Topic 5: Developing Code

Using Java `Math` Class

Lecture Contents

- Java Class Library
- The `Math` Class
  - `Math` Constants
  - `Math` Methods

Java Class Library

- Java Class Library (JCL) is part of the Java Development Kit (JDK)
  - A comprehensive collection of pre-built classes and methods
  - Provides essential functionality for Java applications
  - Includes functionality in classes, for example:
    - `Math` (which we'll discuss in this slide show)
    - `Array` (sorting, etc.)
    - `File` (for file I/O)
    - `ArrayList`
    - `HashMap`

`Math` Constants

- The `Math` class contains constants for π and *e*.
  - Constants in Java are, by convention, in UPPER_CASE

```
public class PlayWithMath {

   public static void main(String args[]) {
      System.out.println("The value of pi is: " + Math.PI);
      System.out.println("The value of e is: " + Math.E);
   }
}
```

```
The value of pi is 3.141592653589793.
The value of e is 2.718281828459045.
```

`Math` Constants

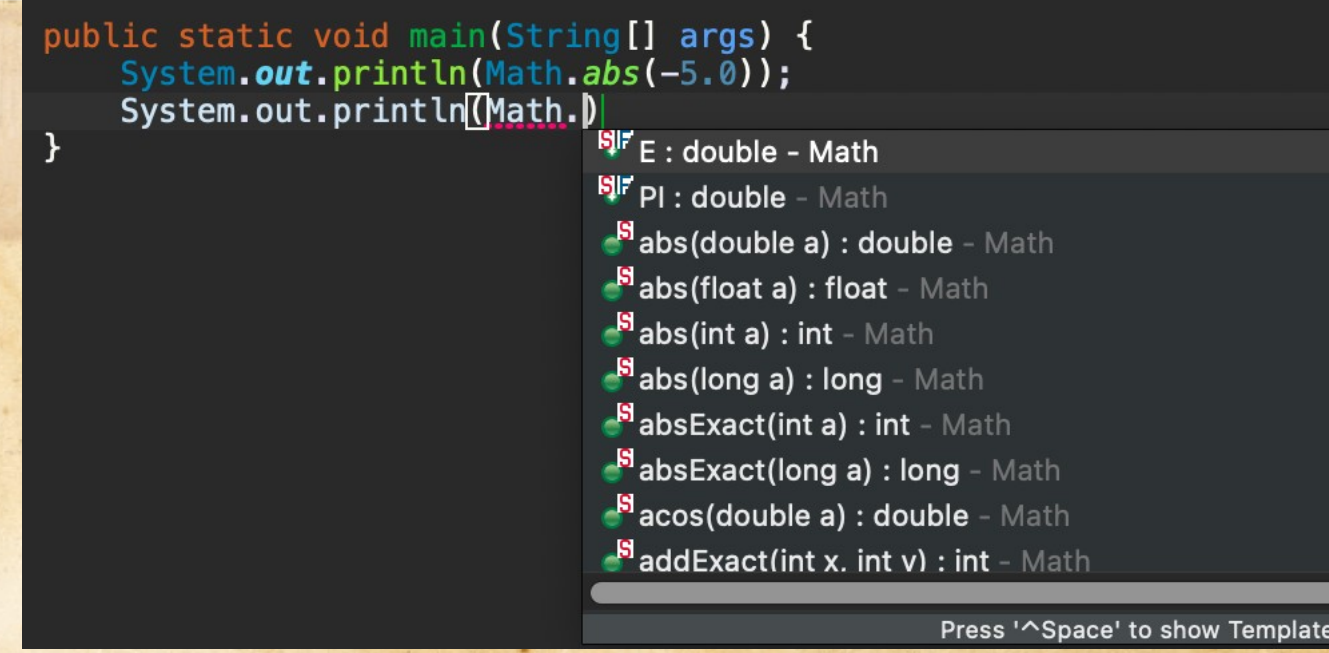
- Use the constant `Math.PI` to write method `calculateCircumference`

```
public static void main(String args[]) {
   double c = calculateCircumference(3.0);
   System.out.println("Circumference: " + c);
}
```

```
Circumference: 18.84955592153876
```

Math Methods

- **Math.abs(-5);**
- **Math.sqrt(2.0);**
- Math.min(3, 5);
- Math.max(3, 5);
- Math.sin(3.14);
- Math.asin(0.5);
- **Math.pow(2, 5);**
- **Math.random();**



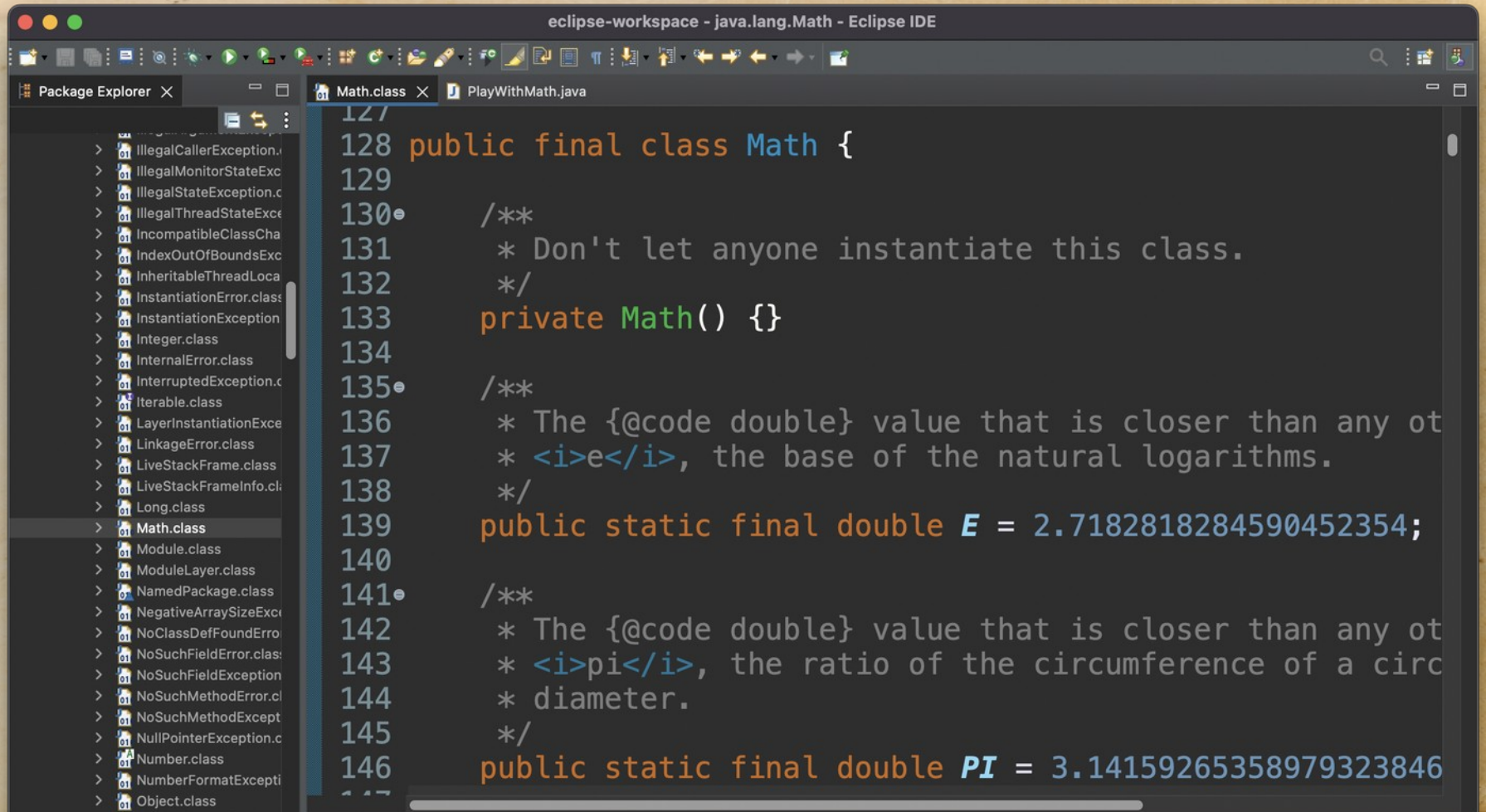
`Math` Methods

- Use the constant `Math.sqrt` (and `Math.pow`, if you wish) to write method `hypotenuseLength`

```
public static void main(String args[]) {
   double len = hypotenuseLength(3.0, 4.0);
   System.out.println("Hypotenuse Length: " + len);
}
```

```
Hypotenuse Length: 5.0
```

- You can view the code of the `Math` class and compare it to the classes we write.



If you take AP CSc A

- AP Java Subset

Math Class	
`static int abs(int x)`	Returns the absolute value of an `int` value
`static double abs(double x)`	Returns the absolute value of a `double` value
`static double pow(double base, double exponent)`	Returns the value of the first parameter raised to the power of the second parameter
`static double sqrt(double x)`	Returns the positive square root of a `double` value
`static double random()`	Returns a `double` value greater than or equal to `0.0` and less than `1.0`

Topic 5: Developing Code

Using Java `Math` Class