EECS1022
MOBILE COMPUTING

JAVA BASICS

DELEGATION & APIs

(SLIDES ADAPTED FROM PROF.H. ROUMANI)

PROF. Y. LESPÉRANCE

Dept. of Electrical Engineering & Computer Science

1

APPROACH

- Determine the available services
  The API tells you what a service does, *not* how it does it.
- Compose your solution accordingly
  Orchestrate what does what you need done…
- Delegate to Method *m*
  - Static Methods: Invoke on the class
    C.m(…)
  - Non-Static Methods: Instantiate then invoke on the object.
    C ref = new C(…); ref.m(…);

2

EXAMPLE OF SERVICES

- Format in String (static)
  Turn something to a string and format it.
  Good for interacting with the UI.
- pow in Math (static)
  Raises a number to an exponent.
- getTime in Date (non-static)
  Number of milliseconds since 1970

3

String.Format API

`%[flags][width][.precision]conversion`

Conversion	
%d	Integer (base 10)
%x	Integer (base 16)
%f	Real, floating point
%e	Real, exponential
%c	Character
%b	Boolean
%tA	Day (a for short)
%tB	Month (b for short)
%tY	Year (y for 2-digit)
%tT	%tH %tM %tS
%%	Escape for %

String Escapes	
\n	New line
\t	tab
\\	Escape for \

width	
0	Zero fill
>0	Right-align
<0	Left-align

Flag	
,	Thousand separator
>0	Right-align
<0	Left-align
+	Include sign
sp	Leading space if >0
(	Enclosing () if <0

4

Math API

Method Summary	
`static double`	**abs**`(double a)` Returns the absolute value of a `double` value.
`static int`	**abs**`(int a)` Returns the absolute value of an `int` value.
`static double`	**pow**`(double a, double b)` Returns the value of the first argument raised to the power of the second argument.

5

Anatomy of an API

The API of a class has three sections:

- A Constructor Section
- A Field Section
- A Method Section

- Each section has a summary and a linked detailed part.
- The job of the constructor is to initialize the state.
- Fields (or public attributes) are used mainly for constants.
- Methods are the ones who "do the work".
- See Links in the course website to basic APIs.

6

The Constructor

- A constructor section *looks like* a method but it has no return and its name is the same as the class.
- There can be more than one (overloaded). The default one creates an object with a default state.
- A class with only a default constructor must have mutators if we are to be able to set a non-default state.
- The constructor is invoked with new in order to instantiate the class and create an object of its type.
- A class with no constructor (i.e. a private one) must have all its fields and methods static or else we cannot use it.

7

Methods

- Each method has a signature = name and parameter types.
- A class cannot have two methods with the same signature (even if the return is different). Hence, can have two methods with the same name (overloaded).
- To bind C.m(…) the compiler locates C (or else issues No Class Definition Found) and then locates m(…) in C (or else issues Cannot Resolve Symbol). If more than one such m is found, it binds with the most specific one.
- Parameters are passed by value. Hence values stored in your variables cannot be inadvertently changed by passing the variables to a method.

8

Behind the Scenes

- Invocation syntax is like two-level addressing: country code then number. The dot separates the two.
- For static methods, the class is the country. For others, a copy is involved. The object reference is the country here.
- Within the object, the address of the object is this.
- Instantiation is slow. So if many state changes are needed and when don't need to retain each, a default constructor and a mutator is better than multiple objects.
- Controller can thus instantiate in *onCreate* and mutate when a button is clicked.

9