

implements- this class
will define the methods
mentioned in the given
interface

public class Mammal extends
Zoo Creature implements
Eater

mammals will exhibit
the behavior
outlined in Eater
interface

new - constructing a
new object of a
class

Bear b = new Bear();

↑
constructor,
like a
method which
builds the
object

this - reference to the object currently being processed.

super - reference to superclass

b is Bear

super refers to "Mammal" aspect of Bear class

integer arithmetic:

+ , - , * , pow()
no problems!

but... / ← division

result is truncated

5 / 2 // 2

(-5) / 2 // -2

truncates toward zero

$(-5.0) / 2 // -2.5$



int gets
promoted to
double

for the purpose of
evaluation.

→ uses "double"
(aka "floating point")
arithmetic

topic: casting

$\text{int } x = (\text{int})2.5674;$
↑

```
int sum;
```

Note: Codepad variables are automatically initialized
in the same way as instance fields.

```
sum
```

```
0 (int)
```

```
sum + sum
```

```
0 (int)
```

```
5/sum
```

Exception: java.lang.ArithmeticException (/ by zero)

```
sum = 2;
```

```
sum
```

```
2 (int)
```

```
5 / sum
```

```
2 (int)
```

```
-5 / sum
```

```
-2 (int)
```

```
(-5) / sum
```

```
-2 (int)
```

```
(-5.0) / sum
```

```
-2.5 (double)
```

```
sum = (-5.0) / sum;
```

Error: possible loss of precision

required: int

found: double

```
int x = 5.2647;
```

Error: possible loss of precision

required: int

found: double

```
x
```

Error: cannot find symbol - variable x

```
int x = (int) 5.2647;
```

```
x
```

```
5 (int)
```

```
x = (int) -24.86;
```

```
x
```

```
-24 (int)
```