

Lab Week11. Data Types

Last Week

- Familiarisation with Eclipse
- Console applications
- Main
- Static methods (procedures and functions)
- Input and Output Streams

Learning Objectives

- Type conversion, casting
- Switch statements
- Enumerated types

Resources

- Lecture Notes - console applications, Data Types
- Java tutorials – see top of moodle unit
- Java enum tutorial :

<http://javarevisited.blogspot.co.uk/2011/08/enum-in-java-example-tutorial.html>

[Please open any portfolio exercises for marking later]

Ex1. Start a new eclipse project Week10 (look back at last week's screencast if required) create a **BankTeller** Class. Write a Bank Teller Console application (started below), which will calculate the optimum combination of notes to be handed out to customers.

A variable **Sterling** contains the number of pounds sterling to be handed out. The program then works out how many **fifties**, how many **twenties**, how many **tens**, **fives** and how many one pound coins should be issued. The program should output to the console these values.

```
public class BankTeller {  
    public static void main(String args[]) {  
        int sterling=189; //try different values to check your code  
        int fifties, twenties, tens, fives ,ones ;  
  
    }  
}
```

The modulus operator '**%**' gives the remainder after division, e.g.

$$5\%2 = 1$$

$$11\%3 = 2$$

Test your program with different values for the sterling variable.

Ex2. Amend your program to provide notes and coins for a floating point sterling value

e.g. float sterling = 26.11;

output should indicate :

1 twenty, 1 five, 1 pound, 0 fifty pence, 0 twenty pence, 1 ten pence, 0 five pence, 1 penniesD

Design ideas

- Split pounds from pence – cast to truncate or use modulus
- Multiply pence by 100
- Deal with pounds
- Deal with pence

Ex3. Enumerated sets. The weather can be in one of three states [*sunny, overcast, raining*], create an enumerated set (call the type *state*) and *state* variable to model the current weather. Extend this into an application that allows a user to indicate if the weather is improving or deteriorating, a suitable message should be output each time the state changes. The application should allow user inputs (a char will do) until an unrecognized char is entered. A partial solution is below

[Hint – see lecture notes, traffic lights example]

```
boolean finished=false;
char command;
while(!finished){
    System.out.println("Weather state :"+current);
    System.out.print("Enter i for improving, d for deteriorating:");

    command =
    switch(command){
        case 'i': current = improveState(current);
                  break;
        case 'd': current = worsenState(current);
                  break;
        default : finished=true; //any other char ends program
    } //case
} //while

public static state improveState(state present)
{
    //return an improved state
}

public static state worsenState(state present)
{
    //return a worsened state
}
```

Ex4 (portfolio) Calendar check. Create an application that allows the user to select a month and enter a day number, the application should output whether the day is a valid day of that month

e.g. [blue text has been entered from keyboard]

Enter a month [1..12] : 2

Enter a day [1..31] : 30

30 is not valid for February

Hints : You do not necessarily need an enumerated set for this particular problem but it may help. You will require a switch statement.

Google “days of the month”, leap years I’ll leave to your discretion.

Extension : Read and try out some of the examples provided in the tutorial

<http://javarevisited.blogspot.co.uk/2011/08/enum-in-java-example-tutorial.html>