{num} == 0) \ || \ (\text{total} / \text{num}) > 2$ returns _____
- $\text{num} != \text{total} \ \&\& \ \text{total} > 1$ returns _____

IF-ELSE STATEMENT

- An if block can only have one if and one else, but multiple else-if are allowed.

- Illegal:

```
if (isFull()) {  
    // do something  
} else {  
    // do something else  
} else {  
    // what else can we do  
}
```

IF-ELSE STATEMENT

- Good practice:

```
if (isFull()) {  
    // do something  
} else if {  
    // do the first task  
} else if {  
    // do the second task  
} else {  
    // execute final resolution  
}
```


SWITCH

```
public void setCoinName(int coinValue) {  
    switch (coinValue) {  
        case 1:  
            nameCoin = "Penny";  
            break;  
        case 5:  
            nameCoin = "Nickel";  
            break;  
        case 10:  
            nameCoin = "Dime";  
        case 25:  
            nameCoin = "Quarter";  
            break;  
        default:  
            nameCoin = "Unidentified";  
            break;  
    }  
}
```

WHILE & DO-WHILE

- while loops are ***pre-text*** loops
 - they only execute after the condition is checked
- do-while loops are ***post-text*** loops
 - they execute before checking the condition